



北京市高等教育精品教材立项项目



中国科学院研究生院教材

Textbooks of Graduate University of Chinese Academy of Sciences

# 图论与网络流理论

■ 高随祥 编著

## Graph Theory and Network Flow Theory



高等教育出版社  
Higher Education Press

## 内容提要

本书系统地阐述图论与网络流理论的基本概念、方法和定理,介绍该领域重要的问题以及典型的算法,展示图论与网络流模型及方法的广泛应用。全书立足基础、兼顾理论与应用,选材精炼,贴近研究和应用前沿,注重思想和方法。主要内容包括图的基本概念、最短路及最小生成树、连通性、匹配、Euler 图、Hamilton 图、支配集、独立集、覆盖集、图的染色、平面图、有向图、网络流等方面的理论与算法。每章配有大量习题和前沿性的专题参考文献。

本书可作为数学、运筹学、系统科学各专业硕士研究生或本科高年级学生的教材或参考书,也可供物理学、化学、生命科学、计算机科学与技术、电子科学与技术、信息科学与网络工程、资源与环境、物流与交通运输、管理科学与工程、过程工程、自动控制等学科专业的本科生、研究生使用,还可供相关领域的科研工作者、广大图论爱好者参考。

## 图书在版编目 (CIP) 数据

图论与网络流理论/高随祥编著. — 北京: 高等教育出版社, 2009. 1

ISBN 978-7-04-020009-6

I. 图… II. 高… III. ①图论 ②网络流 IV. O157.5

中国版本图书馆 CIP 数据核字 (2008) 第 188887 号

策划编辑 张长虹 责任编辑 李 陶 封面设计 王凌波  
版式设计 范晓红 责任校对 王 雨 责任印制 朱学忠

---

出版发行	高等教育出版社	购书热线	010-58581118
社 址	北京市西城区德外大街 4 号	免费咨询	800-810-0598
邮政编码	100120	网 址	<a href="http://www.hep.edu.cn">http://www.hep.edu.cn</a>
总 机	010-58581000		<a href="http://www.hep.com.cn">http://www.hep.com.cn</a>
经 销	蓝色畅想图书发行有限公司	网上订购	<a href="http://www.landraco.com">http://www.landraco.com</a>
印 刷	北京明月印务有限责任公司		<a href="http://www.landraco.com.cn">http://www.landraco.com.cn</a>
		畅想教育	<a href="http://www.widedu.com">http://www.widedu.com</a>
开 本	787×1092 1/16	版 次	2009 年 1 月第 1 版
印 张	23	印 次	2009 年 1 月第 1 次印刷
字 数	450 000	定 价	35.70 元

---

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 20009-00

# 总序

---

在中国科学院研究生院和高等教育出版社的共同努力下，凝聚着中国科学院新老科学家、研究生导师们多年心血和汗水的中国科学院研究生院教材面世了。这套教材的出版，将对丰富我院研究生教育资源、提高研究生教育质量、培养更多高素质的科技人才起到积极的推动作用。

作为科技国家队，中国科学院肩负着面向国家战略需求，面向世界科学前沿，为国家作出基础性、战略性和前瞻性的重大科技创新贡献和培养高级科技人才的使命。中国科学院研究生教育是我国高等教育的重要组成部分，在新的历史时期，中国科学院研究生教育不仅要为我院知识创新工程提供人力资源保障，还担负着落实科教兴国战略和人才强国战略，为创新型国家建设培养一大批高素质人才的重要使命。

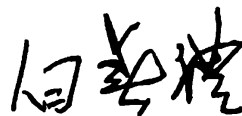
集成中国科学院的教学资源、科技资源和智力资源，中国科学院研究生院坚持教育与科研紧密结合的“两段式”培养模式，在突出科学教育和创新能力培养的同时，重视全面素质教育，倡导文理交融、理工结合，培养的研究生具有宽厚扎实的基础知识、敏锐的科学探索意识、活跃的思维和唯实、求真、协力、创新的良好素质。

研究生教材建设是研究生教育中重要的基础性工作。由一批活跃在科学前沿，同时又具有丰富教学经验的科学家编写的中国

科学院研究生院教材，适合在校研究生学习使用，也可作为高校教师和专业研究人员的参考书。这套研究生教材内容力求科学性、系统性、基础性和前沿性的统一，使学习者不仅能获得比较系统的科学基础知识，也能体会蕴于其中的科学精神、科学思想、科学方法，为进入科学研究的学术殿堂奠定良好的基础；优秀教材不但是体现教学内容和教学方法的知识载体、开展教学的基本条件和手段，也是深化教学改革、提高教育质量、促进科学教育与人文教育结合的重要保证。

“十年树木，百年树人”。我相信，经过若干年的努力，中国科学院研究生院一定能建设起多学科、多类型、多品种、多层次配套的研究生教材体系，为我国研究生教育百花园增添一枝新的奇葩，为我国高级科技人才的培养作出新的贡献。

中国科学院 常务副院长  
中国科学院研究生院 院长  
中国科学院 院士



二〇〇六年二月二十八日

# 前 言

---

图论是研究集合元素间二元关系的学科分支,这种关系可用拓扑图形来表示。图论研究这些拓扑图形的各种结构性质,如连通性、可遍行性、可平面性、匹配性质、染色性质、某些特殊结构、特殊的顶点子集和边子集以及图形上流的性质。

历经数百年的发展,特别是得益于计算机科学和信息科学的有力推动,图论与网络流理论已形成了一门既有趣又实用、既成熟又活跃的学科分支,其理论自成一体,不需要大量的预备知识,各组成部分有关联,但又相互独立,具有自己典型的方法,内容充满思想性和技巧性,是十分适合进行逻辑思维训练的“智力体操”。许多易懂不易解的难题,形成了图论与网络流理论的独特魅力,对研究者和学习者具有巨大的挑战性。图论与网络流理论的应用十分广泛,在运筹学、应用数学、计算机科学与技术、信息科学、生命科学、自动控制、工程建设以及能源、交通、电子、通信、化学、物流、管理、社会科学等众多领域都能找到其应用范例。图论与网络流理论中有大量典型的模型和算法,是许多学科中值得借鉴的模型库和算法基础。图论与网络流理论中有大量的  $NP$ - 难解问题,因而它是算法理论和设计的重要参照系和试验田。

本书是图论与网络流理论的一本入门读物,书中较为系统地阐述图论与网络流理论的基本概念、方法和定理,介绍该领域一些重要的问题以及典型的算法,展示图论与网络流理论模型与方法的广泛应用,试图为学习者从事有关方面的理论研究打下基础,也为进行应用研究的读者提供一种有力的工具。

本书根据笔者多年为研究生授课的讲义整理编写而成。成书时尽量考虑了内容的多学科适用性,力求深入浅出,既照顾初学者的入门需要,又考虑研究者的需求,既重视理论分析,又注意应用举例和内容延伸,既体现数学推理的严密性,又展示算法设计与分析的灵活性,基础知识的阐述与应用技巧的介绍并重。在选材和内容编排上力求系统全面,做到主体内容精炼、外延广泛。在文字表述上力求条理清晰、通俗易懂。

本书立足基础、面向研究和应用前沿。几乎每一章都是一个研究专题,在相应章节中指出重要的研究方向,并配有大量反映最新研究进展和成果的参考文献,以便读者可以从入门很快进入到该专题的研究前沿。每章后配有大量习题,其中既有较容易

的巩固性习题,又有关于正文内容的补充性习题,还有较难的拓展性习题。读者可根据自身情况选做。有些习题技巧性较强,建议初学者不必在某些难题上花费太多精力。

本书若作为本科生或研究生教材,以60学时为宜。有些章节可作为课后阅读材料。

笔者在图论与网络流理论的学习和研究中,得到堵丁柱教授、林诒勋教授、刘彦佩研究员、田丰研究员、胡晓东研究员等专家的指导和帮助,在此向他们表示谢意。本书的工作得到国家自然科学基金的支持并获得北京市精品教材资助。本书原稿和讲义在中科院研究生院图论与网络流理论课程的使用当中,学生们提出了不少很好的建议。本书的编写得到了中科院研究生院数学科学学院彭家贵教授和郭田德教授的鼓励与支持。中科院研究生院教材科张兆华先生为本书的出版做了不少工作。高丽丽女士参与了本书初稿的部分文字录入工作。此外,感谢高等教育出版社对本书出版的重视和支持,感谢本书引用的参考文献的所有作者。特别是本书在某些部分参考或引用了下列中文(中译文)文献的内容,特此致谢。

- [1] D. B. West, *Introduction to Graph Theory* (second edition), Prentice Hall, 2001。(中译本:图论导引。李建中、骆吉周译。北京:机械工业出版社,2006)。
- [2] J. A. Bondy and U. S. Murty, *Graph theory with applications*, 1976。(中译本:图论及其应用。吴望名等译。北京:科学出版社,1987)。
- [3] C. H. Papadimitry and K. Steiglitz, *Combinatorial Optimization — Algorithms & Complexity*, Prentice-Hall, Englewood Cliffs, New Jersey, 1982。(中译本:组合最优化 — 算法和复杂性。刘振宏、蔡茂诚译。北京:清华大学出版社,1988)。
- [4] F. Buckley and M Lewinter, *A Friendly Introduction to Graph Theory*, 2003。(中译本:图论简明教程。李慧霸等译。北京:清华大学出版社,2005)。
- [5] 王树禾。图论。北京:科学出版社,2004。
- [6] 王树禾。图论及算法。合肥:中国科学技术大学出版社,1990
- [7] 张先迪,李正良。图论及其应用。北京:高等教育出版社,2005。
- [8] 蒋长浩。图论与网络流。北京:中国林业出版社,2001。
- [9] 殷剑宏,吴开亚。图论及其算法。合肥:中国科学技术大学出版社,2003。
- [10] 谢政。网络算法与复杂性理论。2版。长沙:国防科技大学出版社,2003。
- [11] 徐俊明。图论及其应用。合肥:中国科学技术大学出版社,1998。
- [12] 孙惠泉。图论及其应用。北京:科学出版社,2004。
- [13] 田丰,马仲蕃。图与网络流理论。北京:科学出版社,1987。
- [14] 王朝瑞。图论。3版。北京:北京理工大学出版社,2001。
- [15] 魏暹菽。图论基础。西安:陕西师范大学出版社,1991。

由于笔者水平所限,书中疏漏和不足在所难免,敬请读者不吝赐教。

高随祥

2008年3月29日于北京

# 目 录

---

<b>第一章 图的基本概念</b> . . . . .	<b>1</b>
§1.1 图的基本概念 . . . . .	1
§1.2 最短路问题 . . . . .	10
§1.3 树及其性质 . . . . .	17
§1.4 生成树与最小生成树 . . . . .	19
§1.5 图的中心与中位点 . . . . .	27
§1.6 图的矩阵表示 . . . . .	34
习题一 . . . . .	38
参考文献 . . . . .	43
<b>第二章 图的连通性</b> . . . . .	<b>53</b>
§2.1 割点和割边 . . . . .	53
§2.2 连通度和边连通度 . . . . .	56
§2.3 2-连通图的性质 . . . . .	60
§2.4 Menger 定理 . . . . .	63
§2.5 可靠通信网络的设计 . . . . .	66
习题二 . . . . .	67
参考文献 . . . . .	70
<b>第三章 匹配理论</b> . . . . .	<b>75</b>
§3.1 匹配与最大匹配 . . . . .	75
§3.2 完美匹配 . . . . .	76
§3.3 二部图的匹配 . . . . .	80
§3.4 二部图中最大匹配与最大权匹配的算法 . . . . .	83

习题三	91
参考文献	93
<b>第四章 Euler 图与 Hamilton 图</b>	<b>97</b>
§4.1 Euler 图	97
§4.2 中国邮递员问题(Chinese Postman Problem)	101
§4.3 Hamilton 图	105
§4.4 旅行商问题 (Traveling Salesman Problem, TSP)	110
习题四	116
参考文献	119
<b>第五章 支配集、独立集、覆盖集和 Ramsey 数</b>	<b>122</b>
§5.1 支配集、点独立集、点覆盖集	122
§5.2 边独立集与边覆盖集	130
§5.3 支配集、点独立集、点覆盖集的求法	135
§5.4 Ramsey 数	139
习题五	147
参考文献	149
<b>第六章 染色理论</b>	<b>157</b>
§6.1 边染色	157
§6.2 点染色	162
§6.3 色多项式	171
§6.4 完美图	175
§6.5 图的边染色算法和点染色算法	182
习题六	194
参考文献	199
<b>第七章 平面图</b>	<b>214</b>
§7.1 平面图的概念	214
§7.2 Euler 公式及其应用	216
§7.3 可平面图的判断	219
§7.4 平面图的对偶图	220
§7.5 外可平面图	222
§7.6 不可平面图的几个研究方向简介	225
§7.7 平面图的面染色和四色猜想	233



---

习题七	240
参考文献	243
<b>第八章 有向图</b>	<b>260</b>
§8.1 有向图的基本概念	260
§8.2 有向路与有向圈	262
§8.3 有向图的连通性及无向图的强连通定向	264
§8.4 Euler 有向图和 Hamilton 有向图	268
§8.5 竞赛图	270
§8.6 根树及其应用	278
习题八	286
参考文献	289
<b>第九章 网络流理论与算法</b>	<b>292</b>
§9.1 网络与网络流的基本概念	292
§9.2 最大流问题及其标号算法	298
§9.3 求最大流的 Dinic 算法	302
§9.4 求最大流的推拉流算法	310
§9.5 最大流问题的一些扩展	314
§9.6 最小费用流问题	319
习题九	331
参考文献	335
<b>名词索引</b>	<b>349</b>



# 第一章 图的基本概念

---

## §1.1 图的基本概念

### 1. 图 (graph)

一集元素及它们之间的某种关系称为图. 具体地说, 图是一个二元组  $(V, E)$ , 其中集合  $V$  称为顶点集, 集合  $E$  是  $V$  中元素组成的某些无序对的集合, 称为边集.

例 1.1.1 给定  $G = (V, E)$ , 其中

$$V = \{v_1, v_2, v_3, v_4, v_5\},$$

$$E = \{(v_1, v_2), (v_2, v_3), (v_3, v_4), (v_3, v_5), (v_1, v_5), (v_1, v_5), (v_5, v_5)\}.$$

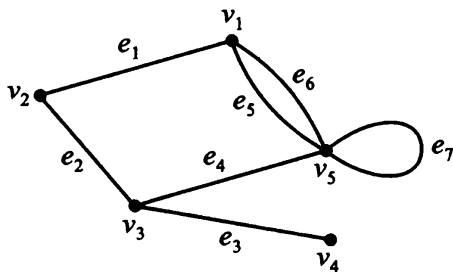
这便定义出一个图.

图的顶点集中的元素称为顶点, 边集中的元素称为边. 在本书中也常将边  $e = (u, v)$  写为  $e = uv$ , 顶点  $u$  和  $v$  称为边  $e$  的端点, 反过来也称边  $e$  连接顶点  $u$  和  $v$ . 图  $G$  的顶点数目  $|V|$  称为图  $G$  的阶, 边的数目  $|E|$  称为图  $G$  的边数. 本书中一般将图的边数记为  $\varepsilon$ , 将图的阶记为  $\nu$ .

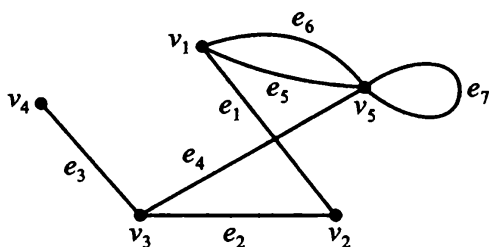
### 2. 图的图示

通常, 图的顶点可用平面上的一个点来表示, 边可用平面上的线段来表示 (直的或曲的). 这样画出的平面图形称为图的图示.

例如, 例 1.1.1 中图的一个图示为



注 (1) 表示顶点的平面点, 其相对位置并无限制. 因此, 同一个图可以画出形状迥异的很多图示. 比如下图是例 1.1.1 中图的另一个图示:



(2) 图的图示直观易懂, 因此以后一般说到一个图, 我们总是画出它的一个图示来表示.

### 3. 一些术语和概念

设  $G = (V, E)$  是一个图, 下述概念中顶点均取自  $V$ , 边均取自  $E$ .

(1) **点与边的关联**(incident): 如果在图  $G$  中点  $v$  是边  $e$  的一个端点, 则称点  $v$  与边  $e$  在图  $G$  中相关联.

(2) **点与点的相邻** (adjacent): 如果图上两点  $u, v$  被同一条边相连, 则称  $u, v$  在图  $G$  中相邻.

(3) **边与边的相邻**: 若图  $G$  中两条边有至少一个公共端点, 则称这两条边在图  $G$  中相邻.

(4) **环边** (loop): 图中两 endpoint 重合的边称为环边.

(5) **重边** (multiple edges): 设  $u$  和  $v$  是图  $G$  的顶点, 图  $G$  中连接  $u$  和  $v$  的两条或两条以上的边称为图  $G$  中  $u, v$  间的重边.

(6) **简单图** (simple graph): 既无环边也无重边的图称为简单图.

(7) **完全图** (complete graph): 任意两点间都有一条边的简单图称为完全图,  $n$  阶完全图记为  $K_n$ .

(8) **空图** (empty graph): 边集为空的图称为空图.

(9) **平凡图** (trivial graph): 边集为空且只有一个顶点的图称为平凡图.

(10) **零图** (null graph): 边集和顶点集都为空的图称为零图. (注: 顶点集为空则边集自然为空.)

(11) **顶点  $v$  的度** (degree): 图  $G$  中顶点  $v$  所关联的边的数目 (环边计两次) 称为顶点  $v$  的度, 记为  $d_G(v)$  或  $d(v)$ .

(12) **图  $G$  的最大度**:  $\Delta(G) = \max\{d_G(v) | v \in V(G)\}$ ;

(13) **图  $G$  的最小度**:  $\delta(G) = \min\{d_G(v) | v \in V(G)\}$ .

(14) **正则图** (regular graph): 各个顶点的度都相等的图称为正则图. 每个顶点的度都等于  $k$  的图称为  $k$ -正则图.

(15) **图的补图** (complement): 设  $G$  是一个图, 以  $V(G)$  为顶点集, 以  $\{(x, y) | (x, y) \notin E(G)\}$  为边集的图称为  $G$  的补图, 记为  $\bar{G}$ .

**定理 1.1.1** 对任何图  $G$ , 其各顶点度数之和等于边数的 2 倍, 即  $\sum_{v \in V(G)} d(v) = 2\varepsilon$ .

**证明** 按每个顶点的度来计数边, 每条边恰数了两次. 证毕.

**推论 1.1.1** 任何图中, 奇度顶点的个数总是偶数 (包括 0).

**证明** 设图  $G$  中有  $k$  个奇度顶点, 不妨设  $v_1, v_2, \dots, v_k$  为奇度顶点, 而  $v_{k+1}, v_{k+2}, \dots, v_\nu$  为偶度顶点, 则由定理 1.1.1,  $\sum_{i=1}^k d(v_i) + \sum_{i=k+1}^{\nu} d(v_i) = 2\varepsilon$ , 即

$$\sum_{i=1}^k d(v_i) = 2\varepsilon - \sum_{i=k+1}^{\nu} d(v_i),$$

此式右端为偶数, 而左端为  $k$  个奇数之和, 因此  $k$  必定为偶数. 证毕.

#### 4. 子图

**子图** (subgraph): 对图  $G$  和  $H$ , 如果  $V(H) \subseteq V(G)$  且  $E(H) \subseteq E(G)$ , 则称图  $H$  是图  $G$  的子图, 记为  $H \subseteq G$ .

**生成子图** (spanning subgraph): 若  $H$  是  $G$  的子图且  $V(H) = V(G)$ , 则称  $H$  是  $G$  的生成子图.

**点导出子图** (induced subgraph): 设  $G$  是一个图,  $V' \subseteq V(G)$ . 以  $V'$  为顶点集, 以  $G$  中两端点均属于  $V'$  的所有边作为边集所组成的子图, 称为  $G$  的由顶点集  $V'$  导出的子图, 简称为  $G$  的点导出子图, 记为  $G[V']$ .

**边导出子图** (edge-induced subgraph): 设  $G$  是一个图,  $E' \subseteq E(G)$ . 以  $E'$  为边集, 以  $E'$  中边的所有端点作为顶点集所组成的子图, 称为  $G$  的由边集  $E'$  导出的子图, 简称为  $G$  的边导出子图, 记为  $G[E']$ .

设  $V' \subseteq V(G)$ ,  $E' \subseteq E(G)$ , 今后经常用  $G - V'$  表示从图  $G$  中删除顶点子集  $V'$  (连同它们关联的边一起删去) 所获得的子图, 用  $G - E'$  表示从图  $G$  中删除边子集  $E'$  (但不删除它们的端点) 所获得的子图. 特别地, 对顶点  $v$  和边  $e$ , 常用  $G - v$  表示  $G - \{v\}$ ,  $G - e$  表示  $G - \{e\}$ .

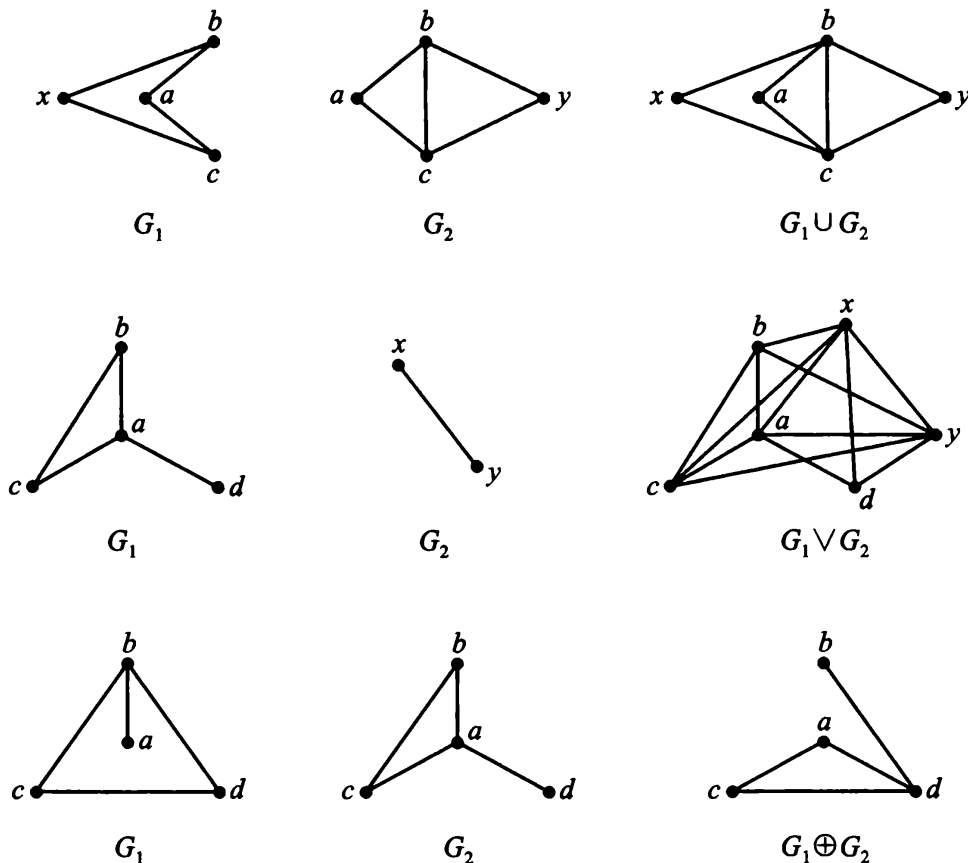
#### 5. 图的并、联和对称差

设  $G_1, G_2$  是两个图,  $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ , 则  $G_1$  与  $G_2$  的**并** (union) 是指图  $(V_1 \cup V_2, E_1 \cup E_2)$ , 记为  $G_1 \cup G_2$ . 特别地, 若  $V_1 \cap V_2 = \emptyset$ , 则图  $G_1$  与  $G_2$  的并  $G_1 \cup G_2$  称为  $G_1$  与  $G_2$  的**不交并** (disjoint union), 不交并有时也称为**和** (addition), 记为  $G_1 + G_2$ .

两个无公共顶点的图  $G_1$ 、 $G_2$  的不交并  $G_1+G_2$  再添加边集  $\{xy|x \in V(G_1), y \in V(G_2)\}$  后得到的图称为  $G_1$  与  $G_2$  的**联** (join, 有时也称为**连接**), 记为  $G_1 \vee G_2$ .

设  $A$ 、 $B$  是两个集合, 集合  $(A-B) \cup (B-A)$  称为  $A$  与  $B$  的**对称差**, 记为  $A \oplus B$ . 容易证明:  $A \oplus B = (A \cup B) - (A \cap B)$ . 设  $G_1 = (V, E_1)$  与  $G_2 = (V, E_2)$  是两个具有相同顶点集的图, 则图  $(V, E_1 \oplus E_2)$  称为  $G_1$  和  $G_2$  的**对称差** (symmetric difference), 记为  $G_1 \oplus G_2$ .

通俗地说, 两个图  $G_1$  与  $G_2$  的并  $G_1 \cup G_2$  就是将两个图的顶点和边都放在一起所形成的图 (重复的顶点和边只取一次); 两个无公共点的图  $G_1$  与  $G_2$  的并即为不交并  $G_1+G_2$ , 它就是将这两个图作为分支放在一起组成的图;  $G_1$  与  $G_2$  的联  $G_1 \vee G_2$  表示将  $G_1$  的每个顶点与  $G_2$  的所有顶点都连边构成的图. 两个具有相同顶点集的图  $G_1$  与  $G_2$  的对称差  $G_1 \oplus G_2$ , 是与  $G_1$ 、 $G_2$  具有相同顶点集的图, 其边集由恰在  $G_1$ 、 $G_2$  之一中出现的边所构成. 下图是并、联、对称差的示例.



## 6. 路和圈

**途径 (walk):** 图  $G$  中一个点边接续交替出现的序列  $w = v_{i_0}e_{i_1}v_{i_1}e_{i_2} \cdots e_{i_k}v_{i_k}$  称为图  $G$  的一条途径, 其中  $v_{i_0}$ 、 $v_{i_k}$  分别称为途径  $w$  的起点和终点,  $w$  上其余顶点称为中途点.

**迹 (trail):** 图  $G$  中边不重复出现的途径称为迹.

**路 (path):** 图  $G$  中顶点不重复出现的迹称为路.

**闭途径 (closed walk):** 图  $G$  中起点和终点相同的途径称为闭途径.

**闭迹 (closed trail):** 图  $G$  中边不重复出现的闭途径称为闭迹, 也称为回路 (circuit).

**圈 (cycle):** 中途点不重复的闭迹称为圈.

**注** (1) 简单图中的路可以完全用顶点来表示, 如  $P = v_{i_0}v_{i_1} \cdots v_{i_k}$ .

(2) 图中从一点  $u$  到另一点  $v$  的路(迹)有时简称为  $(u, v)$  路(迹).

(3) 途径(闭途径)、迹(闭迹)、路(圈)上所含的边的数目称为它们的长度.

(4) 简单图  $G$  中长度为奇数和偶数的圈分别称为**奇圈 (odd cycle)** 和**偶圈 (even cycle)**.

(5) 对任意  $u, v \in V(G)$ , 从  $u$  到  $v$  的具有最小长度的路称为  $u$  到  $v$  的**最短路 (shortest path)**, 其长度称为  $u$  到  $v$  的**距离 (distance)**, 记为  $d_G(u, v)$  或  $d(u, v)$ .

(6) 简单图  $G$  中最短圈的长度称为图  $G$  的**围长 (girth)**, 最长圈的长度称为图  $G$  的**周长 (circumference)**.

**例 1.1.2** 设  $G$  是一个简单图, 若最小度  $\delta(G) \geq 2$ , 则  $G$  中必含有圈.

**证明** 设  $G$  中的最长路为  $P = v_0v_1 \cdots v_k$ . 因  $d(v_0) \geq 2$ , 故存在与  $v_1$  相异的顶点  $v$  与  $v_0$  相邻. 若  $v \notin P$ , 则得到比  $P$  更长的路, 这与  $P$  的取法矛盾. 因此必定  $v \in P$ , 从而  $G$  中有圈. 证毕.

**例 1.1.3** 设  $G$  是简单图, 若最小度  $\delta(G) \geq 3$ , 则  $G$  必有偶圈.

**证明** 设  $P = v_0v_1 \cdots v_k$  是  $G$  的最长路.

因为  $d(v_0) \geq 3$ , 所以存在两个与  $v_1$  相异的顶点  $v'$  和  $v''$  与  $v_0$  相邻.  $v', v''$  必都在路  $P$  上, 否则会得到比  $P$  更长的路. 不妨设  $v' = v_i, v'' = v_j$  ( $2 < i, j \leq k, i < j$ ).

若  $i, j$  中有奇数, 比如  $i$  是奇数, 则路  $P$  上  $v_0$  到  $v_i$  的一段与边  $v_0v_i$  构成一个偶圈;

若  $i, j$  都是偶数, 则路  $P$  上  $v_i$  到  $v_j$  的一段与边  $v_0v_i$  及  $v_0v_j$  构成一个偶圈. 证毕.

**例 1.1.4** 设  $G$  是简单图, 若最小度  $\delta(G) \geq 3$ , 则  $G$  中各个圈长的最大公因数是 1 或 2.

**证明** 取  $G$  中最长路  $P = v_0v_1 \cdots v_k$ . 由上例知,  $G$  中存在  $v_i, v_j \in P$ , 使  $P$  上  $(v_0, v_i)$  段、 $(v_0, v_j)$  段、 $(v_i, v_j)$  段与边  $v_0v_i, v_0v_j$  构成长度分别为  $i+1, j+1$  和  $j-i+2$  的圈. 若  $i+1, j+1, j-i+2$  三数的公因数  $m > 2$ , 则因  $m$  整除前两数的差  $j-i$ , 同时又整除第三个数, 于是  $m$  整除 2, 这是不可能的. 因此  $i+1, j+1, j-i+2$  三数的公因数必不超过 2. 从而各个圈长的最大公因数是 1 或 2. 证毕.

## 7. 二部图

**二部图** (bipartite graph): 若图  $G$  的顶点集可划分为两个非空子集  $X$  和  $Y$ , 使得  $G$  的任一条边都有一个端点在  $X$  中, 另一个端点在  $Y$  中, 则称  $G$  为二部图 (或偶图), 记为  $G = (X \cup Y, E)$  或  $G = (X, Y)$ ,  $(X, Y)$  称为  $G$  的一个顶点二划分.

**完全二部图** (complete bipartite graph): 在二部图  $G = (X \cup Y, E)$  中, 若  $X$  的每个顶点与  $Y$  的每个顶点有边连接, 则称  $G$  为完全二部图; 若  $|X| = m, |Y| = n$ , 则记此完全二部图为  $K_{m,n}$ .

**定理 1.1.2** 一个图是二部图当且仅当它不含奇圈.

**证明 必要性** 设  $C = v_0 v_1 v_2 \cdots v_k v_0$  是二部图  $G = (X \cup Y, E)$  的一个圈. 不妨设  $v_0 \in X$ , 由二部图的定义知,  $v_1 \in Y, v_2 \in X, \cdots$ , 一般地,  $v_{2i} \in X, v_{2i+1} \in Y, (i = 0, 1, \cdots)$ . 又因  $v_0 \in X$ , 故  $v_k \in Y$ , 因而  $k$  是奇数. 注意到圈  $C$  上共有  $k+1$  条边, 因此  $C$  是偶圈.

**充分性** 设  $G$  不含奇圈. 任取  $u \in V(G)$ , 令

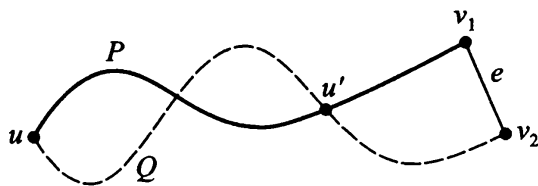
$X = \{v \in V(G) | u, v \text{ 间的距离为奇数}\}, Y = \{v \in V(G) | u, v \text{ 间的距离为偶数}\}.$

任取  $G$  的一条边  $e = v_1 v_2$ , 欲证  $v_1, v_2$  分属于  $X$  和  $Y$ . 设  $P, Q$  分别是  $u$  到  $v_1, v_2$  的最短路.

(1) 如果  $P$  与  $Q$  的最后一个公共顶点是  $v_1$  或  $v_2$  (不妨设为  $v_1$ , 如下图所示), 则因  $P$  和  $Q$  的  $(u, v_1)$  段都是  $u$  到  $v_1$  的最短路, 故这两段长度相等. 因此  $P$  的长度与  $Q$  的长度相差 1, 从而  $u$  到  $v_1, v_2$  的距离奇偶性相反,  $v_1, v_2$  分属于  $X$  和  $Y$ .



(2) 如果  $P$  与  $Q$  的最后一个公共顶点不是  $v_1$  和  $v_2$ , 设其为  $u'$  (如下图), 因  $P$  的  $(u, u')$  段和  $Q$  的  $(u, u')$  段都是  $u$  到  $u'$  的最短路, 故这两段长度相等.

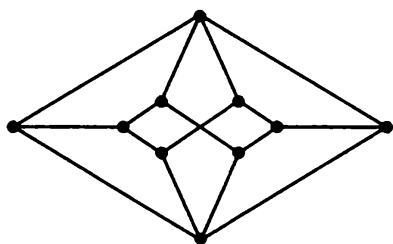


假如  $P, Q$  的奇偶性相同, 则  $P$  的  $(u', v_1)$  段和  $Q$  的  $(u', v_2)$  段奇偶性相同, 这两段与边  $e$  构成一个奇圈, 与定理条件矛盾. 可见  $P, Q$  的奇偶性不同, 从而  $v_1, v_2$  分属于  $X$  和  $Y$ .

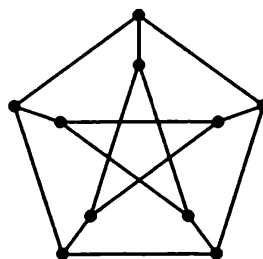
这便证明了  $G$  是一个二部图. 证毕.



例 1.1.5 判断下列图是不是二部图.

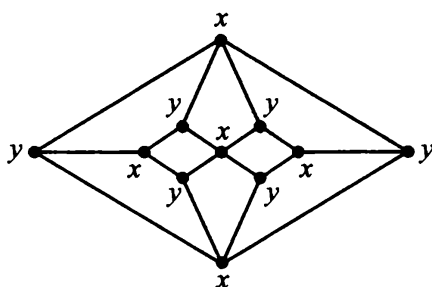


Herschel 图



Peterson 图

解 Herschel 图的一个顶点二划分如下:



可见 Herschel 图是一个二部图.

Peterson 图中含有奇圈, 因此不是二部图.

## 8. 连通性

**图中两点的连通:** 如果在图  $G$  中  $u, v$  两点间有路相通, 则称顶点  $u, v$  在图  $G$  中连通.

**连通图 (connected graph):** 若图  $G$  中任二顶点都连通, 则称图  $G$  是连通图.

**图的连通分支 (connected branch, component):** 若图  $G$  的顶点集  $V(G)$  可划分为若干非空子集  $V_1, V_2, \dots, V_w$ , 使得两顶点属于同一子集当且仅当它们在  $G$  中连通, 则称每个点导出子图  $G[V_i]$  为图  $G$  的一个连通分支 ( $i = 1, 2, \dots, w$ ).  $G$  的连通分支的个数  $w$  称为  $G$  的连通分支数.

**注** (1) 图  $G$  的每个连通分支是  $G$  的一个极大连通子图.

(2) 图  $G$  连通当且仅当连通分支数  $w = 1$ .

下面的定理描述了连通图的一个性质.

**定理 1.1.3** 若图  $G$  连通, 则  $\varepsilon(G) \geq \nu(G) - 1$ .

**证明** 对顶点数  $\nu$  做数学归纳法.

$\nu = 1, 2$  时,  $\varepsilon \geq \nu - 1$  显然成立.

假设  $\nu \leq k$  的连通图都  $\varepsilon \geq \nu - 1$ .

对于  $\nu = k + 1$  的连通图  $G$ , 任取  $v \in V(G)$ , 考虑  $G - v$ .

若  $G - v$  连通, 则由归纳假设,  $\varepsilon(G - v) \geq \nu(G - v) - 1 = k - 1$ , 而

$$\varepsilon(G) \geq \varepsilon(G - v) + 1 \geq (k - 1) + 1 = (k + 1) - 1 = \nu(G) - 1.$$

若  $G - v$  不连通, 设  $G_1, G_2, \dots, G_w$  是其连通分支 ( $w \geq 2$ ). 由归纳假设,  $\varepsilon(G_i) \geq \nu(G_i) - 1$ , ( $i = 1, 2, \dots, w$ ). 故

$$\varepsilon(G - v) = \sum_{i=1}^w \varepsilon(G_i) \geq \sum_{i=1}^w \nu(G_i) - w = \nu(G - v) - w = k - w,$$

而  $\varepsilon(G) \geq \varepsilon(G - v) + w \geq (k - w) + w = (k + 1) - 1 = \nu(G) - 1$ .

归纳法完成. 证毕.

**例 1.1.6** 设有  $2n$  个电话转接台, 每个台与至少  $n$  个台有直通线路, 证明该转接系统中任两台均可实现通话.

**证明** 构造图  $G$  如下: 以转接台作为顶点, 两顶点间连边当且仅当对应的两台间有直通线路. 问题化为: 已知图  $G$  有  $2n$  个顶点, 且最小度  $\delta(G) \geq n$ , 求证  $G$  连通.

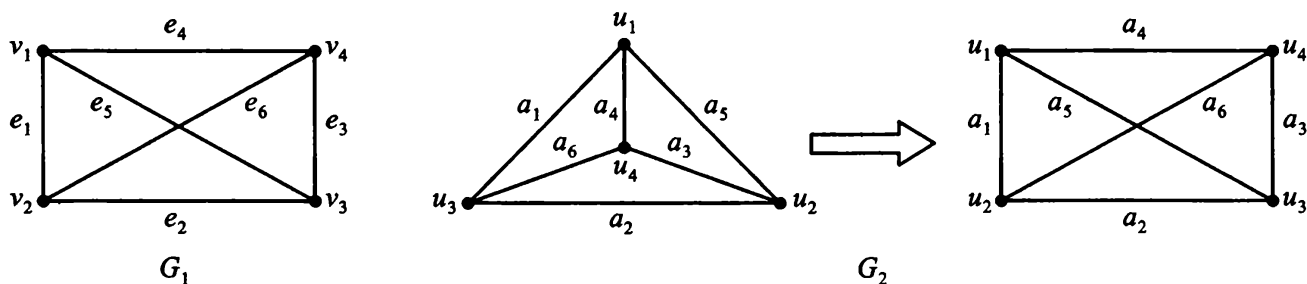
事实上, 假如  $G$  不连通, 则至少有一个连通分支的顶点数不超过  $n$ . 在此连通分支中, 顶点的度至多是  $n - 1$ . 这与  $\delta(G) \geq n$  矛盾. 证毕.

**例 1.1.7** 证明: 若图中只有两个奇度顶点, 则它们必连通.

**证明** 用反证法. 设  $u, v$  是图  $G$  中仅有的两个奇度顶点. 假如  $u$  与  $v$  不连通, 则它们必分属于不同的连通分支. 将每个分支看成一个图时, 其中只有一个奇度顶点. 这与推论 1.1.1 矛盾. 证毕.

## 9. 图的同构

我们已经知道, 同一个图可以有不同形状的图示. 反过来, 两个不同的图也可以有形状相同的图示. 比如:



易见  $G_1$  和  $G_2$  的顶点及边之间都一一对应, 且连接关系完全相同, 只是顶点和边的名称不同而已. 这样的两个图称为是**同构的** (isomorphic).

从数学上看,同构的两个图,其顶点间可建立一一对应,边之间也能建立一一对应,且若一图的两点间有边,则在另一图中对应的两点间有对应的边.严格的数学定义如下.

**定义 1.1.1** 对两个图  $G = (V(G), E(G))$  与  $H = (V(H), E(H))$ , 如果存在两个一一映射:

$$\alpha: V(G) \rightarrow V(H), \quad \beta: E(G) \rightarrow E(H),$$

使得对任意  $e = (u, v) \in E(G)$ , 都有  $(\alpha(u), \alpha(v)) \in E(H)$  且  $\beta(e) = (\alpha(u), \alpha(v))$ , 则称图  $G$  与  $H$  同构, 记为  $G \cong H$ .

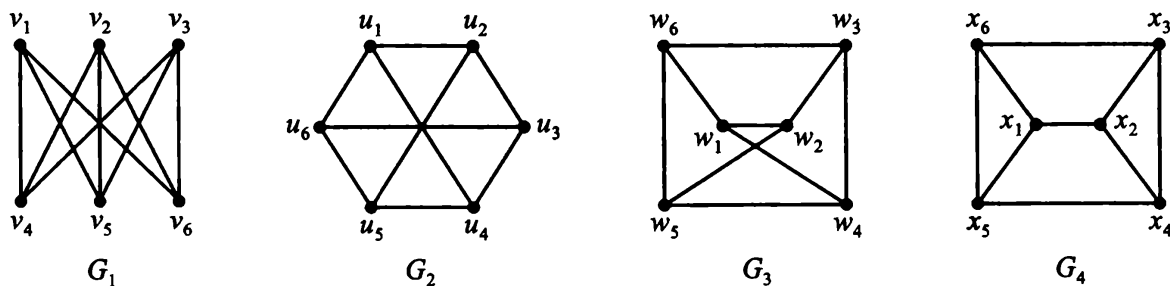
图的同构关系是一种等价关系 (即满足反身性、对称性、传递性), 这种等价关系将  $\nu$  阶图分成若干等价类, 彼此同构的图属于同一类. 同一个等价类中的图有相同的结构, 差别仅在于顶点和边的名称不同. 由于人们关心的是图的结构, 因此, 通常将同构图类中的所有图看成同一个图, 而不在于它们顶点和边的名称以及它们的图示画法. 在图的图示中, 不给顶点和边标定名称的图称为非标定图, 否则称为标定图. 非标定图实际上就是一个同构图类的代表. 在不致误解的情况下, 我们也往往不严格区分标定图与非标定图.

目前尚无简便的方法判别两个图是否同构, 特别是当图的顶点数较大时, 判断两个图是否同构是一件很困难的事情.

两图同构, 首先它们的阶必须相等, 边数必须相等;其次要有相同的环边、重边及圈的状态; 还应保持顶点的度, 即在同构映射下相对应的顶点必须有相同的度. 这些都是两图同构的必要条件, 可用来判断两图不同构.

为判定两图同构, 一般要按定义构造出两图顶点间的一一映射, 然后检验它是否保持邻接关系. 有时也可根据图的特点使用特殊方法.

**例 1.1.8** 判断下列图是否同构.



**解** 图  $G_1$  和  $G_2$  是同构的. 事实上, 定义它们顶点间的映射  $f$ , 分别将顶点  $v_1, v_2, v_3, v_4, v_5, v_6$  映射到  $u_1, u_3, u_5, u_2, u_4, u_6$ , 显然这是  $G_1$  到  $G_2$  的一个一一映射, 且容易验证它保持顶点间的相邻关系.

图  $G_2$  和  $G_3$  也是同构的. 事实上, 定义它们顶点间的映射  $g$ , 分别将顶点  $u_1, u_2, u_3, u_4, u_5, u_6$  映射到  $w_1, w_2, w_3, w_4, w_5, w_6$ , 显然这是  $G_2$  到  $G_3$  的一个一一映射, 且容易验证它保持顶点间的相邻关系.

图  $G_3$  和  $G_4$  不同构, 因为  $G_4$  含有三角形 (即长为 3 的圈), 但  $G_3$  不含.

注 (1)  $G_1, G_2, G_3$  的同构性还可以通过它们的二部图特性来证明. 可以检验, 它们都是完全二部图  $K_{3,3}$ .

(2) 容易证明, 两个图  $G$  和  $H$  同构当且仅当它们的补图  $\bar{G}, \bar{H}$  同构. 利用这一结论, 也可以较简便地判定  $G_1, G_2, G_3$  的同构性, 事实上,  $G_1, G_2, G_3$  的补图都是两个不相交的三角形 (长为 3 的圈  $C_3$ ), 因此同构. 而  $G_4$  的补图是长为 6 的圈  $C_6$ , 故  $G_4$  与前三个图不同构.

关于图的同构有两个猜想至今尚未解决.

**猜想 1 (Ulam 重构猜想, 1929)** 设  $G$  与  $H$  是两个图,  $V(G) = \{u_1, u_2, \dots, u_n\}$ ,  $V(H) = \{v_1, v_2, \dots, v_n\}$ . 若对  $\forall i \in \{1, 2, \dots, n\}$ , 都有  $G - u_i \cong H - v_i$ , 则  $G \cong H$ .

**猜想 2** 设  $G$  与  $H$  都是至少有四条边的图, 若存在一一映射  $\varphi: E(G) \rightarrow E(H)$ , 使得对  $\forall e \in E(G)$ , 都有  $G - e \cong H - \varphi(e)$ , 则  $G \cong H$ .

有关图的重构问题的更多内容可参看文献 [1]、[2].

## §1.2 最短路问题

### 一、赋权图

对图  $G$  的每条边  $e$ , 赋以一个实数  $w(e)$ , 称为边  $e$  的权. 每条边都赋有权的图称为赋权图.

权在不同的问题中会有不同的含义. 例如交通网络中, 权可能表示运费、里程或道路的造价等.

设  $H$  是赋权图  $G$  的一个子图,  $H$  的权定义为  $W(H) = \sum_{e \in E(H)} w(e)$ . 特别地, 对  $G$  中一条路  $P$ , 其权为  $W(P) = \sum_{e \in E(P)} w(e)$ .

### 二、最短路问题

**最短路问题:** 给定赋权图  $G$  及  $G$  中两点  $u, v$ , 求  $u$  到  $v$  的具有最小权的路 (称为  $u$  到  $v$  的最短路).

注 赋权图中路的权也称为路的长, 最短  $(u, v)$  路的长也称为  $u, v$  间的距离, 记为  $d(u, v)$ .

最短路问题是一个优化问题, 属于网络优化和组合优化的范畴. 对这种优化问题的解答一般是一个算法. 最短路问题有很多算法, 其中最基本的一个是 Dijkstra 算法.

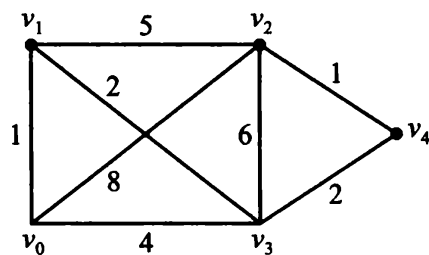
### 三、Dijkstra 算法

#### 1. 算法思想

设赋权图  $G$  中所有边都具有非负权, Dijkstra 算法的目标是求出  $G$  中某个指定顶点  $v_0$  到其他所有点的最短路. 它依据的基本原理是: 若路  $P = v_0v_1 \cdots v_{k-1}v_k$  是从  $v_0$  到  $v_k$  的最短路, 则  $P' = v_0v_1 \cdots v_{k-1}$  必是从  $v_0$  到  $v_{k-1}$  的最短路. 基于这一原理, 算法由近及远地逐次求出  $v_0$  到其他各点的最短路.

下面通过例子说明具体做法.

例 1.2.1 在下图中求  $v_0$  到其他各点的最短路.



解 (1) 令  $S = \{v_0\}$ ,  $\bar{S} = V \setminus S$ , 求  $v_0$  到  $\bar{S}$  中最近点的最短路. 在当前的例子中, 从  $v_0v_1$ 、 $v_0v_2$ 、 $v_0v_3$  中选一条最短的, 结果获得  $v_0$  到  $v_1$  的最短路  $v_0v_1$ .

(2) 令  $S := S \cup \{v_1\}$ ,  $\bar{S} := V \setminus S$ ,  $d(v_0, v_0) = 0$ , 求  $v_0$  到  $\bar{S}$  中最近点的最短路. 这里“最近”是指  $v_0$  到  $\bar{S}$  的直接连边、以及由  $v_0$  到  $S$  中每点  $v_i$  的最短路 (已求出) 再添加  $v_i$  向  $\bar{S}$  的连边所形成的路中最短的. 即选取  $\bar{S}$  中一点  $v$  使得距离

$$d(v_0, v) = \min_{v_i \in S, v \in \bar{S}} \{d(v_0, v_i) + w(v_i v)\}. \quad (*)$$

在当前的例子中, 从  $v_0v_2$ 、 $v_0v_3$ 、 $v_0v_1v_2$ 、 $v_0v_1v_3$  中选一条最短的, 结果获得  $v_0$  到  $v_3$  的最短路  $v_0v_1v_3$ .

(3) 令  $S := S \cup \{v_3\}$ ,  $\bar{S} := V \setminus S$ , 求  $v_0$  到  $\bar{S}$  中最近点的最短路. 即选取  $\bar{S}$  中一点  $v$  使得

$$d(v_0, v) = \min_{v_i \in S, v \in \bar{S}} \{d(v_0, v_i) + w(v_i v)\}.$$

当前应从  $v_0v_2$ 、 $v_0v_1v_2$ 、 $v_0v_1v_3v_2$ 、 $v_0v_1v_3v_4$  中选一条最短的, 结果获得  $v_0$  到  $v_4$  的最短路  $v_0v_1v_3v_4$ .

(4) 令  $S := S \cup \{v_4\}$ ,  $\bar{S} := V \setminus S$ , 求  $v_0$  到  $\bar{S}$  中最近点的最短路. 即选取  $\bar{S}$  中一点  $v$  使得

$$d(v_0, v) = \min_{v_i \in S, v \in \bar{S}} \{d(v_0, v_i) + w(v_i v)\}.$$

当前应从  $v_0 v_2$ 、 $v_0 v_1 v_2$ 、 $v_0 v_1 v_3 v_2$ 、 $v_0 v_1 v_3 v_4 v_2$  中选一条最短的, 结果获得  $v_0$  到  $v_2$  的最短路  $v_0 v_1 v_2$  (或  $v_0 v_1 v_3 v_4 v_2$ ).

一般的, 若  $S = \{v_0, v_1, \dots, v_k\}$  以及相应的最短路已找到, 则可用 (\*) 式来选取  $v$ , 获得  $v_0$  到  $v$  的最短路.

## 2. 算法实现 —— 标号法

在上述算法的过程中, 每轮循环求  $d(v_0, v)$  时, 都要对所有的点  $v_i \in S$  计算  $d(v_0, v_i) + w(v_i v)$  且比较出一个最小的值, 因而在算法循环过程中需要大量重复的路长计算和比较. 为了避免这种重复计算, Dijkstra 算法采用标号方法来实现: 算法执行中, 给每个点  $v$  都附一个标号  $l(v)$ , 表示当前已经算出的从  $v_0$  到该点的最短路的长  $d(v_0, v)$ . 算法每轮循环都考虑修改点  $v$  的标号, 如果通过此前刚刚进入  $S$  集合的点  $v_i$  到  $v$  的连边不能获得更短的  $(v_0, v)$  路, 则该点保持原有标号  $l(v)$  不变; 否则, 修改该点标号为  $l(v) := l(v_i) + w(v_i v)$ , 当前  $v_0$  到  $v$  的最短路应由  $v_0$  到  $v_i$  的最短路及边  $v_i v$  构成.

标号法的基本原理是累进比较. 初始时,  $l(v_0) := 0$ ,  $l(v) := \infty$  ( $v \neq v_0$ ),  $S = \{v_0\}$ ,  $\bar{S} = V \setminus S$ . 然后算法逐步修改  $\bar{S}$  中顶点的标号. 第  $i$  步时, 对  $\bar{S}$  中每个  $v$ , 只对刚进入  $S$  的点  $v_i$  计算  $d(v_0, v_i) + w(v_i v)$  (即  $l(v_i) + w(v_i v)$ ), 并与  $l(v)$  进行比较, 取其较小的一个作为的新标号, 即  $l(v) := \min\{l(v), l(v_i) + w(v_i v)\}$ . 因为对在  $v_i$  之前进入  $S$  的点  $v_0, v_1, \dots, v_{i-1}$ ,  $d(v_0, v_k) + w(v_k v)$  ( $k = 1, 2, \dots, i-1$ ) 的值及其大小信息已含于  $l(v)$  之中, 因此只需计算一个值  $d(v_0, v_i) + w(v_i v)$ , 并将其与此前纪录的最短路的长  $l(v)$  进行比较即可, 而不必对所有的点  $v_i \in S$  都重新计算  $d(v_0, v_i) + w(v_i v)$  且比较出一个最小的, 从而避免了重复计算和比较.

标号法实现算法主要包括两个过程: (1) 修改各点的标号, (2) 从  $\bar{S}$  的所有点中取标号最小的一个点, 放入  $S$  中. 某个点被放入  $S$  集合后, 它的标号成为永久标号, 不再被修改. 算法反复执行上述过程, 直至所有顶点获得永久标号 (被放入  $S$  中) 为止.

算法具体步骤如下:

**Dijkstra 算法:** 求非负权赋权图中  $v_0$  点到其余各点的最短路.

**输入:** 非负权赋权图  $G$ ,  $G$  中指定顶点  $v_0$ .

**输出:**  $G$  中从  $v_0$  到其余各点的最短路.

第 1 步: 令  $l(v_0) := 0$ ,  $l(v) := \infty$  ( $v \neq v_0$ ),  $S := \{v_0\}$ ,  $\bar{S} := V \setminus S$ ,  $i := 0$ .

第 2 步: 对每个  $v \in \bar{S}$ , 令  $l(v) := \min\{l(v), l(v_i) + w(v_i v)\}$ .

取  $v^* \in \bar{S}$  使得  $l(v^*) = \min_{v \in \bar{S}} \{l(v)\}$ . 记  $v_{i+1} = v^*$ , 令  $S := S \cup \{v_{i+1}\}$ ,  
 $\bar{S} := V \setminus S$ .

第 3 步: 令  $i := i + 1$ . 如果  $i = \nu - 1$ , 则停止, 输出各点标号并反向追溯最短路; 否则, 转第 2 步.

### 3. 算法的正确性

**定理 1.2.1** Dijkstra 算法结束时, 对任一个顶点  $v$ , 其标号  $l(v)$  恰是  $v_0$  到  $v$  的最短路的长.

**证明** 按照算法, 顶点  $v$  的最终标号  $l(v)$  是  $v$  进入集合  $S$  时的标号. 假设  $v$  在算法第  $i$  次循环时进入集合  $S$ , 则  $v = v_{i+1}$ , 且  $v$  进入集合  $S$  后  $S = \{v_0, v_1, \dots, v_i, v_{i+1}\}$ . 由于  $l(v)$  是算法在前  $i$  次循环中逐次比较出的当前从  $v_0$  到  $v$  最短路的长, 因此从  $v_0$  到  $v$  仅含  $\{v_0, v_1, \dots, v_i, v_{i+1}\}$  中点的任何路的长都不会小于  $l(v)$ . 任取图  $G$  中一条从  $v_0$  到  $v$  的路  $P$ , 如果  $P$  仅含  $\{v_0, v_1, \dots, v_i, v_{i+1}\}$  中的点, 则如上所述,  $P$  的长不会小于  $l(v)$ ; 如果  $P$  含  $\{v_0, v_1, \dots, v_i, v_{i+1}\}$  之外的点, 不妨设沿着  $P$  从  $v_0$  出发第一个不在  $\{v_0, v_1, \dots, v_i, v_{i+1}\}$  中的顶点是  $v'$ . 因在算法第  $i$  次循环时,  $v_{i+1}$  进入  $S$  而  $v'$  仍未进入  $S$ , 说明当前  $l(v_{i+1}) \leq l(v')$ . 注意  $v'$  只是  $P$  的中途点且图  $G$  中所有边的权都非负, 而当前的  $l(v')$  等于  $v_0$  到  $v'$  仅含  $\{v_0, v_1, \dots, v_i, v_{i+1}\}$  中点的最短路的长, 故路  $P$  的长大于或等于  $P$  上  $v_0$  到  $v'$  段的长 ( $\geq l(v')$ ), 从而  $P$  的长不会小于  $l(v_{i+1})$  (即  $l(v)$ ). 总之, 我们已证明了  $G$  中任一条从  $v_0$  到  $v$  的路  $P$  的长都不会小于  $l(v)$ , 并且按照算法, 从  $v_0$  到  $v$  长为  $l(v)$  的路  $P$  是存在的, 因此标号  $l(v)$  恰是  $v_0$  到  $v$  的最短路的长. 证毕.

按照上述定理, Dijkstra 算法结束时, 每个顶点的标号  $l(v)$  表示  $v_0$  到该点的最短路的长. 此外, 通过检查标号的来源点, 可反向追踪获得  $v_0$  到各点的最短路.

### 4. 算法的有效性

对于一个图论算法, 如果在任何图上施行这个算法所需要的基本计算量 (比如加、减、乘、除运算或比较、判断、赋值等) 称为该算法的计算复杂度或时间复杂度, 有时简称为复杂度 (complexity). 算法的计算复杂度一般都是图的顶点数  $\nu$  的一个函数, 图的顶点数  $\nu$  称为图论问题的规模. 人们总是希望利用计算机执行算法以解决人工难以计算的大规模问题, 因此, 对一个算法人们主要关心它在解决大规模问题时的计算量. 对于图论算法, 主要关心当图的顶点数  $\nu$  增大时, 算法所需计算量的增加情况. 如果一个算法的计算复杂度是规模  $\nu$  的指数函数, 比如  $2^\nu$ 、 $\nu!$  等, 则当问题的规模较大时, 算法所需的计算量快速增大以至于无法接受, 这样的算法在解决大规模问题时实际上是无用的. 当一个算法的计算复杂度是多项式函数时, 算法的计算量随问题规模的增

加其增幅较缓,人们认为一般是可以接受的.因此,一个具有多项式时间复杂度的算法称为**有效算法** (efficient algorithm) 或**好算法** (good algorithm),有时直接称为**多项式时间算法**或**多项式算法**.对于多项式算法,影响其时间复杂度的主项是多项式的最高次项,因此人们在作算法分析时,主要关注其最高次项.一般地,如果一个算法的复杂度是多项式  $a_k\nu^k + a_{k-1}\nu^{k-1} + \cdots + a_1\nu + a_0$  (其中  $a_0, a_1, \cdots, a_k$  是常系数),则我们说它的计算复杂度是  $O(\nu^k)$ ,或是  $O(\nu^k)$  阶的.利用这个记号,如果一个算法的复杂度是  $2^\nu$  或  $\nu!$ ,我们也可写为  $O(2^\nu)$  或  $O(\nu!)$ .

下一定理表明 Dijkstra 算法是多项式算法.

**定理 1.2.2** Dijkstra 算法的计算复杂度为  $O(\nu^2)$ .

**证明** Dijkstra 算法的主要计算量在第 2 步.在第  $i$  次循环中,第 2 步第 1 式需要  $\nu-i-1$  次加法,  $\nu-i-1$  次比较;第 2 式需要不超过  $\nu-i-1$  次比较. ( $i = 0, 1, \cdots, \nu-2$ ).

从而  $\nu-1$  次循环总的计算量不超过  $\sum_{i=0}^{\nu-2} 3(\nu-i-1) = \frac{3\nu(\nu-1)}{2} = O(\nu^2)$ . 证毕.

对一个需要用算法解决的问题,如果存在求解它的多项式算法,则这个问题称为  $P$  问题 (polynomial problem). 如果一个问题,当猜出其答案时可在多项式计算量内验证答案的正确性,则这个问题称为  $NP$  问题 (non-deterministic polynomial problem). 将所有  $P$  问题的集合记为  $P$ , 所有  $NP$  问题的集合记为  $NP$ , 是否  $P = NP$ ? 这是计算机科学和组合最优化领域的一个重要未解难题. 明显地  $P \subseteq NP$ , 但人们目前还不知道是否  $P \supseteq NP$ . Cook(1970) 发现  $NP$  问题中有一类问题,只要其中有一个问题是  $P$  问题,则所有  $NP$  问题全是  $P$  问题 (从而  $P = NP$ ). 这类问题称为  $NP$  完全问题, 简称为  $NPC$  问题 ( $NP$  complete problem). 遗憾的是,对目前已找到的成千上万个  $NPC$  问题,人们还未能证明其中任何一个为  $P$  问题. 图论中有许多问题属于  $NPC$  问题. 对  $NPC$  问题和计算复杂性有兴趣的读者,可参阅文献 [3]~[15].

有关图和网络上的优化问题及有关算法和应用,可参阅著作 [16]~[20].

上述 Dijkstra 算法可以用列表的方式进行计算,下面举例说明.

考虑例 1.2.1 中的赋权图. 首先将顶点  $v_1, v_2, v_3, v_4$  作为一行. 然后将起始点  $v_0$  放在表内第一行左首,将  $v_0$  到其余各点连边上的权 (即标号  $l(v_k)$ ) 放在各顶点下方相应位置上. 从第一行数字中选取最小的一个,用方括号标记,并在斜杠下方标上  $v_0$ , 当前找到  $v_1$  下方的数字最小. 接着,将刚找到的  $v_1$  放在表内第二行左首 (表示  $v_1$  进入集合  $S$ ), 对每个尚未进入  $S$  的顶点  $v_k$ , 将第一行方括号中的数字与边  $v_1v_k$  上的权相加,并与第一行中  $v_k$  列的数字比较,取较小的一个 (即新标号  $l(v_k) := \min\{l(v_k), l(v_1) + w(v_1v_k)\}$ ) 放在第二行中相应列的位置上. 从第二行数字中选取最小的一个,用方括号标记,当前找到  $v_3$  下方的数字最小. 由于该数字 3 是通过  $v_1$  得来的 (即  $\min\{l(v_3), l(v_1) + w(v_1v_3)\} = l(v_1) + w(v_1v_3) = 3$ ), 故在该数字斜杠下方标上  $v_1$ .



$S \backslash V$	$v_1$	$v_2$	$v_3$	$v_4$
$v_0$	[1]/ $v_0$	8	4	$\infty$
$v_1$		6	[3]/ $v_1$	$\infty$
$v_3$		6		[5]/ $v_3$
$v_4$		[6]/ $v_1$		

同样, 将刚找到的  $v_3$  放在表内第三行左首 (表示  $v_3$  进入集合  $S$ ), 对每个尚未进入  $S$  的顶点  $v_k$ , 将第二行方括号中的数字与边  $v_3v_k$  上的权相加, 并与第二行中  $v_k$  列的数字比较, 取较小的一个 (即新标号  $l(v_k) := \min\{l(v_k), l(v_3) + w(v_3v_k)\}$ ) 放在第三行中相应列的位置上. 并从第三行数字中选取最小的一个, 用方括号标记, 当前找到  $v_4$  列的数字最小. 由于该数字 5 是通过  $v_3$  得来的 (即  $\min\{l(v_4), l(v_3) + w(v_3v_4)\} = l(v_3) + w(v_3v_4) = 5$ ), 故在该数字斜杠下方标上  $v_3$ . 最后, 将  $v_4$  放在表内第四行左首 (表示  $v_4$  进入集合  $S$ ), 对尚未进入  $S$  的唯一顶点  $v_2$ , 将第三行方括号中的数字与边  $v_4v_2$  上的权相加, 并与第三行中  $v_2$  列的数字比较, 取较小的一个 (即新标号  $l(v_2) := \min\{l(v_2), l(v_4) + w(v_4v_2)\}$ ) 放在第四行  $v_2$  列的位置上. 并从第四行数字中选取最小的一个, 现在第四行只有  $v_2$  列的数字 6, 用方括号标记之,  $v_2$  进入集合  $S$ . 由于该数字 6 可看成是通过  $v_2$  的旧标号  $l(v_2)$  得来的 (即  $\min\{l(v_2), l(v_4) + w(v_4v_2)\} = l(v_2) = 6$ ), 旧标号  $l(v_2)$  又是在第二行通过  $v_1$  得来的, 故在该数字斜杠下方标上  $v_1$ . 至此, 算法结束. 从每个顶点  $v_k$  所在列的最后一个数字 (方括号内的数字) 可以得知  $v_0$  到  $v_k$  的最短路的长. 比如,  $v_0$  到  $v_4$  的最短路的长为 5.

由斜杠后的标识可反向追踪出最短路. 比如找  $v_0$  到  $v_4$  的最短路: 先从  $v_4$  列下方的标识找到  $v_3$ , 再从  $v_3$  列下方的标识找到  $v_1$ , 最后从  $v_1$  列下方的标识找到  $v_0$ , 从而可知  $v_0$  到  $v_4$  的最短路为  $v_0v_1v_3v_4$ .

Dijkstra 算法也可以通过矩阵形式来进行, 其步骤为:

第 0 步: 将赋权图  $G$  的权矩阵  $(w_{ij})$  的第一列中所有元素全改为  $\times$ , 在第一行剩下的其他元素下面划一条横线.

第 1 步: 在画横线但不带方括号的所有元素中找一个最小的  $w_{ki}$ , 若  $w_{ki} = \infty$ , 则停止, 从  $v_0$  到某些顶点没有路; 否则, 把  $w_{ki}$  用方括号标记, 并把第  $i$  列其他元素都改成  $\times$ , 然后给第  $i$  行中不是  $\times$  的元素都加上  $w_{ki}$ , 并在这些元素下面划一条横线, 转第 2 步.

第 2 步: 如果存在不含  $\times$  的列, 则转第 1 步; 否则结束, 方括号标记的元素  $w_{ij}$  表示从  $v_0$  到  $v_j$  的最短路的长, 而从  $v_0$  到  $v_j$  的最短路由最短  $(v_0, v_i)$  路与边  $v_iv_j$  构成.

仍以例 1.2.1 中的图为例, 其权矩阵为

$$(w_{ij}) = \begin{matrix} & v_0 & v_1 & v_2 & v_3 & v_4 \\ \begin{matrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{matrix} & \begin{pmatrix} \infty & 1 & 8 & 4 & \infty \\ 1 & \infty & 5 & 2 & \infty \\ 8 & 5 & \infty & 6 & 1 \\ 4 & 2 & 6 & \infty & 2 \\ \infty & \infty & 1 & 2 & \infty \end{pmatrix} \end{matrix},$$

$$\begin{pmatrix} \infty & 1 & 8 & 4 & \infty \\ 1 & \infty & 5 & 2 & \infty \\ 8 & 5 & \infty & 6 & 1 \\ 4 & 2 & 6 & \infty & 2 \\ \infty & \infty & 1 & 2 & \infty \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \underline{1} & \underline{8} & \underline{4} & \underline{\infty} \\ \times & \infty & 5 & 2 & \infty \\ \times & 5 & \infty & 6 & 1 \\ \times & 2 & 6 & \infty & 2 \\ \times & \infty & 1 & 2 & \infty \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \underline{1} & \underline{8} & \underline{4} & \underline{\infty} \\ \times & \times & \underline{6} & \underline{3} & \underline{\infty} \\ \times & \times & \infty & 6 & 1 \\ \times & \times & 6 & \infty & 2 \\ \times & \times & 1 & 2 & \infty \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} \times & \underline{1} & \underline{8} & \times & \underline{\infty} \\ \times & \times & \underline{6} & \underline{3} & \underline{\infty} \\ \times & \times & \infty & \times & 1 \\ \times & \times & \underline{9} & \times & \underline{5} \\ \times & \times & 1 & \times & \infty \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \underline{1} & \underline{8} & \times & \times \\ \times & \times & \underline{6} & \underline{3} & \times \\ \times & \times & \infty & \times & \times \\ \times & \times & \underline{9} & \times & \underline{5} \\ \times & \times & \underline{6} & \times & \times \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \underline{1} & \times & \times & \times \\ \times & \times & \underline{6} & \underline{3} & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \underline{5} \\ \times & \times & \times & \times & \times \end{pmatrix}.$$

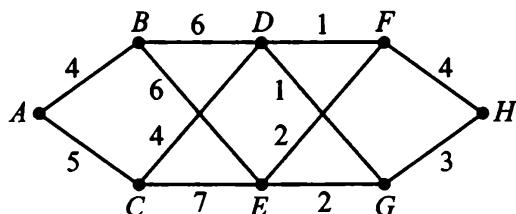
由最后一个矩阵的元素回溯便可获得  $v_0$  到各点的最短路. 比如,  $v_0$  到  $v_4$  的最短路的长是 5; 为了找出从  $v_0$  到  $v_4$  的最短路, 从  $v_4$  所在列 (最后一列) 开始回溯, 因该列非  $\times$  元素在  $v_3$  行 (第 4 行),  $v_3$  列非  $\times$  元素在  $v_1$  行 (第 2 行), 而  $v_1$  列非  $\times$  元素在  $v_0$  行 (第 1 行), 故  $v_0$  到  $v_4$  的最短路为  $v_0v_1v_3v_4$ .

上述最短路算法仅适用于求所有边的权均非负的图中的最短路. 在边允许有负权的图中求最短路, 有 Ford 算法和 Floyd-Warshall 算法. 若图中某个圈上各边的权值之和为负数, 则称该圈为负权圈. Ford 算法用于求无负权圈的图中一点到其他各点的最短路; Floyd-Warshall 算法用于求无负权圈的图中所有顶点之间的最短路. 如果图中存在负权圈, 这两个算法在执行过程中可以检测出负权圈. 它们的计算复杂度分别为  $O(n^2)$  和  $O(n^3)$ . 有兴趣的读者可参看文献 [20]~[25]. 求图的前  $k$  条最短路的算法可参考 [20]、[25].

如果一个图  $G$  的顶点集  $V$  能划分为若干子集  $V_1, V_2, \dots, V_n$ , (每个子集称为一个层), 使得  $G$  的边全在相邻的层之间, 而同层顶点间没有边, 隔层顶点间也没有边, 则称图  $G$  为一个分层图. 注意集合  $V$  划分为子集  $V_1, V_2, \dots, V_n$  的含义是  $V = V_1 \cup V_2 \cup \dots \cup V_n$  且  $V_1 \cap V_2 \cap \dots \cap V_n = \emptyset$ .

对分层图, 也可用动态规划方法来求最短路. 动态规划源于多阶段最优决策过程, 它的基本思想是 Bellman 最优化原理. 该原理可概括地描述为: 如果图  $G$  中从  $u$  点到

$v$  点的最短路  $P$  经过点  $w$ , 则  $P$  上  $w$  到  $v$  的一段必定是该图中  $w$  点到  $v$  点的最短路. 依据这个原理, 可用一个基本的递推关系式使最优决策过程连续地转移. 使用动态规划方法时, 要从最终状态开始逐步递推到起始状态. 例如, 对下图求  $A$  到  $H$  的最短路.



该图的顶点从  $A$  到  $H$  分为 5 层, 同层顶点间互不连边, 隔层顶点间也没有连边, 因此是分层图. 我们从  $H$  点开始考虑. 其前一层顶点  $F$  和  $G$  到  $H$  的最短路的长分别是 4 和 3, 记为  $W(F) = 4$ ,  $W(G) = 3$ . 再考虑  $F$  和  $G$  的前一层顶点  $D$  和  $E$ . 从  $D$  出发有两种选择:  $DFH$ , 路长为  $1 + W(F) = 5$ ;  $DGH$ , 路长为  $1 + W(G) = 4$ ; 故从  $D$  到  $H$  的最短路为  $DGH$ ,  $W(D) = 4$ . 同样可知, 从  $E$  到  $H$  的最短路为  $EGH$ ,  $W(E) = 5$ . 再考虑  $B, C$ .  $B$  到  $H$  可经过  $D$  也可经过  $E$ , 经  $D$  到  $H$  的最短路长为  $6 + W(D) = 10$ , 经  $E$  到  $H$  的最短路长为  $6 + W(E) = 11$ , 故从  $B$  到  $H$  的最短路应经过  $D$ , 且最短路长为  $W(B) = 10$ . 同样可得, 从  $C$  到  $H$  的最短路应经过  $D$ , 且最短路长为  $W(C) = 8$ . 最后考虑顶点  $A$ , 它到  $H$  可经过  $B$  也可经过  $C$ , 经  $B$  到  $H$  的最短路长为  $4 + W(B) = 14$ , 经  $C$  到  $H$  的最短路长为  $5 + W(C) = 13$ , 故从  $A$  到  $H$  的最短路应经过  $C$ , 且最短路长为  $W(A) = 13$ . 因此获得  $A$  到  $H$  的最短路为  $ACDGH$ .

在上述过程中, 决定每层顶点的结果只需要做 4 次加法、2 次比较. 如果将图扩展为  $n$  层, 每个中间层有不超过  $k$  个点, 则决定每层顶点的结果至多需要做  $k^2$  次加法、 $k(k-1)$  次比较. 故算法的计算量为  $[k^2(n-3) + k] + k(k-1)(n-2) = O(k^2n) = O(k\nu)$ , 这低于 Dijkstra 算法的计算量. 由此可见, 针对具体问题的特点可以设计出非常有效的算法.

动态规划是运筹学中的一个分支, 有兴趣的读者可参看 [26] 或其他有关著作. 关于最短路问题读者可进一步参阅文献 [27]~[45].

### §1.3 树及其性质

不含圈的图称为森林 (forest), 不含圈的连通图称为树 (tree).

**定理 1.3.1** 下列命题等价:

- (1)  $G$  是树;
- (2)  $G$  中无环边且任二顶点之间有且仅有一条路;
- (3)  $G$  中无圈且  $\varepsilon = \nu - 1$ ;

- (4)  $G$  连通且  $\varepsilon = \nu - 1$ ;  
 (5)  $G$  连通且对任何  $e \in E(G)$ ,  $G - e$  不连通;  
 (6)  $G$  无圈且对任何  $e \in E(\overline{G})$ ,  $G + e$  恰有一个圈.

**证明** (1)  $\Rightarrow$  (2)

因  $G$  是树, 故  $G$  无圈且连通, 因此  $G$  中无环边, 且对  $\forall u, v \in V(G)$ , 存在路  $P(u, v)$ .

若还存在一条路  $P'(u, v) \neq P(u, v)$ , 则必存在  $x$  和  $y$ , 它们分别是路  $P$  与  $P'$  从  $u$  出发的最后一个分叉点和最后一个汇合点. 则  $P$  的  $(x, y)$  段与  $P'$  的  $(x, y)$  段构成圈, 这与  $G$  是树矛盾. 故只存在唯一的  $(u, v)$  路.

(2)  $\Rightarrow$  (3)

若  $G$  有圈, 则此圈上任二顶点间有两条不同的路, 与前提条件矛盾.

下面用归纳法证明  $\varepsilon = \nu - 1$ .

$\nu = 1$  时,  $\varepsilon = 0$ , 结论真.

假设  $\nu \leq k$  时结论真, 我们来证明当  $\nu = k + 1$  时, 也有  $\varepsilon = \nu - 1$  成立.

当  $\nu = k + 1$  时, 任取边  $uv \in E(G)$ . 考虑图  $G' = G - uv$ , 因  $G$  中  $u, v$  间只有一条路, 即边  $uv$ , 故  $G'$  不连通且只有两个连通分支, 设为  $G_1, G_2$ . 注意到  $G_1, G_2$  分别都连通且任二顶点间只有一条路, 由归纳法假设,  $\varepsilon(G_1) = \nu(G_1) - 1$ ,  $\varepsilon(G_2) = \nu(G_2) - 1$ . 因此

$$\varepsilon(G) = \varepsilon(G_1) + \varepsilon(G_2) + 1 = (\nu(G_1) - 1) + (\nu(G_2) - 1) + 1 = \nu(G) - 1.$$

归纳法完成.

(3)  $\Rightarrow$  (4)

用反证法. 若  $G$  不连通, 设  $G_1, G_2, \dots, G_w$  是其连通分支 ( $w \geq 2$ ), 则  $\varepsilon_i = \nu_i - 1$  (因  $G_i$  是连通无圈图, 由已证明的 (1) 和 (2) 知, 对每个  $G_i$ , (3) 成立). 这样,  $\varepsilon = \sum_{i=1}^w \varepsilon_i = \sum_{i=1}^w \nu_i - w = \nu - w$ , 这与  $\varepsilon = \nu - 1$  矛盾.

(4)  $\Rightarrow$  (5)

因  $\varepsilon(G - e) = \varepsilon(G) - 1 = \nu - 2$ , 而由定理 1.1.3, 每个连通图必满足  $\varepsilon \geq \nu - 1$ , 故图  $G - e$  不连通.

(5)  $\Rightarrow$  (6)

先证  $G$  中无圈: 若  $G$  中有圈, 删去圈上任一边仍连通, 这与条件 (5) 矛盾.

再证对任何  $e \in E(\overline{G})$ ,  $G + e$  恰含一个圈: 因  $G$  连通且已证  $G$  无圈, 故  $G$  是树. 由 (2), 任两不相邻顶点间都有一条路相连, 故  $G + e$  中必有一个含有  $e$  的圈; 另一方面, 因  $G$  无圈,  $G + e$  中的任何圈必含有  $e$ , 但若  $G + e$  中有两个圈含有  $e$ , 则  $(G + e) - e = G$  中仍含有一个圈, 这与  $G$  无圈矛盾. 因此  $G + e$  恰含一个圈.

(6)  $\Rightarrow$  (1)

只需证  $G$  连通. 任取  $u, v \in V(G)$ , 若  $u, v$  相邻, 则  $u$  与  $v$  当然连通. 若  $u, v$  不相邻, 则  $G + uv$  恰含一个圈, 故  $u$  与  $v$  在  $G$  中连通. 由  $u, v$  的任意性, 图  $G$  连通. 证毕.

**推论 1.3.1** 非平凡树至少含两个 1 度顶点 (叶子).

**证明** 设  $T$  是一个非平凡树, 则其顶点数  $\nu$  和边数  $\varepsilon$  满足  $\varepsilon = \nu - 1$ . 因  $T$  连通, 故对每个顶点  $v_i$ , 都有  $d(v_i) \geq 1$ . 若对所有  $v_i$  都有  $d(v_i) \geq 2$ , 则  $\sum_{i=1}^{\nu} d(v_i) \geq 2\nu$ . 但另一方面,  $\sum_{i=1}^{\nu} d(v_i) = 2\varepsilon = 2(\nu - 1) = 2\nu - 2$ . 这两方面矛盾. 故  $T$  至少有一个 1 度顶点, 设为  $u$ . 除此之外, 其余  $\nu - 1$  个顶点的度数之和为  $2\nu - 3$ . 若这些点的度都大于或等于 2, 则其度数之和  $\geq 2(\nu - 1) = 2\nu - 2$ , 这与  $2\nu - 3$  矛盾. 故除  $u$  之外  $T$  还至少有一个度为 1 的顶点. 证毕.

## §1.4 生成树与最小生成树

### 一、生成树 (spanning tree)

**定义 1.4.1** 设  $T$  是图  $G$  的一个子图, 如果  $T$  是一棵树, 且  $\nu(T) = \nu(G)$ , 则称  $T$  是  $G$  的一个生成树.

**定理 1.4.1** 每个连通图都有生成树.

**证明** 设  $G$  是一个连通图. 令  $A = \{G' | G' \text{ 是 } G \text{ 的连通子图且 } \nu(G') = \nu(G)\}$ . 易见  $A$  非空. 从  $A$  中取边数最少的一个, 记为  $T$ . 下证  $T$  是  $G$  的生成树. 显然只需证明  $T$  是树即可.

事实上, 已知  $T$  连通, 下证  $T$  无圈.

若  $T$  有圈  $C$ , 则去掉  $C$  上任一条边  $e$ ,  $T - e$  仍连通, 且显然  $T - e$  是  $G$  的子图并有  $\nu(T - e) = \nu(G)$ . 从而  $T - e \in A$ . 但  $T - e$  比  $T$  少一条边, 这与  $T$  的取法矛盾. 证毕.

利用此定理可得到定理 1.1.3 的一个简便证明: 取  $G$  的生成树  $T$ , 则

$$\varepsilon(G) \geq \varepsilon(T) = \nu(T) - 1 = \nu(G) - 1.$$

### 二、最小生成树问题

**最小生成树问题:** 在赋权图  $G$  中, 求权最小的生成树 (简称为最小生成树). 即: 求

$G$  的一棵生成树  $T$ , 使得

$$W(T) = \min_T \sum_{e \in T} w(e).$$

最小生成树问题是一个优化问题, 需要设计算法寻找其最优解. 求解最小生成树问题的算法较多, 本节主要介绍 Kruskal 算法和 Prim 算法.

### (一) Kruskal 算法 (Joseph Bernard Kruskal, 1956)

#### 1. 算法思想

先从图  $G$  中找出权最小的一条边作为最小生成树的边, 然后逐次从剩余边中选边添入最小生成树中, 选边的原则为: 每次挑选不与已选边构成圈的边中权最小的一条. 直至选出  $\nu(G) - 1$  条边为止.

#### 2. 算法步骤

输入: 赋权连通  $\nu$  阶图  $G$ .

输出:  $G$  的最小生成树  $T$ .

第 1 步: 取  $e_1 \in E(G)$  使得  $w(e) = \min_{e \in G} \{w(e)\}$ , 令  $i := 1$ .

第 2 步: 取  $e_{i+1} \in E(G) \setminus \{e_1, e_2, \dots, e_i\}$  使得

(1)  $G[\{e_1, e_2, \dots, e_i, e_{i+1}\}]$  不含圈; (2)  $e_{i+1}$  是满足 (1) 的权最小的边.

第 3 步: 当  $i + 1 = \nu(G) - 1$  时, 输出最小生成树  $G[\{e_1, e_2, \dots, e_{\nu-1}\}]$ , 算法停止; 否则, 令  $i := i + 1$ , 转第 2 步.

#### 3. 算法的正确性

**定理 1.4.2** 设  $e_1, e_2, \dots, e_{\nu(G)-1}$  是 Kruskal 算法获得的边, 则边导出子图  $G[\{e_1, e_2, \dots, e_{\nu(G)-1}\}]$  是  $G$  的最小生成树.

**证明** 记  $T^* = G[\{e_1, e_2, \dots, e_{\nu(G)-1}\}]$ . 按照算法显然  $T^*$  无圈, 因此  $T^*$  是森林. 设它有  $w$  个连通分支, 则  $\nu(T^*) = \varepsilon(T^*) + w \geq \varepsilon(T^*) + 1 = (\nu(G) - 1) + 1 = \nu(G)$ . 但  $T^*$  是  $G$  的子图, 故  $\nu(T^*) = \nu(G)$ . 于是

$$\varepsilon(T^*) = \nu(G) - 1 = \nu(T^*) - 1.$$

由定理 1.3.1 的 (3),  $T^*$  是一棵树. 又  $\nu(T^*) = \nu(G)$ , 从而是  $G$  的一棵生成树.

下证其最优性, 用反证法.

假设  $T^*$  不是权最小的生成树 (下称最优树). 对  $G$  的任一棵不同于  $T^*$  的生成树  $T$ , 记

$$f(T) = \min\{i | e_i \in \{e_1, e_2, \dots, e_{\nu-1}\} \text{ 且 } e_i \notin T\}.$$

在  $G$  的所有最优树中选取一棵使  $f(T)$  最大的, 记为  $\tilde{T}$  ( $\tilde{T}$  不会是  $T^*$ , 因假设  $T^*$  不是最优树). 设  $f(\tilde{T}) = k$ . 由  $f(\tilde{T})$  的定义,  $e_1, e_2, \dots, e_{k-1}$  既在  $T^*$  上也在  $\tilde{T}$  上, 但  $e_k$  不在  $\tilde{T}$  上. 因此  $\tilde{T} + e_k$  含有一个圈  $C$ .  $C$  上必有一条边  $e'_k \notin T^*$ . 显然  $T' = (\tilde{T} + e_k) - e'_k$  也是一棵生成树, 且  $W(T') = W(\tilde{T}) + w(e_k) - w(e'_k)$ .

按照算法,  $e_k$  是使  $G[\{e_1, e_2, \dots, e_k\}]$  中无圈的边中权最小的. 注意  $G[\{e_1, e_2, \dots, e_{k-1}, e'_k\}]$  是  $\tilde{T}$  的子图, 也无圈. 故由算法规则知:  $w(e'_k) \geq w(e_k)$ . 代入前式得,  $W(T') \leq W(\tilde{T})$ , 这说明  $T'$  也是最优树. 但  $f(T') > k = f(\tilde{T})$  (注意由于  $e_1, e_2, \dots, e_{k-1}$  在  $T^*$  上但  $e'_k \notin T^*$ , 故  $e'_k \notin \{e_1, e_2, \dots, e_{k-1}\}$ ). 这与  $\tilde{T}$  的取法矛盾. 证毕.

#### 4. Kruskal 算法的实现及其计算复杂度分析

Kruskal 算法的计算量主要在第 2 步. 算法共需执行  $\nu - 1$  次第 2 步, 在第  $i$  次执行第 2 步时, 须比较集合  $E(G) \setminus \{e_1, e_2, \dots, e_i\}$  中所有边的权以求得一条权最小的边, 并检验该边是否与已有边构成圈, 如果构成圈, 还需再找新的最小权边, 这是比较浪费的.

在实际应用 Kruskal 算法时, 一般先将所有的边按权由小到大排序, 这需要大约  $\varepsilon \log_2 \varepsilon$  次比较 (见 [46]). 每次执行算法第 2 步时, 不必再比较边的权, 而是直接选取此前尚未考虑过的权最小的边, 检验它是否与已有边构成圈即可. 这样可省去许多次重复比较.

接下来的问题是如何检验所选边是否与已有边构成圈. 这可通过给顶点标号来实现. 算法开始时, 给所有顶点标不同的标号: 顶点  $v_k$  标号为  $k$  ( $k = 1, 2, \dots, \nu$ ). 设算法在  $i$  次循环后选出的边集合为  $E_i$ . 当算法执行第  $i + 1$  次循环选出一边  $e_{i+1}$ , 并将其添加进  $G[E_i]$  形成  $G[E_{i+1}]$  后, 用该边两 endpoint 标号的较小者, 给这条边所连的  $G[E_i]$  的两个连通分支的顶点重新标号 (在  $G[E_{i+1}]$  中成为同一个连通分支). 按这个标号方案, 在任意一步中, 两个顶点属于已选边形成的同一连通分支当且仅当它们有相同的标号. 这意味着当我们考虑向  $G[E_i]$  添加某条边  $e$  是否会形成圈时, 只要检查  $e$  的两个端点是否有相同的标号即可. 如果有相同的标号, 则抛弃该边 (以后的循环中不再使用), 再检验权稍大些的下一个候选边; 如果标号相异, 则取  $e$  作为  $e_{i+1}$  添加进  $G[E_i]$ . 在这个过程中, 对每条候选边只需做一次比较就能决定是否抛弃它. 算法全过程至多需要  $\varepsilon \leq \frac{\nu(\nu-1)}{2}$  次这样的比较.

当添加  $e_{i+1} = uv$  进入  $G[E_i]$  得到  $G[E_{i+1}]$  后, 需要用  $u$  和  $v$  的较小标号给  $e_{i+1}$  所连的两个连通分支重新标号. 不妨设标号  $l(u) \leq l(v)$ , 则实际上只需给  $v$  所在的连通分支中各点重新标号即可. 但该连通分支中各点的标号都等于  $l(v)$ , 而其他分支中点的标号都不会等于  $l(v)$ , 因此只需将标号等于  $l(v)$  的各点的标号改为  $l(u)$ , 而不必判断哪些点属于该连通分支. 可见, 每轮循环中顶点重新标号需要不超过  $\nu$  次判断和  $\nu$  次赋值, 因而算法的  $\nu - 1$  轮循环最多需要  $2\nu(\nu - 1)$  次这样的判断和赋值.

可见算法执行过程约需要  $\varepsilon + 2\nu(\nu - 1)$  次基本运算.

由以上分析可得如下结论.

**定理 1.4.3** 若事先将顶点按权排序, 则 Kruskal 算法的计算复杂度为  $O(\nu^2)$ ; 若加上边集合事先排序的计算量, 则 Kruskal 算法的计算复杂度为  $O(\nu^2 \log_2 \nu)$ .

**证明** 算法执行过程中需要的主要计算量为

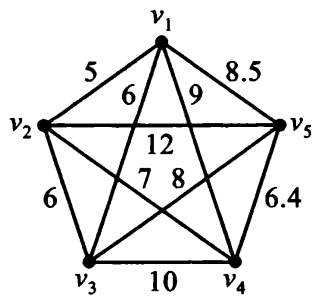
$$\varepsilon + 2\nu(\nu - 1) \leq \frac{\nu(\nu - 1)}{2} + 2\nu(\nu - 1) = \frac{5}{2}\nu(\nu - 1),$$

边集合事先排序需要的计算量为

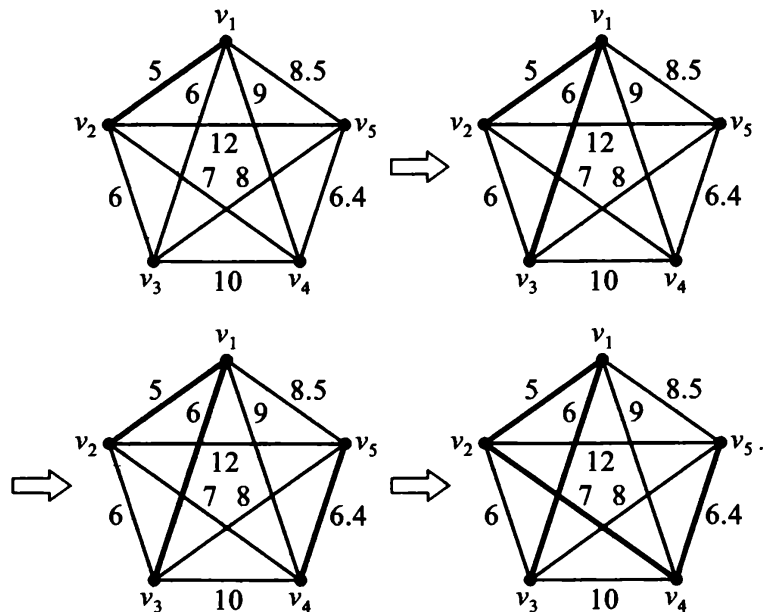
$$\varepsilon \log_2 \varepsilon \leq \frac{\nu(\nu - 1)}{2} \log_2 \frac{\nu(\nu - 1)}{2} \leq \nu^2 \log_2 \nu.$$

故知定理结论成立. 证毕.

**例 1.4.1** 欲建设一个连接 5 座城市的光纤通信网络. 各城市间线路的造价如图所示, 求一个使总造价最少的线路建设方案.



**解** 问题归结为求所给赋权图的最小权生成树. 使用 Kruskal 算法, 算法执行过程如下所示.





其中粗线所示为逐步形成的最小权生成树.

## (二) Prim 算法 (Robert Clay Prim, 1957)

### 1. 算法思想

先从图  $G$  中找出权最小的一条边作为最小生成树的边, 在算法任一循环中, 设已经选出的边导出的子图为  $G'$ , 从  $G'$  的顶点向  $G'$  以外顶点的连边为  $E'$ , 则选择  $E'$  中权最小的边向  $G'$  中添加, 如此反复循环直至选出  $\nu(G) - 1$  条边为止.

Prim 算法与 Kruskal 算法的根本区别在于: Kruskal 算法在保持无圈的基础上选边, 而 Prim 算法在保持连通的基础上选边. Prim 算法的添边过程实际上是树的生长过程. Kruskal 算法的添边过程一般情况下是森林合并为树的过程.

在一些文献中, Kruskal 算法也称为避圈法, Prim 算法也称为边割法.

### 2. 算法步骤

输入: 赋权连通  $\nu$  阶图  $G$ .

输出:  $G$  的最小生成树  $T$ .

第 1 步: 任取  $v_0 \in V(G)$ , 令  $S_0 = \{v_0\}$ ,  $\bar{S}_0 = V(G) \setminus S_0$ ,  $i := 0$ ,  $E_0 = \emptyset$ .

第 2 步: 求  $S_i$  到  $\bar{S}_i$  间权最小的边  $e_{i+1}$ , 设  $e_{i+1}$  的属于  $\bar{S}_i$  的端点为  $v_{i+1}$ , 令  $S_{i+1} := S_i \cup \{v_{i+1}\}$ ,  $\bar{S}_{i+1} = V(G) \setminus S_{i+1}$ ,  $E_{i+1} := E_i \cup \{e_{i+1}\}$ .

第 3 步: 当  $i + 1 = \nu(G) - 1$  时, 输出最小生成树  $G[E_{i+1}] = G[\{e_1, e_2, \dots, e_{\nu-1}\}]$ , 算法停止; 否则, 令  $i := i + 1$ , 转第 2 步.

### 3. 算法的正确性

**定理 1.4.4** 设  $e_1, e_2, \dots, e_{\nu-1}$  是 Prim 算法获得的边, 则边导出子图  $G[\{e_1, e_2, \dots, e_{\nu-1}\}]$  是  $G$  的最小生成树.

**证明** 用  $T_k$  表示 Prim 算法第  $k$  次循环得到的边集所构成的子图, 即

$$T_k = G[E_k] = G[\{e_1, e_2, \dots, e_k\}], k = 1, 2, \dots, \nu - 1.$$

我们用归纳法来证明, 对每个  $k(k = 1, 2, \dots, \nu - 1)$ ,  $T_k$  必含于  $G$  的某个最小生成树中.

当  $k=1$  时,  $T_1$  仅由算法中从初始点  $v_0$  发出的边  $e_1$  生成. 对  $G$  的任一棵最小生成树  $T^*$ , 若  $T^*$  不含有  $e_1$ , 则  $T^* + e_1$  有一个圈  $C$ : 设  $C$  上与  $v_0$  关联且异于  $e_1$  的边为  $e$ , 则  $T' = T^* + e_1 - e$  也是  $G$  的一棵生成树. 按照算法,  $e_1$  是  $G$  中与  $v_0$  关联的边中权最小的, 因此, 权  $w(e) \geq w(e_1)$ , 从而  $w(T') \leq w(T^*)$ . 可见  $T'$  也是  $G$  的一棵最小生成树, 并且含有边  $e_1$ . 即  $T_1$  含于最小生成树  $T'$  中.

假设  $k = m$  时 ( $1 \leq m < \nu - 1$ ) 结论成立, 即  $T_m$  含于  $G$  的某个最小生成树  $T^*$  中. 考虑最后一条进入  $T_{m+1} = G[\{e_1, e_2, \dots, e_m, e_{m+1}\}]$  的边  $e_{m+1}$ . 设  $e_{m+1} = v_m v_{m+1}$ , 则按照算法, 必定其一个端点  $v_m$  在  $T_m$  中而另一个端点  $v_{m+1}$  不在  $T_m$  中.

若  $e_{m+1}$  在  $T^*$  中, 则  $T_{m+1} = T_m + e_{m+1}$  含于最小生成树  $T^*$  中.

若  $e_{m+1}$  不在  $T^*$  中, 则  $T^* + e_{m+1}$  有一个圈  $C$ ,  $e_{m+1}$  在  $C$  上.  $C$  上以  $v_{m+1}$  为端点且异于  $e_{m+1}$  的边不属于  $T_m$  (因  $v_{m+1}$  不在  $T_m$  中). 因此  $C$  上必可找到一条异于  $e_{m+1}$  的边  $e = uv$ , 它不属于  $T_m$  且满足  $u \in T_m$  而  $v \notin T_m$  (因  $e_{m+1}$  的端点  $v_m$  在  $C$  上且  $v_m \in T_m$ ). 由算法步骤知, 权  $w(e) \geq w(e_{m+1})$ . 于是  $T' = T^* + e_{m+1} - e$  也是  $G$  的一棵最小生成树, 并且含有边  $e_{m+1}$ . 根据归纳假设,  $T_m$  的边全在  $T^*$  中, 因此也全在  $T'$  中, 从而  $T_{m+1} = T_m + e_{m+1}$  的边全在最小生成树  $T'$  中. 归纳法完成. 证毕.

#### 4. 算法的计算复杂度分析

Prim 算法的主要计算量在第 2 步. 在算法第  $i$  轮循环执行第 2 步时,  $S_i$  中有  $i$  个顶点,  $\bar{S}_i$  中有  $\nu - i$  个顶点, 故  $S_i$  到  $\bar{S}_i$  间最多有  $i(\nu - i)$  条边. 从这些边中选出一条权最小的边  $e_{i+1}$ , 需要  $i(\nu - i) - 1$  次比较. 算法需循环执行第 2 步  $\nu - 1$  次, 因此总的计算量为

$$\sum_{i=1}^{\nu-1} [i(\nu - i) - 1] \leq \sum_{i=1}^{\nu-1} i(\nu - i) = \frac{1}{6} \nu(\nu - 1)(\nu + 1).$$

由此可见, Prim 算法的计算复杂度为  $O(\nu^3)$ .

#### (三) Prim 算法的矩阵实现 —— 求最小生成树的权矩阵法

Prim 算法可以通过权矩阵来实现. 给定一个赋权的连通  $\nu$  阶简单图  $G$ , 其各边上的权组成的  $\nu$  阶矩阵  $W = (w_{ij})_{\nu \times \nu}$  称为  $G$  的权矩阵, 其中元素  $w_{ij}$  定义如下:

- (1)  $w_{ii} = \infty, i = 1, 2, \dots, \nu;$
- (2) 若边  $v_i v_j \in E(G)$ , 则  $w_{ij} = w(v_i v_j);$
- (3) 若边  $v_i v_j \notin E(G)$ , 则  $w_{ij} = \infty.$

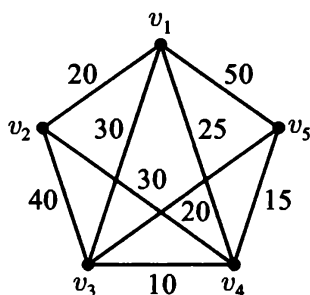
#### 算法步骤

第 0 步: 划掉矩阵  $W$  的第一列 (将  $W$  第一列元素全改为  $\times$ ), 并在第一行剩下的每个元素下面画一条横线.

第 1 步: 在画横线的元素中找一个最小的元素  $w_{ki}$ , 并将其圈起来. 然后把第  $i$  列其他元素全部划去 (改为  $\times$ ), 并在第  $i$  行没有被划掉的元素下面画上横线.

第 2 步: 若矩阵  $W$  的所有元素要么已被划去要么已被圈起来, 则算法结束, 圈起来的元素即对应于所求最小生成树的边. 否则, 转第 1 步.

**例 1.4.2** 用权矩阵法求如下赋权图  $G$  的一棵最小生成树.



解 求解过程如下.

$$\begin{aligned}
 W &= \begin{pmatrix} \infty & 20 & 30 & 25 & 50 \\ 20 & \infty & 40 & 30 & \infty \\ 30 & 40 & \infty & 10 & 20 \\ 25 & 30 & 10 & \infty & 15 \\ 50 & \infty & 20 & 15 & \infty \end{pmatrix} \rightarrow \begin{pmatrix} \times & \underline{20} & \underline{30} & \underline{25} & \underline{50} \\ \times & \infty & 40 & 30 & \infty \\ \times & 40 & \infty & 10 & 20 \\ \times & 30 & 10 & \infty & 15 \\ \times & \infty & 20 & 15 & \infty \end{pmatrix} \\
 &\rightarrow \begin{pmatrix} \times & (20) & \underline{30} & \underline{25} & \underline{50} \\ \times & \times & \underline{40} & \underline{30} & \underline{\infty} \\ \times & \times & \infty & 10 & 20 \\ \times & \times & 10 & \infty & 15 \\ \times & \times & 20 & 15 & \underline{\infty} \end{pmatrix} \rightarrow \begin{pmatrix} \times & (20) & \underline{30} & (25) & \underline{50} \\ \times & \times & \underline{40} & \times & \underline{\infty} \\ \times & \times & \infty & \times & 20 \\ \times & \times & \underline{10} & \times & \underline{15} \\ \times & \times & 20 & \times & \infty \end{pmatrix} \\
 &\rightarrow \begin{pmatrix} \times & (20) & \times & (25) & \underline{50} \\ \times & \times & \times & \times & \underline{\infty} \\ \times & \times & \times & \times & \underline{20} \\ \times & \times & (10) & \times & \underline{15} \\ \times & \times & \times & \times & \infty \end{pmatrix} \rightarrow \begin{pmatrix} \times & (20) & \times & (25) & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & (10) & \times & (15) \\ \times & \times & \times & \times & \times \end{pmatrix}.
 \end{aligned}$$

最后得到的最小生成树由边  $v_1v_2$ 、 $v_1v_4$ 、 $v_4v_3$ 、 $v_4v_5$  构成, 其权为  $20+25+10+15=70$ .

#### (四) Prim 算法的标号形式 —— 标号 Prim 算法

与 Kruskal 算法类似, 使用顶点标号法可进一步降低 Prim 算法的计算复杂度.

##### 1. 算法思想

给图的顶点赋以标号, 该标号与边的权有关, 在执行 Prim 算法的过程中, 通过修改顶点标号和比较顶点的标号大小, 来选择满足 Prim 算法要求的最小权边.

顶点  $v$  的标号记为  $l(v)$ . 用  $w(uv)$  表示边  $e=uv$  的权. 若  $u, v$  两点间没有边, 则  $w(uv)=\infty$ .

## 2. 算法步骤

输入: 赋权连通  $\nu$  阶图  $G$ .

输出:  $G$  的最小生成树  $T$ .

第 1 步: 任取  $v_0 \in V(G)$ , 令  $l(v_0) = 0$ ,  $l(v) = \infty$  ( $v \neq v_0$ ),  $S_0 = \{v_0\}$ ,  $\bar{S}_0 = V(G) \setminus S_0$ ,  $T_0 = v_0$ ,  $i := 0$ .

第 2 步: 对  $\forall v \in N_G(v_i) \cap \bar{S}_i$ , 若  $w(v_i v) < l(v)$ , 则令  $l(v) := w(v_i v)$ .

第 3 步: 选取  $v_{i+1} \in \bar{S}_i$  使得  $l(v_{i+1}) = \min_{v \in \bar{S}_i} l(v)$ . 设  $e_{i+1} = uv_{i+1}$ , ( $u \in S_i$ ), 使得  $w(e_{i+1}) = l(v_{i+1})$ , 令  $S_{i+1} := S_i \cup \{v_{i+1}\}$ ,  $\bar{S}_{i+1} = V(G) \setminus S_{i+1}$ ,  $T_{i+1} := T_i \cup \{e_{i+1}\}$ .

第 4 步: 当  $i+1 = \nu(G) - 1$  时, 输出最小生成树  $T_{\nu-1}$ , 算法停止; 否则, 令  $i := i+1$ , 转第 2 步.

## 3. 算法的正确性

标号 Prim 算法是 Prim 算法的标号实现, 因此由 Prim 算法的正确性即知标号 Prim 算法结束时必能得到图  $G$  的最小生成树.

## 4. 算法的计算复杂度分析

算法的主要计算量在第 2 步和第 3 步. 算法第一次执行第 3 步至多需做  $\nu - 2$  次比较, 第二次执行第 3 步至多需做  $\nu - 3$  次比较,  $\dots$ , 因此执行第 3 步至多共需做

$$(\nu - 2) + (\nu - 1) + \dots + 2 + 1 = \frac{1}{2}(\nu - 1)(\nu - 2)$$

次比较; 同样, 执行第 2 步共需不超过  $\frac{1}{2}(\nu - 1)(\nu - 2)$  次比较. 于是标号 Prim 算法的时间复杂度为  $O(\nu^2)$ .

标号 Prim 算法将 Prim 算法每次循环中比较  $S_{i+1}$  到  $\bar{S}_{i+1}$  间所有边的权, 改为修改  $N_G(v_i) \cap \bar{S}_i$  中顶点的标号并比较  $\bar{S}_i$  中顶点的标号, 因此节省了计算量.

一些文献中将标号 Prim 算法称为 Dijkstra 最小生成树算法.

在实际应用中, 往往需要求图的满足某种约束条件的最小生成树. 这一类问题统称为约束最小生成树问题 (constrained minimum spanning tree problems). 常见的有顶点度约束最小生成树、直径约束最小生成树、边容量约束最小生成树、最多叶子最小生成树等, 详细可参阅文献 [47]~[67]. 约束最小生成树问题大多数是 NPC 问题, 因此对这些问题主要是设计好的近似算法.

有时需要考虑欧氏距离下图的最小生成树问题, 这方面可参阅文献 [68]~[71]. 有关随机生成树问题也已得到研究, 有兴趣的读者可参阅文献 [72]~[76]. 关于最小生成树的更多文献请参看文献 [77]~[108].

## §1.5 图的中心与中位点

### 一、基本概念

设  $G$  是一个赋权图, 每条边  $v_i v_j$  上的权为  $w_{ij} \geq 0$ , 每个顶点  $v_i$  也有一个权  $q(v_i) \geq 0$ .  $d(u, v)$  表示在考虑边赋权下图  $G$  中顶点  $u$  到  $v$  的距离.

图  $G$  的半径 (radius):  $\text{rad } G = \min_{u \in V(G)} \max_{v \in V(G)} d(u, v)$ .

图  $G$  的直径 (diameter):  $\text{diam } G = \max_{u \in V(G)} \max_{v \in V(G)} d(u, v) = \max\{d(u, v) \mid \forall u, v \in V(G)\}$ .

图中顶点  $u$  的离心率 (eccentricity):  $e(u) = \max_{v \in V(G)} d(u, v)$ .

图  $G$  的中心 (center): 图  $G$  中具有最小离心率的顶点.

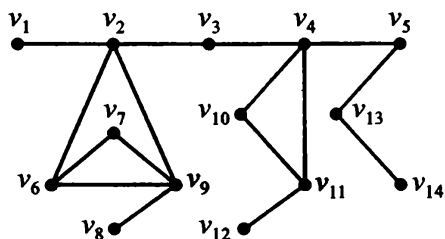
图  $G$  的中位点 (median): 设  $g(u) = \sum_{v \in V(G)} q(v)d(u, v)$ , 使  $g(u)$  达到最小的顶点称为图  $G$  的中位点. 图的中位点有时也称为中值点或重心 (barycenter).

**注** (1) 一个图的中心一般不唯一, 中位点一般也不唯一; 中心和中位点未必相同.

(2) 对于顶点无赋权的图, 可将每个顶点的权看作 1, 此时, 离心率  $e(u) = \max_{v \in V(G)} d(u, v)$ , 半径  $\text{rad } G = \min_{u \in V(G)} e(u)$ , 直径  $\text{diam } G = \max_{u \in V(G)} e(u)$ , 即半径是各点离心率的最小值, 直径是各点离心率的最大值. 而且, 中心处的离心率就是半径. 进一步地, 如果图的边也无赋权, 则将每条边上的权视为 1, 便得到半径、直径、离心率、中心、中位点的相应定义. 此时, 距离  $d(u, v)$  实际上是在边非赋权情况下顶点  $u$  到  $v$  的距离.

(3) 对于一般图  $G$ , 未必有  $\text{rad } G = \frac{\text{diam } G}{2}$ . 但对于树, 有  $\text{rad } G = \left\lceil \frac{\text{diam } G}{2} \right\rceil$  成立 (习题 1.62).

**例 1.5.1** 求如下非赋权图  $G$  的半径、直径、所有的中心、中位点以及点  $v_{10}$  的离心率.



**解** 按定义可求得,  $v_3$  和  $v_4$  都是中心, 半径  $\text{rad } (G) = 4$ , 直径  $\text{diam } (G) = 7$ , 点  $v_{10}$  的离心率  $e(v_{10}) = 5$ .  $v_3$  是唯一的中位点, 因为  $g(v_3) = 30$  是所有  $g(v)$  中最小的.

### 二、求图的中心和中位点的算法

根据中心和中位点的定义不难得出如下算法.

### 1. 求中心的算法

输入: 赋权图  $G$ , 各顶点的权  $q(v), v \in V(G)$ .

输出:  $G$  的中心.

第 1 步: 利用最短路算法求出  $G$  中所有顶点间的距离.

第 2 步: 对  $G$  的每个顶点  $u$ , 计算离心率  $e(u) = \max_{v \in V(G)} q(v)d(u, v)$ .

第 3 步: 比较各顶点的离心率, 求出离心率最小的顶点.

第 4 步: 输出离心率最小的顶点, 结束.

该算法可以通过距离矩阵实现.

#### 算法的距离矩阵形式

第 1 步: 利用最短路算法求出  $G$  中所有顶点间的距离, 并组成距离矩阵  $D = (d_{ij})$ , 其元素  $d_{ij}$  是图  $G$  中顶点  $v_i$  与顶点  $v_j$  间的距离,  $i, j = 1, 2, \dots, \nu$ .

第 2 步: 对  $j = 1, 2, \dots, \nu$ , 给矩阵  $D$  的第  $j$  列乘以  $q(v_j)$ ; 并求出所得矩阵每一行的最大元素:  $e(v_i), i = 1, 2, \dots, \nu$ .

第 3 步: 比较所有  $e(v_i), (i = 1, 2, \dots, \nu)$ , 其中最小者所在的行对应的顶点即为  $G$  的中心.

第 4 步: 输出中心, 结束.

上述算法的正确性是显然的. 如果第 1 步中求距离使用 Dijkstra 最短路算法或 Floyd-Warshall 最短路算法, 则上述算法的时间复杂度为  $O(\nu^3)$ .

### 2. 求中位点的算法

输入: 赋权图  $G$ , 各顶点的权  $q(v), v \in V(G)$ .

输出:  $G$  的中位点.

第 1 步: 利用最短路算法求出  $G$  中所有顶点间的距离.

第 2 步: 对  $G$  的每个顶点  $u$ , 计算  $g(u) = \sum_{v \in V(G)} q(v)d(u, v)$ .

第 3 步: 比较各顶点的  $g(u)$  值, 求出使  $g(u)$  达到最小的顶点.

第 4 步: 输出时  $g(u)$  达到最小的顶点, 结束.

#### 算法的距离矩阵形式

第 1 步: 利用最短路算法求出  $G$  中所有顶点间的距离, 并组成距离矩阵  $D = (d_{ij})$ , 其元素  $d_{ij}$  是图  $G$  中顶点  $v_i$  与顶点  $v_j$  间的距离,  $i, j = 1, 2, \dots, \nu$ .

第 2 步: 对  $j = 1, 2, \dots, \nu$ , 给矩阵  $D$  的第  $j$  列乘以  $q(v_j)$ , 并求出所得矩阵各行的行和:

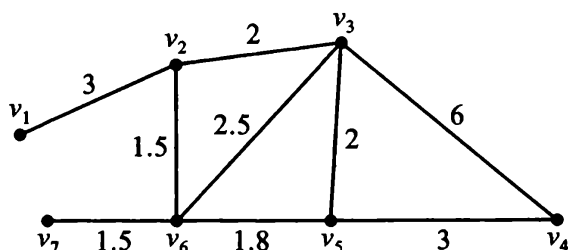
$$g(v_i) = \sum_{j=1}^{\nu} q(v_j)d_{ij}, i = 1, 2, \dots, \nu.$$

第 3 步: 比较所有行和  $g(v_i)(i = 1, 2, \dots, \nu)$ , 其中最小者所在的行对应的顶点即为  $G$  的中位点.

第 4 步: 输出中位点, 结束.

该算法的正确性也是显然的. 如果第 1 步中求距离使用 Dijkstra 最短路算法或 Floyd-Warshall 最短路算法, 则上述算法的时间复杂度为  $O(\nu^3)$ .

**例 1.5.2** 设某市有 7 个化工厂, 已知这些工厂间的公路及距离 (单位: km) 如图所示. 现拟建一个消防队负责这些工厂的消防工作. 问消防队应设在哪个工厂, 以便任一个工厂失火时都能及时赶去灭火?

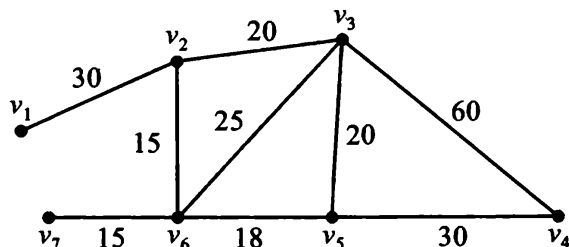


**解** 为了使消防队能在最短时间赶到任一个工厂灭火, 需要消防队离最远工厂的距离尽可能短. 因此这是一个确定图的中心的问题. 此例中图的顶点无赋权, 故可将所有顶点的权视为 1. 利用最短路算法求得图  $G$  的距离矩阵为

$$D = \begin{pmatrix} 0 & 3 & 5 & 9.3 & 6.3 & 4.5 & 6 \\ 3 & 0 & 2 & 6.3 & 3.3 & 1.5 & 3 \\ 5 & 2 & 0 & 5 & 2 & 2.5 & 4 \\ 9.3 & 6.3 & 5 & 0 & 3 & 4.8 & 6.3 \\ 6.3 & 3.3 & 2 & 3 & 0 & 1.8 & 3.3 \\ 4.5 & 1.5 & 2.5 & 4.8 & 1.8 & 0 & 1.5 \\ 6 & 3 & 4 & 6.3 & 3.3 & 1.5 & 0 \end{pmatrix}$$

$D$  的各行最大元分别为 9.3, 6.3, 5, 9.3, 6.3, 4.8, 6.3, 其最小者为 4.8, 对应于第 6 行, 故中心为  $v_6$ , 因此应将消防队设在  $v_6$  处.

**例 1.5.3** 某能源集团有若干煤矿, 集团打算在这些矿点之一建一个火力发电厂. 已知这些煤矿间的道路连接及距离 (单位: km) 如图所示, 煤矿  $v_1 \sim v_7$  每年可供应给该电厂的原煤预计产量依次为 0.5, 1.08, 0.7, 1.13, 0.3, 0.6 (万吨). 问发电厂应建在哪个矿点, 才能使每年供应发电厂的原煤运费最省?



解 这是一个确定图的中位点的问题. 利用最短路算法求得图  $G$  的距离矩阵为

$$D = \begin{pmatrix} 0 & 30 & 50 & 93 & 63 & 45 & 60 \\ 30 & 0 & 20 & 63 & 33 & 15 & 30 \\ 50 & 20 & 0 & 50 & 20 & 25 & 40 \\ 93 & 63 & 50 & 0 & 30 & 48 & 63 \\ 63 & 33 & 20 & 30 & 0 & 18 & 33 \\ 45 & 15 & 25 & 48 & 18 & 0 & 15 \\ 60 & 30 & 40 & 63 & 33 & 15 & 0 \end{pmatrix}$$

各列分别乘以相应顶点的产量, 得

$$D_1 = \begin{pmatrix} 0 & 32.4 & 35 & 93 & 81.9 & 13.5 & 36 \\ 15 & 0 & 14 & 63 & 42.9 & 4.5 & 18 \\ 25 & 21.6 & 0 & 50 & 26 & 7.5 & 24 \\ 46.5 & 68.04 & 35 & 0 & 39 & 14.4 & 37.8 \\ 31.5 & 35.64 & 14 & 30 & 0 & 5.4 & 19.8 \\ 22.5 & 16.2 & 17.5 & 48 & 23.4 & 0 & 9 \\ 30 & 32.4 & 28 & 63 & 42.9 & 4.5 & 0 \end{pmatrix}$$

$D_1$  的行和依次为: 291.8, 157.4, 154.1, 240.74, 136.34, 136.6, 200.8, 其最小者为 136.34, 对应于第 5 行, 故中位点为  $v_5$ . 因此发电厂应设在  $v_5$  处, 每年原煤总运费为 136.34 万吨千米乘以单位运价.

### 三、树的中心和中位点

本部分我们对顶点和边均无赋权的树讨论其中心和中位点. 在这种情况下树  $T$  中顶点的离心率为  $e(u) = \max_{v \in V(T)} d(u, v)$ , 树的中心是具有最小离心率的顶点, 而树的中位点是使  $g(u) = \sum_{v \in V(T)} d(u, v)$  达到最小的顶点.

**定理 1.5.1 (Jordan, 1869)** 一棵树要么只有一个中心, 要么有两个相邻的中心.

**证明** 当  $\nu \leq 2$  时, 结论显然成立.

设结论对所有  $\nu \leq k$  的树都成立. 我们来证明结论对  $\nu = k + 1$  的树也成立.

设树  $T$  是一棵顶点数  $\nu = k + 1 > 2$  的树. 由于  $\nu(T) > 2$ , 故  $T$  的中心不会是叶子 (1 度顶点). 设  $V_T'$  是  $T$  的所有叶子的集合, 则删去  $T$  的所有叶子所得之图  $T' = T - V_T'$  仍是树. 我们来证明  $T'$  与  $T$  具有相同的中心.

首先因  $\nu(T) > 2$ , 故  $T$  的顶点不全是 1 度的, 因而  $V(T')$  非空. 对  $\forall u \in V(T')$ , 必



有  $u \in V(T)$ . 设  $v^*$  是  $T$  中到  $u$  距离最远的顶点, 即

$$d_T(u, v^*) = \max_{v \in V(T)} d_T(u, v) = e_T(u) \quad (e_T(u) \text{ 表示 } u \text{ 在 } T \text{ 中的离心率}),$$

则  $v^*$  必是  $T$  的叶子 (否则  $u$  到  $v^*$  的最短路通过  $v^*$  仍可延伸, 这与  $v^*$  是距  $u$  的最远点矛盾). 由于  $T'$  是由  $T$  删去所有叶子顶点后形成的, 上述结论表明  $T'$  中每个顶点的离心率比  $T$  中减少 1, 即

$$e_{T'}(u) = e_T(u) - 1, \quad \forall u \in V(T').$$

可见  $T'$  中某点  $u^*$  使得

$$e_{T'}(u^*) = \min_{u \in V(T')} e_{T'}(u) \quad \text{当且仅当} \quad e_T(u^*) = \min_{u \in V(T)} e_T(u),$$

因此  $T'$  与  $T$  具有相同的中心.

由于  $\nu(T') \leq k$ , 按照归纳假设,  $T'$  要么只有一个中心, 要么有两个相邻的中心. 因此  $T$  也只有一个中心或有两个相邻的中心. 归纳法完成, 证毕.

对树的中位点有类似的结论.

**定理 1.5.2** 一棵树要么只有一个中位点, 要么有两个相邻的中位点.

证明留作习题.

设  $T$  是一棵树. 对  $T$  中任一顶点  $u$ , 若  $d(u) = k$ , 则  $T - u$  有  $k$  个连通分支, 其中具有最多边数的那个分支中所含的边数, 称为顶点  $u$  的分支权, 记为  $\rho(u)$

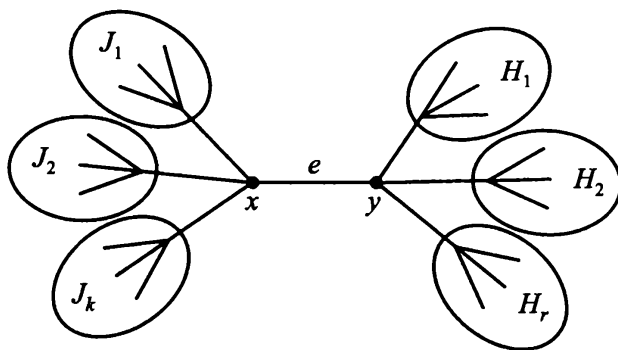
**定理 1.5.3** 设  $T$  是一棵树,  $u \in T$ .  $u$  是  $T$  的中位点当且仅当

$$\rho(u) = \min_{v \in V(T)} \rho(v) \quad (\text{即 } u \text{ 是 } T \text{ 中具有最小分支权的顶点}).$$

**证明** 按定义, 树的中位点是使  $g(u) = \sum_{v \in V(T)} d(u, v)$  达到最小的顶点. 我们分三步来证明.

(1) 对  $\forall e = xy \in E(T)$ ,  $g(x) \leq g(y)$  当且仅当  $\rho(x) \leq \rho(y)$ .

事实上, 假定  $T - x$  中不含  $y$  的分支有  $k$  个, 分别为  $J_1, J_2, \dots, J_k$ ;  $T - y$  中不含  $x$  的分支有  $r$  个, 分别为  $H_1, H_2, \dots, H_r$ , 并设  $J = J_1 \cup J_2 \cup \dots \cup J_k$  共含有  $n_1$  个顶点,  $H = H_1 \cup H_2 \cup \dots \cup H_r$  共含有  $n_2$  个顶点.



则显然 
$$g(x) = \sum_{v \in V(T)} d(x, v) = \sum_{u \in V(J)} d(x, u) + \sum_{v \in V(H)} d(y, v) + n_2 + 1, \quad (*)$$

$$g(y) = \sum_{v \in V(T)} d(y, v) = \sum_{u \in V(J)} d(x, u) + \sum_{v \in V(H)} d(y, v) + n_1 + 1. \quad (**)$$

若  $g(x) \leq g(y)$ , 则  $n_2 \leq n_1$ , 因此  $T - y$  的最大分支必定是含  $x$  的那个分支, 因而  $\rho(x) \leq \rho(y)$ .

反之, 设  $\rho(x) \leq \rho(y)$ , 则  $T - y$  的最大分支必是含  $x$  的那个分支 (若不然, 假定  $T - y$  的最大分支是  $H_i$ , 则  $\rho(x) \geq \nu(H) + 1 > \nu(H_i) = \rho(y)$ , 这与前提条件矛盾). 如果  $T - x$  的最大分支是含  $y$  的那个分支, 则

$$\rho(x) = \nu(H) + 1 = n_2 + 1, \quad \rho(y) = \nu(J) + 1 = n_1 + 1.$$

由  $\rho(x) \leq \rho(y)$  知  $n_2 \leq n_1$ , 从而由 (\*) 和 (\*\*) 式,  $g(x) \leq g(y)$ ; 如果  $T - x$  的最大分支是不含  $y$  的某个分支, 不妨设为  $J_1$ , 则显然

$$n_1 \geq \nu(J_1) \geq T - x \text{ 的含有 } y \text{ 的那个分支的顶点数} \geq n_2,$$

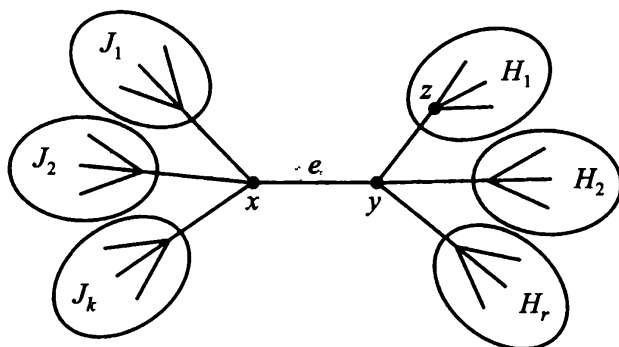
从而由 (\*) 和 (\*\*) 式,  $g(x) \leq g(y)$ .

(2) 对  $\forall e = xy \in E(T)$ , 若  $\rho(x) \leq \rho(y)$ , 则对  $\forall z \in N(y) - \{x\}$ , 都有  $\rho(x) < \rho(z)$  ( $N(y)$  表示  $y$  的邻点集合).

事实上, 不妨设  $z \in H_1$ , 沿用 (1) 中的记号. 前面已证明, 若  $\rho(x) \leq \rho(y)$ , 则  $T - y$  的最大分支必是含  $x$  的那个分支. 于是

$$n_2 + 1 = \nu(H) + 1 \leq \rho(x) \leq \rho(y) = \nu(J) + 1 = n_1 + 1,$$

从而  $n_1 \geq n_2 \geq \nu(H_1)$ , 可见  $T - z$  的最大分支必是  $T - z$  中含  $y$  的那个分支, 它由  $J, H_2, \dots, H_r$  与  $x, y$  以及相互间的边组成 (即  $T - H_1$ ). 因此,  $\rho(z) \geq n_1 + 2 > \rho(y) \geq \rho(x)$ .



(3) 对  $\forall e = xy \in E(T)$ , 若  $g(x) \leq g(y)$ , 则对  $\forall z \in N(y) - \{x\}$ , 都有  $g(x) < g(z)$ .

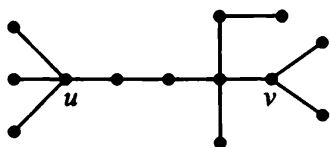
沿用 (2) 中的图形和记号. 因  $g(x) \leq g(y)$ , 由 (1) 知,  $\rho(x) \leq \rho(y)$ , 由 (2) 的证明,  $n_1 \geq n_2 \geq \nu(H_1)$ . 注意到  $T - H_1$  中每点到  $z$  的距离比到  $y$  的距离增加 1, 而  $H_1$  中每点到  $z$  的距离比到  $y$  的距离减少 1, 因此

$$g(z) - g(y) = \nu(T - H_1) - \nu(H_1) \geq \nu(J) + 2 - \nu(H_1) \geq n_1 + 2 - n_2 > 0,$$

故  $g(z) > g(y) \geq g(x)$ .

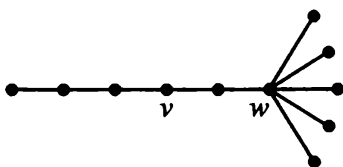
由上述 (1)、(2)、(3) 易证,  $g(u_0) = \min_{u \in V(T)} g(u)$  当且仅当  $\rho(u_0) = \min_{u \in V(T)} \rho(u)$ . 事实上, 若  $g(u_0) = \min_{u \in V(T)} g(u)$ , 则对  $\forall v \in N(u_0)$ ,  $g(u_0) \leq g(v)$ , 由 (1),  $\rho(u_0) \leq \rho(v)$ . 由 (2) 递推知, 对  $\forall v \in T$ ,  $\rho(u_0) \leq \rho(v)$ , 从而  $\rho(u_0) = \min_{u \in V(T)} \rho(u)$ ; 反之, 若  $\rho(u_0) = \min_{u \in V(T)} \rho(u)$ , 则对  $\forall v \in N(u_0)$ ,  $\rho(u_0) \leq \rho(v)$ , 由 (1),  $g(u_0) \leq g(v)$ . 由 (3) 递推知, 对  $\forall v \in T$ ,  $g(u_0) \leq g(v)$ , 从而  $g(u_0) = \min_{u \in V(T)} g(u)$ . 证毕.

有趣的是, 对树  $T$  中两个不相邻的点  $u, v$ ,  $\rho(u) < \rho(v)$  未必一定有  $g(u) < g(v)$ , 反之亦然. 例如, 读者可以考察下图中  $u, v$  两点的情况.



上述定理的结论使我们能够较容易地确定无赋权树的中位点. 实际上, 该充分必要条件往往作为树的中位点的另一种定义. 一些著作中将这样定义的中位点称为树的形心 (centroid).

值得一提的是, 树的中心和中位点未必重合, 例如下图中所示的树的中心为  $v$ , 而中位点为  $w$ . 这也说明一般情况下, 图的中心和中位点未必重合.



还需指出, 树  $T$  中所有顶点对的距离之和  $W(T) = \sum_{u,v \in V(T)} d(u,v)$  称为该树的 Wiener 指数, 它是树的一个重要参数, 在化学和通信理论中有重要的应用, 读者可参看文献 [109]~[117].

图的中心和中位点属于图论和网络最优化中的一个研究专题 —— 选址问题 (location problem) 的研究范畴, 实际应用中常见的  $k$  中心问题、 $k$  中位点问题, 以及各种限制条件下求中心和中位点的问题, 都属于这一研究方向. 在这一研究专题上已有丰富的研究成果, 有兴趣的读者可查阅参考文献 [118]~[155], 其中 [118]~[127] 是关于选址问题的专著和综述, 特别地, 文献 [124]~[125] 和中文文献 [126]~[127] 是几篇很好的近期综述.

## §1.6 图的矩阵表示

设图  $G = (V, E)$ , 其中  $V = \{v_1, v_2, \dots, v_\nu\}$ ,  $E = \{e_1, e_2, \dots, e_\varepsilon\}$ . 定义  $G$  的关联矩阵和邻接矩阵如下:

### 1. 关联矩阵

$$M(G) = \begin{matrix} & e_1 & e_2 & \cdots & e_\varepsilon \\ \begin{matrix} v_1 \\ v_2 \\ \vdots \\ v_\nu \end{matrix} & \begin{pmatrix} m_{11} & m_{12} & \cdots & m_{1\varepsilon} \\ m_{21} & m_{22} & & m_{2\varepsilon} \\ \vdots & \vdots & & \vdots \\ m_{\nu 1} & m_{\nu 2} & \cdots & m_{\nu\varepsilon} \end{pmatrix} \end{matrix},$$

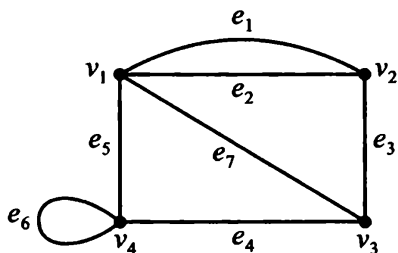
其中  $m_{ij}$  表示顶点  $v_i$  与边  $e_j$  关联的次数.

### 2. 邻接矩阵

$$A(G) = \begin{matrix} & v_1 & v_2 & \cdots & v_\nu \\ \begin{matrix} v_1 \\ v_2 \\ \vdots \\ v_\nu \end{matrix} & \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1\nu} \\ a_{21} & a_{22} & & a_{2\nu} \\ \vdots & \vdots & & \vdots \\ a_{\nu 1} & a_{\nu 2} & \cdots & a_{\nu\nu} \end{pmatrix} \end{matrix},$$

其中  $a_{ij}$  表示顶点  $v_i$  与  $v_j$  相邻的次数.

**例 1.6.1** 求下图的关联矩阵和邻接矩阵.



解

$$M(G) = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{pmatrix}, A(G) = \begin{pmatrix} 0 & 2 & 1 & 1 \\ 2 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & \textcircled{1} \end{pmatrix} \rightarrow 2$$

通过观察容易看出, 图  $G$  的关联矩阵  $M(G)$  和邻接矩阵  $A(G)$  有如下特点:

(1)  $M(G)$  的元素是 0, 1 或 2, 其中元素 2 是由环边导致的;  $A(G)$  的元素也是 0, 1 或 2, 其中 2 是由重边导致的 (若两点间有多于两条重边, 则会出现大于 2 的元素). 如果图  $G$  是简单图, 则关联矩阵  $M(G)$  和邻接矩阵  $A(G)$  中只有元素 0 和 1.

(2) 邻接矩阵  $A(G)$  是对称矩阵.

(3)  $M(G)$  中每列之和 = 2;  $M(G)$  中第  $i$  行之和 =  $v_i$  的度.

(4) 若  $G$  中无环边, 则  $A(G)$  的主对角线元素全为 0, 且  $A(G)$  中第  $i$  行 (列) 之和 =  $v_i$  的度.

**定理 1.6.1** 设  $A$  是  $\nu$  阶图  $G$  的邻接矩阵, 则  $A^n$  的第  $i$  行第  $j$  列元素  $a_{ij}^{(n)}$  等于  $G$  中从  $v_i$  到  $v_j$  的长度为  $n$  的途径的数目 ( $1 \leq n < \nu$ ).

**证明** 对  $n$  作数学归纳法.

$n = 1$  时,  $A$  的第  $i$  行第  $j$  列元素  $a_{ij}^{(1)} = a_{ij}$  等于顶点  $v_i$  到  $v_j$  的边数, 即  $v_i$  到  $v_j$  的长度为 1 的途径的数目, 结论成立.

假设  $n = k$  时结论成立, 即  $A^k$  的第  $i$  行第  $j$  列元素  $a_{ij}^{(k)}$  等于  $G$  中从  $v_i$  到  $v_j$  的长度为  $k$  的途径的数目.

当  $n = k + 1$  时,  $A^n = A^{k+1} = A^k A$ , 故

$$a_{ij}^{(n)} = a_{ij}^{(k+1)} = \sum_{r=1}^{\nu} a_{ir}^{(k)} a_{rj}.$$

由于  $a_{ir}^{(k)}$  是  $G$  中顶点  $v_i$  到  $v_r$  的长度为  $k$  的途径的数目,  $a_{rj}$  是  $G$  中顶点  $v_r$  到  $v_j$  的长度为 1 的途径 (边) 的数目, 故  $a_{ir}^{(k)} a_{rj}$  可看作是  $G$  中从顶点  $v_i$  到  $v_r$  长度为  $k$  的所有途径再经过  $v_r$  与  $v_j$  间的连边前行一步到达  $v_j$ , 所形成的长为  $k + 1$  的途径数目.

因而  $\sum_{r=1}^{\nu} a_{ir}^{(k)} a_{rj}$  是  $G$  中从顶点  $v_i$  到  $v_j$  的长度为  $k + 1$  的途径的总数, 可见定理结论

对  $n = k + 1$  成立.

归纳法完成. 证毕.

**推论 1.6.1** 设  $A$  是简单图  $G$  的邻接矩阵, 则

- (1)  $A^2$  的对角线元素  $a_{ii}^{(2)}$  的值等于  $G$  中顶点  $v_i$  的度;
- (2)  $A^3$  的对角线元素  $a_{ii}^{(3)}$  的值等于  $G$  中含顶点  $v_i$  的三角形数目的两倍;
- (3) 图  $G$  中三角形的数目等于矩阵  $A^3$  的迹除以 6.
- (4) 若  $G$  是连通图, 则  $G$  中任两相异点  $v_i$  与  $v_j$  间的距离等于使  $A^n$  的元素  $a_{ij}^{(n)} \neq 0$  的最小的整数  $n$ .

**证明** (1) 按照定理 1.6.1,  $a_{ii}^{(2)}$  的值等于  $G$  中顶点  $v_i$  到自己长为 2 的途径的数目. 这种长为 2 的途径只能是从  $v_i$  到其邻点再沿原边返回  $v_i$  的途径, 因此其数目等于  $v_i$  的度.

(2) 由定理 1.6.1,  $a_{ii}^{(3)}$  的值等于  $G$  中顶点  $v_i$  到自己长为 3 的途径的数目. 这种长为 3 的途径是以  $v_i$  为一个顶点的三角形. 但在计数过程中每个这种三角形从正、反两个方向算作两条途径, 因此,  $a_{ii}^{(3)}$  的值等于  $G$  中含顶点  $v_i$  的三角形数目的两倍.

(3) 由 (2) 知,  $a_{ii}^{(3)}$  的值等于  $G$  中以  $v_i$  为顶点的三角形数目的 2 倍. 由于每个三角形有三个顶点, 故在计数矩阵  $A^3$  的迹  $a_{11}^{(3)} + a_{22}^{(3)} + \cdots + a_{\nu\nu}^{(3)}$  的过程中, 每个三角形又被重复计算 3 次, 因此  $G$  中三角形的数目等于矩阵  $A^3$  的迹除以 6.

(4)  $a_{ij}^{(n)} \neq 0$  表明点  $v_i$  与  $v_j$  间有长为  $n$  的途径, 因而  $v_i$  与  $v_j$  间的距离不超过  $n$ . 另一方面, 由于  $n$  是使得  $a_{ij}^{(n)} \neq 0$  的最小整数, 因此,  $v_i$  与  $v_j$  间没有长为  $1, 2, \cdots, n-1$  的途径, 从而  $v_i$  与  $v_j$  间的距离不小于  $n$ . 于是结论得证.

**推论 1.6.2** 设  $A$  是  $\nu$  阶图  $G$  的邻接矩阵,  $A + A^2 + \cdots + A^k = (r_{ij})$ , 则  $r_{ij}$  等于  $G$  中从顶点  $v_i$  到  $v_j$  的长度不超过  $k$  的途径的数目.

**证明** 由定理 1.6.1, 这个推论是显然的.

**推论 1.6.3** 设  $A$  是  $\nu$  阶图  $G$  的邻接矩阵 ( $\nu \geq 3$ ),  $R = A + A^2 + \cdots + A^{\nu-1}$ , 则图  $G$  连通的充分必要条件是矩阵  $R$  中每个元素均不为零.

**证明** 设  $A = (a_{ij})$ ,  $A^{(k)} = (a_{ij}^{(k)})$ ,  $R = (r_{ij})$ .

**必要性** 设  $G$  是连通图, 则  $G$  中任两相异顶点  $v_i$  与  $v_j$  之间都有路相连, 取一条  $(v_i, v_j)$  路, 设其长度为  $k$  (显然  $1 \leq k \leq \nu - 1$ ), 则  $a_{ij}^{(k)} \geq 1$ , 故由  $R$  的定义,  $r_{ij} \geq a_{ij}^{(k)} \geq 1$ ; 而对  $\forall i \in \{1, 2, \cdots, \nu\}$ , 因  $G$  连通且  $\nu \geq 3$ , 故  $r_{ii} \geq a_{ii}^{(2)} = d(v_i) \geq 1$ . 可见  $R$  中没有零元素.

**充分性** 设  $v_i$  与  $v_j$  是  $G$  中任两相异顶点. 因  $r_{ij} = a_{ij} + a_{ij}^{(2)} + \cdots + a_{ij}^{(\nu-1)} \neq 0$ , 故必有某  $a_{ij}^{(k)} \neq 0$ . 因此在  $G$  中有至少一条长不超过  $k$  的  $(v_i, v_j)$  路, 由此可见  $G$  是连通图. 证毕.

**推论 1.6.4** 设  $G$  是点、边均无赋权的连通图,  $A$  是  $G$  的邻接矩阵,  $R_k = A + A^2 + \cdots + A^k$ , 则  $G$  的顶点  $v_i$  的离心率  $e(v_i)$  等于使得矩阵  $R_k$  的第  $i$  行没有零元素的最小  $k$  值.

**证明** 在图  $G$  中, 顶点  $v_i$  的离心率  $e(v_i) = \max_{v_j \in V(G)} d(v_i, v_j)$ . 设  $e(v_i) = k_0$ , 则  $v_i$  到  $G$  中任何其他顶点的距离都不超过  $k_0$ , 即对  $\forall v_j \in V(G) (v_j \neq v_i)$ , 都存在某个  $m: 1 \leq m \leq k_0$ , 使得  $A^m$  的元素  $a_{ij}^{(m)} > 0$ . 因此矩阵  $R_{k_0}$  的第  $i$  行元素都不为零. 另一方面, 不可能存在  $k' < k_0$ , 使得  $R_{k'} = A + A^2 + \cdots + A^{k'}$  中第  $i$  行没有零元素. 否则, 对  $\forall v_j \in V(G), (v_j \neq v_i)$ , 因为在  $R_{k'}$  中第  $i$  行第  $j$  列元素不为零, 故必存在某个  $A^m (1 \leq m \leq k' < k_0)$ , 其第  $i$  行第  $j$  列元素  $a_{ij}^{(m)} \neq 0$ . 由推论 1.6.1 的结论 (4), 这意味着  $v_i$  到每个顶点的距离都小于  $k_0$ , 与  $k_0 = e(v_i) = \max_{v_j \in V(G)} d(v_i, v_j)$  矛盾. 证毕.

**推论 1.6.5** 设  $A$  是连通图  $G$  的邻接矩阵,  $R_k = A + A^2 + \cdots + A^k$ , 则  $G$  的半径  $\text{rad}(G)$  等于使得矩阵  $R_k$  中至少有一行没有零元素的最小  $k$  值.

**证明** 因半径  $\text{rad}G = \min_{v \in V(G)} e(v)$ , 其中  $e(v)$  表示顶点  $v$  的无赋权离心率. 设  $\text{rad}G = k_0$ , 则必有某个  $v_i$  使得  $e(v_i) = k_0$ , 由推论 1.6.4,  $k_0$  是使得矩阵  $R_k$  的第  $i$  行没有零元素的最小  $k$  值. 由于  $e(v_i)$  是所有  $e(v)$  中最小的 ( $v \in V(G)$ ), 故  $k_0$  是第 1 个使得  $R_k$  中至少有一行全为非零元的  $k$  值. 证毕.

**推论 1.6.6** 设  $A$  是连通图  $G$  的邻接矩阵,  $R_k = A + A^2 + \cdots + A^k$ , 则  $G$  的直径  $\text{diam}G$  等于使得矩阵  $R_k$  中没有零元素的最小  $k$  值.

**证明** 设  $\text{diam}G = k_0$ . 因直径  $\text{diam}G = \max\{d(u, v) | \forall u, v \in V(G)\}$ , 对  $G$  中任意两点  $v_i, v_j$ , 都有  $d(v_i, v_j) \leq k_0$ , 从而存在某个  $m: 1 \leq m \leq k_0$ , 使得  $A^m$  的元素  $a_{ij}^{(m)} > 0$ , 因此  $R_{k_0}$  的第  $i$  行第  $j$  列位置上的元素不为 0. 由  $v_i, v_j$  的任意性,  $R_{k_0}$  中所有元素都不为 0. 另一方面, 因  $k_0 = \text{diam}G = \max\{d(u, v) | \forall u, v \in V(G)\}$ , 对  $\forall k': 1 \leq k' < k_0$ , 必存在点  $v_i, v_j$  使得  $d(v_i, v_j) > k'$ , 因而  $v_i$  与  $v_j$  之间没有长为  $1, 2, \dots, k'$  的路. 故对所有  $r: 1 \leq r \leq k'$ , 矩阵  $A^r$  的元素  $a_{ij}^{(r)}$  都等于 0, 从而  $R_{k'}$  中第  $i$  行第  $j$  列位置上的元素等于 0. 因此,  $k_0$  是使得矩阵  $R_k$  中没有零元素的最小  $k$  值. 证毕.

推论 1.6.5 不仅给出了求图的半径的方法, 同时也给出了求无赋权图的中心的方法: 在依次计算  $R_1, R_2, \dots, R_{\nu-1}$  的过程中, 当第一次遇到某个  $k$  对应的  $R_k$  中至少有一行全为非零元素时, 非零行对应的顶点即为图的中心.

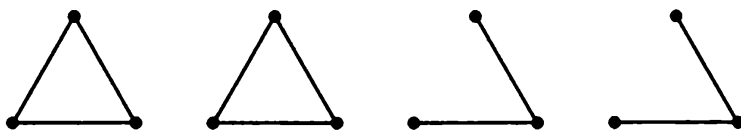
图在计算机上处理时, 可以以关联矩阵或邻接矩阵的形式存于计算机中. 因邻接矩阵比关联矩阵小且又是对称矩阵, 故在计算机存储时通常使用邻接矩阵的上三角部分.

## 习题一

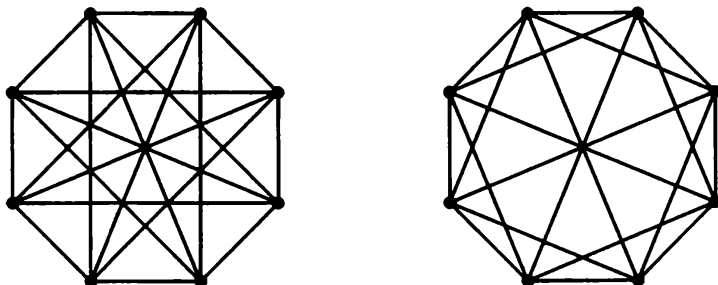
- 1.1 对任何简单图  $G$ , (1) 证明:  $\varepsilon(G) \leq \frac{\nu(\nu-1)}{2}$ ;  
 (2)  $\varepsilon(G) = \frac{\nu(\nu-1)}{2}$  当且仅当  $G \cong K_\nu$ .
- 1.2 证明 (1)  $\varepsilon(K_{m,n}) = mn$ ; (2) 若  $G$  是完全二部图, 则  $\varepsilon(G) \leq \frac{\nu^2}{4}$ .
- 1.3 设图  $G$  有 21 条边, 12 个 3 度顶点, 其余顶点的度均为 2, 求图  $G$  的阶数.
- 1.4 证明: 任何简单图必有至少两个顶点具有相等的度.
- 1.5 设  $G$  是简单图, 求  $G$  的所有不同的生成子图的个数 (包括  $G$  本身和空图).
- 1.6 证明: 在任一次聚会上握过奇数次手的人必为偶数个.
- 1.7 证明: 若  $\nu \geq 4$  且  $\varepsilon = \nu + 1$ , 则存在  $v \in V(G)$  使得  $d(v) \geq 3$ . 由此证明:  $n$  个球队比赛 ( $n \geq 4$ ), 已赛完  $n + 1$  场, 则必定有一个球队已参加过至少 3 场比赛.
- 1.8 在一次体育比赛中, 将所有运动队组织成两个赛区, 每个赛区有 13 个队, 能否恰当安排比赛使得每个队在其所在赛区中进行 9 场比赛而与另一个赛区中的运动队进行 4 场比赛?
- 1.9 在平面上有  $n$  个点  $S = \{x_1, x_2, \dots, x_n\}$ , 其中任两个点之间的距离至少是 1. 证明在这  $n$  个点中, 距离为 1 的点对数不超过  $3n$ .
- 1.10 某次会议有  $n$  人参加, 其中有些人互相认识, 但每两个互相认识的人, 都没有共同的熟人, 每两个互不认识的人都恰好有两个共同的熟人. 证明每一个参加者都有同样数目的熟人.
- 1.11 在一个化学实验室里, 有  $n$  个药箱, 其中每两个不同的药箱恰有一种相同的化学品, 而且每种化学品恰好在两个药箱中出现, 问: (1) 每个药箱有几种化学品?  
 (2) 这  $n$  个药箱中共有几种不同的化学品?
- 1.12 在一次舞会中, A、B 两国留学生各  $n$  ( $n > 2$ ) 人, A 国每个学生都与 B 国一些 (不是所有) 学生跳过舞, B 国每个学生至少与 A 国一个学生跳过舞. 证明一定可以找到 A 国两个学生  $a_1, a_2$  及 B 国两个学生  $b_1, b_2$ , 使得  $a_1$  和  $b_1, a_2$  和  $b_2$  跳过舞, 而  $a_1$  和  $b_2, a_2$  和  $b_1$  没有跳过舞.
- 1.13 设图  $G$  有  $\nu$  个顶点  $\varepsilon$  条边,  $\delta$  和  $\Delta$  分别是  $G$  的最小度和最大度, 证明:  $\delta \leq \frac{2\varepsilon}{\nu} \leq \Delta$  (其中  $\frac{2\varepsilon}{\nu}$  称为图的顶点平均度).
- 1.14 令  $G$  是至少有两个顶点的图. 证明或反驳:  
 (1) 删除一个度为  $\Delta(G)$  的顶点不会增加图的顶点平均度;  
 (2) 删除一个度为  $\delta(G)$  的顶点不会减小图的顶点平均度.



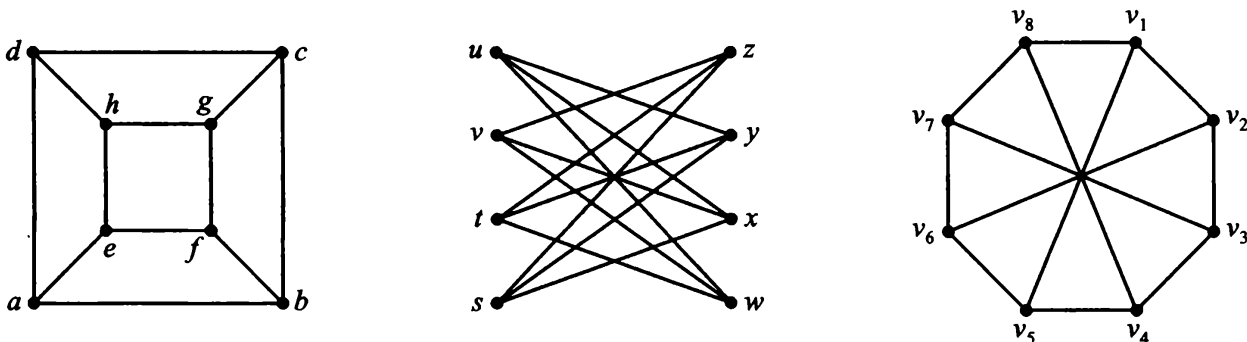
- 1.15 令  $G$  是一个顶点平均度为  $a = \frac{2\varepsilon}{\nu}$  的无环边图.
- (1) 证明:  $G - x$  的顶点平均度至少为  $a$  当且仅当  $d(x) \leq \frac{a}{2}$ ;
  - (2) 利用 (1) 的结果给出一个算法来证明: 如果  $a > 0$ , 则  $G$  有一个最小度大于  $\frac{a}{2}$  的子图.
- 1.16 设  $G$  是连通图, 如果  $\nu(G) \geq 2$  且  $\varepsilon(G) < \nu(G)$ , 则  $G$  至少有两个 1 度顶点.
- 1.17 令  $u$  和  $v$  是简单图  $G$  中的相邻顶点. 证明: 边  $uv$  至少属于  $G$  中的  $d(u) + d(v) - \nu(G)$  个三角形.
- 1.18 证明: 含有  $\nu$  个顶点的  $k$ -正则图有  $\frac{k\nu}{2}$  条边.
- 1.19 证明: 在  $k$ -正则图  $G$  中,  $k\nu \equiv 0 \pmod{2}$ , 即正则图的阶数和度数不可能同时为奇数.
- 1.20 证明: 围长为 4 的  $k$ -正则图至少含有  $2k$  个顶点; 围长为 5 的  $k$ -正则图至少有  $k^2 + 1$  个顶点.
- 1.21 证明: 若  $G = (X \cup Y, E)$  是一个  $k$ -正则二部图, 则  $|X| = |Y|$ .
- 1.22 证明: 连通图中两条最长路必有公共顶点.
- 1.23 若  $G$  是简单图且  $\delta(G) \geq k$ , 则  $G$  有长为至少  $k$  的路; 如果  $k \geq 2$ , 则  $G$  还包含一个长至少为  $k + 1$  的圈.
- 1.24 每个无环边图  $G$  都有一个二部子图至少包含  $\frac{\varepsilon(G)}{2}$  条边.
- 1.25 证明: (1)  $\varepsilon(G) > \nu(G)$  时, 图  $G$  中有圈;  
(2)  $\varepsilon(G) \geq \nu(G) + 4$  时, 图  $G$  中有两个无公共边的圈.
- 1.26 证明: 若  $G$  是  $\delta(G) \geq \frac{\nu(G) - 1}{2}$  的简单图, 则  $G$  是连通图.
- 1.27 (1) 证明: 若  $e \in E(G)$ , 则  $\omega(G) \leq \omega(G - e) \leq \omega(G) + 1$  ( $\omega(G)$  表示图  $G$  的连通分支数);  
(2) 试举例说明: 若  $v \in V(G)$ , 则  $\omega(G) \leq \omega(G - v) \leq \omega(G) + 1$  未必成立.
- 1.28 证明: 若  $G$  是连通图, 且每个顶点都是偶度点, 则  $\omega(G - v) \leq \frac{1}{2}d(v)$ .
- 1.29 证明或反驳: 如果  $G$  是  $\nu$  个顶点的简单图, 且其最大度是  $\left\lceil \frac{\nu}{2} \right\rceil$ 、最小度是  $\left\lfloor \frac{\nu}{2} \right\rfloor - 1$ , 则  $G$  是连通图.
- 1.30 设  $G$  和  $H$  是两个图, 证明  $G \cong H$  当且仅当  $\bar{G} \cong \bar{H}$ .
- 1.31 证明: 如果图  $G$  不连通, 则其补图  $\bar{G}$  必连通.
- 1.32 如果图  $G$  满足  $G \cong \bar{G}$ , 则称  $G$  是自补图. 证明: 若  $\nu$  阶图  $G$  是自补图, 则  $\nu = 0, 1 \pmod{4}$ .
- 1.33 令  $G$  是一个 4 顶点图, 分别删除其中的一个点后得到的子图系列如下, 试确定  $G$ .



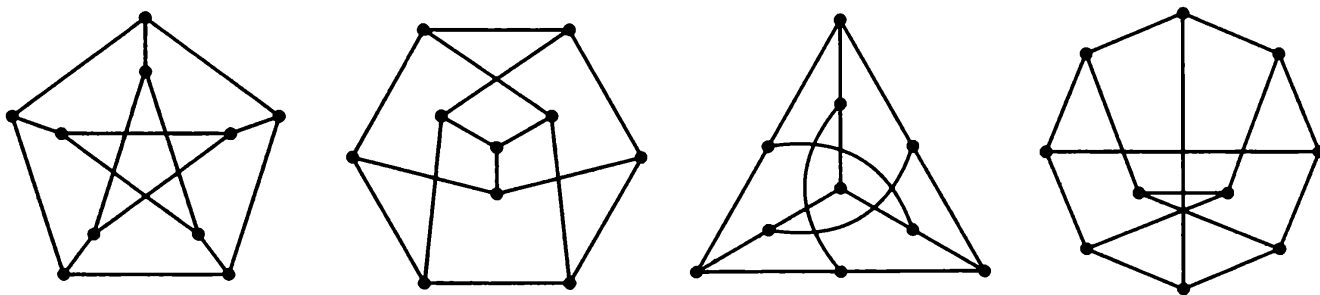
1.34 下面两图是否同构?



1.35 讨论下列三个图的同构性.



1.36 证明: 下列第一个图 (Peterson 图) 与后三个图同构.



1.37 已知  $\nu$  阶简单图  $G$  有  $\varepsilon$  条边, 各顶点的度数均为 3.

(1) 若  $\varepsilon = 3\nu - 6$ , 证明  $G$  在同构意义下唯一, 并求  $\varepsilon, \nu$ ; (2) 若  $\nu = 6$ , 证明  $G$  在同构意义下不唯一.

1.38 若简单图  $G$  有  $\nu$  个顶点,  $\nu - 1$  条边, 则  $G$  一定是树吗? 说明理由.

1.39 证明: 非平凡树中最长路的起点和终点都是叶子 (1 度点).

1.40 证明: 恰有两个叶子的树必是路.

1.41 证明: 若  $G$  是树且  $\Delta \geq k$ , 则  $G$  至少有  $k$  个 1 度顶点.

1.42 证明: 每个树都是二部图.

1.43 设  $n_i$  表示树  $T$  中度为  $i$  的顶点的个数.

(1) 证明对非平凡树  $T$  的叶子数  $n_1$  有下面公式成立:  $n_1 = 2 + n_3 + 2n_4 + 3n_5 + \cdots = 2 + \sum_{i=3}^{\infty} (i-2)n_i$ .

(2) 证明  $\sum_{i=1}^{\infty} in_i$  只依赖于树  $T$  的顶点数.

1.44 证明: (1)  $G$  是森林当且仅当  $\varepsilon = \nu - w$ ;

(2) 无孤立点的森林至少有  $2w$  个 1 度顶点. 这里  $w$  表示  $G$  的连通分支数, 孤立点指零度顶点.

1.45 证明: 具有  $\varepsilon$  条边的  $\nu$  顶点图至少有  $\varepsilon - \nu + 1$  个圈.

1.46 令  $T$  是一棵  $\nu$  顶点树, 对于  $2 \leq i \leq k$  的每个  $i$  值, 树中有一个度为  $i$  的顶点, 其余的  $\nu - k + 1$  个顶点都是叶子. 试确定  $\nu$  并将其表示成  $k$  的表达式.

1.47 设  $T$  是一棵非平凡树, 其中所有与叶子相邻的顶点的度至少为 3, 证明:  $T$  中必有某一对叶子有公共的相邻顶点.

1.48 设  $G$  是一个连通的  $\nu$  顶点图, 证明:  $G$  恰有一个圈当且仅当  $G$  恰有  $\nu$  条边.

1.49 设  $T$  是一棵树, 证明:  $T$  的顶点全为奇度点当且仅当对  $\forall e \in E(T), T - e$  的两个连通分支都具有奇数的阶.

1.50 令  $T$  是一棵阶为偶数的树, 证明:  $T$  恰有一个生成子图使得其中每个顶点的度均为奇数.

1.51 证明: 若  $T$  是一个具有  $k$  条边的树,  $G$  是一个简单图且  $\delta(G) \geq k$ , 则  $G$  含有与  $T$  同构的子图.

1.52 令  $G$  是一棵树, 其中  $2k$  个顶点具有奇数度, 证明:  $G$  可分解成  $k$  条路.

1.53 对  $\nu \geq 4$ , 令  $G$  是满足  $\varepsilon(G) \geq 2\nu - 3$  的  $\nu$  点简单图, 证明:  $G$  有两个等长度的圈.

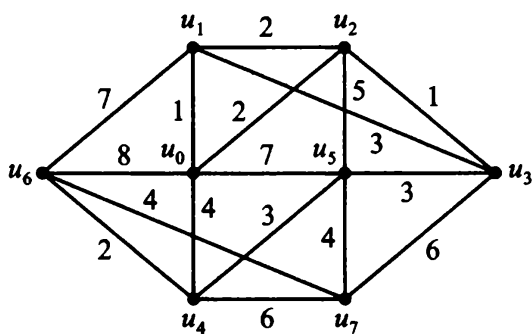
1.54 令  $u$  是连通图  $G$  的一个顶点, 证明可以选出从  $u$  到其他各顶点的最短路使得这些路的并是一棵树.

1.55 令  $T, T'$  是连通图  $G$  的两棵生成树. 对于  $e \in E(T) - E(T')$ , 证明存在一条边  $e' \in E(T') - E(T)$ , 使得  $T' + e - e'$  和  $T - e + e'$  都是  $G$  的生成树.

1.56 设  $G$  是一个赋权连通图并且其各边上的权值互不相同. 不使用 Kruskal 算法, 证明:  $G$  只有一棵权值最小的生成树 (提示: 利用上题结论).

1.57 给  $n$  阶完全图  $K_n$  的所有边任意赋以正整数权值. 一个圈的权值定义为该圈上所有边的权值之和. 证明: 所有圈的权值均为偶数当且仅当具有奇数权值的那些边构成的子图是一个生成二部图. (提示: 对于具有偶数权值的边构成的子图, 其连通分支都是完全图).

1.58 求下图中从  $u_0$  到其余各点的最短路.



1.59 修改 Kruskal 算法用以:

- (1) 求赋权连通图中的最大权生成树;
- (2) 求不连通赋权图中的最小权生成森林.

1.60 设  $G$  是直径为  $k$ 、围长为  $2k+1$  的图, 证明:

- (1) 对  $\forall x \in V(G)$ ,  $x$  必在  $G$  的长为  $2k+1$  的圈上;
- (2)  $G$  必定是正则图.

1.61 证明: 对任一简单连通图  $G$  都有  $\text{rad}G \leq \text{diam}G \leq 2\text{rad}G$ .

1.62 证明: (1) 树  $T$  只有一个中心当且仅当  $\text{diam}T = 2\text{rad}T$ ;

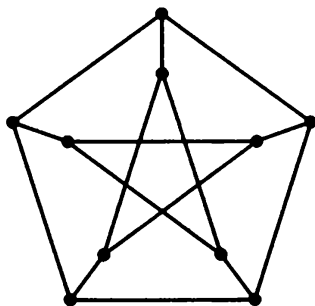
(2) 若树  $T$  有两个中心, 则  $\text{diam}T = 2\text{rad}T - 1$ .

1.63 设  $G$  是一个简单图. 证明: 若  $\text{diam}G \geq 3$ , 则  $\text{diam}\bar{G} \leq 3$ ; 若  $\text{rad}G \geq 3$ , 则  $\text{rad}\bar{G} \leq 2$ .

1.64 证明: 若  $G$  是自补图, 则  $\text{diam}G \leq 3$ .

1.65 设  $G$  是直径为 2 的简单图,  $\nu(G) \geq 4$  且  $\Delta(G) = \nu(G) - 2$ , 证明  $\varepsilon(G) \geq 2\nu(G) - 4$ .

1.66 Peterson 图 (下图所示) 是否是二部图? 试求其围长、半径、直径.



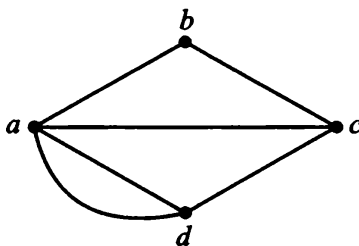
1.67 设  $x$  和  $y$  是图  $G$  中顶点  $v$  的两个不同的邻点, 证明: 如果  $G$  是一棵树, 则离心率  $2e(v) \leq e(x) + e(y)$ .

1.68 证明: 一棵树要么只有一个中位点, 要么有两个相邻的中位点. (提示: 对相邻顶点  $u$  和  $v$ , 考虑  $g(u) - g(v)$ .)

1.69 令  $G$  是一棵树, 它有  $\nu$  个顶点、 $k$  个叶子且最大度为  $k$ . (1) 证明  $G$  是  $k$  条具有

同一个公共端点的路的并. (2) 确定  $\text{diam}G$  的可能的最大值和最小值.

- 1.70 证明: 具有  $\nu + 1$  条边的任意  $\nu$  顶点图的围长最大值为  $\left\lfloor \frac{2\nu + 2}{3} \right\rfloor$ .
- 1.71 证明: 在所有直径为  $k$  但不是树的图中,  $2k + 1$  是围长的最大值. (提示: 证明若  $G$  有一个长至少为  $2k + 2$  的圈, 则  $G$  必有长度更小的圈.)
- 1.72 已知图  $G$  如下:



- (1) 求  $G$  的邻接矩阵  $A$  和关联矩阵  $M$ ;
- (2) 求  $A^2, A^3, A^4$ , 说明从顶点  $b$  到  $d$  长度为 1, 2, 3, 4 的途径分别有几条.

## 参考文献

- [1] C.St.J.A. Nash-Williams, The reconstruction problem, in *Selected Topics in Graph Theory* (L. W. Beineke and R.J. Wilson eds.), Academic Press, London, (1978) 205-236.
- [2] W.T. Tutte, *Graph Theory*, Cambridge University Press, (2001) 115-124. (影印版: 图论, 机械工业出版社, 2004).
- [3] <http://www.nada.kth.se/~viggo/problemlist/>
- [4] M. R. Garey & D. S. Johnson, *Computers and Intractability - a guid to the theory of NP-completeness*, W. H. Freedman, San Francisco, 1979. (中译本: 计算机和难解性 —— NP 完全性理论导引. 张立昂, 等译, 北京: 科学出版社, 1987).
- [5] D.S. Hochbaum, *Approximation Algorithms for NP-hard Problems*, International Thomson Publishing Company, 1995.
- [6] Ding-zhu Du and Ker-I Ko, *Theory of Computational Complexity*, John Wiley & Sons, INC., New York, 2000.
- [7] Feng Cao, Ding-zhu Du, Biao Gao, Peng-jun Wan, and P. M. Pardalos, Minimax problems in combinatorial optimization, *Minmax and Applications* (eds. by Ding-zhu Du & P. M. Pardalos), Kluwer Academic Publishers, 1995.
- [8] D.B.Shmoys, Computing near-optimal solutions to combinatorial optimization problems, *Combinatorial Optimization* (W.Cook, L.Lovasz and P.Seymour eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol.20, (1995) 355-397.
- [9] C.H. Papadimitriou & K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.

- [10] M. R. Garey & D. S. Johnson and L. Stockmeyer, Some simplified NP-complete graph problems, *Theoretic Computer Science*, I 237-267, 1976.
- [11] Cook S. The complexity of theorem-proving procedure, *Conference Record of Third ACM Symposium on Theory of Computing*, 1970, pp151-158.
- [12] 王树禾. 图论及其算法. 合肥: 中国科学技术大学出版社, 1990.
- [13] 谢政. 网络算法与复杂性理论. 2 版. 长沙: 国防科技大学出版社, 2003.
- [14] 陈志平, 徐宗本. 计算机数学 —— 计算复杂性理论与 NPC、NP 难问题的求解. 北京: 科学出版社, 2001.
- [15] 赵瑞清, 孙崇智. 计算复杂性概论. 北京: 气象出版社, 1989.
- [16] S. Even, *Graph Algorithms*, Computer Science Press, 1979.
- [17] A. Gibbons, *Algorithmic Graph Theory*, Cambridge University Press, 1985.
- [18] Christoph Mandl, *Applied Network Optimization*, Academic Press, 1979.
- [19] Gary Chartrand, O.R. Oellermann, *Applied and Algorithmic Graph Theory*, McGraw-Hill Companies, Inc., New York, 1993.
- [20] N. Christofidès, *Graph Theory: An Algorithmic Approach*, Academic Press Inc., London, 1990.
- [21] L.R. Ford, *Network flow theory*, Rand Corporation Report, (1956) 923.
- [22] R.W. Floyd, Algorithm 97- Shortest Path, *Comm., ACM*, 5(6) (1962) 345.
- [23] S. Warshall, A theorem on Boolean matrices, *J. ACM*, 9(1) (1962), 11-12.
- [24] 王朝瑞. 图论. 3 版. 北京: 北京理工大学出版社, 2001.
- [25] 刘纘武. 应用图论. 长沙: 国防科技大学出版社, 2006.
- [26] 甘应爱, 等. 运筹学. 北京: 清华大学出版社, 2003.

### 最短路

- [27] E. Dijkstra, A note on two problem in connection with graphs, *Numer. Math.*, 1(1959) 269-271.
- [28] F. Glover, R. Glover, and D. Klingman, The threshold shortest path algorithm, *Networks*, 14(1984) 25-36.
- [29] M. Desrochers, and F. Soumis, a reoptimization algorithm for the shortest path problem with time windows, *European Journal of Operational Research*, 35(1988) 242-254.
- [30] G. Gallo, and S. Pallottino, Shortest path algorithms, *Annals of Operations Research*, 7(1988) 73-79.
- [31] D. Bertsekas, A simple and fast label correcting algorithm for shortest paths, *Networks*, 23(1993) 703-709.
- [32] M. Dror, Note on the complexity of the shortest path models for column generation, *Operations Research*, 42(1994) 977-978.

- [33] W.B. Powell, Zhi-Long Chen, A generalized threshold algorithm for the shortest path problem with time windows, in *Network Design: Connectivity and Facilities Location* (P. M. Pardalos and Ding-zhu Du eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol.40, (1998) 303-318.
- [34] R. K. Ahuja, J. B. Orlin, S. Pallottino, M. G. Scutella, Minimum time and minimum cost path problems in street networks with traffic lights, *Transportation Science*, 36(2002), 326-336.
- [35] R. K. Ahuja, J. B. Orlin, S. Pallottino, M. G. Scutellà, Dynamic shortest paths minimizing travel times and costs, *Networks*, 41(4)(2003), 197-205.
- [36] C. Duin, A branch-checking algorithm for all-pairs shortest paths, *Algorithmica*, 41(2) (2004), 131-145.
- [37] S. Pettie and V. Ramachandran, A shortest path algorithm for real-weighted undirected graphs, *SIAM Journal on Computing*, 34(2004-2005), 1398-1431.
- [38] T. Relebach, S. Stefanakos, Wavelength conversion in all-optical networks with shortest-path routing, *Algorithmica*, 43(1-2)(2005), 43-61.
- [39] H.D. Sherali, C. Jeenanunta, A. G. Hobeika, The approach-dependent, time-dependent, label-constrained shortest path problem, *Networks*, 48(2)(2006), 57-67.
- [40] U. Zwick, A slightly improved sub-cubic algorithm for the all pairs shortest paths problem with real edge length, *Algorithmica*, 46(2)(2006), 181-192.
- [41] G.A. Klunder, H.N. Post, The shortest path problem on large-scale real-road networks, *Networks*, 48(4) (2006), 182-194.
- [42] P. D'Alberto, A. Nicolau, R-Kleene: A high-performance divide-and-conquer algorithm for the all-pair shortest path for densely connected networks, *Algorithmica*, 47(2)(2007), 203-213.
- [43] L. Gualà, G. Proietti, Exact and approximate truthful mechanisms for the shortest paths tree problem, *Algorithmica*, 49(3)(2007), 171-191.
- [44] T. M. Chan, All-pairs shortest paths with real weights in  $O(n^3/\log n)$  time, *Algorithmica*, 50(2) (2008), 236-243.
- [45] A. V. Goldberg, A practical shortest path algorithm with linear expected time, *SIAM Journal on Computing*, 37(5)(2008), 1637-1655.
- [46] D.E. Knuth, *The art of computer programming, Vol.3: Sorting and Searching*, Addison-Wesley, Reading, Mass., (1973) 184.

#### 约束生成树

- [47] D. Kleitman and D. West, Spanning trees with many leaves, *SIAM Journal on Discrete Mathematics*, 4(1) (1991) 99-106.
- [48] B.M.E. Moret, and H.D. Shapiro, An empirical analysis of algorithms for constructing minimum spanning tree, *Lecture Notes on Computer Science*, 519(1991) 400-411.

- [49] H. I. Lu, R. Ravi, The power of local optimization: Approximation algorithms for maximum-leaf spanning tree, in *Proceedings of the 30<sup>th</sup> Allerton Conference*, (1992) 533-542.
- [50] G. Galbiati, F. Maffioli, A. Morzenti, A short note on the approximability of the maximum leaves spanning tree problem, *Info. Proc. Lett.* 52(1994) 45-49.
- [51] L. Gouveia, A  $2n$  constraint formulation for the capacitated minimal spanning tree problem, *Operations Research*, 43(1995) 130-141.
- [52] G. Robins and J.S. Salowe, Low-degree minimum spanning trees, *Discrete and Computational Geometry*, 14: 2(1995) 151-166.
- [53] B. Boldon, N. Deo, and N. Kumar, Minimum-weight degree-constrained spanning tree problem: heuristics and implementation on an SIMD parallel machine, *Parallel Comput.* 22(1996)369-382.
- [54] G. Craig, M. Krishnamoorthy, and M. Palaniswami, Comparison of heuristic algorithms for the degree constrained minimum spanning tree. In *Meta-heuristics: Theory and Applications* (I.H. Osman and J.P. Kelly eds.), (1996)83-96.
- [55] N. Kumar, *Parallel computation of constrained spanning trees: heuristics and SIMD implementations*, Ph. D. Dissertation, Department of Computer Science, University of Central Florida, Orlando, (1997).
- [56] N. Deo and N. Kumar, Computation of constrained spanning trees: a unified approach, *Network Optimization* (P.M. Pardalos, D.W. Hearn, and W.W. Hager eds.), Lecture Notes in Economics and Mathematical Systems, 450, Springer-Verlag, Berlin Heidelberg, 1997.
- [57] A. Abdallah, N. Deo, N. Kumar, and T. Terry, Parallel computation of a diameter-constrained MST and related problems, *Technical Report CS-TR-97-04*, Department of Computer Science, University of Central Florida, Orlando, (1997).
- [58] N. Deo and N. Kumar, Constrained spanning tree problems: approximate methods and parallel computation, in *Network Design: Connectivity and Facilities Location* (P.M. Pardalos and Ding-zhu Du eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol.40, (1998) 191-216.
- [59] Kaneko, Spanning trees with constraints on the leaf degree, *Discrete Applied Mathematics*, 115(2001), 73-76.
- [60] C. Ribeiro and M. C. Souza, Variable neighborhood search for the degree-constrained minimum spanning tree problem, *Discrete Applied Mathematics*, 118(2002), 43-54.
- [61] R. Hassin and A. Levin, An efficient polynomial time approximation scheme for the constrained minimum spanning tree problem using matroid intersection, *SIAM Journal on Computing*, 33(2003-2004), 261-268.
- [62] R. Hassin and A. Levin, Minimum restricted diameter spanning trees, *Discrete Applied Mathematics*, 137(2004), 343-357.



- [63] J. Könemann, A. Levin, A. Sinha, Approximating the degree-bounded minimum diameter spanning tree problem, *Algorithmica*, 41(2)(2004), 117-129.
- [64] R. Andrade, A. Lucena and N. Maculan, Using Lagrangian dual information to generate degree constrained spanning trees, *Discrete Applied Mathematics*, 154(2006), 703-717.
- [65] M. Laszlo and S. Mukherjee, A class of heuristics for the constrained forest problem, *Discrete Applied Mathematics*, 154(2006), 6-14.
- [66] A. Kaneko, M. Kano, K. Suzuki, Spanning trees with leaf distance at least four, *Journal of Graph Theory*, 55(2007), 83-90.
- [67] A. Salles da Cunha, A. Lucena, Lower and upper bounds for the degree-constrained minimum spanning tree problem, *Networks*, 50(1)(2007), 55-66.

#### 欧氏生成树

- [68] J.M. Ho, D.T. Lee, C.H. Chang, C.K. Wong, Minimum diameter spanning trees and related problems, *SIAM J. Computing*, 20:5(1991) 987-997.
- [69] C. Monna, and S. Suri, Transitions in geometric minimum spanning trees, *Discrete and Computational Geometry*, 8: 2(1992)265-293.
- [70] G. Robins and J.S. Salowe, On the maximum degree of minimum spanning trees, *Discrete and Computational Geometry*, 14: 2(1995)151-166.
- [71] S. Khuller, B. Raghavachari, and N. Young, Low degree spanning tree of small weights, *SIAM J. Computing*, 25: 2(1996) 355-368.

#### 随机生成树

- [72] H. Ishii, S. Shiode, T. Nishida, Y. Namasuya, Stochastic spanning tree problem, *Discrete Applied Mathematics*, 3(1981) 263-273.
- [73] D. J. Bertsimas, The probabilistic minimum spanning tree problem, *Networks*, 20 (1990), 245-275.
- [74] S. Geetha, K. P. K. Nair, On stochastic spanning tree problem, *Networks*, 23(1993)675-679.
- [75] I. B. Moret, Interval elimination method for stochastic spanning tree problem, *Appl. Math. and Computation*, 66(1994) 325-331.
- [76] D. R. Karger, P. N. Klein, and R. E. Tarjan, A randomized linear-time algorithm to find minimum spanning trees, *J. Assoc. Comp. Mach.*, 42(1995) 321-328.

#### 生成树更多文献

- [77] J. B. Jr. Kruskal, On the shortest spanning subtree of a graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical Society*, 7(1956) 48-50.
- [78] R. C. Prim, Shortest connection networks and some generalizations, *Bell Systems Technical Journal*, 36(1957) 1389-1401.

- [79] W. Xu, *Quadratic minimum spanning tree problems and related topics*, Ph.D. Dissertation, Dept. of Math., Univ. of Maryland, College Park, MD, (1984).
- [80] R.E.Tarjan, A simple version of Karzanov's blocking flow algorithm, *Operation Research Letters*. 2(1984) 265-268.
- [81] H.N.Gabow, Z.Galil, T.Spencer, and R.E.Tarjan, Efficient algorithms for finding minimum spanning trees in undirected and directed graphs, *Combinatorica*, 6(1986) 109-122.
- [82] K. Malik, G. Yu, A branch and bound algorithm for the capacitated minimum spanning tree problem, *Networks*, 23(1993) 525-532.
- [83] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows*, Prentice Hall, 1993.
- [84] S. Khuller, B. Raghavachari, N. Young, Balancing minimum spanning and shortest path trees, *Algorithmica*, 14: 4(1995)305-321.
- [85] D. Jungnickel, *Graphs, Networks and Algorithms*, (Algorithms and Computation in Mathematics, Vol.5), Springer, 1998.
- [86] Guoliang Xue and Shangzhi Sun, Optimal multicast tree in communication systems with channel capacities and channel reliabilities, *IEEE Transactions on Communications*, 47: 5(1999), pp662-663.
- [87] R.K. Ahuja, J.B. Orlin, Minimum spanning trees, shortest paths, maximum flow, minimum cost flow, *CRC Handbook of Discrete and Combinatorial Mathematics*, K. H. Rosen (Editor-in chief), CRC Press, New York, 2000, 629-633, 652-683.
- [88] Bang Ye Wu, Kun-Mao Chao and Chuan Yi Tang, Approximation algorithms for the shortest total path length spanning tree problem, *Discrete Applied Mathematics*, 105 (2000), 273-289.
- [89] R.K. Ahuja, J.B. Orlin, A faster algorithm for the inverse spanning tree problem, *Journal of Algorithms*, 34(1)(2000), 177-193.
- [90] M.R. Henzinger and V. King, Maintaining minimum spanning forests in dynamic graphs, *SIAM J. on Computing*, 31: 2(2001) 364-374.
- [91] R.K. Ahuja, T. L. Magnanti, J.B. Orlin, Minimum spanning tree problem, *Handbook of Applied Optimization*, edited by M. Resende and P. Pardalos, Oxford University Press, 2002, 363-374.
- [92] R. K. Ahuja, J.B. Orlin, D. Sharma, A composite very large-scal neighborhood structure for the capacitated minimum spanning tree problem, *Operations Research Letters*, 31(3)(2003), 185-194.
- [93] K. W. Chong, Y. Han, Y. Igarashi and T. W. Lam, Improving the efficiency of parallel minimum spanning tree algorithms, *Discrete Applied Mathematics*, 126(2003), 33-54.
- [94] Bang Ye Wu, Approximation algorithms for the optimal  $p$ -source communication spanning tree, *Discrete Applied Mathematics*, 143(2004), 31-42.

- [95] Bang Ye Wu, An improved algorithm for the  $k$ -source maximum eccentricity spanning trees, *Discrete Applied Mathematics*, 143(2004), 342-350.
- [96] H. B. McMahan and A. Proskurowski, Multi-source spanning trees: algorithms for minimizing source eccentricities, *Discrete Applied Mathematics*, 137(2004), 213-222.
- [97] R. Hassin, A. Levin, Approximation algorithms for quickest spanning tree problems, *Algorithmica*, 41(1)(2004), 43-52.
- [98] Yi-Ting Chiang, Ching-Chi Lin, and Hsueh-I Lu, Orderly spanning trees with applications, *SIAM Journal on Computing*, 34(2004-2005), 924-945.
- [99] D. Benoit, E. D. Demaine, J. I. Munro, R. Raman, V. Raman, S. S. Rao, Representing trees of higher degree, *Algorithmica*, 43(4)(2005), 275-292.
- [100] J. Jansson, J. H. - K. Ng, K. Sadakane, W. - K. Sung, Rooted maximum agreement supertrees, *Algorithmica*, 43(4) (2005), 293-307.
- [101] M. Elkin, A unconditional lower bound on the time-approximation trade-off for the distributed minimum spanning tree problem, *SIAM Journal on Computing*, 36(2)(2006), 433-456.
- [102] S. Arora, G. Karakostas, A  $2 + \varepsilon$  approximation algorithm for the  $k$ -MST problem, *Mathematical Programming*, 107(2006), 491-504.
- [103] Yen Hung Chen, Bang Ye Wu, Chuan Yi Tang, Approximation algorithms for some  $k$ -source shortest paths spanning tree problems, *Networks*, 47(3)(2006), 147-156.
- [104] V. Chepoi, B. Estellon, K. Nouioua, Y. Vaxès, Mixed covering of trees and the augmentation problem with odd diameter constraints, *Algorithmica*, 45(2)(2006), 209-226.
- [105] Bang Ye Wu, Chih-Yuan Hsiao, Kun-Mao Chao, The swap edges of a multiple-sources routing tree, *Algorithmica*, 50(3)(2008), 299-311.
- [106] V. Berry, Z. S. Peng, H. F. Ting, From constrained to unconstrained maximum agreement subtree in linear time, *Algorithmica*, 50(3)(2008), 369-385.
- [107] Mao-Cheng Cai, C.W. Duin, Xiaoguang Yang, Jianzhong Zhang, The partial inverse minimum spanning tree problem when weight increase is forbidden, *European Journal on Operational Research*, 188(2008), 348-353.
- [108] G. Bergantiños, S. Lorenzo-Freire, "Optimistic" weighted shapely rules in minimum cost spanning tree problems, *European Journal on Operational Research*, 185(2008), 289-298.

### Wiener 指数

- [109] I. Gutman, E. Estrada, and O. Ivanciuc, Some properties of the Wiener polynomial of trees, *Graph Theory Notes of New York*, 36(1999) 7-13.
- [110] A.A. Dobrynin, R. Entringer, and I. Gutman, Wiener Index of trees: theory and applications, *Acta Applicandae Mathematicae*, 66(2001) 211-249.
- [111] M. Diudea, I. Gutman, and L. Jantschi, *Molecular Topology*, Nova Publishers, Huntington, New York, 2001.

- [112] M. Fischermann, A. Hoffmann, D. Rautenbach, L. Székely and L. Volkmann, Wiener index versus maximum degree in trees, *Discrete Applied Mathematics*, 122(2002), 127-137.
- [113] W.C. Shiu, P.C. Bor Lam, and K.K. Poon, On Wiener numbers of polygonal nets, *Discrete Applied Mathematics*, 122(2002), 251-261.
- [114] Xueliang Li and Lusheng Wang, Solutions for two conjectures on the inverse problem of the Wiener index of peptoids, *SIAM Journal on Discrete Mathematics*, 17(2003-2004), 210-218.
- [115] J. Rada, Variation of the Wiener index under tree transformations, *Discrete Applied Mathematics*, 148(2005), 135-146.
- [116] J. Rada, and C. Uzcátehui, Randić ordering of chemical tree, *Discrete Applied Mathematics*, 150(2005), 232-250.
- [117] C. R. Johnson, A. L. Duarte, Converse to the Parter-Wiener theorem: The case of non-trees. *Discrete Mathematics*, 306(2006), 3125-3129.

#### 中心、中位点、选址问题

- [118] G.Y. Handler, P.B. Mirchandani, *Location in Networks: Theory and Algorithms*, MIT Press, Massachusetts, 1979.
- [119] S.C. Narula, Minisum hierarchical location-allocation problems on a network: a survey, *Annals of Operations Research*, 6(1986), 257-272.
- [120] R.F. Love, *Facility Location - Models and Methods*, North-Holland, 1988.
- [121] P.B. Mirchandani, R.L. Francis, *Discrete Location Theory*, John Wiley & Sons, Inc., New York, 1990.
- [122] Nicos Christofides, *Graph Theory - an Algorithmic Approach*, Academic Press, INC., 1990.
- [123] Z. Drezner, *Facility Location - a Survey of Applications and Methods*, Springer Ser. Oper. Res., Springer, New York, 1995.
- [124] J. Reese, Solution methods for the  $p$ -median problem: an annotated bibliography, *Networks*, 48(3)(2006), 125-142.
- [125] C.S. ReVelle, H.A. Eiselt, M.S. Daskin, A bibliography for some fundamental problem categories in discrete location science, *European Journal on Operational Research*, 184(2008), 817-848.
- [126] 杨丰梅, 华国伟, 邓猛, 黎建强. 选址问题研究的若干进展. *运筹与管理*, 14(6)(2005), 1-7.
- [127] 黎青松, 杨伟, 曾传华. 中心问题与中位问题的研究现状. *系统工程*, 5(2005).
- [128] J. Plesnik, A heuristic for the  $p$ -center problem in graphs, *Discrete Applied Mathematics*, 17(1987), 262-268.
- [129] J. Bar-Ilan et al, How to allocate network center, *J. Algorithms*, 15(1993), 385-415.

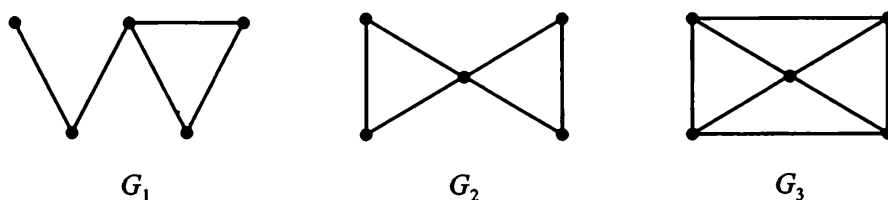
- 
- [130] S. Guha, S. Khuller, Connected facility location problems, in *Network Design: Connectivity and Facilities Location* (P.M. Pardalos and Ding-zhu Du eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 40 (1998), 179-190.
- [131] S. Khuller, and Y. J. Sussmann, The capacitated  $k$ -center problem, *SIAM Journal on Discrete Mathematics*, 13(2000), 403-418.
- [132] A. Tamir, The  $k$ -centrum multi-facility location problem, *Discrete Applied Mathematics*, 109(2001), 293-307.
- [133] T. Feder, R. Motwani, R. Panigrahy, C. Olston, and J. Widom, Computing the median with uncertainty, *SIAM Journal on Computing*, 32(2002-2003), 538-547.
- [134] R. R. Mettu and C. G. Plaxton, The online median problem, *SIAM Journal on Computing*, 32(2002-2003), 816-832.
- [135] J. A. Mesa, J. Puerto and A. Tamir, Improved algorithms for several network location problems with equality measures, *Discrete Applied Mathematics*, 130(2003), 437-448.
- [136] F. A. Chudak and D. B. Shmoys, Improved approximation algorithms for the uncapacitated facility location problem, *SIAM Journal on Computing*, 33(2003-2004), 1-25.
- [137] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit, Local Search Heuristics for  $k$ -median and facility location problems, *SIAM Journal on Computing*, 33(2003-2004), 544-562.
- [138] R.K. Ahuja, J.B. Orlin, S. Pallottino, M.P. Scaparra, M.G. Scutella, A multi-exchange heuristic for the single source capacitated facility location, *Management Science*, 50(2004), 749-760.
- [139] A. Ageev, Yinyu Ye, and Jiawei Zhang, Improved combinatorial approximation algorithms for the  $k$ -level facility location problem, *SIAM Journal on Discrete Mathematics*, 18(2004-2005), 207-217.
- [140] M. Thorup, Quick  $k$ -median,  $k$ -center, and facility location for sparse graphs, *SIAM Journal on Computing*, 34(2004-2005), 405-432.
- [141] M. Charikar and S. Guha, Improved combinatorial algorithms for facility location problems, *SIAM Journal on Computing*, 34(2004-2005), 803-824.
- [142] R. Fleischer, M. J. Golin, Y. Zhang, Online maintenance of  $k$ -medians and  $k$ -covers on a line, *Algorithmica*, 45(4)(2006), 549-567.
- [143] M. Mahdian, Yinyu Ye, Jiawei Zhang, Approximation algorithms for metric facility location problems, *SIAM Journal on Computing*, 36(2)(2006), 411-432.
- [144] R. Benkoczi, B. K. Bhattacharya, D. Breton, Efficient computation of 2-medians in a tree network with positive/negative weights. *Discrete Mathematics*, 306(2006), 1505-1514.
- [145] R. J. Faudree, R. H. Schelp, A. Saito, I. Schiermeyer, Degree conditions for hamiltonicity: counting the number of missing edges. *Discrete Mathematics*, 307(2007), 873-877.

- [146] P. Avella, A. Sassano. I. Vasil'ev, Computational study of large-scale  $p$ -median problems, *Mathematical Programming*, 109(2007), 89-114.
- [147] Z. Stanimirovic, J. Kratica, D. Dugosija, Genetic algorithms for solving the discrete ordered median problem, *European Journal on Operational Research*, 182(2007), 983-1001.
- [148] R. Aboolian, O. Berman, D. Krass, Competitive facility location model with concave demand, *European Journal on Operational Research*, 181(2007), 598-619.
- [149] R. Aboolian, O. Berman, D. Krass, Competitive facility location and design problem, *European Journal on Operational Research*, 182(2007), 40-62.
- [150] J. Kratica, Z. Stanimirovic, D. Tošić, V. Filipovic, Two genetic algorithms for solving the uncapacitated single allocation  $p$ -hub median problem, *European Journal on Operational Research*, 182(2007), 15-28.
- [151] S. de Vries, M.E. Posner, R. V. Vohra, Polyhedral properties of the  $k$ -median problem on a tree, *Mathematical Programming*, 110(2007), 261-285.
- [152] M. P. Scaparra, R. L. Church, An exact solution approach for the interdiction median problem with fortification, *European Journal of Operational Research*, 189(2008), 76-92.
- [153] M. Labbé, H. Yaman, Solving the hub location problem in a star-star network, *Networks*, 51(1)(2008), 19-33.
- [154] O. Berman, Z. Drezner, The  $p$ -median problem under uncertainty, *European Journal on Operational Research*, 189(2008), 19-30.
- [155] D. Fotakis, On the competitive ratio for online facility location, *Algorithmica*, 50(1)(2008), 1-57.

## 第二章 图的连通性

---

在第一章中已经定义连通图是任二顶点间都有路相连的图. 对于连通图, 其连通的程度也有高有低. 例如, 下列三个图都是连通图. 对于图  $G_1$ , 删除一条边或一个顶点便可使其变得不连通; 而对于图  $G_2$ , 至少需要删除两条边才能使其不连通, 也可以删除一个顶点使其不连通; 对于图  $G_3$ , 要破坏其连通性, 则至少需要删除三条边或三个顶点. 明显地,  $G_2$  的边连通性能好于  $G_1$ , 而  $G_3$  的边连通性能和点连通性能都好于前两者.



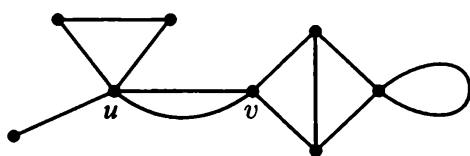
本章主要讨论如何通过图的顶点集、边集和不交的路集合的结构性质来获知图的连通性能好坏. 通过研究割边和割点来刻画 1- 连通图的特性; 定义连通度和边连通度来度量连通图连通程度的高低; 通过不交路结构和元素的共圈性质来反映图的 2- 连通和  $k$ - 连通性.

### §2.1 割点和割边

**定义 2.1.1** 设  $G$  是一个图,  $v \in V(G)$ , 如果  $w(G - v) > w(G)$ , 则称  $v$  为  $G$  的一个割点.

**注** 该定义与某些著作中的定义有所不同, 主要是在环边的顶点是否算作割点上有所区别.

例如, 下图中  $u, v$  两点是其割点.



由定义, 下述定理显然成立.

**定理 2.1.1** 对连通图  $G$ , 顶点  $v$  是  $G$  的割点当且仅当  $G - v$  不连通.

**定理 2.1.2** 如果点  $v$  是简单图  $G$  的一个割点, 则边集  $E(G)$  可划分为两个非空子集  $E_1$  和  $E_2$ , 使得边导出子图  $G[E_1]$  和  $G[E_2]$  恰好有一个公共顶点  $v$ .

**证明** 图  $G$  的割点  $v$  必位于  $G$  的某个连通分支中, 且从  $G$  中删除  $v$  后, 该连通分支必断开成为至少两个新的连通分支. 设  $G_1$  是连通分支之一. 以  $G_1$  的所有边以及  $G$  中  $v$  与  $G_1$  间的所有连边组成边子集  $E_1$ ,  $G$  的其余边组成边子集  $E_2$ , 则显然  $E_1$  和  $E_2$  形成  $G$  的边集的一个划分, 且  $v$  是边导出子图  $G[E_1]$  和  $G[E_2]$  的唯一一个公共顶点. 证毕.

**定理 2.1.3** 设  $v$  是树  $T$  的顶点, 则  $v$  是  $T$  的割点当且仅当  $d(v) > 1$ .

**证明 必要性** 设  $v$  是  $T$  的割点, 下面用反证法证明  $d(v) > 1$ .

(1) 若  $d(v) = 0$ , 则  $T \cong K_1$ , 显然  $v$  不是割点.

(2) 若  $d(v) = 1$ , 则  $T - v$  是有  $\nu(T - v) - 1$  条边的无圈图, 故是树. 从而  $w(T - v) = 1 = w(T)$ . 因此  $v$  不是割点.

以上均与条件矛盾. 因此  $d(v) > 1$ .

**充分性** 设  $d(v) > 1$ , 则  $v$  至少有两个邻点  $u, w$ . 路  $uvw$  是  $T$  中一条  $(u, w)$  路. 因  $T$  是树,  $uvw$  是  $T$  中唯一的  $(u, w)$  路, 所以  $T - v$  中无  $(u, w)$  路, 从而  $w(T - v) > 1 = w(T)$ , 故  $v$  是割点. 证毕.

**推论 2.1.1** 每个非平凡无环连通图至少有两个顶点不是割点.

**证明** 设  $T$  是  $G$  的生成树, 则  $T$  至少有两个叶子  $u, v$ , 由上一定理知,  $u, v$  都不是  $T$  的割点, 即  $w(T - u) = w(T) = 1$ . 由于  $T - u$  是图  $G - u$  的生成树, 故

$$w(G - u) = w(T - u) = w(T) = 1 = w(G),$$

因此  $u$  不是  $G$  的割点. 同理  $v$  也不是  $G$  的割点. 证毕.

**定理 2.1.4** 设  $v$  是连通图  $G$  的一个顶点, 则下列命题等价:

(1)  $v$  是  $G$  的割点;

(2) 存在  $u, w \in V(G)$ , 使得  $u, w$  异于  $v$ , 且  $v$  在每条  $(u, w)$  路上;

(3) 存在  $V(G) \setminus \{v\}$  的一个划分:  $V(G) \setminus \{v\} = U \cup W, U \cap W = \emptyset$ , 使得对  $\forall u \in U$  和  $\forall w \in W, v$  在每条  $(u, w)$  路上.

**证明** (1)  $\Rightarrow$  (3) 因  $v$  是割点, 故  $G - v$  至少有两个连通分支  $G_1, G_2$ . 令  $U = V(G_1)$  而  $W = V(G) \setminus (V(G_1) \cup \{v\})$ , 则对  $\forall u \in U$  和  $\forall w \in W, u, w$  分别属于  $G - v$  的不同



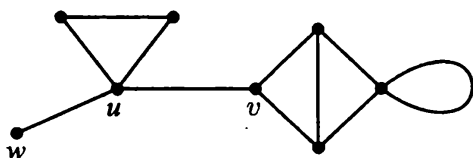
的连通分支. 可见  $G$  中所有的  $(u, w)$  路必经过  $v$  (否则  $G - v$  中仍有  $(u, w)$  路, 这与  $u, w$  分别属于  $G - v$  的不同的连通分支矛盾).

(3)  $\Rightarrow$  (2) 显然.

(2)  $\Rightarrow$  (1) 若  $v$  在每条  $(u, w)$  路上, 则  $G - v$  中不存在  $(u, w)$  路, 即  $G - v$  不连通, 故  $v$  是  $G$  的割点. 证毕.

**定义 2.1.2** 设  $G$  是一个图,  $e \in E(G)$ , 如果  $w(G - e) > w(G)$ , 则称  $e$  为  $G$  的一条割边.

例如下图中, 边  $uv$  和边  $wu$  都是其割边.



**定理 2.1.5** 边  $e$  是图  $G$  的割边当且仅当  $e$  不在  $G$  的任何圈中.

**证明** 证其逆否命题:  $e$  不是割边当且仅当  $e$  含在  $G$  的某个圈中.

**必要性** 设  $e = xy$  不是割边. 假定  $e$  位于  $G$  的某个连通分支  $G_1$  中, 则  $G_1 - e$  仍连通. 故在  $G_1 - e$  中有  $(x, y)$  路  $P$ ,  $P + e$  便构成  $G_1$  中一个含有  $e$  的圈.

**充分性** 设  $e$  含在  $G$  的某个圈  $C$  中, 而  $C$  含于某连通分支  $G_1$  中, 则  $G_1 - e$  仍连通. 故  $w(G - e) = w(G)$ , 这说明  $e$  不是割边. 证毕.

**定理 2.1.6** 一个连通图是树当且仅当它的每条边都是割边.

**证明** 设  $G$  是连通图, 则  $G$  是树  $\Leftrightarrow G$  无圈  $\Leftrightarrow$  任何边  $e$  不含在圈中  $\Leftrightarrow$  任何边  $e$  是  $G$  的割边. 证毕.

**定理 2.1.7** 设  $e$  是连通图  $G$  的一条边, 则下列命题等价:

- (1)  $e$  是  $G$  的割边;
- (2)  $e$  不在  $G$  的任何圈上;
- (3) 存在  $u, v \in V(G)$ , 使得  $e$  在每条  $(u, v)$  路上;
- (4) 存在  $V(G)$  的一个划分:  $V(G) = U \cup W, U \cap W = \emptyset$ , 使得对  $\forall u \in U$  和  $\forall w \in W, e$  在每条  $(u, w)$  路上.

**证明** (1)  $\Leftrightarrow$  (2) 定理 2.1.5 已证. (1)  $\Rightarrow$  (4)  $\Rightarrow$  (3)  $\Rightarrow$  (1) 的证明与定理 2.1.4 的证明类似.

## §2.2 连通度和边连通度

**定义 2.2.1** 对图  $G$ , 若  $V(G)$  的子集  $V'$  使得  $w(G - V') > w(G)$ , 则称  $V'$  为图  $G$  的一个点割集. 含有  $k$  个顶点的点割集称为  $k$ -点割集. 若  $V'$  是图  $G$  的一个点割集, 而  $V'$  减少任意一个点都不再是  $G$  的点割集, 则称  $V'$  是  $G$  的一个极小点割集.  $G$  中含点数最少的点割集称为  $G$  的最小点割集.

注 (1) 割点是 1-点割集.

(2) 完全图没有点割集.

**定义 2.2.2** 图  $G$  的连通度定义为  $\kappa(G) = \min\{|V'| \mid V' \text{ 是连通图 } G \text{ 的点割集}\}$ . 特别地,  $\nu$  阶完全图的连通度定义为  $\kappa(K_\nu) = \nu - 1$ , 不连通图的连通度定义为 0. 若  $\kappa(G) \geq k$ , 则称  $G$  为  $k$ -连通的.

注 (1) 对于不是完全图的连通图  $G$ , 连通度  $\kappa(G)$  就是  $G$  的最小点割集所含的点数.

(2) 按  $k$ -连通的定义, 图  $G$  是  $k$  连通的当且仅当  $G$  的最小点割集至少含  $k$  个顶点, 当且仅当  $G$  中没有  $(k - 1)$ -点割集, 当且仅当从  $G$  中任意去掉  $k - 1$  个顶点后, 所剩图仍连通.

(3) 按  $k$ -连通的定义, 若图  $G$  是  $k$ -连通的, 则它也是  $(k - 1)$ -连通、 $(k - 2)$ -连通、 $\dots$ 、1-连通的. 因此, 所有非平凡连通图都是 1-连通的.

**定义 2.2.3** 设  $G$  是一个图,  $S \subset V(G)$ ,  $\bar{S} = V(G) \setminus S$ , 用  $[S, \bar{S}]$  表示图  $G$  中一端在  $S$  中另一端在  $\bar{S}$  中的所有边的集合, 它称为为图  $G$  的一个边割集. 含有  $k$  条边的边割集称为  $k$ -边割集. 若  $E'$  是图  $G$  的一个边割集, 而  $E'$  减少任意一条边都不再是  $G$  的边割集, 则称  $E'$  是  $G$  的一个极小边割集.  $G$  中含边数最少的边割集称为  $G$  的最小边割集.

例如, 下图 1 中的粗边集合构成了一个 3-边割集; 图 2 中的粗边所成之集不是边割集; 图 3 中的粗边构成一个极小边割集; 图 4 中的粗边构成一个最小边割集.

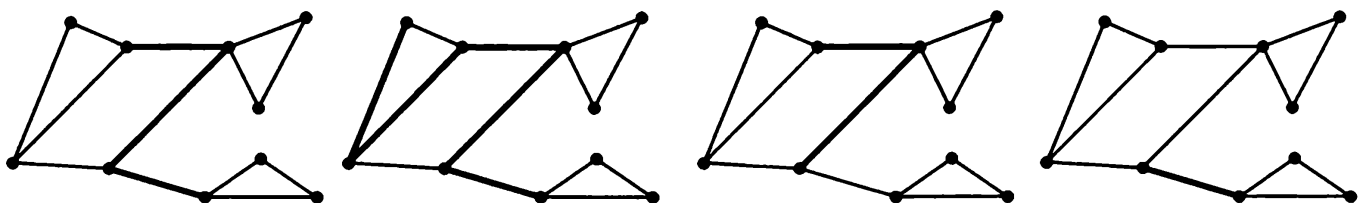


图 1. 边割集

图 2. 非边割集

图 3. 极小边割集

图 4. 最小边割集

注 (1) 对非平凡连通图  $G$ , 若  $E'$  是一个边割集, 则  $G \setminus E'$  不连通. 反之不然. 比如对上述图 2 中粗边所成之集  $E'$ ,  $G \setminus E'$  不连通, 但它不是边割集.

(2) 一条割边构成一个 1- 边割集.

**定义 2.2.4** 图  $G$  的边连通度定义为  $\kappa'(G) = \min\{|E'| \mid E' \text{ 是连通图 } G \text{ 的边割集}\}$ . 完全图的边连通度定义为  $\kappa'(K_\nu) = \nu - 1$ , 不连通图的边连通度定义为 0. 若  $\kappa'(G) \geq k$ , 则称  $G$  为  $k$ - 边连通的.

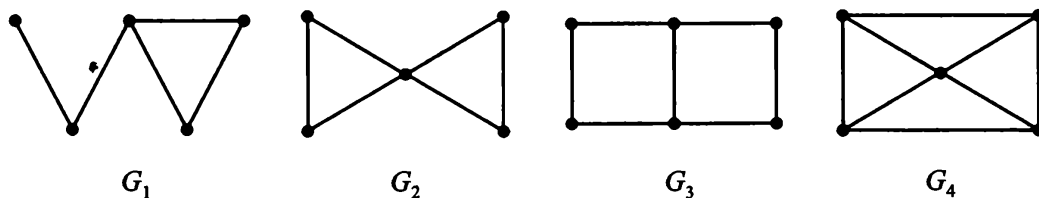
**注** (1) 连通图  $G$  的边连通度  $\kappa'(G)$  就是  $G$  的最小边割集所含的边数.

(2) 若  $G$  是含有割边的连通图, 则  $\kappa'(G) = 1$ .

(3) 按  $k$ - 边连通的定义, 图  $G$  是  $k$ - 边连通的, 当且仅当  $G$  的最小边割集至少含  $k$  条边, 当且仅当  $G$  中没有  $(k-1)$ - 边割集, 当且仅当从  $G$  中任意去掉  $k-1$  条边后, 所剩图仍连通.

(4) 按  $k$ - 边连通的定义, 若图  $G$  是  $k$ - 边连通的, 则它也是  $(k-1)$ - 边连通、 $(k-2)$ - 边连通、 $\dots$ 、1- 边连通的. 因此, 所有非平凡连通图都是 1- 边连通的.

例如, 下列图中,  $G_1$  是连通的且有割点和割边, 因此是 1- 连通的和 1- 边连通的;  $G_2$  的最小点割集含 1 个点, 最小边割集含 2 条边, 故它是 1- 连通的和 2- 边连通的;  $G_3$  的最小点割集和最小边割集分别含 2 个点和 2 条边, 因此是 2- 连通的和 2- 边连通的;  $G_4$  的最小点割集和最小边割集分别含 3 个点和 3 条边, 因此是 3- 连通的和 3- 边连通的.



**定理 2.2.1**  $\kappa(G) \leq \kappa'(G) \leq \delta(G)$ .

**证明** 先证  $\kappa(G) \leq \kappa'(G)$ .

对图的边连通度  $\kappa'(G)$  作数学归纳法.

对  $\kappa'(G) = 1$  的图  $G$ ,  $G$  连通且至少含有两个顶点. 若  $G$  仅含两个顶点, 则显然  $\kappa(G) = \nu - 1 = 1$ ; 若  $G$  至少含三个点, 设  $e = uv$  是  $G$  的一条割边, 则  $u$  或  $v$  必是割点, 故  $\kappa(G) = 1$ . 总之, 此时  $\kappa(G) = \kappa'(G) = 1$ .

假设对所有  $\kappa' = k$  的图, 都有  $\kappa \leq \kappa'$ , 则对  $\kappa'(G) = k+1$  的图  $G$ , 设  $E_1$  是它的一个  $k+1$  边割集. 任取边  $e = uv \in E_1$ , 则  $E_1 - e$  是  $G - e$  的最小边割集, 故  $\kappa'(G - e) = k$ . 由归纳假设,  $\kappa(G - e) \leq \kappa'(G - e)$ . 取  $G - e$  的最小点割集  $T$ , 则

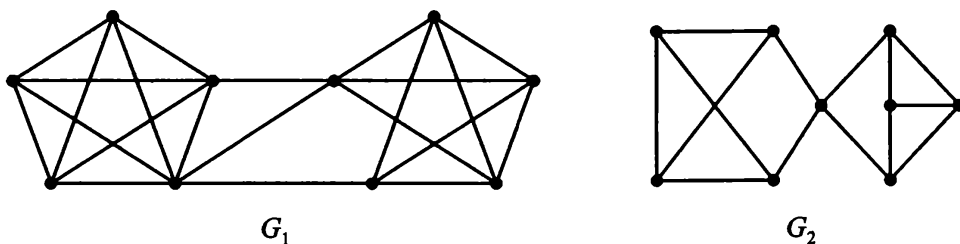
$$|T| = \kappa(G - e) \leq \kappa'(G - e) = k,$$

且  $T \cup \{u\}$  构成  $G$  的一个点割集. 故  $\kappa(G) \leq |T \cup \{u\}| \leq |T| + 1 \leq k + 1 = \kappa'(G)$ . 归纳完成.

再证  $\kappa'(G) \leq \delta(G)$ .

设  $d(v) = \delta$ . 删去与  $v$  关联的  $\delta$  条边后,  $G$  变成不连通图, 故这  $\delta$  条边构成  $G$  的一个边割集. 因此  $\kappa'(G) \leq \delta(G)$ . 证毕.

下图  $G_1$  是一个  $\kappa = 2$ ,  $\kappa' = 3$  和  $\delta = 4$  的图;  $G_2$  是一个  $\kappa = 1$ ,  $\kappa' = 2$  和  $\delta = 3$  的图.



**定理 2.2.2** 对具有  $\nu$  个顶点  $\varepsilon$  条边的连通图  $G$ , 有  $\kappa(G) \leq \left\lfloor \frac{2\varepsilon}{\nu} \right\rfloor$ .

**证明** 因  $2\varepsilon = \sum_{v \in V(G)} d(v) \geq \delta\nu$ , 故  $\delta \leq \frac{2\varepsilon}{\nu}$ , 由定理 2.2.1,  $\kappa \leq \delta \leq \frac{2\varepsilon}{\nu}$ . 由于  $\kappa$  是整数, 因此  $\kappa \leq \left\lfloor \frac{2\varepsilon}{\nu} \right\rfloor$ . 证毕.

**定理 2.2.3** 设  $G$  是一个简单图,  $k$  是一个自然数, 若  $\delta(G) \geq \frac{\nu + k - 2}{2}$ , 则  $G$  是  $k$ -连通的.

**证明** 用反证法. 假如  $G$  不是  $k$ -连通的, 则  $G$  的连通度  $\kappa < k$ , 即存在  $G$  的点割集  $S$ , 使得  $|S| < k$ , 且  $G - S$  不连通. 因  $G - S$  有  $\nu - |S|$  个顶点, 且至少有两个连通分支, 故必有  $G - S$  的某个连通分支  $G'$  含有不超过  $\frac{\nu - |S|}{2}$  个顶点. 注意到  $G'$  中任一个顶点只可能与  $G'$  内的点及  $S$  中的点相邻, 因而其在  $G$  中的顶点度  $\leq \frac{\nu - |S|}{2} - 1 + |S| = \frac{\nu + |S| - 2}{2}$ . 结合  $|S| < k$ , 这意味着  $\delta(G) \leq \frac{\nu + |S| - 2}{2} < \frac{\nu + k - 2}{2}$ , 与定理条件矛盾. 证毕.

**推论 2.2.1** 设  $G$  是一个简单图, 若  $\delta(G) \geq \frac{\nu - 1}{2}$ , 则  $G$  是连通图.

**证明** 在定理 2.2.3 中取  $k = 1$  即可.

**定理 2.2.4** 设  $G$  是一个直径为 2 的简单图, 则  $\kappa'(G) = \delta(G)$ .

**证明** 因  $G$  的直径为 2, 故  $G$  连通. 设  $E'$  是  $G$  的一个最小边割集, 将  $G - E'$  中顶点数最少的一个连通分支记为  $G_1$ , 另一个连通分支记为  $G_2$ .

设  $u$  是  $G_1$  中一个顶点, 如果它在  $G$  中不与  $G_2$  的顶点相邻, 则对  $G_2$  中任意顶点  $v$ , 由  $d_G(u, v) = 2$  知, 在  $G$  中  $v$  必有邻点属于  $G_1$ . 由此可知, 在  $G$  中来看, 要么

$G_1$  中每个点都在  $G_2$  中有邻点, 要么  $G_2$  中每个点都在  $G_1$  中有邻点. 在前一种情况下,  $|E'| \geq \nu(G_1)$ , 在后一种情况下,  $|E'| \geq \nu(G_2)$ . 总之,

$$\kappa' = |E'| \geq \min\{\nu(G_1), \nu(G_2)\} = \nu(G_1). \quad (*)$$

对  $G_1$  中任意顶点  $u$ , 用  $d_1(u)$  和  $d_2(u)$  分别表示在  $G$  中  $u$  向  $G_1$  和向  $G_2$  连的边数. 则

$$\delta(G) \leq d_G(u) = d_1(u) + d_2(u) \leq \nu(G_1) - 1 + d_2(u).$$

根据定理 2.2.1,

$$\kappa' \leq \delta(G) \leq \nu(G_1) - 1 + d_2(u). \quad (**)$$

结合 (\*) 式可得  $d_2(u) \geq 1$ . 任取  $G_1$  中一个顶点  $u_0$ , 由于

$$\kappa' = |E'| = \sum_{u \in G_1} d_2(u) = \sum_{u \in G_1 - u_0} d_2(u) + d_2(u_0) \geq \nu(G_1) - 1 + d_2(u_0),$$

结合 (\*\*) 式得

$$\nu(G_1) - 1 + d_2(u_0) \leq \kappa' \leq \delta(G) \leq \nu(G_1) - 1 + d_2(u_0)$$

可见  $\kappa' = \delta(G)$ . 证毕.

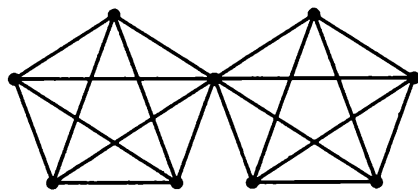
**推论 2.2.2** 设非空简单图  $G$  中任二不相邻顶点  $u$  和  $v$  都满足  $d(u) + d(v) \geq \nu - 1$ , 则  $\kappa'(G) = \delta(G)$ .

**证明** 若  $G$  中任意两点都相邻, 则  $G$  是完全图,  $\kappa'(G) = \nu - 1 = \delta(G)$ , 结论成立. 若  $G$  中有不相邻顶点, 则  $\text{diam } G \geq 2$ . 但由于任二不相邻顶点  $u$  和  $v$  都满足  $d(u) + d(v) \geq \nu - 1$ , 即任二不相邻顶点  $u$  和  $v$  都有公共的邻点. 因此又有  $\text{diam } G \leq 2$ . 从而  $\text{diam } G = 2$ . 由定理 2.2.4 便得  $\kappa'(G) = \delta(G)$ . 证毕.

按照推论 2.2.2, 下面的结论是显然的.

**推论 2.2.3** 设  $G$  是一个简单图, 若  $\delta(G) \geq \frac{\nu - 1}{2}$ , 则  $\kappa'(G) = \delta(G)$ .

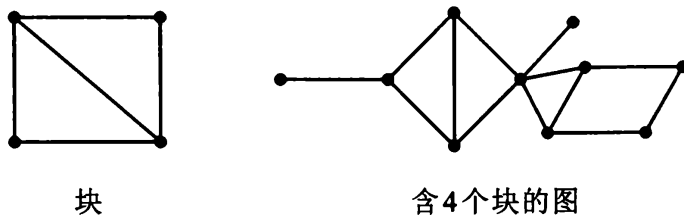
虽然直径为 2 的图能使得边连通度达到上界  $\delta(G)$ , 但其点连通度仍然可能很小. 例如下图的直径为 2, 其连通度只有 1.



### §2.3 2-连通图的性质

**定义 2.3.1** 无割点的连通图称为块 (block). 设  $G$  是一个图,  $H$  是  $G$  的一个子图, 若  $H$  本身是一个块且它是  $G$  中具有此性质的极大子图, 则称  $H$  是图  $G$  的一个块.

下面是块及图的块的例子.



**注** 至少有三个顶点的图是块当且仅当它是 2-连通图. (若只有两个顶点, 则有反例, 例如  $K_2$  是个块, 但不是 2-连通的.)

关于 2-连通图 (块), 有下列重要结论.

**定理 2.3.1 (Whitney, 1932)**  $\nu \geq 3$  的图  $G$  是 2-连通图 (块) 当且仅当  $G$  中任二顶点共圈.

**证明 充分性** 设  $G$  中任二顶点在同一圈上, 欲证  $G$  是 2-连通的.

对  $\forall w \in V(G)$ , 任取  $u, v \in V(G - w)$ . 由条件,  $u, v$  在  $G$  中共处于某个圈  $C$  上. 若  $w \notin C$ , 则在  $G - w$  中  $u, v$  仍在圈  $C$  上; 若  $w \in C$ , 则  $G - w$  中  $u, v$  在路  $C - w$  上. 总之  $u, v$  在  $G - w$  中有路相连. 由  $u, v$  的任意性,  $G - w$  是连通图, 故  $w$  不是  $G$  的割点. 再由  $w$  的任意性知,  $G$  无割点, 即  $G$  是 2-连通的.

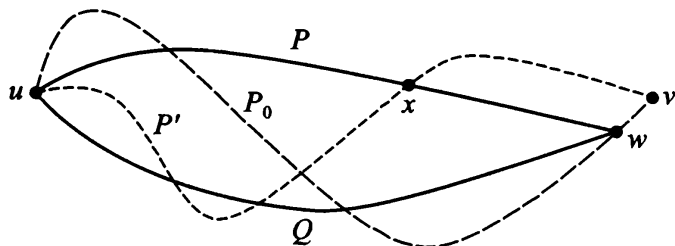
**必要性** 设  $G$  是 2-连通图, 欲证任二顶点  $u, v$  都在同一圈上.

对距离  $d(u, v)$  作归纳法.

$d(u, v) = 1$  时的情况. 因  $G$  是 2-连通图, 故  $\kappa' \geq \kappa \geq 2$ , 因而  $G$  中无割边,  $G - uv$  仍连通. 于是  $G - uv$  中存在一条  $(u, v)$  路  $P_1$ . 这表明在  $G$  中  $u, v$  都在圈  $P_1 + uv$  上.

假定  $d(u, v) < k$  时, 结论成立. 下证  $d(u, v) = k$  时结论也成立.

当  $d(u, v) = k$  时, 设  $P_0 = u \cdots wv$  是长为  $k$  的一条  $(u, v)$  路, 则  $d(u, w) = k - 1$ . 由归纳法假设,  $u, w$  在同一圈上, 故在  $u, w$  间有两条无公共内部顶点的路  $P$  和  $Q$ .



因  $G$  是 2-连通图, 故  $G - w$  仍连通, 在  $G - w$  中存在  $(u, v)$  路  $P'$ . 令  $x$  是  $P'$  上最后一个与  $P \cup Q$  的公共顶点 (因  $u \in P \cup Q$ , 这样的  $x$  存在). 不妨设  $x \in P$ , 则  $P$  上

$(u, x)$  段 +  $P'$  上  $(x, v)$  段与  $Q + wv$  是两条内部无公共点的  $(u, v)$  路. 故  $u, v$  在同一圈上. 归纳法完成. 证毕.

由定理 2.3.1, 下面一个推论显然成立.

**推论 2.3.1**  $\nu \geq 3$  的图  $G$  是 2-连通图 (块) 当且仅当  $G$  中任二顶点被至少两条内部无公共顶点的路所连.

**推论 2.3.2** 若  $\nu \geq 3$  的图  $G$  是 2-连通图 (块), 则  $G$  的任二边都位于同一圈上.

**证明** 设  $G$  是  $\nu \geq 3$  的 2-连通图, 且  $e_1, e_2 \in E(G)$ , 在  $e_1$  和  $e_2$  上各添加一个新的顶点  $v_1$  和  $v_2$ , 构成一个新图  $G'$ , 则  $G'$  仍是 2-连通的. 由 Whitney 定理,  $v_1$  和  $v_2$  在  $G'$  中位于同一个圈上, 故  $e_1$  和  $e_2$  在  $G$  中位于同一个圈上. 证毕.

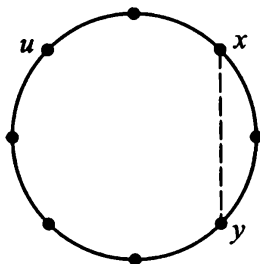
关于块的一些等价命题总结在下一个定理中.

**定理 2.3.2** 设  $G$  是  $\nu \geq 3$  的连通图, 则下列命题等价:

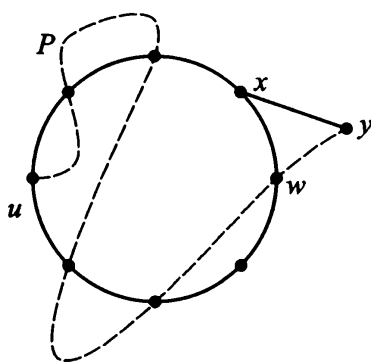
- (1)  $G$  是 2-连通的 (块);
- (2)  $G$  的任二顶点共圈;
- (3)  $G$  的任一顶点与任一边共圈;
- (4)  $G$  的任二边共圈;
- (5) 对  $\forall u, v \in V(G)$  及  $\forall e \in E(G)$ , 存在  $(u, v)$  路含有边  $e$ ;
- (6) 对  $\forall u, v, w \in V(G)$ , 存在  $(u, v)$  路含有顶点  $w$ ;
- (7) 对  $\forall u, v, w \in V(G)$ , 存在  $(u, v)$  路不含有顶点  $w$ .

**证明** (1)  $\Leftrightarrow$  (2): 见 Whitney 定理.

(2)  $\Rightarrow$  (3): 设  $G$  中任二顶点共圈. 对  $\forall u \in V(G)$  及  $\forall e = xy \in E(G)$ , 若  $x = u$  或  $y = u$ , 则结论显然. 以下假定  $x, y \neq u$ . 设  $C$  是含  $u$  与  $x$  的圈. 若  $y \in C$ , 则  $C$  上含有  $u$  的  $(x, y)$  路与边  $xy$  形成一个圈, 它含有  $u$  及边  $e$ , 如下图所示.

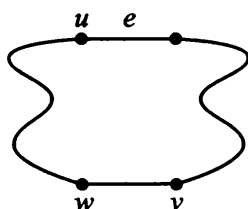


若  $y \notin C$ , 则由 Whitney 定理,  $x$  不是割点, 故存在不含  $x$  的  $(u, y)$  路  $P$ , 令  $w$  是  $P$  上从  $y$  出发第一个与  $C$  公共的顶点, 则  $C$  上  $x - u - w$  段 +  $P$  上  $(w, y)$  段 +  $xy$  构成一个含  $u$  和  $e = xy$  的圈, 如下图所示.



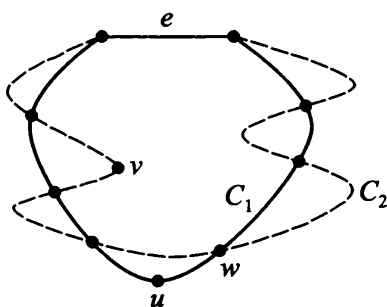
(3)  $\Rightarrow$  (4): 与 (2)  $\Rightarrow$  (3) 类似可证.

(4)  $\Rightarrow$  (5): 设  $G$  中任二边共圈. 对  $\forall u, v \in V(G)$  及  $\forall e \in E(G)$ , 如果  $e = uv$ , 结论显然成立; 如果  $u$  或  $v$  之一是  $e$  的端点, 比如  $u$  是  $e$  的端点. 设  $v$  的一个邻点是  $w$ , 则  $e$  与边  $wv$  共圈, 故显然有  $(u, v)$  路含有边  $e$ , 如下图所示.



下面假定  $u$  和  $v$  都不是  $e$  的端点.

因任二边共圈显然任二点共圈, 故由 (2)  $\Rightarrow$  (3) 知  $u$  与  $e$  共圈,  $v$  也与  $e$  共圈. 设这二圈分别是  $C_1$  和  $C_2$ . 若  $u \in C_2$  或  $v \in C_1$ , 则结论显然成立; 若  $u \notin C_2$  且  $v \notin C_1$ , 则如下构造含  $e$  的  $(u, v)$  路: 从  $u$  出发沿  $C_1$  到达  $C_1$  与  $C_2$  的第一个公共顶点  $w$ , 再从  $w$  出发沿  $C_2$  含  $e$  的部分到达  $v$ , 即可 (如下图所示).



(5)  $\Rightarrow$  (6): 对  $\forall u, v, w \in V(G)$ , 设与  $w$  相关联的一条边为  $e$ . 由 (5), 存在  $(u, v)$  路含有边  $e$ , 于是含有  $w$ .

(6)  $\Rightarrow$  (7): 对  $\forall u, v, w \in V(G)$ , 存在  $(u, w)$  路  $P$  含有顶点  $v$ , 则  $P$  的从  $u$  到  $v$  的一段不含有  $w$ .

(7)  $\Rightarrow$  (1): 因为对  $\forall w \in V(G)$  及  $\forall u, v \in V(G)$ , 存在  $(u, v)$  路不含有顶点  $w$ , 故  $w$  不是割点. 由  $w$  的任意性,  $G$  无割点. 从而  $G$  是块. 证毕.



## §2.4 Menger 定理

上节的 Whitney 定理表明, 对一个图  $G$ ,  $G$  的最小点割集含至少 2 个点当且仅当  $G$  中任意两点间都有至少 2 条内部无公共顶点的路, 即“分离” $G$  中两点所需删除的最少顶点数和这两点间内部无公共顶点的路的最大条数要么都大于等于 2, 要么都不超过 2. 这涉及到图的连通性能的两种度量: 删除顶点时图保持连通的能力和顶点间不交路径的重数. Whitney 定理说明对 2- 连通图这两种连通性度量是等价的. 下列 Menger 定理显示, 这一结论对任意  $k$ - 连通图都成立.

设  $G$  是一个简单图,  $x, y$  是  $G$  中任二不同顶点. 如果从  $G$  中删去一组顶点后不再有  $(x, y)$  路, 则称这组顶点分离  $x$  和  $y$ , 并将这组顶点称为一个  $(x, y)$  分离集.  $G$  中含点数最少的  $(x, y)$  分离集称为最小  $(x, y)$  分离集, 其顶点数称为  $G$  的  $(x, y)$  分离数, 记为  $s(x, y)$ . 此外, 将  $G$  中两两内部无公共顶点的  $(x, y)$  路的最大条数记为  $r(x, y)$ .

**定理 2.4.1 (Menger, 1927)** 设  $x, y$  是图  $G$  中两个不相邻顶点, 则  $G$  中分离  $x, y$  所需的最少顶点数等于  $G$  中两两内部无公共顶点的  $(x, y)$  路的最大条数, 即  $s(x, y) = r(x, y)$ .

**证明** 首先, 对任意图  $G$  和  $G$  中任意两个不相邻顶点  $x, y$ , 显然有  $s(x, y) \geq r(x, y)$ .

下面用反证法证明对任意图  $G$  和  $G$  中任意两个不相邻顶点  $x, y$ ,  $s(x, y) \leq r(x, y)$ .

假设结论不真. 则存在图  $G$  和  $G$  中两个不相邻顶点  $x, y$ , 使得  $s(x, y) > r(x, y)$ . 取  $G$  为具有这种性质的所有图中顶点数最少并且边数最少的一个, 即: 对具有上述性质的任何一个图  $H$ , 要么  $\nu(G) < \nu(H)$ , 要么  $\nu(G) = \nu(H)$  且  $\varepsilon(G) \leq \varepsilon(H)$ .

下面证明该图  $G$  的四条结论.

(1) 按照上述取法, 由于  $s(x, y) > r(x, y) \geq 0$ , 可知  $x$  与  $y$  在  $G$  中是连通的. 如果  $s(x, y) = 1$ , 则必须  $r(x, y) = 0$ , 这与  $x, y$  连通矛盾. 因此  $s(x, y) > 1$ .

(2) 对  $G$  的任二异于  $x, y$  的相邻顶点  $u$  和  $v$ , 存在  $G$  的顶点子集  $U$ , 满足  $|U| = s(x, y) - 1$  且  $U \cup \{u\}$  和  $U \cup \{v\}$  都是  $G$  的  $(x, y)$  分离集.

事实上, 设  $e = uv$ ,  $H = G - e$ .  $H$  的  $(x, y)$  分离数

$$s_H(x, y) \geq s(x, y) - 1, \quad (*)$$

(否则存在  $H$  的  $(x, y)$  分离集  $R$  使得  $|R| \leq s(x, y) - 2$ , 则  $R \cup \{u\}$  是  $G$  的  $(x, y)$  分离集, 且只含  $s(x, y) - 1$  个点, 这与  $G$  的  $(x, y)$  分离数为  $s(x, y)$  矛盾). 另一方面, 由于  $s(x, y) > r(x, y)$ ,  $H$  中两两内部无公共顶点的  $(x, y)$  路的条数

$$r_H(x, y) \leq r(x, y) \leq s(x, y) - 1. \quad (**)$$

由  $G$  的最小性可知,  $H$  中的  $(x, y)$  分离数等于  $H$  中两两内部无公共顶点的  $(x, y)$  路的最大条数, 即  $s_H(x, y) = r_H(x, y)$ . 结合 (\*) 式和 (\*\*) 式, 知  $s_H(x, y) = r_H(x, y) = s(x, y) - 1$ . 设  $U$  是  $H$  的一个最小  $(x, y)$  分离集, 由于  $H$  与  $G$  只相差一条边  $e = uv$ , 易见  $U \cup \{u\}$  和  $U \cup \{v\}$  都是  $G$  的  $(x, y)$  分离集.

(3) 用  $N_G(x)$  和  $N_G(y)$  分别表示  $x$  和  $y$  在  $G$  中的邻点集合, 则  $N_G(x) \cap N_G(y) = \emptyset$ .

事实上, 假如  $N_G(x) \cap N_G(y) \neq \emptyset$ , 则  $\exists u \in N_G(x) \cap N_G(y)$ . 设  $H = G - u$ . 由  $G$  的最小性,  $H$  中  $(x, y)$  分离数等于  $H$  中两两内部无公共顶点的  $(x, y)$  路的最大条数. 与上述 (2) 的证明类似可知, 图  $H$  的  $(x, y)$  分离数  $s_H(x, y) = s(x, y) - 1$ , 故  $H$  中存在  $s(x, y) - 1$  条两两内部无公共顶点的  $(x, y)$  路. 显然它们和路  $xuy$  一起形成  $G$  的  $s(x, y)$  条两两内部无公共顶点的  $(x, y)$  路. 这与  $G$  的取法矛盾.

(4) 记  $k = s(x, y)$ . 如果  $W = \{w_1, w_2, \dots, w_k\}$  是  $G$  的一个最小  $(x, y)$  分离集, 则  $W \subseteq N_G(x)$  或者  $W \subseteq N_G(y)$ .

事实上, 设  $G_x$ 、 $G_y$  为图  $G - W$  中分别包含  $x$  和  $y$  的连通分支, 用  $G_1$  表示由  $V(G_x) \cup W$  在  $G$  中导出的子图,  $G_2$  表示由  $V(G_y) \cup W$  在  $G$  中导出的子图. 显然  $V(G_1) \cap V(G_2) = W$ , 且  $\nu(G_1) \geq k + 1$ ,  $\nu(G_2) \geq k + 1$ . 如果  $\nu(G_1) = k + 1$ , 则  $V(G_x) = \{x\}$ , 此时因  $W$  是  $G$  的最小  $(x, y)$  分离集, 故  $x$  与  $W$  中每个点都相邻, 结论显然成立. 同理,  $\nu(G_2) = k + 1$  时结论成立.

下面用反证法证明不可能  $\nu(G_1)$  和  $\nu(G_2)$  都大于  $k + 1$ . 假设  $\nu(G_1) \geq k + 2$ , 且  $\nu(G_2) \geq k + 2$ . 给  $G_1$  添加一个新顶点  $v^*$ , 将它与  $G_1$  中属于  $W$  的每个顶点都连新边, 这样获得的图记为  $G_1^*$ . 若  $G_1^*$  中存在  $(x, v^*)$  分离集  $S$  满足  $|S| < k$ , 则易见  $S$  也是  $G$  的  $(x, y)$  分离集, 这与  $G$  的最小  $(x, y)$  分离集  $W$  含  $k$  个点矛盾. 由此可知,  $G_1^*$  的  $(x, v^*)$  分离数为  $k$ . 因  $G_1^*$  的阶小于  $G$  的阶, 由  $G$  的最小性,  $G_1^*$  中任二点的分离数等于  $G_1^*$  中该两点间两两内部无公共顶点的路的最大条数. 因此  $G_1^*$  中有  $k$  条两两内部无公共顶点的  $(x, v^*)$  路. 由  $G_1^*$  的结构可知, 这  $k$  条路从  $x$  出发分别连到  $W$  的不同顶点. 同理可知, 从  $y$  出发也有  $k$  条两两内部无公共顶点的路分别连到  $W$  的不同顶点. 这些路连接成  $G$  中  $k$  条两两内部无公共顶点的  $(x, y)$  路, 这与  $G$  的取法矛盾.

现在继续完成定理的证明.

设  $P$  是  $G$  中一条最短的  $(x, y)$  路. 由结论 (3),  $P$  的长至少为 3. 设  $P = xv_1v_2 \cdots v_t y$ . 按照结论 (2), 存在  $G$  的顶点子集  $U$ , 满足  $|U| = s(x, y) - 1$  且  $U \cup \{v_1\}$  和  $U \cup \{v_2\}$  都是  $G$  的  $(x, y)$  最小分离集. 由结论 (1),  $s(x, y) > 1$ , 故  $U \neq \emptyset$ . 由结论 (4),  $U \cup \{v_1\} \subseteq N_G(x)$  或者  $U \cup \{v_1\} \subseteq N_G(y)$ , 但由于  $xv_1 \in E(G)$  及结论 (3), 可知  $U \subseteq N_G(x)$ , 且  $U \not\subseteq N_G(y)$ ; 同样, 由结论 (4),  $U \cup \{v_2\} \subseteq N_G(x)$  或者  $U \cup \{v_2\} \subseteq N_G(y)$ , 但上面已知  $U \subseteq N_G(x)$ , 由结论 (3) 可知必定  $U \cup \{v_2\} \subseteq N_G(x)$ , 从而  $v_2 \in N_G(x)$ . 这与  $P$  是  $G$  的最短  $(x, y)$  路矛盾. 证毕.

由 Menger 定理可得到 Whitney 定理的如下推广

**推论 2.4.1**  $\nu \geq k+1$  的图  $G$  是  $k$ -连通图当且仅当  $G$  中任二顶点至少被  $k$  条两两内部无公共顶点的路所连.

**证明 必要性** 用反证法. 设  $G$  是  $k$ -连通图. 假定  $G$  中存在两个顶点  $x, y$ , 它们之间两两内部无公共顶点的路最多只有  $m$  条,  $m < k$ . 分两种情况考虑.

(1) 如果  $xy \notin E(G)$ , 则因  $G$  中最小点割集的点数不超过  $G$  中最小  $(x, y)$  分离集的点数, 由 Menger 定理,  $\kappa(G) \leq s(x, y) = m < k$ . 这与  $G$  是  $k$ -连通图矛盾.

(2) 如果  $xy \in E(G)$ , 则  $H = G - xy$  中两两内部无公共顶点的  $(x, y)$  路最多只有  $m - 1 < k - 1$  条. 由 Menger 定理,  $\kappa(G - xy) \leq s_H(x, y) = m - 1 < k - 1$ . 因此, 存在  $G - xy$  的一个顶点子集  $U$ , 使得  $|U| \leq k - 2$ , 且  $(G - xy) - U$  不连通. 故图  $G - (U \cup \{x\})$  和  $G - (U \cup \{y\})$  至少有一个是不连通图, 这表明  $G$  有不超过  $k - 1$  个点的点割集, 与  $G$  是  $k$  连通图矛盾.

**充分性** 用反证法. 设  $G$  中任二顶点至少被  $k$  条两两内部无公共顶点的路所连. 假如  $G$  不是  $k$ -连通图, 则  $\kappa(G) < k$ . 设  $W$  是  $G$  的一个最小点割集, 则  $|W| = \kappa(G) < k$ . 从  $G - W$  的两个不同连通分支中各取一个点  $x$  和  $y$ . 由充分性的前提条件,  $x$  和  $y$  在  $G$  中至少被  $k$  条两两内部无公共顶点的路所连, 即  $r(x, y) \geq k$ . 由 Menger 定理,  $G$  中  $(x, y)$  分离数  $s(x, y) = r(x, y) \geq k$ , 但因  $|W| < k$ , 故  $x, y$  在  $G - W$  中仍连通, 这与  $x, y$  的取法矛盾. 证毕.

关于边连通, 有与 Menger 定理类似的结果, 称为边形式的 Menger 定理. 它的证明与 Menger 定理类似.

设  $G$  是一个简单图,  $x, y$  是  $G$  中任二不同顶点. 如果从  $G$  中删去一组边后不再有  $(x, y)$  路, 则称这组边分离  $x$  和  $y$ , 并将这组边称为一个  $(x, y)$  边分离集.  $G$  中含边数最少的  $(x, y)$  边分离集称为最小  $(x, y)$  边分离集, 其边数称为  $G$  的  $(x, y)$  边分离数, 记为  $s'(x, y)$ . 此外, 将  $G$  中两两无公共边的  $(x, y)$  路的最大条数记为  $r'(x, y)$ .

**定理 2.4.2 (边形式 Menger 定理)** 设  $x, y$  是图  $G$  中两个不相邻顶点, 则  $G$  中分离  $x, y$  所需的最少边数等于  $G$  中两两无公共边的  $(x, y)$  路的最大条数, 即  $s'(x, y) = r'(x, y)$ .

利用边形式 Menger 定理能够给出下列与推论 2.4.1 类似的结果, 其证明与推论 2.4.1 类似.

**推论 2.4.2** 图  $G$  是  $k$ -边连通图当且仅当  $G$  中任二顶点至少被  $k$ -条两两无公共边的路所连.

关于求图中内部无公共顶点的路和无公共边的路的算法以及相关问题的推广和应用, 读者可参看文献 [1]~[7].

## §2.5 可靠通信网络的设计

**可靠网络设计问题:** 给定赋权图  $G$  和正整数  $m$ , 求  $G$  的具有最小权的  $m$ - 连通生成子图.

当  $m = 1$  时, 就是求最小生成树问题.

当  $m > 1$  时, 问题尚未解决.

当  $G = K_n$  且每边的权皆为 1 时, 问题成为: 求  $K_n$  中边数最少的  $m$ - 连通生成子图. 这一问题于 1962 年由 Harary 所解决. 下面来描述这种生成子图的构造.

令  $f(m, n)$  表示有  $n$  个顶点的  $m$ - 连通图所能具有的最少边数 ( $m < n$ ). 设  $G$  是一个具有这种性质且有  $f(m, n)$  条边的图. 因  $2\varepsilon = \sum_{v \in V(G)} d(v)$ , 且  $\kappa \leq \kappa' \leq \delta$ . 故

$$f(m, n) = \frac{1}{2} \sum_{v \in V(G)} d(v) \geq \frac{n\delta}{2} \geq \frac{mn}{2}.$$

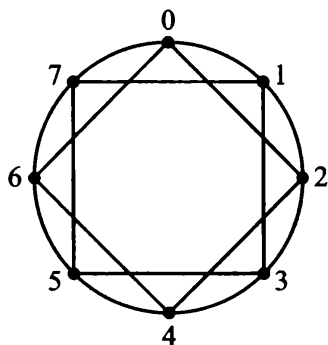
Harary 找到了具有  $n$  个顶点、 $\lceil \frac{mn}{2} \rceil$  条边的  $m$ - 连通图. 这种图记为  $H_{m,n}$ , 构造如下:

记  $H_{m,n}$  的顶点集为  $\{0, 1, 2, \dots, n-1\}$ .

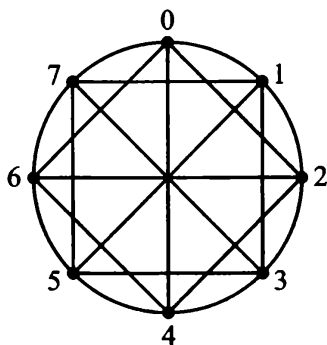
(1) 若  $m$  为偶数, 设  $m = 2r$ . 当且仅当  $0 \leq j - i + r \leq 2r \pmod{n}$  时, 顶点  $i$  与  $j$  连边;

(2) 若  $m$  为奇数 (设  $m = 2r + 1$ ), 而  $n$  是偶数, 则先构作  $H_{2r,n}$ , 然后在顶点  $i$  与  $i + \frac{n}{2}$  间加上一条边 ( $i = 1, 2, \dots, \frac{n}{2}$ );

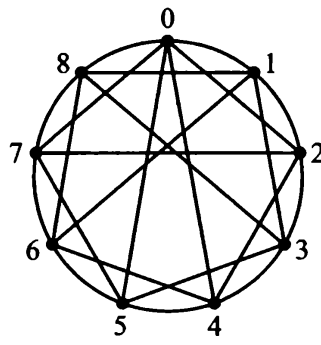
(3) 若  $m$  为奇数 (设  $m = 2r + 1$ ), 且  $n$  也是奇数, 则先构作  $H_{2r,n}$ , 然后在顶点  $i$  与  $i + \frac{n+1}{2}$  间加上一条边 ( $i = 1, 2, \dots, \frac{n-1}{2}$ ), 再在顶点  $0$  与  $\frac{n-1}{2}$ 、 $0$  与  $\frac{n+1}{2}$  之间各加上一边.



$H_{4,8}(m=4, r=2, n=8)$



$H_{5,8}(m=5, r=2, n=8)$



$H_{5,9}(m=5, r=2, n=9)$

**定理 2.5.1**  $H_{m,n}$  具有  $\lceil \frac{mn}{2} \rceil$  条边, 且是  $m$ - 连通的.

**证明** 在情形 (1) 中, 按连边规则, 每个顶点  $i$  恰与  $r$  个顶点  $i+1, i+2, \dots, i+r \pmod{n}$  连边 ( $i = 0, 1, 2, \dots, n-1$ ), 故共连出  $n \cdot r$  条边, 这恰等于  $\lceil \frac{mn}{2} \rceil$ . 在情形 (2) 中, 除了按情形 (1) 连出  $n \cdot r$  条边外, 另外又添加了  $\frac{n}{2}$  条边, 故共有  $n \cdot r + \frac{n}{2} = \frac{n(2r+1)}{2} = \frac{mn}{2}$  条边. 在情形 (3) 中, 除了按情形 (1) 连出  $n \cdot r$  条边外, 另外又添加了  $\left(\frac{n-1}{2} - 1\right) + 2 = \frac{n+1}{2}$  条边, 故共有  $n \cdot r + \frac{n+1}{2} = \frac{n(2r+1)+1}{2} = \frac{mn+1}{2} = \lceil \frac{mn}{2} \rceil$  条边. 总之,  $H_{m,n}$  具有  $\lceil \frac{mn}{2} \rceil$  条边.

下面证明  $H_{m,n}$  是  $m$ -连通的. 只证  $m = 2r$  的情况,  $m = 2r+1$  的情况类似可证. 我们用反证法来证明  $H_{m,n}$  中没有少于  $m = 2r$  个顶点的点割集.

若不然, 设  $V'$  是一个点割集且  $|V'| < 2r$ . 又设  $i$  与  $j$  是  $H_{2r,n} - V'$  中属于不同连通分支的两个顶点. 将  $H_{m,n}$  的所有顶点依次排成一圈, 以  $i$  和  $j$  作为分界点, 将所有顶点依次分为两个集合:

$$S = \{i, i+1, \dots, j-1, j\} \text{ 和 } T = \{j, j+1, \dots, i-1, i\}.$$

这里加法取模  $n$ . 因  $|V'| < 2r$  且  $i, j \notin V'$ , 故要么  $|S \cap V'| < r$ , 要么  $|T \cap V'| < r$ . 不妨设  $|S \cap V'| < r$ . 则从有序顶点集合  $S$  中减去  $V'$  至多删去  $r-1$  个点, 因此在  $S - V'$  中必存在一个始于  $i$  而终于  $j$  的顶点序列, 其任二相继项之差  $\leq r$ . 但由  $H_{2r,n}$  的构成 (1), 这样的序列中相邻项之间存在边. 因此这个序列构成了  $H_{2r,n} - V'$  中一条  $(i, j)$  路. 这与  $i, j$  属于不同连通分支矛盾. 证毕.

**推论 2.5.1**  $f(m, n) = \lceil \frac{mn}{2} \rceil$ .

**证明** 由上一定理,  $H_{m,n}$  是  $m$ -连通的, 且  $\varepsilon(H_{m,n}) = \lceil \frac{mn}{2} \rceil$ . 故

$$f(m, n) \leq \varepsilon(H_{m,n}) = \lceil \frac{mn}{2} \rceil.$$

另一方面, 前已述及  $f(m, n) \geq \lceil \frac{mn}{2} \rceil$ . 故  $f(m, n) = \lceil \frac{mn}{2} \rceil$ . 证毕.

**注** 因  $\kappa \leq \kappa'$ , 故  $H_{m,n}$  也是  $m$  边连通的. 若  $g(m, n)$  表示  $n$  个顶点的  $m$  边连通图可能的最少边数, 则与以上类似可证  $g(m, n) = \lceil \frac{mn}{2} \rceil$ .

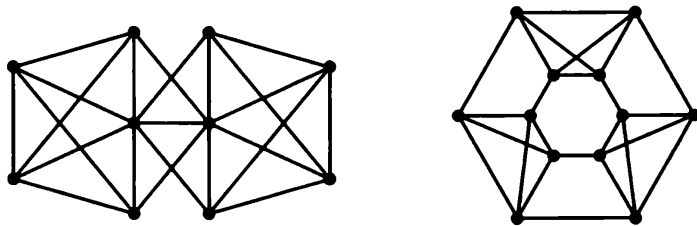
关于图的连通性和边连通性的更多内容可参看文献 [8]~[67], 其中 [15] 是连通性方面一篇较好的综述; 文献 [16]~[20] 主要研究图的边连通性, [21]~[24] 是边连通性的算法方面的文献; 文献 [25]~[45] 是图的点连通性的有关研究进展, 文献 [46]~[60] 与点连通性的算法有关; [61]~[68] 则主要关注动态网络中边连通性和点连通性的在线维持.

## 习题二

2.1 证明: 若  $G$  是连通简单图但不是完全图, 则  $G$  中必存在三个顶点  $u, v$  和  $w$ , 使

得  $uv, vw \in E(G)$  而  $uw \notin E(G)$ .

- 2.2 证明: 点  $v$  是图  $G$  的割点当且仅当  $v$  至少属于  $G$  的两个不同的块.
- 2.3 证明或反驳: 如果直径为 2 的简单图有一个割点, 则其补图有一个孤立顶点.
- 2.4 证明: 若一个图  $G$  至少有一条非环边, 则  $G$  至少有两个非割点.
- 2.5 证明: 若  $v$  是简单图  $G$  的一个割点, 则  $\bar{G} - v$  是连通的.
- 2.6 设  $G$  是自补图, 证明:  $G$  有割点当且仅当  $G$  有 1 度点.
- 2.7 证明: 恰有两个顶点不是割点的连通简单图是一条路.
- 2.8 (1) 举反例说明如下命题不成立: 如果  $e$  是图  $G$  的一条割边, 则  $e$  至少有一个端点是  $G$  的割点.  
(2) 试添加一个假设条件使其成立, 并证明所得命题的正确性.
- 2.9 (1) 证明: 顶点度全为偶数的图没有割边.  
(2) 对任意  $k \geq 1$ , 构造具有一条割边的  $2k + 1$  正则简单图.
- 2.10 证明: 当  $k \geq 2$  时,  $k$ - 正则二部图没有割边.
- 2.11 设图  $G$  连通且  $e \in E(G)$ . 证明:  $e$  在  $G$  的每棵生成树中当且仅当  $e$  是  $G$  的割边.
- 2.12 设  $G$  是一个图. 证明:  $G$  是一棵树当且仅当它是连通的且其每条边都是割边.
- 2.13 令  $C$  是赋权图  $G$  中的一个圈, 设  $e$  是  $C$  上权值最大的边, 证明:  $G$  中存在一棵不包含  $e$  的最小生成树. 并由此证明: 不断删除图中权值最大的非割边, 直到剩下的图没有圈为止, 最后将得到一棵最小权生成树.
- 2.14 设  $G$  是一个直径为 2 的图,  $S$  是  $G$  的一个最小边割集. 证明:  $G - S$  的连通分支至少有一个同构于  $K_1$  或  $K_{\delta(G)}$ .
- 2.15 设  $A, B$  是两个集合, 集合  $(A - B) \cup (B - A)$  称为  $A$  与  $B$  的对称差, 记为  $A \oplus B$ . 证明: (1)  $A \oplus B = (A \cup B) - (A \cap B)$ ;  
(2) 图  $G$  的两个极小边割集的对称差仍是  $G$  的边割集.
- 2.16 证明: 对任何图  $G$ , 有  $\kappa'(G) \leq \left\lfloor \frac{2\varepsilon}{\nu} \right\rfloor$ .
- 2.17 证明或否定: 如果一个简单图  $G$  的最大度是  $\left\lfloor \frac{\nu}{2} \right\rfloor$ , 最小度是  $\left\lfloor \frac{\nu}{2} \right\rfloor - 1$ , 则  $G$  是连通的.
- 2.18 (1) 证明: 若  $G$  是简单图且  $\delta(G) \geq \nu - 2$ , 则  $\kappa(G) = \delta(G)$ ;  
(2) 找出一个简单图, 使得  $\delta(G) = \nu - 3$  且  $\kappa(G) < \delta(G)$ .
- 2.19 对下面的每个图  $G$ , 确定其连通度、边连通度和最小顶点度.



- 2.20 构造一个连通度为 1 的 3-正则简单图.
- 2.21 证明: 对图  $G$  的任一条边  $e$ ,  $\kappa(G - e) \geq \kappa(G) - 1$ .
- 2.22 证明: 若  $G$  是简单图且  $\Delta(G) \leq 3$ , 则  $\kappa(G) = \kappa'(G)$ .
- 2.23 设  $H$  是连通图  $G$  的子图. 举例说明: 有可能  $\kappa(H) > \kappa(G)$ .
- 2.24 设  $G$  是度至少为 2 的正则二部图, 证明:  $\kappa(G) \neq 1$ .
- 2.25 一个单圈图是指仅含有一个圈的图. 证明: 如果  $G$  是单圈图, 则  $\kappa(G) \leq 2$  且  $\kappa'(G) \leq 2$ .
- 2.26 证明: 图  $G$  是  $k$ -连通的当且仅当  $G \vee K_n$  是  $(k+n)$ -连通的. 这里  $G \vee K_n$  表示将  $G$  的每个顶点与完全图  $K_n$  的所有顶点都连边构成的图.
- 2.27 证明或反驳: (1) 任何  $k$ -连通图都是  $k$ -边连通图;  
(2) 任何  $k$ -边连通图都是  $k$ -连通图.
- 2.28 (1) 设图  $G$  是  $k$ -边连通的, 且  $k > 0$ , 证明: 若  $E'$  是  $G$  的  $k$  条边的集合, 则  $G - E'$  的连通分支数  $w(G - E') \leq 2$ ;  
(2) 对  $k > 0$ , 找出一个  $k$ -连通图  $G$  及  $G$  的  $k$  个顶点的子集  $V'$ , 使得  $G - V'$  的连通分支  $w(G - V') > 2$ .
- 2.29 证明: 若  $G$  是  $k$ -边连通的, 则  $\varepsilon \geq \frac{k\nu}{2}$ .
- 2.30 设  $G$  是连通图, 且对其每条边  $e$ , 都存在两个包含  $e$  的圈  $C_1$  和  $C_2$  使得二者的唯一公共边为  $e$ . 证明  $G$  是 3-边连通的. 由此证明 Peterson 图是 3-边连通的.
- 2.31 证明: 连通图  $G$  是  $k$ -边连通的当且仅当它的每个块都是  $k$ -边连通的.
- 2.32 证明: 若简单图  $G$  无偶圈, 则  $G$  的每个块要么是  $K_2$  要么是奇圈.
- 2.33 证明: 对不是块的连通图, 至少有两个块, 它们每个恰含有  $G$  的一个割点.
- 2.34 对非平凡连通图  $G$ , 定义  $G$  的块-割点图  $B(G)$  为:  $B(G)$  的顶点集由  $G$  的所有块和割点组成,  $B(G)$  中两个顶点  $x$  和  $y$  连边当且仅当  $x$  和  $y$  一个是  $G$  的割点而另一个是  $G$  的包含该割点的块. 证明  $B(G)$  是一棵树.
- 2.35 证明: 若简单图  $G$  是一个  $\delta(G) \geq 3$  的块, 则  $G$  中必存在顶点  $v$ , 使得  $G - v$  也是块.
- 2.36 设  $G$  是连通图并且其所有的块为  $B_1, B_2, \dots, B_k$ , 证明  $G$  恰有  $\left( \sum_{i=1}^k \nu(B_i) \right) - k + 1$  个顶点.

- 2.37 设  $G$  是至少含有 3 个顶点的连通图, 在  $G$  中距离为 2 的顶点对间全部添加新边, 所得之图记为  $G'$ . 证明  $G'$  是 2-连通的.
- 2.38 设  $G$  是一个连通图. 试给出一个公式, 用  $G$  的块的生成树数目表示  $G$  的生成树的数目.
- 2.39 设  $G$  是一个块,  $x$  和  $y$  是  $G$  的不同顶点. 如果  $P$  是  $G$  中一条  $(x, y)$  路, 问  $G$  中是否一定存在一条与  $P$  内部无公共顶点的  $(x, y)$  路?
- 2.40 证明: 图  $G$  是 2-边连通的当且仅当  $G$  的任二顶点至少由两条无公共边的路所连.
- 2.41 设  $G$  是一个  $k$ -连通图,  $v_1, v_2, \dots, v_k$  是  $G$  中  $k$  个不同的顶点. 给  $G$  添加一个新顶点  $w$ , 并将  $w$  到上述  $k$  个顶点都连上边, 这样对  $G$  修改之后所得之图记为  $G'$ . 证明  $\kappa(G') = k$ .
- 2.42 设  $G$  是  $k$ -连通图, 且联图  $H = G \vee K_1$  (由完全图  $K_1$  的顶点向  $G$  的所有顶点连边构成之图), 证明  $H$  是  $(k+1)$ -连通的.
- 2.43 设  $G$  是一个  $k$ -连通图且  $v$  是  $G$  的一个顶点. 对任意正整数  $m$ , 定义  $G_m$  是由  $G$  添加  $m$  个新顶点  $w_1, w_2, \dots, w_m$  和所有形如  $w_i u$ , ( $1 \leq i \leq m, u \in N_G(v)$ ) 的边后所得之图. 证明  $G_m$  是  $k$ -连通的.
- 2.44 证明或否定: 如果  $G$  是一个  $k$ -边连通图且  $v, v_1, v_2, \dots, v_k$  是  $G$  的  $k+1$  个不同顶点, 那么对每个  $i = 1, 2, \dots, k$ , 都存在  $(v, v_i)$  路  $P_i$ ,  $P_i$  恰好包含  $\{v_1, v_2, \dots, v_k\}$  中一个顶点  $v_i$ , 且当  $i \neq j$  时,  $P_i$  和  $P_j$  无公共边.
- 2.45 设图  $G$  满足:  $\kappa(G) = 3$ , 且存在不相邻的顶点  $x$  和  $y$  使得  $G$  的  $(x, y)$  分离数为 4. 问:  
(1) 在  $G$  中内部无公共顶点的  $(x, y)$  路的最大数目是多少?  
(2) 给出一个满足上述性质的图的例子.
- 2.46 设图  $G$  的连通度  $\kappa(G) = k$ , 证明对  $G$  中任何一个顶点  $v, \kappa(G - v) \geq k - 1$ .

## 参考文献

- [1] N. Robertson, and P. D. Seymour, Outline of a disjoint paths algorithm, in *Paths, Flows and VLSI-Layout* (B. Korte, L. Lovasz, H. J. Promel, and A. Schrijver editors), Springer-Verlag, Berlin, 1990.
- [2] J. M. Kleinberg, Approximation algorithms for disjoint paths problems, *Ph. D thesis, MIT, Cambridge, MA, May 1996*.
- [3] A. Srinivasan, Improved approximations for edge-disjoint paths, unsplittable flow and related routing problems, in *Proceedings of the 38<sup>th</sup> Annual Symposium on the Foundations*



- of Computer Science*, 1997, 416-425.
- [4] S. G. Kolliopoulos, and C. Stein, Approximating disjoint-path problems using greedy algorithms and packing integer programs, in *Proceedings of 6<sup>th</sup> International IPCO Conference*, Houston, Texas, 1998.
- [5] T. Erlebach and K. Jansen, The maximum edge-disjoint paths problem in bidirected trees, *SIAM J. on Discrete Mathematics*, 14(2001), 326-355.
- [6] D. Huygens, A. R. Mahjoub, and P. Pesneau, Two edge-disjoint hop-constrained paths and polyhedra, *SIAM J. on Discrete Mathematics*, 18(2004-2005), 287-312.
- [7] P. D. Seymour, Disjoint paths in graphs. *Discrete Mathematics*, 306(2006), 979-991.
- [8] T. Gomes, J. Craveirinha, Efficient calculation of the most reliable pair of link disjoint paths in telecommunication networks, *European Journal on Operational Research*, 181(2007), 1055-1064.
- [9] J. A. Bondy, and U. S. R. Murty, *Graph Theory with Applications*, The Macmillan Press LTD. 1976. (中译本: 吴望名、李念祖、吴兰芳、谢伟如、梁文沛译, 图论及其应用, 北京: 科学出版社, 1984).
- [10] 蒋长浩, 图论与网络流, 北京: 中国林业出版社, 2001.
- [11] D. Reinhard, *Graph Theory* (second edition), Springer-Verlag, New York, (2000)43-67. (影印版: 图论, 世界图书出版公司, 2003).
- [12] W. T. Tutte, *Graph Theory*, Cambridge University Press, (2001) 32-114. (影印版: 图论, 机械工业出版社, 2004).
- [13] 张先迪, 李正良. 图论及其应用. 北京: 高等教育出版社, 2005.
- [14] D. B. West, *Introduction to Graph Theory* (second edition), Prentice-Hall, Inc. , 2001.(中译本: 图论导引. 李建中、骆吉周译. 北京: 机械工业出版社, 2006).
- [15] A. R. Vetta, Graph Connectivity: Relaxations and Algorithms, *Ph. D Thesis, Department of Mathematics, MIT*, 2002.
- [16] W. Mader, Connectivity and edge-connectivity in finite graphs, in *Surveys in Combinatorics, London Mathematical Society Lecture Note Series 38*, Cambridge University Press, 38(1979) 66-95.
- [17] J. Bang-Jensen, H. N. Gabow, T. Jordán, and Z. Szigeti, Edge-Connectivity Augmentation with Partition Constraints, *SIAM J. on Discrete Mathematics*, 12(1999), 160-207.
- [18] T. Jordán, On minimally  $k$ -edge-connected graphs and shortest  $k$ -edge-connected Steiner networks, *Discrete Applied Mathematics*, 131(2003), 421-432.
- [19] A. Frank, Augmenting graphs to meet edge-connectivity requirements, *SIAM J. Discrete Mathematics*, 5(1992), 25-53.
- [20] R. Carr and R. Ravi, "A new bound for the 2-edge connected subgraph problem", *Integer programming and combinatorial optimization*, Springer-Verlag, Lecture Notes in Computer Science, 1412, 1998, pp112-125.

- [21] H. Nagamochi, An approximation for finding a smallest 2-edge-connected subgraph containing a specified spanning tree, *Discrete Applied Mathematics*, 126(2003), 83-113.
- [22] J. Cheriyan, A. Sebö and Z. Sigei, “Improving on the 1.5-approximation of a smallest 2-edge connected spanning subgraph”, *SIAM J. Discrete Mathematics*, 14(2001), 170-180.
- [23] D. W. Matula, Determine edge connectivity in  $O(mn)$ , *Proceedings of the 28<sup>th</sup> Annual Symposium on the Foundations of Computer Science*, 1987, 249-251.
- [24] H. N. Gabow, An Improved Analysis for Approximating the smallest  $k$ -edge connected spanning subgraph of a multigraph, *SIAM J. on Discrete Mathematics*, 19(2005), pp. 1-18.
- [25] W. Mader, “On  $k$ -critically  $n$ -connected graphs”, in J. Bondy and K. Murty (eds.), *Progress in Graph Theory*, Academic Press, Toronto, 1984, pp389-398.
- [26] C. Monma, B. Munsun and W. Pulleyblank, “Minimum-weight two-connected spanning networks”, *Math. Programming*, (Series A), 46(2)(1990), 153-171.
- [27] Frank, “Connectivity augmentation problems in network design”, in J. Birge and K. Murty (eds. ), *Mathematical Programming: State of the Art 1994*, University of Michigan, Ann Arbor, 1994, pp34-63.
- [28] J. Cheriyan, T. Jordan and Z. Nutov, “On rooted node-connectivity problems”, *Algorithmica*, 30(2001), 353-375.
- [29] H. Nagamochi and T. Ibaraki, Graph connectivity and its augmentation: applications of MA orderings, *Discrete Applied Mathematics*, 123(2002), 447-472.
- [30] P. Dankelmann and O. R. Oellermann, Bounds on the average connectivity of a graph, *Discrete Applied Mathematics*, 129 (2003), 305-318.
- [31] E. Oh and J. Chen, On strong Menger-connectivity of star graphs, *Discrete Applied Mathematics*, 129 (2003), 499-511.
- [32] N. Lichiardopol, Broadcast time and connectivity, *Discrete Applied Mathematics*, 143 (2004), 359-363.
- [33] M. Katz, N. A. Katz, A. Korman, and D. Peleg, Labeling schemes for flow and connectivity, *SIAM J. on Computing*, 34(2004-2005), 23-40.
- [34] M. Kriesell, A constructive characterization of 3-connected triangle-free graphs, *Journal of Combinatorial Theory*, (Ser. B), 97(2007), 358-370.
- [35] A. Hellwig, L. Volkmann, On connectivity in graphs with given clique number, *Journal of Graph Theory*, 52(2006), 7-14.
- [36] M. J. Stein, Forcing highly connected subgraphs, *Journal of Graph Theory*, 54(2007), 331-349.
- [37] Zemin Jin, Xingxing Yu, Xiaoyan Zhang, Contractible subgraphs in  $k$ -connected graphs, *Journal of Graph Theory*, 55(2007), 121-136.

- [38] N. Bougard, G. Joret, Turán's theorem and  $k$ -connected graphs, *Journal of Graph Theory*, 58(2008), 1-13.
- [39] L. S. Chandran and L. S. Ram, On the number of minimum cuts in a graph, *SIAM Journal on Discrete Mathematics*, 18(2004-2005), 177-194.
- [40] C. Balbuena, M. Cera, A. Diáñez, P. García-Vázquez, X. Marcote, Connectivity of graphs with given girth pair. *Discrete Mathematics*, 307(2007), 155-162.
- [41] C. Balbuena, M. Cera, A. Diáñez, P. García-Vázquez, X. Marcote: On the restricted connectivity and superconnectivity in graphs with given girth. *Discrete Mathematics*, 307(2007), 659-667.
- [42] L. Cereceda, J. van den Heuvel, M. Johnson, Connectedness of the graph of vertex-colourings. *Discrete Mathematics*, 308(2008), 913-919.
- [43] C.-K. Lin, H.-M. Huang, J. J. M. Tan, L.-H. Hsu, On spanning connected graphs. *Discrete Mathematics*, 308(2008), 1330-1333.
- [44] J. C. Bermond, N. Homobono and C. Peyrat, Large fault-tolerant interconnection networks, *Graphs and Combinatorics*, Springer, 5(1989), 107-123.
- [45] D. F. Hsu, On container width and length in graphs, groups and networks, *IEICE Trans. Fundam*, 1994, E(77A), 668-680.
- [46] N. Garg, A. Singla and S. Vempala, "Improved approximations for biconnected subgraphs via better lower bounding techniques", *Proc. 6<sup>th</sup> Annual SODA*, 1993, pp103-111.
- [47] S. Khuller and B. Raghavachari, "Improved approximation algorithms for uniform connectivity problems", *Journal of Algorithms*, 21(1996), 434-450.
- [48] C. Fernandes, "A better approximation ratio for the minimum  $k$ -edge-connected spanning subgraph problem", *Proc. 8<sup>th</sup> Annual SODA*, 1997, pp629-638.
- [49] R. Ravi and D. Williamson, "An approximation algorithm for minimum-cost vertex-connectivity problems", *Algorithmica*, 18(1997), 21-43.
- [50] V. Auletta, Y. Dinitz, Z. Nutov and D. Parente, "A 2-approximation algorithm for finding an optimum 3-vertex-connected spanning subgraph", *Journal of algorithms*, 32(1999), 21-30.
- [51] Czumaj and A. Lingas, "On approximability of the minimum-cost  $k$ -connected spanning subgraph problem", *Proc. 10<sup>th</sup> Annual SODA*, 1999, pp281-290.
- [52] K. Jain, I. Mandoiu, V. Vazirani and D. Williamson, "A primal-dual schema based approximation algorithm for the element connectivity problem", *Proc. 10th Annual SODA*, 1999, pp484-489.
- [53] G. Kortsarz and Z. Nutov, "Approximating node connectivity problems via setcovers", in K. Jansen and S. Khuller (eds. ), *Approximation Algorithms for combinatorial Optimization*, Springer-Verlag, Lecture Notes in Computer Science, Vol. 1913, 2000, pp194-205.

- 
- [54] S. Vempala and A. Vetta, “Factor  $\frac{4}{3}$  Approximations for Minimum 2-Connected Subgraphs”, in K. Jansen and S. Khuller (eds. ), *Approximation Algorithms for Combinatorial Optimization*, Springer-Verlag, Lecture Notes in Computer Science, 2000, pp262-273.
- [55] H. Gabow, “Using expander graph to find vertex connectivity”, *Proc. 41st POCS*, 2000, pp410-420.
- [56] L. Fleischer, K. Jain, and D. Williamson, “ An iterative rounding 2-approximation algorithm for the element connectivity problem”, *Proc. 42<sup>nd</sup> IEEE FOCS*, 2001, pp339-347.
- [57] J. Cheriyan, S. Vempala, and A. Vetta, An approximation algorithm for the minimum-cost  $k$ -vertex connected subgraph, *SIAM Journal on Computing*, 32(2002-2003), 1050-1055.
- [58] G. Kortsarz, R. Krauthgamer, J. R. Lee, Hardness of approximation for vertex-connectivity network design problems, *SIAM J. on Computing*, 33(2003-2004), 704-720.
- [59] H. Koskinen, J. Karvo, O. Apilo, On improving connectivity of static ad hoc networks by adding nodes, *Proceedings of the 4<sup>th</sup> Annual Mediterranean Workshop on Ad Hoc Networks*, 2005, 169-178.
- [60] R. Gentilini, C. Piazza, A. Policriti, Symbolic graphs : linear solutions to connectivity related problems, *Algorithmica*, 50(1)(2008), 120-158.
- [61] G. N. Frederickson, Ambivalent data structures for dynamic 2-edge-connectivity and  $k$  smallest spanning trees, in *Proceedings of 32<sup>th</sup> Annual Symposium Foundation of Computer Science*, 1991, 632-641.
- [62] Z. Galil, G. F. Italiano, Fully dynamic algorithms for 2-edge-connectivity. *SIAM Journal on Computing*, 21(1992), 1047-1069.
- [63] Z. Galil, G. F. Italiano, Maintaining the 3-edge-connected components of a graph on-line. *SIAM Journal on Computing*, 22(1993), 11-28.
- [64] E. A. Dinitz, Maintaining the 4-edge-connected components of a graph on-line, in *Proceedings of the 2<sup>nd</sup> Israel Symposium Theory of Computing and Systems*, 1993, 88-99.
- [65] J. A. La Poutre, Maintenance of triconnected components of graphs. *Proc. 19<sup>th</sup> Int. Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science 623, Springer-Verlag, Berlin, 1992, pp354-365.
- [66] J. A. La Poutre, J. van Lecuwen, M. H. Overmars, Maintenance of 2- and 3-connected components of graphs, part I: 2- and 3-edge-connected components. *Discrete Mathematics*, 114(1993), 329-359.
- [67] Kanevsky, R. Tamassia, G. Di Battista, and J. Chen, On-line maintenance of the four-connected components of a graph, *Proceedings of 32<sup>nd</sup> Annual Symposium Foundation of Computer Science*, 1991, 793-801.
- [68] M. Rauch, Fully dynamic biconnectivity in graphs. *Proceedings of 33<sup>rd</sup> Annual Symposium Foundation of Computer Science*, 1992, 50-59.

# 第三章 匹配理论

## §3.1 匹配与最大匹配

**定义 3.1.1** 设  $G$  是一个图, 由  $G$  中一些不相邻的边组成的集合  $M$  称为  $G$  的一个**匹配** (matching). 对匹配  $M$  中的每条边  $e = uv$ , 其两 endpoints  $u$  和  $v$  称为被匹配  $M$  所匹配, 而  $u$  和  $v$  都称为是  $M$  **饱和的** (saturated vertex).

**注** (1) 更为确切地说,  $M$  是图  $G$  的一个匹配是指:  $M \subseteq E(G)$ , 且对  $\forall e_i, e_j \in M$ , 若  $e_i$  与  $e_j$  不相同, 则  $e_i$  与  $e_j$  在  $G$  中不相邻.

(2) 设  $M$  是图  $G$  的一个匹配, 则  $G$  中每个顶点要么未被  $M$  饱和, 要么仅被  $M$  中一条边饱和.

(3) 一个图的匹配一般不唯一. 特别地,  $G$  中每一条边构成  $G$  的一个匹配.

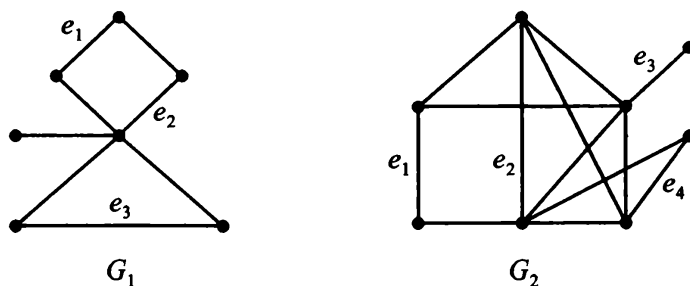
**定义 3.1.2** 图  $G$  的含边数最多的匹配称为  $G$  的**最大匹配** (maximum matching).

**注** (1) 若用  $|M|$  表示匹配  $M$  所含的边数, 则  $G$  的最大匹配  $M$  可更为确切地描述为:  $M$  是  $G$  的一个匹配, 且  $G$  中不存在匹配  $M'$  使得  $|M'| > |M|$ .

(2) 如果  $G$  中每个点都是  $M$  饱和的, 则称  $M$  是  $G$  的**完美匹配** (perfect matching).

(3) 任何图的完美匹配必是它的最大匹配.

例如, 在下图  $G_1$  中, 边集  $\{e_1\}$ 、 $\{e_1, e_2\}$ 、 $\{e_1, e_2, e_3\}$  都构成匹配, 其中  $\{e_1, e_2, e_3\}$  是  $G_1$  的一个最大匹配. 在  $G_2$  中, 边集  $\{e_1, e_2, e_3, e_4\}$  是一个完美匹配, 也是一个最大匹配.



**定义 3.1.3** 设  $M$  是图  $G$  的一个匹配,  $G$  的一条  $M$  交错路 (alternating path) 是指其边在  $M$  和  $E(G) - M$  中交替出现的路. 如果  $G$  的一条  $M$  交错路的起点和终点都是  $M$  非饱和的, 则称其为一条  $M$  可扩展路或  $M$  增广路 (augmenting path).

**定理 3.1.1 (Berge, 1957)** 图  $G$  的一个匹配  $M$  是最大匹配的充分必要条件是  $G$  中不存在  $M$  可扩展路.

**证明 必要性** 设  $M$  是  $G$  的一个最大匹配. 如果  $G$  中存在一个  $M$  可扩展路  $P$ , 则以  $P$  上所有不属于  $M$  的边构成集合  $M'$ . 显然  $M'$  也是  $G$  的一个匹配, 且比  $M$  多一条边. 这与  $M$  是最大匹配相矛盾.

**充分性** 设  $G$  中不存在  $M$  可扩展路. 若匹配  $M$  不是最大匹配, 则存在另一匹配  $M'$ , 使得  $|M'| > |M|$ . 令

$$H = G[M \oplus M'], \text{ (其中 } M \oplus M' = M \cup M' - M \cap M' \text{ 称为 } M \text{ 与 } M' \text{ 的对称差).}$$

即  $H$  是边集合  $M \oplus M'$  导出的子图 (如下图中所示), 则  $H$  中每个顶点的度非 1 即 2 (这是因为图  $H$  中一个顶点最多只与  $M$  的一条边及  $M'$  的一条边相关联). 故  $H$  的每个连通分支要么是  $M$  的边与  $M'$  的边交替出现的一个偶长度圈, 要么是  $M$  的边与  $M'$  的边交替出现的一条路 (习题 3.2). 由于  $|M'| > |M|$ ,  $H$  的边集  $M \oplus M'$  中  $M'$  的边多于  $M$  的边, 故必有  $H$  的某个连通分支是一条  $M$  的边与  $M'$  的边交替出现的路, 且始于  $M'$  的边又终止于  $M'$  的边. 明显地, 这条路是一条  $M$  交错路, 且两端点是  $M$  非饱和的, 因此是一条  $M$  可扩展路. 这与前提条件矛盾. 证毕.

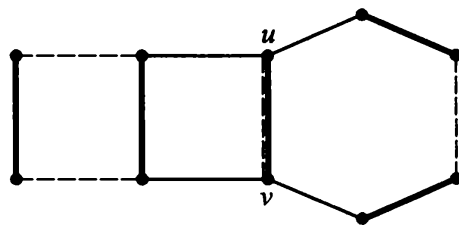


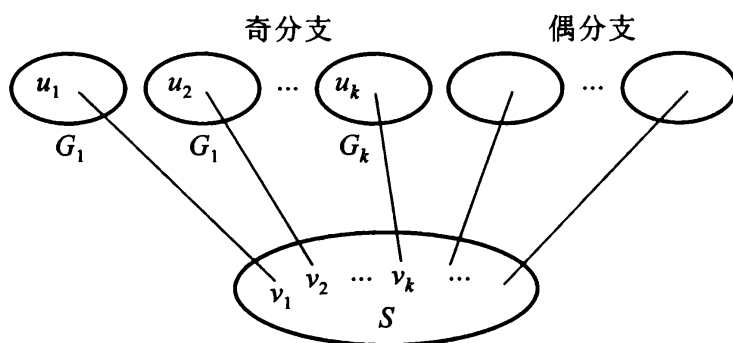
图  $G$  及其两个匹配: 虚线边属于匹配  $M$ , 粗线边属于匹配  $M'$ . 虚线边与粗线边 (不计  $u, v$  间的连边) 形成  $H = G[M \oplus M']$ .

## §3.2 完美匹配

**定义 3.2.1** 图  $G$  的含有奇数个顶点的连通分支称为  $G$  的奇分支.  $G$  的奇分支的个数用  $o(G)$  表示.

**定理 3.2.1 (Tutte, 1947)** 图  $G$  有完美匹配的充分必要条件是 对  $\forall S \subset V(G)$ ,  $o(G - S) \leq |S|$ . (这里  $|S|$  表示集合  $S$  含顶点的个数).

**证明 (Lovász, 1973) 必要性** 设图  $G$  有完美匹配  $M$ . 对  $\forall S \subset V(G)$ , 若  $G - S$  无奇分支, 则  $o(G - S) = 0$ , 结论显然成立; 否则, 设  $G_1, G_2, \dots, G_k$  是  $G - S$  的所有奇分支. 注意每个奇分支中至少有一个顶点在  $M$  下与  $S$  中的顶点匹配 (因  $G_i$  是奇分支,  $M$  是完美匹配), 设奇分支  $G_i$  中某个顶点  $u_i$  在  $M$  下与  $S$  中的某个顶点  $v_i$  匹配 ( $i = 1, 2, \dots, k$ ), 则  $v_1, v_2, \dots, v_k$  必互不相同. 故  $o(G - S) = k = |\{v_1, v_2, \dots, v_k\}| \leq |S|$  (如下图所示).



**充分性 (反证法)** 假设某图  $G$  满足: 对  $\forall S \subset V(G), o(G - S) \leq |S|$ , 但  $G$  没有完美匹配. 首先, 取  $S = \phi$ , 知  $o(G) = 0$ , 故  $V(G)$  是偶数. 现在, 给  $G$  添加边以获得一个边尽可能多的没有完美匹配的图  $G^*$  (称为边极大图). 因  $G$  是  $G^*$  的生成子图, 故对  $\forall S \subset V(G), G - S$  是  $G^* - S$  的生成子图, 从而

$$o(G^* - S) \leq o(G - S) \leq |S|. \quad (*)$$

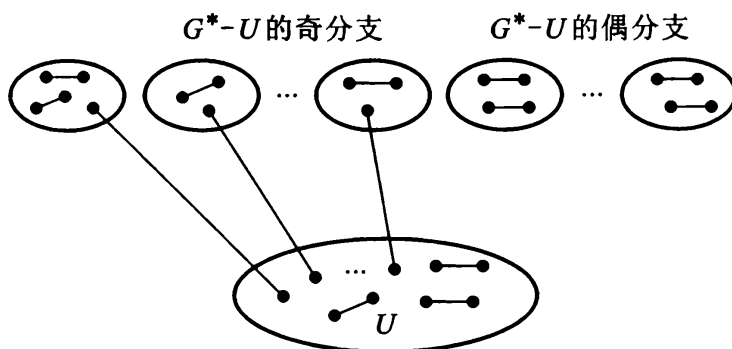
令

$$U = \{u \mid u \in V(G^*), d_{G^*}(u) = \nu - 1\}.$$

若  $U = V(G^*)$ , 则  $G^*$  是偶数阶完全图, 有完美匹配. 这与  $G^*$  的性质矛盾. 因此,  $U \neq V(G^*)$ , 可以证明, 此时  $G^* - U$  的每个连通分支都是完全图 (记为命题 A, 另证).

由 (\*) 式,  $o(G^* - U) \leq |U|$ , 即  $G^* - U$  的奇分支个数最多是  $|U|$ . 但这样一来,  $G^*$  就有一个完美匹配:

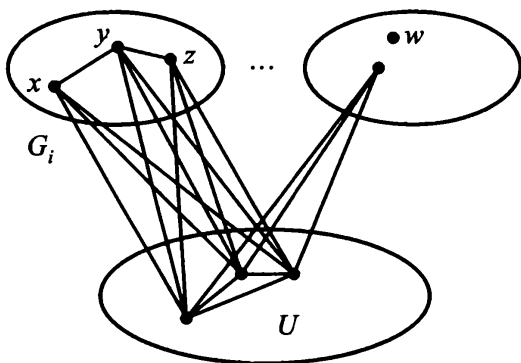
$G^* - U$  的各奇分支中的一个顶点和  $U$  的一个顶点配对;  $U$  中余下的顶点以及  $G^* - U$  的各分支中余下的顶点在本分支内配对 (由于各分支及  $U$  都是完全图), 如下图所示.



这与  $G^*$  无完美匹配矛盾. 因此, 反证法假设的情况不会发生, 充分性得证. 证毕.

**命题 A 的证明** 在上述充分性证明的条件下, 命题 A 可叙述为当  $U \neq V(G^*)$  时,  $G^* - U$  的每个连通分支都是完全图.

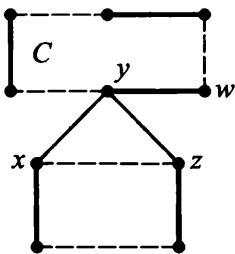
用反证法证明. 若不然, 设  $G^* - U$  中某个连通分支  $G_i$  不是完全图, 则  $|V(G_i)| \geq 3$ . 必存在  $x, y, z \in V(G_i)$ , 使得  $xy, yz \in E(G_i)$ , 且  $xz \notin E(G_i)$  (习题 2.1). 由于  $y \notin U$ , 故必有与  $y$  不相邻的顶点, 即必存在  $w \in V(G^* - U)$ , 使得  $yw \notin E(G^*)$ . 如下图所示 ( $w$  也可能位于  $G_i$  中). 由于  $G^*$  是不含完美匹配的边极大图, 所以  $G^* + xz$  和  $G^* + yw$



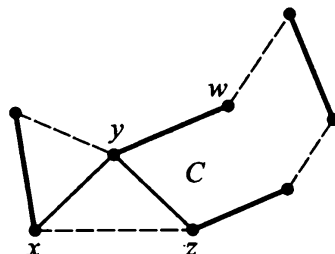
$G^*$ 及 $G^*-U$ 的连通分支结构示意图

都含有完美匹配, 分别设为  $M_1$  和  $M_2$  ( $M_1$  必含有边  $xz$ ,  $M_2$  必含有边  $yw$ ). 用  $H$  表示  $G^* \cup \{xz, yw\}$  中由对称差  $M_1 \oplus M_2$  导出的子图. 由于对  $\forall u \in V(H)$ ,  $d_H(u) = 2$  或  $0$  (由  $M_1$  和  $M_2$  都是完美匹配知), 故  $H$  的每个非平凡连通分支都是其边在  $M_1$  和  $M_2$  中交替出现的偶长度圈 (习题 3.2). 下分两种情形:

(1) 边  $xz$  和  $yw$  分别在  $H$  的不同分支中. 如下图情形 (1) 所示, 设  $yw$  在  $H$  的某个非平凡连通分支 (偶长度圈)  $C$  上, 则  $M_1$  在  $C$  上的边连同  $M_2$  不在  $C$  上的边构成  $G^*$  的一个完美匹配 (边  $xz$  和  $yw$  不在这个匹配中). 这与  $G^*$  的选择矛盾.



情形(1):  $xz$  和  $yw$  分别在  $H$  的不同分支中.  
虚线表示  $M_1$  的边, 粗线表示  $M_2$  的边.



情形(2):  $xz$  和  $yw$  分别在  $H$  的同一分支中.  
虚线表示  $M_1$  的边, 粗线表示  $M_2$  的边.

(2)  $xz$  和  $yw$  在  $H$  的同一连通分支 (偶长度圈)  $C$  中, 如上图情形 (2) 所示. 不妨设  $x, y, w, z$  在  $C$  中依顺时针次序出现, 并设  $M_1$  在  $C$  的  $yw \cdots z$  段中的边集为  $M'_1$ ,  $M_2$  在  $C$  的  $yw \cdots z$  段中的边集为  $M'_2$ , 于是  $M'_1 \cup \{yz\} \cup (M_2 - M'_2)$  是  $G^*$  的完美匹配, 又与  $G^*$  的选择矛盾.



综合 (1)、(2) 两种情形, 便证明了  $G^* - U$  的每个连通分支都是完全图. 证毕.

**推论 3.2.1** 偶数阶  $(k-1)$ -边连通的  $k$  正则图有完美匹配.

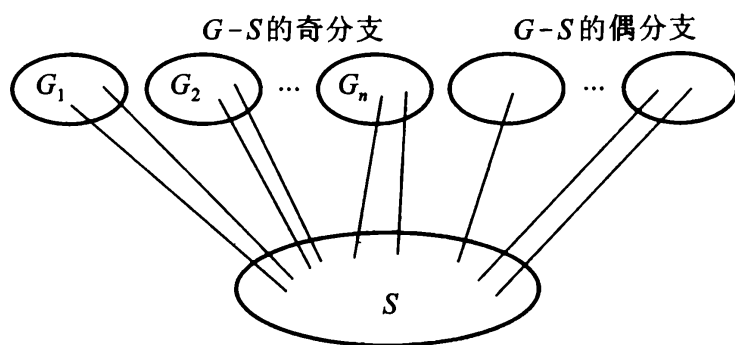
**证明** 设  $G$  是命题中所述的  $k$  正则图.

当  $k=1$  时, 结论显然.

以下假定  $k \geq 2$ . 任取  $S \subseteq V(G)$ . 若  $S = \emptyset$ , 则  $G - S = G$ , 无奇分支,  $o(G - S) = |S| = 0$ . 下设  $S$  是  $G$  的任一非空顶点集,  $G_1, G_2, \dots, G_n$  是  $G - S$  的奇分支. 若  $n=0$ , 则  $o(G - S) = 0$ , 显然  $o(G - S) < |S|$ ; 否则, 令

$$\nu_i = |V(G_i)|, \quad m_i = |\{e | e \text{ 是 } G_i \text{ 与 } S \text{ 之间的连边}\}|.$$

由于  $\kappa' \geq k-1$ , 故  $m_i \geq k-1, (i=1, 2, \dots, n)$ . 如图所示.



若存在  $i$ , 使得  $m_i = k-1$ , 则因  $\sum_{v \in V(G_i)} d_G(v) = k\nu_i$ , 从而

$$m_i = \sum_{v \in V(G_i)} d_G(v) - \sum_{v \in V(G_i)} d_{G_i}(v) = k\nu_i - 2\varepsilon(G_i).$$

因此,  $2\varepsilon(G_i) = k\nu_i - m_i = k\nu_i - (k-1) = k(\nu_i - 1) + 1$ . 但因  $\nu_i - 1$  是偶数 ( $G_i$  是奇分支), 上式两端不可能相等. 这个矛盾说明  $m_i \geq k, (i=1, 2, \dots, n)$ , 于是  $\sum_{i=1}^n m_i \geq kn$ .

且由正则性知  $\sum_{v \in S} d_G(v) = k|S|$ . 故有

$$o(G - S) = n \leq \frac{1}{k} \cdot \sum_{i=1}^n m_i \leq \frac{1}{k} \sum_{u \in S} d_G(u) = |S|.$$

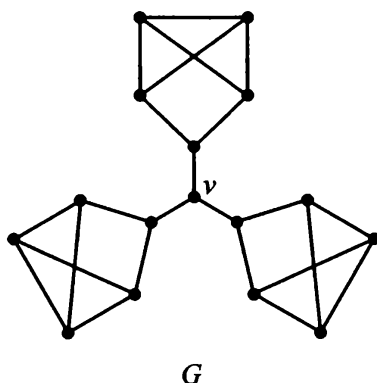
由 Tutte 定理,  $G$  有完美匹配. 证毕.

**推论 3.2.2 (Peterson, 1891)** 2-边连通 (无割边) 的 3-正则图有完美匹配.

**证明** 设  $G$  是 2-边连通的 3-正则图. 因  $2\varepsilon = \sum_{v \in V(G)} d(v) = 3\nu$ , 故  $\nu$  为偶数. 由

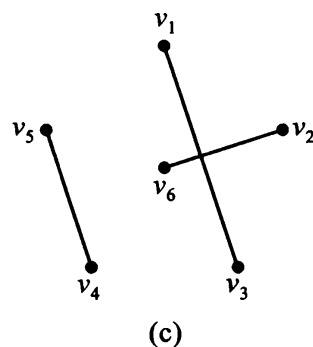
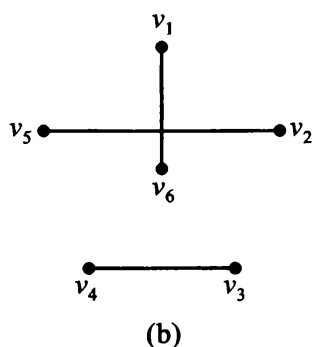
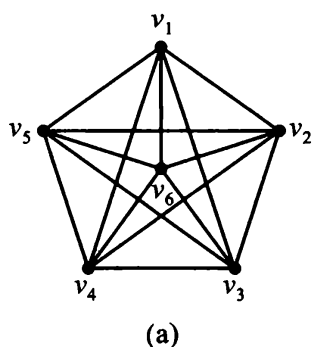
推论 3.2.1,  $G$  有完美匹配. 证毕.

注 有割边的 3 正则图未必有完美匹配. 例如, 对如下所示的图  $G$ , 因  $o(G-v) = 3$ , 故无完美匹配.



**推论 3.2.3** 偶数阶完全图  $K_{2n}$  有  $2n - 1$  个边不重的完美匹配.

**证明** 设  $K_{2n}$  的顶点集为  $\{v_1, v_2, \dots, v_{2n}\}$ . 以  $v_1, v_2, \dots, v_{2n-1}$  为顶点构造一个正  $2n - 1$  边形, 并将  $v_{2n}$  置于正  $2n - 1$  边形的中心位置. 用直线段连接每个顶点对, 将边  $v_i v_{2n}$  和所有与  $v_i v_{2n}$  垂直的边之集记为  $M_i, i = 1, 2, \dots, 2n - 1$ . 易检验每个  $M_i$  都是  $G$  的完美匹配, 且不同的  $M_i$  无公共边. 例如, 按这种方法构造出  $K_6$  的两个完美匹配如下图 (b)、(c) 所示. 显然,  $v_{2n}$  关联的每条边对应这样一个完美匹配, 故共有  $2n - 1$  个边不重的完美匹配. 证毕.



### §3.3 二部图的匹配

设  $M$  是二部图  $G = (X, Y)$  的一个匹配, 若  $M$  饱和  $X$  中每个顶点, 则称  $M$  饱和  $X$ .

**定理 3.3.1 (Hall, 1935)** 设  $G$  是具有二划分  $(X, Y)$  的二部图, 则  $G$  有饱和  $X$  的匹配当且仅当对  $\forall S \subseteq X, |N(S)| \geq |S|$ , 其中  $N(S)$  表示  $S$  的所有邻点之集.

**证明 必要性** 设  $G$  有饱和  $X$  的匹配  $M$ , 则对  $\forall S \subseteq X$ , 因  $S$  的顶点在  $M$  下与  $N(S)$  中某些顶点一一配对, 故  $|N(S)| \geq |S|$ .

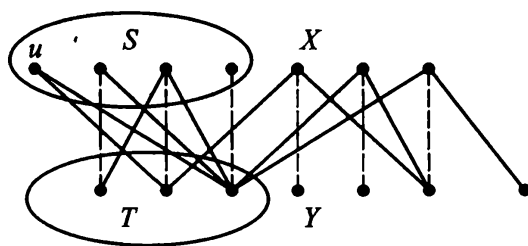
**充分性** 设  $G = (X, Y)$  是二部图, 且对  $\forall S \subseteq X, |N(S)| \geq |S|$ . 下用反证法证明  $G$  有饱和  $X$  的匹配.

假如  $G$  中不存在饱和  $X$  的匹配, 则  $G$  的最大匹配  $M^*$  不能饱和  $X$  的全部顶点. 设  $u$  是  $X$  的一个  $M^*$  非饱和点, 并设

$$A = \{v | v \in V(G) \text{ 且 } u \text{ 到 } v \text{ 有 } M^* \text{ 交错路}\}.$$

由于  $M^*$  是最大匹配, 故由 Berge 定理,  $u$  是  $A$  中唯一的  $M^*$  非饱和点. 如下图所示, 令

$$S = A \cap X, \quad T = A \cap Y.$$



二部图  $G$  及集合  $S$  与  $T$ ,  $A = S \cup T$ ,  
其中虚线边为  $M^*$  的边.

注意  $S - \{u\}$  中的顶点在  $M^*$  下与  $T$  中的顶点一一配对 (因  $u \in S$ , 且对  $\forall t \in T, u$  与  $t$  有  $M^*$  交错路  $P_t$  相连, 而且  $t$  是  $M^*$  饱和的, 故交错路  $P_t$  上最后一条边必是  $M^*$  的边, 它将  $S$  中一个顶点与  $t$  配对. 而且不同的  $t$  会有  $S$  中不同的顶点相配, 否则会有两条  $M^*$  的边关联到  $S$  中同一顶点). 因此

$$|T| = |S| - 1. \quad (*)$$

此外,  $N(S) \supseteq T$  (对  $T$  中每个顶点  $t$ , 从  $u$  到  $t$  有  $M^*$  交错路, 在交错路上顶点  $t$  的前一个顶点必属于  $S$ , 且与顶点  $t$  相邻, 因此  $t \in N(s)$ ), 且  $N(S) \subseteq T$  (对  $N(S)$  中每个顶点  $t$ , 设它是  $S$  中顶点  $s$  的邻点, 若从  $u$  到  $s$  的  $M^*$  交错路经过  $t$ , 则该交错路的  $(u, t)$  段是  $u$  到  $t$  的  $M^*$  交错路, 因此  $t \in T$ ; 否则, 因  $u$  到  $s$  的  $M^*$  交错路上最后一条边必属于  $M^*$ , 故该交错路必可延伸为  $u$  到  $t$  的  $M^*$  交错路, 因此  $t \in T$ ). 故

$$N(S) = T \quad (**)$$

由  $(*)$ 、 $(**)$  知:  $|N(S)| = |T| = |S| - 1 < |S|$ , 这与定理条件矛盾. 证毕.

**推论 3.3.1** 设  $G = (X, Y)$  是二部图, 若  $X$  中每个顶点至少关联  $k$  条边 ( $k \geq 1$ ) 而  $Y$  中每个顶点至多关联  $k$  条边, 则  $G$  中存在饱和  $X$  的匹配.

**证明** 由条件知, 对  $\forall S \subseteq X, S$  至少关联  $k|S|$  条边. 这  $k|S|$  条边至少关联  $Y$  中  $|S|$  个顶点, 即  $|N(S)| \geq |S|$ . 由 Hall 定理,  $G$  有饱和  $X$  的匹配. 证毕.

**推论 3.3.2 (Frobenius, 1917)** 具有二划分  $(X, Y)$  的二部图  $G$  有完美匹配的充分必要条件是  $|X| = |Y|$  且对  $\forall S \subseteq X$  (或  $Y$ ), 均有  $|N(S)| \geq |S|$ .

**证明** 由 Hall 定理, 结论是显然的.

**推论 3.3.3 (König, 1916)** 设  $G$  是  $k$  正则二部图 ( $k > 0$ ), 则  $G$  有  $k$  个边不重的完美匹配.

**证明** (1) 先证  $G$  有完美匹配.

设  $G = (X, Y)$  是  $k$  正则二部图, 则  $k|X| = |E(G)| = k|Y|$ , 因  $k > 0$ , 故  $|X| = |Y|$ . 任取  $S \subseteq X$ , 令

$$E_1 = \{G \text{ 中与 } S \text{ 关联的边}\}, \quad E_2 = \{G \text{ 中与 } N(S) \text{ 关联的边}\},$$

则  $E_1 \subseteq E_2$ . 因而  $k|N(S)| = |E_2| \geq |E_1| = k|S|$ , 即  $|N(S)| \geq |S|$ . 由推论 3.3.2,  $G$  有完美匹配.

(2) 再证  $G$  中有  $k$  个边不重的完美匹配 (用归纳法).

当  $k = 1$  时, 显然.

设对所有  $k$  正则二部图, 结论成立. 下证对  $(k+1)$  正则二部图  $G$ , 结论也成立.

设  $M$  是  $G$  的一个完美匹配. 令  $G' = G - M$ . 则  $G'$  是  $k$  正则二部图. 由归纳假设,  $G'$  中有  $k$  个边不重的完美匹配, 故  $G$  中有  $k+1$  个边不重的完美匹配. 证毕.

上述证明中结论 (1) 的一种简便证法: 由推论 3.3.1,  $G$  中有饱和  $X$  的匹配. 因  $|X| = |Y|$  (如前所证), 故这个匹配就是完美匹配. 下一推论是显然的.

**推论 3.3.4** 完全二部图  $K_{n,n}$  中存在  $n$  个边不重的完美匹配.

**推论 3.3.5** 设  $G = (X, Y)$  是二部图, 且  $|X| = |Y| = n$ . 若最小度  $\delta(G) \geq \frac{n}{2}$ , 则  $G$  有完美匹配.

**证明** 用反证法. 若  $G$  没有完美匹配, 则由推论 3.3.2, 存在  $S \subseteq X$ ,  $S \neq \phi$ , 使  $|N(S)| < |S|$ . 因  $G$  是二部图且  $\delta(G) \geq \frac{n}{2}$ , 故  $|S| > |N(S)| \geq \delta(G) \geq \frac{n}{2}$ , 且  $Y - N(S) \neq \phi$  (因  $|N(S)| < |S| \leq |X| = |Y|$ ). 令  $u \in Y - N(S)$ , 则  $N(u) \subseteq X - S$ , 因此,

$$\delta(G) \leq d_G(u) = |N(u)| \leq |X| - |S| < n - \frac{n}{2} = \frac{n}{2}.$$

这与条件矛盾. 故  $G$  有完美匹配. 证毕.

**注**  $G$  的  $k$  正则生成子图称为  $G$  的  $k$ -因子. 若  $G$  存在无公共边的  $k$ -因子  $H_1, H_2, \dots, H_n$ , 使得  $G = H_1 \cup H_2 \cup \dots \cup H_n$ , 则称  $G$  是可  $k$ -因子分解的.

推论 3.2.3 证明了完全图  $K_{2n}$  是可 1-因子分解的; 推论 3.3.4 证明了完全二部图  $K_{n,n}$  是可 1-因子分解的; 推论 3.3.3 证明了每个  $k$  正则二部图是可 1-因子分解的.

若图  $G$  有 1 因子, 则它显然应是偶数阶图. 因此奇数阶完全图  $K_{2n+1}$  不可能有 1 因子.

因子分解是图论的一个重要研究方向, 对这方面进一步的了解可参看文献 [1]~[6], 其中文献 [6] 是一篇很好的综述.

## §3.4 二部图中最大匹配与最大权匹配的算法

### 一、求最大匹配的匈牙利算法

#### 1. 背景与问题

**指派问题 (assignment problem):** 欲安排  $n$  个人  $x_1, x_2, \dots, x_n$  从事  $n$  项工作  $y_1, y_2, \dots, y_n$ . 已知每个人能胜任其中一项或几项工作. 试问: 能否给每个人分配一项他所胜任的工作? 若能, 如何求出这种安排?

**图论描述:** 构造一个二部图  $G = (X, Y): X = \{x_1, x_2, \dots, x_n\}, Y = \{y_1, y_2, \dots, y_n\}$ , 当且仅当  $x_i$  胜任工作  $y_j$  时,  $x_i y_j \in E(G)$ . 问:  $G$  中是否有完美匹配? 若有, 如何求之?

一般地, 对于一个二部图  $G = (X, Y): X = \{x_1, x_2, \dots, x_n\}, Y = \{y_1, y_2, \dots, y_m\}$ , 如何判断  $G$  中是否有饱和  $X$  的匹配? 若有, 如何求之? 更一般地, 如何求出  $G$  中一个具有尽可能多边的匹配 (最大匹配)?

#### 2. 理论基础

**Berge 定理 (定理 3.1.1):** 图  $G$  的匹配  $M$  是最大匹配的充要条件是  $G$  中不存在  $M$  可扩展路.

**Hall 定理 (定理 3.3.1):** 设  $G$  是具有二划分  $(X, Y)$  的二部图, 则  $G$  有饱和  $X$  的匹配当且仅当对  $\forall S \subseteq X, |N(S)| \geq |S|$ , 其中  $N(S)$  表示  $S$  的所有邻点之集.

#### 3. 匈牙利算法

匈牙利算法由匈牙利数学家 Egerváry 首先提出, 后来由 Edmonds (1965) 基于 Berge 定理和 Hall 定理进行了改进<sup>[7]</sup>. 这种算法既能判定一个二部图  $G = (X, Y)$  中是否存在饱和  $X$  的匹配, 又能在存在的情况下求出一个饱和  $X$  的匹配.

**算法思想:** 从二部图  $G = (X, Y)$  的任何匹配  $M$  开始.

若  $M$  饱和  $X$ , 则算法结束.

若  $M$  不饱和  $X$ , 在  $X$  中选择一个  $M$  不饱和点  $x$ . 若  $G$  中不存在以  $x$  为起点的  $M$  增广路, 则可找到与  $x$  由  $M$  交错路相连的顶点集合  $A$ , 而  $S = A \cap X$  满足  $|N(S)| < |S|$  (见 Hall 定理的证明), 此时由 Hall 定理,  $G$  不存在饱和  $X$  的匹配. 若存在以  $x$  为起点的  $M$  增广路  $P$ , 则由 Berge 定理知  $M$  不是最大匹配, 且  $M' = M \oplus E(P)$

是比  $M$  更大的匹配, 用  $M'$  替代  $M$ . 反复进行上述过程, 使匹配的边数逐步增加, 直至得到  $|X|$  条匹配边为止.

**算法步骤:**

**输入:** 二部图  $G = (X, Y)$ .

**输出:**  $G$  的一个饱和  $X$  的匹配.

**第 0 步:** 若  $|X| > |Y|$ , 则  $G$  中不存在饱和  $X$  的匹配, 停止. 否则, 任取图  $G$  的一个匹配  $M$ .

**第 1 步:** 若  $M$  饱和  $X$ , 则停止, 输出  $M$ ; 否则, 取  $X$  中一个  $M$  非饱和点  $x$ , 记  $S := \{x\}$ ,  $T := \emptyset$ .

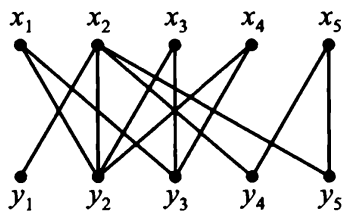
**第 2 步:** 若  $N(S) \subseteq T$ , 则停止,  $G$  中不存在饱和  $X$  的匹配 (因  $|N(S)| \leq |T| = |S| - 1 < |S|$ ); 否则取  $y \in N(S) - T$ .

**第 3 步:** 若  $y$  是  $M$  饱和的, 设  $yz \in M$ , 令  $S := S \cup \{z\}$ ,  $T := T \cup \{y\}$ , 转第 2 步 (此时仍保持  $|T| = |S| - 1$ ). 否则, 获得一条  $M$  增广路  $P(x, y)$ , 令  $M := M \oplus E(P)$ , 转第 1 步.

**注** (1) 按照算法过程, 进入  $T$  中的点必与  $S$  中某点相邻, 故总有  $N(S) \supseteq T$ , 因此第 2 步中判断  $N(S) \subseteq T$  可改为判断  $N(S) = T$ , 进而可转化为判断是否  $|N(S)| = |T|$ .

(2) 当  $|X| = |Y|$  时, 上述算法所求的就是二部图  $G = (X, Y)$  的完美匹配.

**例 3.4.1** 判断如下二部图是否存在完美匹配. 若存在, 求出其一个完美匹配; 若不存在, 给出满足  $N(S) = T$  的集合  $S$  和  $T$ .



**解** 从一个初始匹配  $M = \{x_2y_2, x_3y_3, x_5y_5\}$  (图 1 中粗边) 出发, 执行匈牙利算法. 因  $M$  尚未饱和  $X$ , 找到  $X$  中一个  $M$  未饱和点  $x_1$ . 从  $x_1$  出发, 反复执行算法第 2 步和第 3 步, 找到一条  $M$  增广路  $x_1y_2x_2y_1$  (图 2). 按第 3 步, 沿增广路交换进入匹配和不进入匹配的边, 匹配扩大为  $M = \{x_1y_2, x_2y_1, x_3y_3, x_5y_5\}$  (图 3).

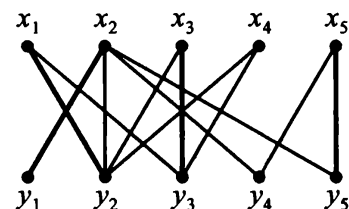
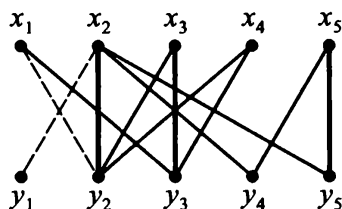
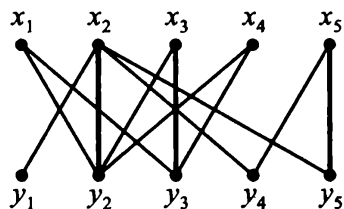


图 1. 图  $G = (X, Y)$  及一个匹配  $M$       图 2:  $M$  增广路  $x_1y_2x_2y_1$       图 3. 无完美匹配

因  $M$  仍未饱和  $X$ , 按算法第 1 步, 找到  $X$  中一个  $M$  未饱和点  $x_4$ . 然后算法令  $S = \{x_4\}, T = \emptyset$ , 并转入第 2 步试图找一条从  $x_4$  出发的  $M$  增广路. 因当前  $N(S) = \{y_2, y_3\}, N(S) \neq T$ , 从  $N(S) - T$  中任意取出一个, 比如  $y_3$ , 转到第 3 步. 因  $y_3$  是  $M$  饱和的, 其匹配点为  $x_3$ , 按算法第 3 步,  $S$  和  $T$  更新为  $S = \{x_4, x_3\}, T = \{y_3\}$ . 此后, 转第 2 步判断是否  $N(S) = T$ . 因当前  $N(S) = \{y_2, y_3\} \neq T$ , 故取  $y_2 \in N(S) - T$ , 再次循环执行第 3 步, 找到  $y_2$  的匹配点  $x_1$ ,  $S$  和  $T$  进一步更新为  $S = \{x_4, x_3, x_1\}, T = \{y_3, y_2\}$ , 再转第 2 步. 此时发现  $N(S) = \{y_2, y_3\} = T$ , 因此该图不存在完美匹配. 算法找到的满足  $N(S) = T$  的集合为  $S = \{x_4, x_3, x_1\}, T = \{y_3, y_2\}$ .

匈牙利算法的正确性由 Berge 定理和 Hall 定理可知.

匈牙利算法是多项式时间算法. 事实上, 设  $|X| = n, |Y| = m$ . 算法每找到一条增广路更新一次匹配, 匹配边增加一条, 故最多需执行  $n$  次增广路的循环.

算法每找一条增广路需反复执行第 2 步和第 3 步“生长”交错路, 这实际上是反复扩充集合  $S$  和  $T$  的过程. 算法每循环一次第 2 步和第 3 步,  $S$  和  $T$  的元素各增加 1 个. 由于  $|X| = n$ , 故这种扩张不会超过  $n$  次.

而算法每执行一次第 2 步和第 3 步, 除了需要做两次赋值 ( $S := S \cup \{z\}, T := T \cup \{y\}$ ) 和一次判断 ( $y$  是否  $M$  饱和的) 外, 主要计算量在于判断是否  $|N(S)| = |T|$  (见前注).  $|T|$  的计数可通过每次循环加 1 来实现,  $|N(S)|$  的计数可用上一循环的  $|N(S)|$  值加上本次循环新进入  $S$  集合的点  $z$  在  $Y - N(S)$  中的邻点数计算, 这需要不超过  $|Y| = m$  次判断. 因此算法总的计算复杂度约为  $n \cdot n \cdot (m + 6) = O(n^3)$ .

#### 4. 求二部图最大匹配的算法

前述匈牙利算法稍加修改后可用于求二部图的最大匹配.

##### 求二部图最大匹配的匈牙利算法

输入: 二部图  $G = (X, Y)$ .

输出:  $G$  的一个最大匹配.

第 0 步: 任取图  $G$  的一个匹配  $M$ , 设  $X$  中  $M$  非饱和点的集合为  $A$ .

第 1 步: 若  $A = \emptyset$ , 则停止, 输出当前的  $M$  (最大匹配); 否则, 任取  $x \in A$  (一个  $M$  非饱和点), 记  $S := \{x\}, T := \emptyset$ , 转下步.

第 2 步: 若  $N(S) \subseteq T$ , 则不存在从  $x$  出发的  $M$  增广路, 令  $A := A - \{x\}$ , 转第 1 步; 否则, 取  $y \in N(S) - T$ , 转下步.

第 3 步: 若  $y$  是  $M$  饱和的, 设  $yz \in M$ , 令  $S := S \cup \{z\}, T := T \cup \{y\}$ , 转第 2 步. 否则, 获得一条  $M$  增广路  $P(x, y)$ , 令  $M := M \oplus E(P), A := A - \{x, y\}$ , 转第 1 步.

在上述算法的第 3 步中, 通过运算  $M := M \oplus E(P)$  将匹配扩大一条边. 记  $M' = M \oplus E(P)$ , 此时原有的  $M$  饱和点仍是扩大后的匹配  $M'$  的饱和点, 而增广路的两个端点由原来的  $M$  非饱和点变为  $M'$  饱和点. 此外, 对此前在执行第 2 步时由于无  $M$

增广路而从  $A$  中退出的  $M$  非饱和点  $x$ , 现在必定也无  $M'$  增广路 (因当前的增广路  $P$  不可能用到从这样的  $x$  出发的  $M$  交错路上的边). 因此, 算法通过第 1 步至第 3 步的循环, 反复寻找  $M$  增广路, 增加匹配边, 减少  $M$  非饱和点, 直至不存在增广路为止, 此时便得到  $G$  的最大匹配.

**注** 求二部图最大匹配的匈牙利算法, 其计算复杂性仍为  $O(\nu^3)$ . 可在匈牙利算法中引入标号方法, 使算法的计算复杂度为  $O(\nu \cdot \varepsilon)$  (参看文献 [8]、[9]).

求二部图最大匹配目前已知的最好算法是 Hopcroft 和 Karp 提出的一个  $O(\nu^{2.5})$  算法. 读者可参看文献 [9]、[10].

求一般图的最大匹配也有多项式时间算法. 第一个多项式时间算法是由 Edmonds 给出的一个  $O(\nu^4)$  阶算法. 读者可参看文献 [11], 也可在文献 [8]、[9] 或 [12]~[14] 中找到对该算法的描述. Ahuja、Magnanti 和 Orlin 对这一算法进行了改进<sup>[15]</sup>, 将其复杂度降低为  $O(\nu^3)$ . Even 和 Kariv 提出的一个算法<sup>[16]</sup> 将计算复杂度改进到  $O(\nu^{2.5})$ . 目前已知的最快算法是 Micali 和 Vazirani 提出的  $O(\varepsilon\nu^{0.5})$  阶算法<sup>[17,18]</sup>, 该算法对稀疏图运算时, 计算复杂度低于  $O(\nu^{2.5})$ .

## 二、求赋权二部图最大权匹配的 Kuhn-Munkres 算法<sup>[19,20]</sup>

### 1. 背景与问题

**最优指派问题:** 欲安排  $n$  个人员  $x_1, x_2, \dots, x_n$  从事  $n$  项工作  $y_1, y_2, \dots, y_n$ . 已知每个人能胜任其中一项或几项工作, 各人做不同工作的效率不同. 求一种工作安排, 使得每个人分配一项不同于他人的工作, 且使总的工作效率达到最大.

**图论描述:** 给定赋权完全二部图  $K_{n,n} = (X, Y) : X = \{x_1, x_2, \dots, x_n\}, Y = \{y_1, y_2, \dots, y_n\}$ , 边  $x_i x_j$  有权  $w_{ij}$  (权可以为 0, 这表示  $x_i$  不胜任工作  $y_j$ ). 求  $K_{n,n}$  的一个具有最大权的完美匹配.

一般地, 给定一个赋权二部图  $G = (X, Y)$ , (未必  $|X| = |Y|$ ), 求  $G$  中权最大的匹配, 这一问题可以通过增添一些顶点及一些权为 0 的边, 化为上述对赋权完全二部图求最大权完美匹配的问题.

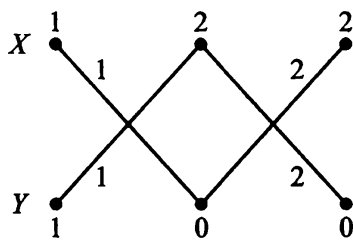
### 2. 算法的理论基础

#### • 顶点标号与可行顶点标号 (feasible vertex labeling)

图  $G$  的顶点标号是从顶点集到正整数集的一个映射. 用  $l(v)$  表示顶点  $v$  的标号,  $w(uv)$  表示边  $(u, v)$  的权. 对于赋权二部图  $G = (X, Y)$ , 若对每条边  $e = xy$ , 均有  $l(x) + l(y) \geq w(xy)$ , 则称这个标号为  $G$  的一个可行顶点标号.

下图显示了一个赋权二部图以及它的一个可行顶点标号.





一个赋权二部图及其可行顶点标号

注 赋权二部图的可行顶点标号总是存在的. 一种平凡的可行顶点标号是: 对  $\forall v \in V$ ,

$$l(v) = \begin{cases} \max_{y \in Y} w(vy), & v \in X \\ 0, & v \in Y \end{cases}.$$

• 相等子图

设  $G$  是一个赋权二部图,  $l$  是  $G$  的可行顶点标号, 边  $(u, v)$  上的权为  $w(uv)$ . 令

$$E_l = \{xy \in E(G) | l(x) + l(y) = w(xy)\},$$

$G$  中以  $E_l$  为边集的生成子图称为  $G$  的  $l$ 相等子图, 记为  $G_l$ . 注意  $G_l$  的顶点集与  $G$  的顶点集相同.

下面的例子显示了一个赋权完全二部图  $G = K_{4,4}$  的平凡顶点标号, 以及在这种标号下的相等子图.

$w =$	$\begin{matrix} & y_1 & y_2 & y_3 & y_4 \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix} & \begin{pmatrix} 3 & 5 & 5 & 4 \\ 2 & 2 & 0 & 2 \\ 2 & 4 & 4 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}$
-------	--

图  $G$  的平凡标号

对应的相等子图

**定理 3.4.1** 设  $l$  是赋权二部图  $G$  的一个可行顶点标号. 若相等子图  $G_l$  有完美匹配  $M^*$ , 则  $M^*$  是  $G$  的最大权完美匹配.

**证明** 由于相等子图  $G_l$  是  $G$  的生成子图, 故  $G_l$  的完美匹配  $M^*$  也是  $G$  的完美匹配, 而且

$$w(M^*) = \sum_{e \in M^*} w(e) = \sum_{v \in V} l(v).$$

另一方面, 对  $G$  的任何完美匹配  $M$ , 有:

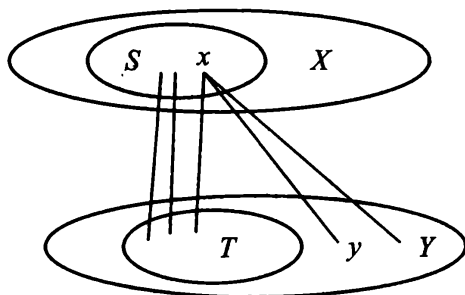
$$w(M) = \sum_{e \in M} w(e) \leq \sum_{v \in V} l(v)$$

可见  $W(M^*) \geq W(M)$ , 即  $M^*$  是  $G$  的最大权完美匹配. 证毕.

对任一个赋权二部图, 总可以给它添加一些顶点和权为 0 的边将其转化为赋权完全二部图  $K_{n,n}$ .  $K_{n,n}$  中总存在完美匹配, 且原赋权二部图的最大权匹配与  $K_{n,n}$  中的最大权完美匹配相对应. 由定理 3.4.1 知, 欲求赋权完全二部图  $G = K_{n,n}$  的最大权完美匹配, 只需求出其某个相等子图  $G_l$  中的完美匹配即可, 这可用匈牙利方法求得. 问题是, 如果相等子图  $G_l$  中无完美匹配时怎么办? Kuhn 和 Munkres 给出修改标号的一个方法, 使得新的相等子图的最大匹配逐渐扩大, 最终得到有完美匹配的相等子图.

### 3. 算法思想

首先给出赋权二部图  $G$  的任意一个可行顶点标号 (如平凡标号), 然后决定相等子图  $G_l$ , 在  $G_l$  中执行匈牙利算法. 若在  $G_l$  中找到完美匹配, 则由定理 3.4.1, 它就是  $G$  的最大权完美匹配. 否则, 匈牙利算法终止于  $S \subset X, T \subset Y$ , 且  $N_{G_l}(S) = T$ , 如下图所示. 设当前找到的匹配为  $M$ .



$$\text{令 } \alpha_l = \min\{l(x) + l(y) - w(xy) \mid x \in S, y \in Y - T\}, \quad (*)$$

对每个顶点  $u$ , 修改其标号如下

$$l'(u) = \begin{cases} l(u) - \alpha_l, & u \in S, \\ l(u) + \alpha_l, & u \in T, \\ l(u), & \text{其他.} \end{cases} \quad (**)$$

可以检验  $l'$  仍是  $G$  的一个可行顶点标号. 用  $l'$  替代  $l$ , 获得新的相等子图  $G_{l'}$ .

注意  $S$  与  $T$  间及  $X - S$  与  $Y - T$  间每条边的端点标号之和未变, 因而原来在相等子图中的边仍会保留在新标号下的相等子图中;  $S$  与  $Y - T$  间的边原来都不在相等子图中, 修改标号后, 每条边的端点标号之和减小  $\alpha_l$ , 因而至少有一条边进入新的相等子图;  $T$  与  $X - S$  间每条边的端点标号之和增加  $\alpha_l$ , 可能有一些原来属于相等子图的边会退出新的相等子图, 但这些边中不会有原相等子图的匹配  $M$  中的边. 因此,  $M$  中的边全在新的相等子图  $G_{l'}$  中, 且  $G_{l'}$  中有  $M$  可扩展路, 因而可得到更大的匹配.

反复进行上述过程, 直到获得一个相等子图含有完美匹配为止.

## 4. 算法步骤

Kuhn-Munkres 算法: 求赋权二部图中的最大权匹配.

输入: 二部图  $G = (X, Y)$  及各边上的权 (矩阵).

输出:  $G$  的最大权匹配.

第 0 步: 给  $G = (X, Y)$  添加一些顶点和权为 0 的边, 使其成为赋权完全二部图, 仍记为  $G$ .

第 1 步: 从  $G$  的任一可行的顶点标号  $l$  (比如平凡标号) 开始, 求出相等子图  $G_l$ .

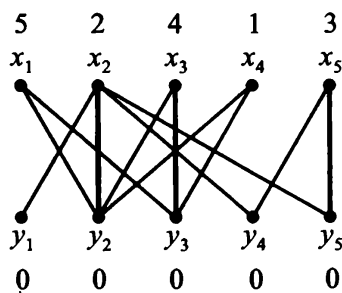
第 2 步: 在  $G_l$  中执行匈牙利算法, 如果求得  $G_l$  的一个完美匹配  $M$ , 则输出  $M$  ( $M$  即为  $G$  的最大权完美匹配, 删去其中权为 0 的边及其端点后, 即得原始图的最大权匹配), 算法停止; 否则, 匈牙利算法必终止于两个集合  $S \subset X$  和  $T \subset Y$  使得  $N_{G_l}(S) = T$ , 此时转下步.

第 3 步: 按 (\*) 式计算  $\alpha_l$ , 按 (\*\*) 计算  $G$  的新的可行顶点标号  $l'$ , 以  $l'$  替代  $l$ ,  $G_{l'}$  替代  $G_l$ , 转第 2 步.

例 3.4.2 设赋权完全二部图  $K_{5,5} = (X, Y)$  的权矩阵如下, 求其最大权完美匹配.

$$\begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{array} \begin{array}{ccccc} y_1 & y_2 & y_3 & y_4 & y_5 \\ \left( \begin{array}{ccccc} 3 & 5 & 5 & 4 & 1 \\ 2 & 2 & 0 & 2 & 2 \\ 2 & 4 & 4 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 3 & 3 \end{array} \right) \begin{array}{l} \rightarrow 5 \\ \rightarrow 2 \\ \rightarrow 4 \\ \rightarrow 1 \\ \rightarrow 3 \end{array} \end{array}$$

解 求出  $X$  中每个顶点关联的最大权 (权矩阵每行的最大元), 依次为 5, 2, 4, 1, 3. 给定初始可行顶点标号为平凡标号, 获得相等子图  $G_l$  如下. 取初始匹配  $M = \{x_2y_2, x_3y_3, x_5y_5\}$ .

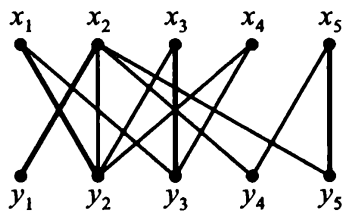


$K_{5,5}$  的平凡标号及相应的相等子图  $G_l$ ,  
 $G_l$  的一个匹配  $M$  (粗边)

在相等子图  $G_l$  中执行匈牙利算法, 对匹配进行一次增广路扩张后, 得新的匹配

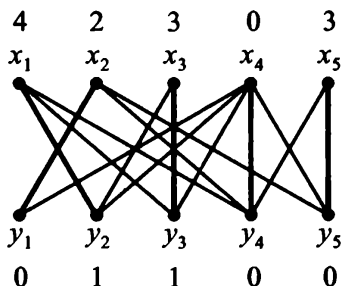
$$M = \{x_1y_2, x_2y_1, x_3y_3, x_5y_5\},$$

并发现当前的  $G_l$  无完美匹配, 匈牙利算法终止时获得  $S = \{x_1, x_3, x_4\}, N(S) = T = \{y_2, y_3\}$  (见例 3.4.1).



经一次增广路扩张后发现  $G_l$  无完美匹配,  
 $S = \{x_1, x_3, x_4\}, N(S) = T = \{y_2, y_3\}$

按公式  $\alpha_l = \min\{l(x) + l(y) - w(xy) \mid x \in S, y \in Y - T\}$  计算得  $\alpha_l = 1$ , 端点标号之和与边的权之差达到该最小值的有 4 条边:  $x_1y_4, x_4y_1, x_4y_4, x_4y_5$ . 按照 (\*\*) 式修改标号, 得到新的可行顶点标号及其相等子图  $G_l$ . 新的  $G_l$  比原来增加了四条边  $x_1y_4, x_4y_1, x_4y_4, x_4y_5$ . 在新的相等子图中以原匹配边作为初始匹配再执行匈牙利算法, 经一次增广路扩张后, 得到  $G_l$  的完美匹配  $M = \{x_1y_2, x_2y_1, x_3y_3, x_4y_4, x_5y_5\}$  (如下图), 此即所给赋权完全图  $K_{5,5}$  的最大权完美匹配.



修改标号后, 新的标号及其相等子图  $G_l$ .  
 $G_l$  比原相等子图增加了 4 条边:  $x_1y_4, x_4y_1,$   
 $x_4y_4, x_4y_5$ . 粗边是当前  $G_l$  的一个完美匹配

Kuhn-Munkres 算法的正确性证明可参看文献 [9], 其计算复杂度为  $O(\nu^3)$ [8].

二部图最大权匹配的其他算法 (如网络流方法) 可参看文献 [8]、[15]、[21]. 对于求一般赋权图中的最大权匹配, Edmonds 首先给出一个算法[22], 其后 Gabow 和 Lawler 在  $O(\nu^3)$  时间内实现了该算法[23,24]. 更快的算法已由 Gabow 和 Tarjan 给出[25,26].

为了在高速骨干通信网络中设计高性能路由器调度算法, 人们已经研究并提出了二部图最大匹配和最大权匹配的若干快速启发式算法. 有兴趣的读者可参看文献 [27]~[35] 及 [39]~[42].

设在一个完全二部图  $K_{n,n} = (X, Y)$  中,  $X$  中每个点对  $Y$  中各点都有偏好值,  $Y$  中每个点对  $X$  中各点也有偏好值. 对  $K_{n,n}$  的一个匹配  $M$ , 如果存在  $x \in X$  及  $y \in Y$ , 使得  $x$  对  $y$  的偏好值大于它对当前匹配点的偏好值,  $y$  对  $x$  的偏好值也大于它对当前匹配点的偏好值, 则称  $M$  是一个不**稳定匹配**, 而称  $(x, y)$  为一个**不稳定对**. 如果一个完美匹配没有不稳定对, 则称其为**稳定匹配**. 求一个完全二部图的稳定匹配有  $O(\nu^2)$  多项式时间算法[9][15][36][37]. 关于稳定匹配的综述和新进展可参看文献 [37]、[38]. 稳

定匹配在通信网络路由器调度算法中得到了应用<sup>[39]~[42]</sup>.

[43]~[45] 是与指派问题有关的文献, 其中文献 [44] 对指派问题进行了较全面的综述. 关于匹配理论和应用的进一步了解, 读者可参阅文献 [42]~[64], 其中 [46]~[48] 是图的匹配方面的几本较好著作.

### 习题三

3.1 设  $G$  是一个连通的非平凡图, 证明:

- (1) 若  $0 < \Delta(G) \leq 2$ , 则  $G$  是一个圈或一条路;
- (2)  $G$  是一个圈当且仅当  $G$  是 2-正则图.

3.2 (1) 设  $M_1, M_2$  是简单图  $G$  的两个匹配, 证明  $M_1 \oplus M_2$  在  $G$  中的边导出子图  $G[M_1 \oplus M_2]$  的每个非平凡连通分支必定是下列两种情况之一: ① 边在  $M_1$  和  $M_2$  中交替出现的偶长度圈; ② 边在  $M_1$  和  $M_2$  中交替出现的路.

(2) 设  $M_1, M_2$  是简单图  $G$  的两个完美匹配, 证明边导出子图  $G[M_1 \oplus M_2]$  的每个非平凡连通分支都是  $M_1$  和  $M_2$  的边交替出现的偶长度圈.

3.3 证明: 一棵树最多只有一个完美匹配.

3.4 证明一棵树  $T$  有完美匹配当且仅当对  $\forall v \in V(G)$ , 奇分支数  $o(T - v) = 1$ .

3.5 两人在图  $G$  上做游戏: 交替选择相异的顶点  $v_0, v_1, v_2, \dots$ , 使得对每个  $i > 0$ ,  $v_i$  与  $v_{i-1}$  相邻, 选择最后一个顶点者获胜. 证明: 第一选点人有一个得胜策略当且仅当图  $G$  没有完美匹配.

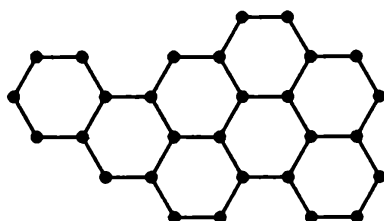
3.6 设  $M$  和  $N$  是图  $G$  中两个无公共边的匹配, 且  $|M| > |N|$ . 证明:  $G$  中必存在两个不相交匹配  $M'$  和  $N'$ , 使得  $M' \cup N' = M \cup N$ , 且  $|M'| = |M| - 1$ ,  $|N'| = |N| + 1$ .

3.7 设  $G = (X, Y)$  是一个二部图,  $U_1, U_2$  是  $X$  的两个子集,  $V_1, V_2$  是  $Y$  的两个子集 (注意  $U_1 \cap U_2, V_1 \cap V_2$  可能非空).  $M_1$  是  $G$  中  $U_1$  与  $V_1$  间的一个完美匹配,  $M_2$  是  $G$  中  $U_2$  与  $V_2$  间的一个完美匹配. 证明: 存在匹配  $M \subseteq M_1 \cup M_2$ , 它饱和  $U_1$  和  $V_2$ .

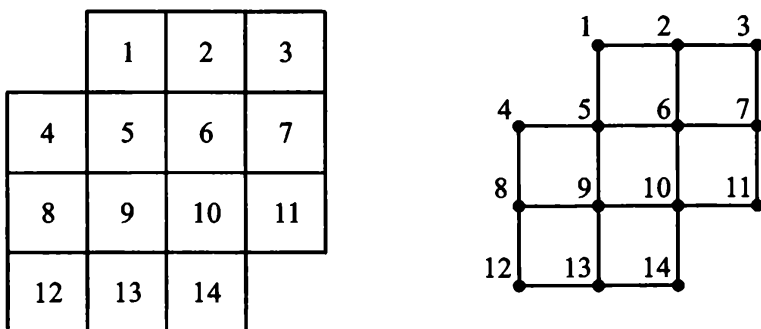
3.8 设  $G = (X, Y)$  是一个二部图, 利用上题的结论证明:

- (1)  $G$  中存在一个匹配饱和  $G$  的所有最大度顶点;
- (2)  $G$  的边集可划分成  $\Delta(G)$  个匹配 ( $\Delta(G)$  表示图  $G$  的最大度).

3.9 证明下图没有完美匹配.



3.10 下图所示的是 14 个大小相同的正方形组成的图形. 试证明: 不论如何用剪刀沿着图形中所画的直线对它进行裁剪, 总剪不出 7 个由相邻的两个小正方形组成的矩形来.



- 3.11 设  $G$  是一个 3 正则图且  $G$  有一个包含所有顶点的圈 (Hamilton 圈), 证明  $G$  是可 1- 因子分解的.
- 3.12 证明: 每个无割边的 3 正则图可分解为一个 1 因子和一个 2 因子的并.
- 3.13 证明: 若 3 正则图有割边, 则不能 1- 因子分解.
- 3.14 证明: 奇数阶完全图  $K_{2n+1}$  是可 2- 因子分解的, 但偶数阶完全图  $K_{2n+1}$  不可 2- 因子分解.
- 3.15 证明: 一个连通图是可 2- 因子化的当且仅当它是偶数度正则的. 利用此结论证明: 偶数阶完全图可分解为一个 1 因子和  $n - 1$  个 2 因子的并.
- 3.16 已知工人  $x_1, x_2, x_3, x_4, x_5$  做工作  $y_1, y_2, y_3, y_4, y_5$  的效率  $w_{ij}$  如下矩阵所示:

$$W = (w_{ij}) = \begin{matrix} & \begin{matrix} y_1 & y_2 & y_3 & y_4 & y_5 \end{matrix} \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{matrix} & \begin{pmatrix} 3 & 4 & 1 & 2 & 2 \\ 0 & 1 & 2 & 3 & 1 \\ 0 & 5 & 5 & 1 & 3 \\ 2 & 2 & 1 & 0 & 5 \\ 1 & 4 & 2 & 1 & 3 \end{pmatrix} \end{matrix} .$$

试给出一种工作效率最大的工作分派方案.

- 3.17 设有  $n$  个人竞聘  $m$  个岗位, 已知每个人至少对  $r$  个岗位提出了申请 ( $r \geq 1$ ), 而每个岗位至多收到  $r$  个人的申请, 设每个岗位最多录用 1 个人, 且各岗位录用的人不能相同, 问是否可以给每个人都安排一个自己所申请的岗位? 为什么?
- 3.18 盟军某飞行大队有若干名来自不同国家的飞行员, 飞行队的每架飞机必须有两名飞行员驾驶. 但由于语言和训练习惯等原因, 某些飞行员适合同机飞行, 另一些飞行员不适宜合同机飞行. 飞行队想要知道如何进行飞行员的搭配, 才能使尽可能多的飞机同时起飞. 请将这一问题化为图论问题.

## 参考文献

- [1] G. Chartrand, and O. R. Ollermann, *Applied and Algorithmic Graph Theory*, McGraw-Hill, New York, 1993.
- [2] J. Akiyama, M. Kano, Factors and factorizations of graphs — a survey, *J. Graph Theory*, 9(1985), 1-42.
- [3] W. D. Wallis, *One-factorizations*, Kluwer Academic Publishers, Dordrecht, Boston, 1997.
- [4] Guizhen Liu, and Binhai Zhu, Some problems on factorizations with constraints in bipartite graphs, *Discrete Applied Mathematics*, 128(2003), 421-434.
- [5] J. R. Correa, M. Matamala, Some remarks about factors of graphs, *Journal of Graph Theory*, 57(2008), 265-274.
- [6] M. D. Plummer, Graph factors and factorization: 1985-2003: A survey. *Discrete Mathematics*, 307(2007), 791-821.
- [7] J. Edmonds, Path, tree and flowers, *Canad. J. Math.*, 17(1965)449-467.
- [8] 谢政. 网络算法与复杂性理论. 第 2 版. 长沙: 国防科技大学出版社, 2003.
- [9] D. B. West, *Introduction to Graph Theory* (second edition), Prentice Hall, 2001. (中译本: 图论导引. 李建中, 骆吉周译. 北京: 机械工业出版社, 2006.)
- [10] J. E. Hopcroft and R. E. Karp, An  $O(n^{2.5})$  algorithm for maximum matching in bipartite graphs, *SIAM on Journal of Computing*, 2(1973), pp225-231.
- [11] J. Edmonds, Paths, trees, and flowers, *Canad. J. Math.*, 17(1965), 449-467.
- [12] C. H. Papadimitriou & K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.
- [13] 田丰, 马仲蕃. 图与网络流理论. 北京: 科学出版社, 1987.
- [14] 蒋长浩. 图论与网络流. 北京: 中国林业出版社, 2001.
- [15] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, 1993. (影印版: 网络流: 理论算法与应用. 北京: 机械工业出版社, 2005 年.)
- [16] S. Even, and O. Kariv, An  $O(n^{2.5})$  algorithm for maximum matching in general graphs, in *Proceedings of 16<sup>th</sup> IEEE Annual Symposium on Foundations of Computer Science*, 1975, 100-112.
- [17] S. Micali, and V. V. Vazirani, An  $O(\sqrt{|V|} \cdot |E|)$  algorithm for finding maximum matching in general graphs. in *Proceedings of 21<sup>th</sup> IEEE Annual Symposium on Foundations of Computer Science*, 1980, 17-27.
- [18] V. V. Vazirani, A theory of alternating paths and blossoms for proving correctness of the  $O(\sqrt{|V|} \cdot |E|)$  general graph matching algorithm, *Combinatorica*, 14(1994), 71-91.
- [19] H. W. Kuhn, The Hungarian method for the assignment problem, *Naval Research Logistics Quarterly*, 2(1955), 83-97.

- [20] J. Munkres, Algorithms for the assignment and transportation problems, *J. Soc. Indust. Appl. Math.* 5(1957), 32-38.
- [21] R. E. Tarjan, Data structures and network algorithms, *Society of Industrial and Applied Mathematics*, Pennsylvania, Nov. 1983. (二部图最大权匹配的  $O(n^3 \log N)$  算法)
- [22] J. Edmonds, Maximum matchings and a polyhedron with 0,1 vertices, *J. Res. Nat. Bur. Standards*, 69B(1965), 125-130.
- [23] H. N. Gabow, An efficient implementation of Edmonds' algorithm for maximum matchings on graph, *J. Assoc. Comp. Math.*, 23(1975), 221-234.
- [24] E.L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart, and Winston, 1976.
- [25] H.N. Gabow, Data structures for weighted matching and nearest common ancestors with linking. In *Proceedings of the 1th ACM-SIAM Symp. Disc. Algs. SIAM*, (1990), 434-443.
- [26] H.N. Gabow, and R.E. Tarjan, Faster scaling algorithms for general graph matching problems. *Tech. Rept. CU-CS-432-89 Dept. Comp. Sci. Univ. Colorado-Boulder*, 1989.

#### 路由器调度中的匹配算法

- [27] T. Anderson, S. Owicki, J. Saxe, and C. Thacker, High speed switch scheduling for local area networks, *ACM Transactions on Computer Systems*, Nov., (1993) 319-352.
- [28] M. Ali, H. Nguyen, A neural network implementation of an input access scheme in a high-speed packet switch, in *Proc. of GLOBECOM 1989*, 1192-1196.
- [29] Ge Nong, J.K Muppala, and M. Hamdi, Analysis of Nonblocking ATM switches with multiple input queues, *IEEE/ACM Transactions on Networking*, 7: 1(1999) 60-74.
- [30] N. Mckeown, *Scheduling Algorithms for Input-queued Cell Switches*, Ph.D Dissertation, University of California at Berkley, May 1995.
- [31] N. Mckeown, The iSLIP scheduling algorithm for input-queued switches, *IEEE/ACM Transactions on Networking*, 7(2)(1999), 188-201.
- [32] A. Mekittikul, N. Mckeown, A practical scheduling algorithm to achieve 100% throughput in input-queued switches, in *Proc. IEEE INFOCOM'98*, San Francisco, CA, Apr. (1998)792-799.
- [33] N. Mckeown, V. Anantharam, and J. Walrand, Achieving 100% throughput in an input-queued switch, in *Proc. IEEE INFOCOM'96*, San Francisco, Mar., 1(1996) 296-302
- [34] D. Shah, M. Kopikare, Delay bounds for approximate maximum weight matching algorithms for input queued switches, in *Proc. IEEE INFOCOM 2002*, (2002) 1-8
- [35] P. Giaccone, B. Prabhakar, and D. Shah, Randomized scheduling algorithms for high-aggregate bandwidth switches, *IEEE Journal on Selected Areas in Communications*, 21(4)(2003), 546-559.



**稳定匹配及其应用**

- [36] D. Gale , and L.S. Shapley, College admissions and the stability of marriage, *American Mathematical Monthly*, 69(1962), 9-15.
- [37] D. Gusfield, and R.W. Irving, *The Stable Marriage Problem: Structure and Algorithms*, MIT Press, Cambridge, MA, 1989.
- [38] K. Cechlárová, and D. F. Manlov, The exchange-stable marriage problem, *Discrete Applied Mathematics*, 152(2005) 109-122
- [39] E. Leonardi, M. Mellia, F. Neri, and M.A. Marsan, On the stability of input-queued switches with speed-up, *IEEE/ACM Transactions on Networking*, 9: 1(2001), 104-118.
- [40] S.T. Chuang, A. Goel, N. Mckeown, and B. Prabhakar, Matching output queueing with combined input and output queueing, *IEEE Journal on Selected Areas in Communications*, 17(1999) 1030-1039.
- [41] Ge Nong, and M. Hamdi, On the provision of integrated QoS guarantees of unicast and multicast traffic in input-queued switches, in *Proceedings of the Global Telecommunications Conference-Globecom'99*, 1742-1746.
- [42] M. Ajmone Marsan, E. Leonardi, M. Mellia, and F. Neri, Stability of maximal size matching scheduling in input-queued cell switches, in *IEEE ICC*, New Orleans, LA, June (2000) 1758-1763.

**指派问题**

- [43] D. P. Bertsekas, The auction algorithm: a distributed relaxation method for the assignment problem, *Annals of Operations Research*, 14(1988), 105-123.
- [44] R. E. Burkard, Selected topics on assignment problems, *Discrete Applied Mathematics*, 123(2002) 257-302.
- [45] Meng Xu, Anthony Chen, Ziyou Gao, An improved origin-based algorithm for solving the combined distribution and assignment problem, *European Journal of Operational Research*, 188(2008), 354-369.

**匹配问题更多文献**

- [46] A. Gibbons, *Algorithmic Graph Theory*, Cambridge University Press, 1985.
- [47] L. Lováze, and M. D. Plummer, Matching Theory, *Annals of Discrete Mathematics* 29, North-Holland, Amsterdam, 1986.
- [48] A. Schrijver, *Combinatorial Optimization Polyhedra and Efficiency*, New York, NY: Springer-Verlag, 2003.
- [49] P.M. Vaidya, Geometry helps in matching, *SIAM Journal on Computing*, 18(1989) 1201-1225.
- [50] P.M. Vaidya, Approximate minimum weight matching on points in k-dimensional space, *Algorithmica*, 1990.

- [51] A. Frieze, M. Karonski, and L. Thoma, On perfect matchings and Hamilton cycles in sums of random trees, *SIAM Journal on Discrete Mathematics*, 12(1999), 208-216.
- [52] M. Hanckowiak, M. Karonski, and A. Panconesi, On the distributed complexity of computing maximal matchings, *SIAM Journal on Discrete Mathematics*, 15(2001-2002), 41-57.
- [53] R. Hassin and S. Rubinstein, Robust Matchings, *SIAM Journal on Discrete Mathematics*, 15(2001-2002), 530-537.
- [54] T. Yi, K. G. Murty and C. Spera, Matchings in colored bipartite networks, *Discrete Applied Mathematics*, 121(2002), 261-277.
- [55] N. Pippenger, Enumeration of Matchings in the Incidence Graphs of Complete and Complete Bipartite Graphs, *SIAM Journal on Discrete Mathematics*, 16(2002-2003), 47-64.
- [56] A. Frank, Restricted  $t$ -matchings in bipartite graphs, *Discrete Applied Mathematics*, 131(2003), 337-346.
- [57] A. Czygrinow, M. Hańckowiak and E. Szymańska, Distributed algorithm for approximating the maximum matching, *Discrete Applied Mathematics*, 143(2004), 62-71.
- [58] M. Huber, Exact sampling from perfect matchings of dense regular bipartite graphs, *Algorithmica*, 44(3)(2006), 183-193.
- [59] M. Mucha, P. Sankowski, Maximum matchings in planar graphs via Gaussian elimination, *Algorithmica*, 45(1)(2006), 3-20.
- [60] S. Fujita, K. I. Kawarabayashi, C. L. Lucchesi, K. Ota, M. D. Plummer and A. Saito, A pair of forbidden subgraphs and perfect matchings, *Journal of Combinatorial Theory*, (Ser.B), 96(2006), 315-324.
- [61] D. Hartvigsen, Finding maximum square-free 2-matchings in bipartite graphs, *Journal of Combinatorial Theory*, (Ser.B), 96(2006), 693-705.
- [62] T. Kaiser, A note on interconnecting matchings in graphs. *Discrete Mathematics*, 306(2006), 2245-2250.
- [63] Haicheng Ma, Haizhen Ren, The new methods for constructing matching-equivalence graphs. *Discrete Mathematics*, 307(2007), 125-131.
- [64] R. E. L. Aldred, R. P. Anstee, S. C. Locke, Perfect matchings after vertex deletions. *Discrete Mathematics*, 307(2007), 3048-3054.

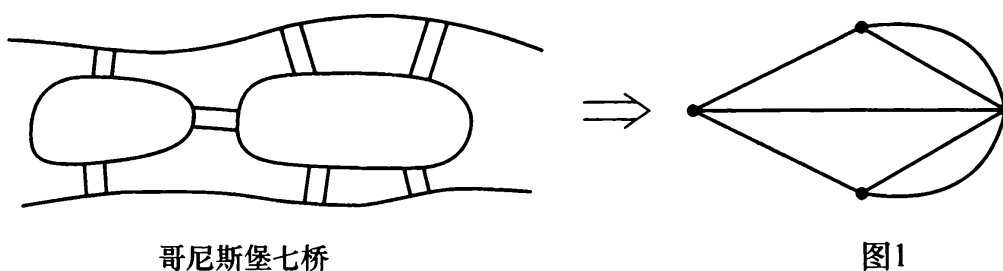
# 第四章 Euler 图与 Hamilton 图

---

## §4.1 Euler 图

### 一、基本概念

通常认为, 图论见诸于文献的开端性研究之一是瑞士数学家欧拉关于七桥问题的研究. 在 18 世纪普鲁士的哥尼斯堡城 (Königsberg), 普雷格尔河穿城而过, 河中有两个河心岛, 有七座桥将小岛与河岸连接起来 (如下图所示). 有市民尝试从河岸或岛屿的任一陆地点出发, 经过每座桥一次且仅一次回到出发点, 但一直未能获得成功. 人们怀疑, 这样的走法是否存在? 这便是七桥问题.



哥尼斯堡七桥

图1

1741 年, 欧拉经过对七桥问题的研究, 发表了第一篇有关图论的论文, 从而使七桥问题闻名于世. 欧拉将四块陆地用平面上四个点来表示, 两块陆地间有一座桥相连, 就在两个相应的点间连一条边, 最终获得如图 1 所示的一个图  $G$ . 于是七桥问题转化为一个图论问题: 在图  $G$  中从任一顶点出发, 经过每条边恰好一次回到出发点, 是否可能?

为了进一步的讨论, 我们需要定义如下术语.

**Euler 闭迹** (closed trail, tour, circuit): 经过图  $G$  的每条边恰好一次的闭迹.

**Euler 迹** (trail): 经过每条边恰好一次的迹.

**Euler 图**: 有 Euler 闭迹的图.

Euler 图、Euler 迹、Euler 闭迹有时也分别称为欧拉图, 欧拉迹, 欧拉闭迹.

利用这些术语,七桥问题可叙述为:图 1 中的图  $G$  是否为 Euler 图? 欧拉对此做出了否定的回答. 事实上,欧拉研究了更一般的情况,获得了任意一个图是否为 Euler 图的判定条件.

## 二、Euler 图的判定

**定理 4.1.1 (Euler, 1736)** <sup>[1]</sup> 一个非空连通图是 Euler 图当且仅当它没有奇度顶点.

**证明 必要性** 设图  $G$  是 Euler 图,  $C$  是  $G$  的一条 Euler 闭迹. 对  $\forall v \in V(G)$ ,  $v$  必在  $C$  上出现. 注意  $C$  每经过  $v$  一次,就有两条与  $v$  关联的边被使用. 设  $C$  经过  $v$  共  $k$  次,则  $d(v) = 2k$ .

**充分性** 无妨设  $\nu(G) > 1$ . 因  $G$  连通,故至少有一条边. 下面用反证法证明充分性结论.

假设图  $G$  无奇度顶点,但它不是 Euler 图. 令

$$S = \{G \mid G \text{ 是至少有一条边的连通图, 无奇度顶点, 且不是 Euler 图}\},$$

则  $S \neq \emptyset$ . 取  $S$  中边数最少的一个图,记为  $G'$ . 因  $\delta(G') \geq 2$ ,故  $G'$  含有圈(例 1.1.2),因而含有闭迹. 设  $C$  是  $G'$  中一条最长的闭迹. 由假设,  $C$  不是  $G'$  的 Euler 闭迹. 因此  $G' - E(C)$  必有一个连通分支至少含有一条边. 记这个连通分支为  $G_0$ . 由于  $C$  是闭迹,故  $G_0$  中没有奇度顶点,且  $\varepsilon(G_0) < \varepsilon(G')$ . 由  $G'$  的选择可知,  $G_0$  必有 Euler 闭迹  $C_0$ . 由于  $G'$  连通,故  $C$  必经过  $G_0$  中至少一个顶点,从而  $V(C) \cap V(C_0) \neq \emptyset$ . 因此  $C + C_0$  是  $G'$  的一条闭迹,且  $\varepsilon(C + C_0) > \varepsilon(C)$ ,这与  $C$  的选取矛盾. 证毕.

**充分性的另一种证法 (数学归纳法):**

无妨设  $\nu(G) > 1$ . 因  $G$  连通且无奇度顶点,故  $\delta(G) \geq 2$ ,因而必含有圈.

当  $\nu(G) = 2$  时,设仅有的两点为  $u, v$ . 因  $u, v$  的度都是偶数,故  $u, v$  间必有偶数条边,它们显然构成 Euler 闭迹.

假设  $\nu(G) = k$  时,结论成立.

当  $\nu(G) = k + 1$  时,任取  $v \in V(G)$ . 令  $S = \{v \text{ 的所有关联边}\}$ . 记  $S$  中的边为  $e_1, e_2, \dots, e_m$ ,其中  $m = d(v)$  为偶数. 记  $G' = G - v$ . 对  $G'$  做如下操作:

(1) 任取  $e_i, e_j \in S$ ,设在  $G$  中  $e_i = v_i v, e_j = v_j v$ ,则对  $G$  和  $S$  做如下操作:

$$G' := G' + v_i v_j, S := S - \{e_i, e_j\};$$

(2) 若  $S = \emptyset$ ,则停止;否则转 (1) 继续.

这个过程最终得到的  $G'$  有  $k$  个顶点,且每个顶点在  $G'$  中的度与在  $G$  中完全一样. 由归纳假设,  $G'$  中有 Euler 闭迹,设为  $C$ . 在  $G$  中从任一点  $u$  出发 ( $u \neq v$ ),沿  $C$

的对应边前行. 每当遇到上述添加边  $v_i v_j$  时都用对应的两条边  $e_i, e_j$  代替, 这样显然可获得  $G$  的一条 Euler 闭迹. 证毕.

**定理 4.1.2** 一个连通图有 Euler 迹当且仅当它最多有两个奇度顶点.

**证明 必要性** 设连通图  $G$  有 Euler 迹  $T$ . 若  $T$  是 Euler 闭迹, 则  $G$  中无奇度顶点. 否则, 设  $T$  的起点和终点分别为  $u, v$ . 在  $G$  的基础上, 给  $u, v$  间添加一条边  $e$  (若  $G$  中有边  $uv$ , 则  $e$  是重边), 所得之图记为  $G^*$ , 则  $T + e$  是  $G^*$  的 Euler 闭迹. 由定理 4.1.1, 图  $G^*$  无奇度顶点. 而  $G$  可看作是  $G^*$  删除了边  $e$  后得到的, 因此,  $G$  最多只可能有两个奇度顶点.

**充分性** 若  $G$  无奇度顶点, 则由定理 4.1.1,  $G$  有 Euler 闭迹, 自然有 Euler 迹.

若  $G$  只有两个奇度顶点, 设其为  $u, v$ , 则给  $G$  添加一条新边  $e = uv$ , 所得之图  $G + e$  的每个顶点都是偶度顶点, 由定理 4.1.1,  $G + e$  有 Euler 闭迹, 因此  $G$  有 Euler 迹. 证毕.

一个图  $G$  如果有一条 Euler 迹或 Euler 闭迹, 则我们可以沿着 Euler 迹或 Euler 闭迹连续而不重复地把  $G$  的边画完. 因此存在 Euler 迹或 Euler 闭迹的图通常称为可一笔画的图, 或者说它可一笔画成. 如果图  $G$  可分解为两条迹或闭迹的并, 则  $G$  的边可用两笔连续且不重复地画完. 一般地, 如果图  $G$  可分解为  $k$  条迹或闭迹的并, 则称  $G$  可  $k$  笔画成.

定理 4.1.1 和定理 4.1.2 表明, 一个图  $G$  可一笔画成的充分必要条件是  $G$  至多有 2 个奇度顶点. 一般地, 有下述推论.

**推论 4.1.1** 一个连通图可不超过  $k$  笔画成当且仅当它最多有  $2k$  个奇度顶点.

证明留作习题.

Toida 于 1973 年发现, 在一个 Euler 图中, 对任何一条边, 经过它的圈的个数都是奇数. 1984 年, Mckee 又证明这个条件是充分的. 这样便形成了对 Euler 图的另一种刻画.

**定理 4.1.3 (Toida, Mckee)<sup>[2,3]</sup>** 一个非平凡连通图  $G$  是 Euler 图的充分必要条件是  $G$  的每条边含在奇数个圈上.

**证明 必要性** 设  $G$  是 Euler 图, 则  $G$  的所有顶点都是偶数度的. 任取  $G$  的一条边  $e = uv$ , 因  $G$  有 Euler 闭迹, 故  $G' = G - e$  是连通图. 为叙述方便, 将  $G'$  中从  $u$  到  $v$  的迹简称为  $u-v$  迹. 我们来考察  $G'$  中所有不同的  $u-v$  迹的数目, 这里两条迹相同是指两条迹的边数相同且每一步使用的边相同, 因此两条迹不同要么至少其中一条迹上有另一条迹所没有的边, 要么两条迹使用边的次序不同. 注意到在  $G'$  中从顶点  $u$  出发有奇数条边可作为迹的始边. 一旦始边被确定, 则这条迹就要沿选定的边继续到

下一个顶点  $w$ . 由于除  $uw$  外  $w$  关联的边有奇数条, 所以迹的第二条边有奇数种选择. 如此继续, 从每个点前行都有奇数条边可选择, 直至到达迹的另一个端点  $v$ . 这表明在  $G' = G - e$  中有奇数条不同的  $u - v$  迹.

假设  $T$  是  $G'$  中一条  $u - v$  迹, 但不是一条  $u - v$  路, 则  $T$  必定重复经过了某些顶点. 如果它重复经过了某个顶点  $v_1$ , 则意味着  $T$  包含一条  $v_1 - v_1$  闭迹  $C$ , 此时将  $C$  “逆向” 使用,  $T$  上其他边不变, 便得到另一条  $u - v$  迹. 如果两条  $u - v$  迹能通过若干次 (含 0 次) 这样的闭迹段 “逆向” 变换互相转化, 则称它们是同类的. 可见, 如果一条  $u - v$  迹重复经过  $k$  个点 (多次经过同一点的计重复经过次数), 则经这种逆向变换可获得  $2^k$  个同类  $u - v$  迹. 这种分类构成一个等价关系 (满足自反性、对称性、传递性), 因此形成了对有重复点的  $u - v$  迹集合的划分. 划分出的每一个等价类有偶数条  $u - v$  迹 ( $2^k$  条,  $k$  是该等价类中每条迹的重复点个数). 这说明有重复点的  $u - v$  迹总共有偶数条.

由以上两段的结论知,  $G' = G - e$  中总共有奇数条顶点无重复的  $u - v$  迹 (即  $u - v$  路), 因此,  $G$  中有奇数个含有边  $e$  的圈.

**充分性** 设  $G$  的每条边含在奇数个圈上, 希望证明  $G$  的每个顶点都是偶数度的. 任取顶点  $v$ , 设  $v$  关联的边共有  $k$  条, 分别为  $e_1, e_2, \dots, e_k$ . 与这些边相对应, 构造一个有重边的图  $H$  如下: 顶点集为  $\{u_1, u_2, \dots, u_k\}$ , 对于每个  $u_i$ , 相应于每个既含有  $e_i$  也含有某个  $e_j$  的圈, 在  $u_i$  和  $u_j$  之间连一条边.

由于  $e_i$  含在奇数个圈上, 且每个经过  $e_i$  的圈必经过  $e_1, e_2, \dots, e_k$  中某条其他边, 因此  $H$  中顶点  $u_i$  的度为奇数. 由  $u_i$  的任意性,  $H$  中每个点的度都是奇数. 由公式  $2\varepsilon(H) = \sum_{i=1}^k d_H(u_i)$  知,  $H$  的顶点个数  $k$  必须是偶数, 从而可知在  $G$  中顶点  $v$  关联偶数条边. 由  $v$  的任意性,  $G$  中所有点都是偶数度顶点, 故为 Euler 图. 证毕.

### 三、求 Euler 图中的 Euler 闭迹 - Fleury 算法

对于复杂一些的图, 即使判定出它有 Euler 闭迹, 也未必能很快地找出一条 Euler 闭迹. 在许多大规模应用中, 需要借助于算法来找图的 Euler 闭迹. Fleury 给出了一个在 Euler 图中找 Euler 闭迹的多项式时间算法. 其基本思想是, 从图中一个顶点出发, 用深度优先方法找图的迹, 在任何一步, 尽可能不使用剩余图的割边, 除非没有别的选择.

**Fleury 算法的步骤如下:**

**输入:** Euler 图  $G$

**输出:**  $G$  的 Euler 闭迹.

**第 1 步:** 任取  $v_0 \in V(G)$ , 令  $w_0 := v_0, i := 0$ .

**第 2 步:** 设迹  $w_i = v_0 e_1 v_1 \cdots e_i v_i$  已取定. 从  $E - \{e_1, e_2, \dots, e_i\}$  中选取一条边

$e_{i+1}$ , 使得

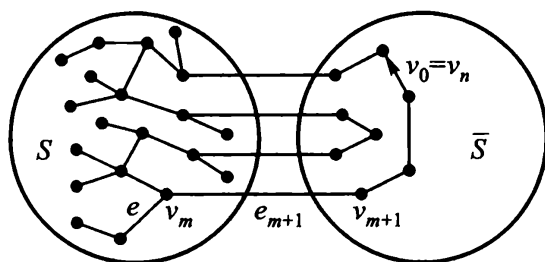
- (1)  $e_{i+1}$  和  $v_i$  相关联;
- (2)  $e_{i+1}$  不选  $G_i = G - \{e_1, e_2, \dots, e_i\}$  的割边, 除非没有别的选择.

第 3 步: 当第 2 步不能再执行时, 停止.

**定理 4.1.4** 若  $G$  是 Euler 图, 则 Fleury 算法终止时得到的是  $G$  的 Euler 闭迹.

**证明** 设  $G$  是 Euler 图,  $W_n = v_0 e_1 v_1 e_2 \cdots e_n v_n$  是 Fleury 算法终止时得到的迹. 则显然  $v_n$  在  $G_n = G - \{e_1, e_2, \dots, e_n\}$  中的度数是 0 (因算法终止在  $v_n$ , 若不是 0, 则算法应继续). 故  $v_0 = v_n$  (否则  $v_n$  在  $G$  中是奇度顶点, 这不可能), 即  $W_n$  是闭迹. 下面用反证法证明  $W_n$  是 Euler 闭迹.

若  $W_n$  不是  $G$  的 Euler 闭迹, 设  $S = \{G_n \text{ 中度} > 0 \text{ 的所有顶点}\}$ . 则  $S \neq \emptyset$  (因  $W_n$  不是  $G$  的 Euler 闭迹, 有边不在  $W_n$  上), 且  $W_n$  上有  $S$  中的点 (否则  $W_n$  上的点都是  $G_n$  的孤立点, 这与  $G$  是 Euler 图 (从而连通) 矛盾), 但  $v_n \in \bar{S} = V(G) - S$ . 设  $m$  是  $W_n$  上使得  $v_m \in S$  而  $v_{m+1} \in \bar{S}$  的最大整数. 因  $W_n$  终止于  $\bar{S} = V(G) - S$ , 故  $e_{m+1} = v_m v_{m+1}$  是  $G_m$  中  $[S, \bar{S}]$  的仅有的一条边, 因而是  $G_m$  的一条割边. 如下图所示.



设  $e$  是  $G_m$  中与  $v_m$  关联的另一条边 (因  $d_{G_n}(v_m) > 0$ , 这样的边  $e$  一定存在), 则由算法第 2 步知:  $e$  必定也是  $G_m$  的一条割边 (否则, 按算法应选  $e$  而不选  $e_{m+1}$ ), 因而  $e$  是  $G_m[S]$  的割边.

但由于  $W_n$  是闭迹,  $v_0 = v_n \in \bar{S}$ , 且对  $\forall v \in S, d_G(v)$  是偶数. 故  $G_m[S] = G_n[S]$  中每个顶点都是偶数度顶点, 由此又推出  $G_m[S]$  无割边 (第二章习题 2.9).

以上两方面矛盾. 证毕.

Fleury 算法的计算复杂度分析留给读者.

有关欧拉问题的更多内容可参看文献 [4]~[9].

## §4.2 中国邮递员问题 (Chinese Postman Problem)

**问题** (管梅谷, 1960): 一位邮递员从邮局出发投递邮件, 经过他所管辖的每条街道至少一次后, 回到邮局. 请为他选择一条路线, 使其所行路程尽可能短.

**图论模型:** 求赋权连通图中含所有边且权最小的闭途径. 这样的闭途径称为**最优邮路**.

**基本思想:**

- (1) 若  $G$  是 Euler 图, 则  $G$  的 Euler 闭迹便是最优邮路, 可用 Fleury 算法求得;
- (2) 若  $G$  不是 Euler 图, 则含有所有边的闭途径必须重复经过一些边, 最优邮路要求重复经过的边的权之和达到最小. 闭途径重复经过一些边, 实质上可看成给图  $G$  添加了一些重复边 (其权与原边的权相等), 最终消除了奇度顶点形成一个 Euler 图. 因此, 在这种情况下求最优邮路可分为两步进行: 首先给图  $G$  添加一些重复边得到 Euler 图  $G^*$ , 使得添加边的权之和  $\sum_{e \in F} w(e)$  最小, (其中  $F = E(G^*) - E(G)$ ); 然后用 Fleury 算法求  $G^*$  的一条 Euler 闭迹. 这样便得到  $G$  的最优邮路. 问题是: 如何给图  $G$  添加重复边得到 Euler 图  $G^*$ , 使得添加边的权之和  $\sum_{e \in F} w(e)$  最小?

下面介绍解决这一问题的两种方法. 为叙述方便, 将上述的 Euler 图  $G^*$  称为图  $G$  的**最优邮路 Euler 图**.  $G$  的任一条最优邮路对应一个最优邮路 Euler 图.

**方法一: 图上作业法**

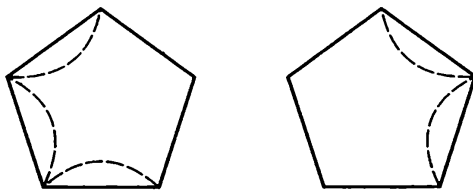
**定理 4.2.1** 设  $C$  是一条经过赋权连通图  $G$  的每条边至少一次的闭途径, 则  $C$  是  $G$  的最优邮路当且仅当  $C$  对应的最优邮路 Euler 图  $G^*$  满足:

- (1)  $G$  的每条边在  $G^*$  中至多重复出现一次;
- (2)  $G$  的每一个圈上, 在  $G^*$  中重复出现的边的权之和不超过该圈总权的一半.

**证明 必要性** 设  $C$  是最优邮路, 即  $C$  是它所对应的  $G^*$  的一个 Euler 闭迹.

先证 (1), 用反证法. 假设  $G$  中的边  $e = uv$  被  $C$  经过了  $m$  次, 且  $m \geq 3$ , 即在  $G^*$  中  $u, v$  之间有  $m$  条重边. 则在  $G^*$  中删去这  $m$  条边中的两条, 不改变  $u, v$  的度数的奇偶性. 所得图  $G'$  仍为 Euler 图, 但  $w(G') < w(G^*)$ , 即  $G'$  中的 Euler 闭迹是比  $C$  更优的一条投递路线, 这是矛盾的.

下证 (2), 也用反证法. 假设  $C_1$  是  $G$  中一个圈, 且在  $C$  所对应的 Euler 图  $G^*$  中,  $C_1$  上重复出现的边的权之和  $> C_1$  的权的一半. 将  $C_1$  上重复的边改为不重复而未重复的边改为重复边 (如下图所示).

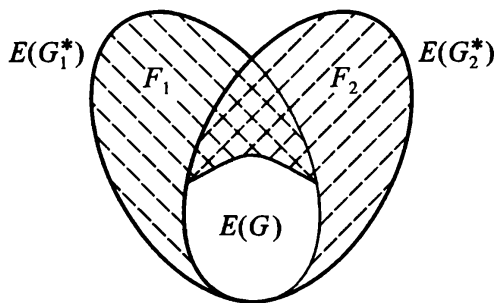


这样做不改变  $G^*$  中  $C_1$  上各顶点的度数的奇偶性. 设所得图为  $\tilde{G}$ , 则  $\tilde{G}$  仍是 Euler 图, 但  $w(\tilde{G}) < w(G)$ ,  $\tilde{G}$  的 Euler 闭迹比  $C$  更优. 这是不可能的.



**充分性** 设  $C_1$  是  $G$  的一条投递路线 (经过每条边至少一次的闭途径), 它对应的 Euler 图  $G_1^*$  满足 (1) 和 (2). 设  $C_2$  是  $G$  的一个最优邮路, 我们来证明  $w(C_1) = w(C_2)$ .

设  $G_2^*$  是  $C_2$  对应的 Euler 图 (由必要性知,  $G_2^*$  也满足 (1) 和 (2)),  $F_1$ 、 $F_2$  分别为  $G_1^*$  和  $G_2^*$  的重复边集合. 令  $F = (F_1 \cup F_2) - (F_1 \cap F_2)$ , 而  $[F]$  为  $F$  在  $G_1^* \cup G_2^*$  中的边导出子图 ( $[F]$  中的边全是  $F_1$  和  $F_2$  的边, 且无一边既在  $F_1$  中又在  $F_2$  中).  $F_1$ 、 $F_2$ 、 $E(G)$  和  $F$  的关系如下图所示, 其中  $F_1$ 、 $F_2$  分别是两个椭圆中的虚线部分, 而  $F$  是无交叉的虚线部分.



由于在从  $G$  得到  $G_1^*$  和  $G_2^*$  的过程中, 给一个顶点  $v$  添加边的条数的奇偶性与度数  $d_G(v)$  的奇偶性相同, 故  $[F]$  中各顶点的度数均为偶数. 这表明  $[F]$  各连通分支均为 Euler 图. 从而  $[F]$  是若干个无公共边的圈 (习题 4.7). 而且  $[F]$  中各圈均既有  $F_1$  的边又有  $F_2$  的边 (因  $F_1$  的边不会形成圈,  $F_2$  的也是). 由条件 (2),  $[F]$  中任一个圈  $C'$  上  $F_1$  的边的权之和与  $F_2$  的边的权之和均不超过  $\frac{1}{2}w(C')$ . 于是  $F_1$ 、 $F_2$  在  $C'$  上边的权之和必都等于  $\frac{1}{2}w(C')$ . 这说明  $[F]$  中属于  $F_1$  的边的权之和 =  $[F]$  中属于  $F_2$  的边的权之和. 因此  $w(G_2^*) = w(G_1^*)$ , 从而  $w(C_2) = w(C_1)$ . 证毕.

由定理 4.2.1 可得求最优邮路的一个算法, 称为奇偶点图上作业法. 算法口诀如下:

### 奇偶点图上作业法

先分奇偶点, 奇点对对连; 连线不重迭, 重迭要改变; 圈上连线长, 不得过半圈.

奇偶点图上作业法虽然通俗易懂, 但使用时需要检查图的每个圈, 对于有很多圈的图, 计算量会非常大, 因此实际当中用得较少.

### 方法二: Edmonds-Johnson 方法

**定理 4.2.2** 设  $G$  是赋权连通图,  $G$  中有  $2k$  个奇度顶点.  $G^*$  是  $G$  的一个最优邮路 Euler 图, 令  $E' = \{e | e \in E(G), \text{在由 } G \text{ 得到 } G^* \text{ 时, } e \text{ 被添加重复边}\}$ . 则边导出子图  $G[E']$  是以  $G$  的奇度顶点为起点和终点的  $k$  条无公共边的最短路之并.

**证明** 记  $H = G[E']$ . 在  $G$  中给  $E'$  的边添加重复边得到了无奇度顶点的图  $G^*$ , 这意味着在  $G$  的每个奇度顶点处, 添加了奇数条  $E'$  中的边; 而在  $G$  的每个偶度顶点处, 添加了偶数条  $E'$  中的边 (含零条). 因此,  $G$  中的奇度顶点在  $H$  中仍为奇度顶点,

而  $G$  中的偶度顶点要么在  $H$  中不出现, 要么在  $H$  中仍为偶度顶点. 可见,  $H$  中奇度顶点仍有  $2k$  个.

$H$  中不会出现圈. 否则, 若  $H$  中有一个圈  $C$ , 则也对应  $G^*$  中的一个圈, 从  $G^*$  中删去  $C$  上各边对应的重复边, 得到的图也是一个包含  $G$  的 Euler 图 (无奇度顶点), 但比  $G^*$  有更小的权值, 这与  $G^*$  的最优性矛盾.

在  $H$  中从任一个奇度顶点出发, 沿着未曾走过的边前行, 一直到无法继续时为止, 所经过的点和边形成一条迹  $P_1$ , 且终止点必定是  $H$  的奇度顶点. 由于  $H$  中无圈, 故  $P_1$  一定是连接两个奇度顶点的路, 而且是这两个奇度顶点间的最短路 (否则, 若有一条更短的路, 则用更短路上的边替换  $E'$  中  $P_1$  的边, 添加重复边, 可得到比  $G^*$  权值更小的 Euler 图, 这与  $G^*$  的最优性矛盾.)

再考虑图  $H_1 = H - E(P_1)$ . 显然  $H_1$  中有  $2k - 2$  个奇度顶点. 与以上同样道理, 可在  $H_1$  中找到一条连接两个奇度顶点的最短路  $P_2$ . 重复这个过程, 得到  $k$  条最短路  $P_1, P_2, \dots, P_k$  和  $k$  个子图  $H_0, H_1, \dots, H_{k-1}$ , 其中

$$H_0 = H, H_i = H_{i-1} - E(P_i), (i = 1, 2, \dots, k-1).$$

最终得到  $H_k = H_{k-1} - E(P_k)$ , 因  $H_k$  中无奇度顶点, 且不含圈, 故  $H_k$  是空图. 于是  $H = P_1 \cup P_2 \cup \dots \cup P_k$ . 证毕.

按照定理 4.2.2, 要想给  $G$  添加重复边得到最优邮路 Euler 图  $G^*$ , 必须沿着  $G$  中连接奇度顶点的  $k$  条无公共边的最短路添加重复边, 当然, 这些最短路的权之和应当尽可能地小. 基于这一原理, Edmonds 和 Johnson 于 1973 年给出了求解邮递员问题的有效算法<sup>[4]</sup>. 算法的基本思想是: 先求出  $G$  中所有  $2k$  个奇度顶点间的最短路. 为了从中选出  $k$  条权最小且无公共端点的最短路, 以  $G$  的所有奇度点为顶点构造赋权完全图  $K_{2k}$ , 两顶点连边上的权为这两个奇度点在  $G$  中最短路的权. 求赋权完全图  $K_{2k}$  的最小权完美匹配, 该完美匹配的  $k$  条边对应的  $k$  对顶点间的最短路必定无公共边 (否则, 若两条路有公共边, 则删去公共边后可得到两条连接奇度顶点的更短路, 从而对应出  $K_{2k}$  中权更小的完美匹配, 矛盾). 沿这  $k$  条最短路添加重复边, 便可得到最优邮路 Euler 图  $G^*$ . 然后找出  $G^*$  的 Euler 闭迹, 即得到最优邮路.

### Edmonds-Johnson 算法

输入: 赋权连通图  $G$ .

输出:  $G$  的最优邮路 (经过  $G$  的每条边至少一次的权最小的闭途径).

第 1 步: 若  $G$  中无奇度顶点, 令  $G^* = G$ , 转第 2 步; 否则转第 3 步.

第 2 步: 求  $G^*$  中的 Euler 闭迹, 并按该 Euler 闭迹输出  $G$  的最优邮路, 结束.

第 3 步: 求  $G$  中所有奇度顶点对之间的最短路.

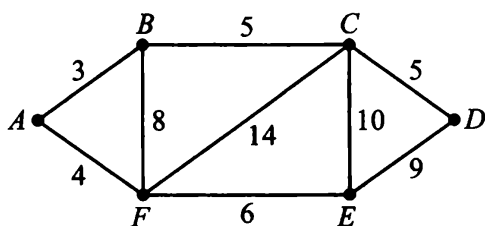
第 4 步: 以  $G$  中奇度顶点集  $V'$  为顶点集, 构作赋权完全图  $K_n$ , ( $n = |V'|$ ), 使得对  $\forall v_i, v_j \in V', K_n$  中边  $v_i v_j$  的权为  $G$  中  $v_i$  至  $v_j$  最短路的权.

第 5 步: 求  $K_n$  的最小权完美匹配  $M$ .

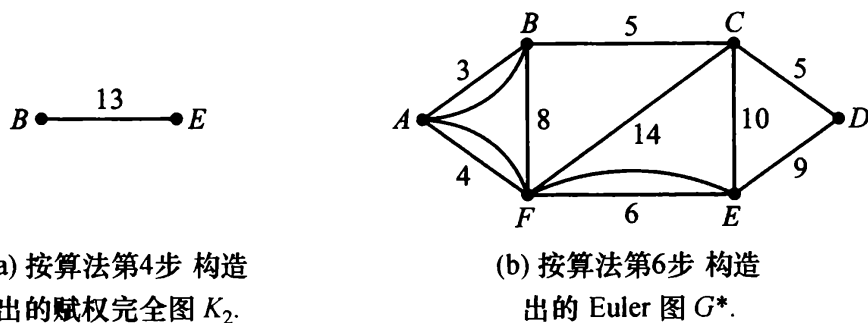
第 6 步:  $M$  中每条边所匹配的两点在  $G$  中都对应有一条最短路, 在  $G$  中沿这些最短路添加重复边, 得 Euler 图  $G^*$ , 转第 2 步.

考虑 Edmonds-Johnson 算法的计算复杂度. 注意图  $G^*$  与  $G$  的顶点数相同, 为  $\nu \cdot G^*$  的边数不超过  $\varepsilon(G)$  的 2 倍. 第 1 步的计算量为  $O(\varepsilon)$ . 第 2 步调用 Fleury 算法, 计算复杂度为  $O(\varepsilon)$ . 第 3 步若使用 Floyd 最短路算法, 计算量为  $O(\nu^3)$ . 第 4 步的计算量不超过  $O(\nu^2)$ . 第 5 步若使用最快的最大权匹配算法, 计算复杂度为  $O(\nu^3)$ . 第 6 步的计算量为  $O(\varepsilon)$ . 因此 Edmonds-Johnson 算法的计算复杂度为  $O(\nu^3)$ .

例 4.2.1 求下图的最优投递路线, 其中  $A$  点为邮局.



解 该图只有两个奇度点  $B$  和  $E$ ,  $B$  到  $E$  的最短路为  $BAFE$ , 权为 13. 按算法第 4 步, 令  $V' = \{B, E\}$ , 构造出赋权完全图  $K_2$  (如下图 a), 其最小权完美匹配由其唯一的边  $BE$  构成. 再按照算法的第 6 步, 沿匹配边  $BE$  在  $G$  中对应的最短路  $BAFE$  添加重复边, 得到相应的 Euler 图  $G^*$  (如图 b). 易求得  $G^*$  的 Euler 闭迹, 从而得到最优邮路.



(a) 按算法第 4 步构造出的赋权完全图  $K_2$ .

(b) 按算法第 6 步构造出的 Euler 图  $G^*$ .

有关中国邮递员问题的更多内容可参看文献 [1]、[2] 及 [7]~[12]

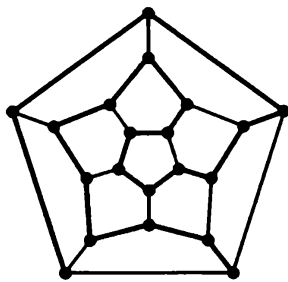
## §4.3 Hamilton 图

**定义 4.3.1** 经过图  $G$  的每个顶点恰一次的路称为  $G$  的 **Hamilton 路**, 简称为  $H$  路. 经过图  $G$  的每个顶点恰一次的圈称为  $G$  的 **Hamilton 圈**, 简称为  $H$  圈. 具有 Hamilton 圈的图称为 **Hamilton 图**, 简称为  $H$  图

Hamilton 图的研究起源于一种十二面体上的游戏. 1857 年, 爱尔兰著名数学家 William Rowan Hamilton 爵士 (他也是第一个给出复数的代数描述的人) 制作了一种

玩具, 它是一个木制的正十二面体, 在正十二面体的每个顶点上有一个木栓, 并标有世界著名城市的名字. 游戏者用一条细线从一个顶点出发, 设法沿着十二面体的棱找出一条路, 通过每个城市恰好一次, 最后回到出发点. 这个游戏当时称为 Icosian 游戏, 也称为周游世界游戏.

将正十二面体从一个面剖开并铺展到平面上得到的图形如下图所示, 称为十二面体图. 周游世界游戏用图论术语来说就是判断十二面体图是否为 Hamilton 图, 并设法找出其 Hamilton 圈. 其中一条 Hamilton 圈如图中粗边所示.



十二面体图是  $H$  图

判断一个图是否 Hamilton 图与判断一个图是否 Euler 图似乎很相似, 然而二者却有本质的不同. 目前为止尚没有找到判别一个图是不是 Hamilton 图的有效充分必要条件. 这是图论和计算机科学中未解决的重要难题之一.

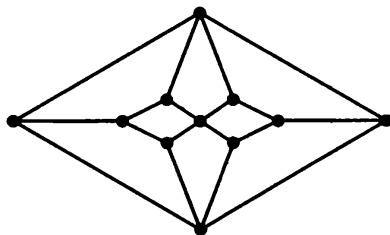
本节给出一些判别 Hamilton 图的经典的充分条件和必要条件.

### 一、必要条件

**定理 4.3.1** 设  $G$  是二部图, 若  $G$  是  $H$  图, 则  $G$  必有偶数个顶点.

**证明** 设  $G = (X, Y)$ , 由于  $G$  的边全在  $X$  和  $Y$  之间, 因此如果  $G$  有 Hamilton 圈  $C$ , 则  $G$  的所有顶点全在  $C$  上, 且必定是  $X$  的点和  $Y$  的点交替在  $C$  上出现, 因此  $G$  必有偶数个顶点. 证毕.

这个定理给出了一个二部图不是 Hamilton 图的简单判断条件: 如果一个二部图有奇数个顶点, 则它必定不是 Hamilton 图. 例如, 下列 Herschel 图是二部图, 但有奇数个顶点, 故不是  $H$  图.



Herschel 图不是  $H$  图

**定理 4.3.2** 若  $G$  是  $H$  图, 则对  $V(G)$  的每个非空真子集  $S$ , 均有: 连通分支数

$$W(G - S) \leq |S|.$$

**证明** 设  $C$  是  $G$  的一个  $H$  圈, 则对  $V(G)$  的每个非空真子集  $S$ , 均有

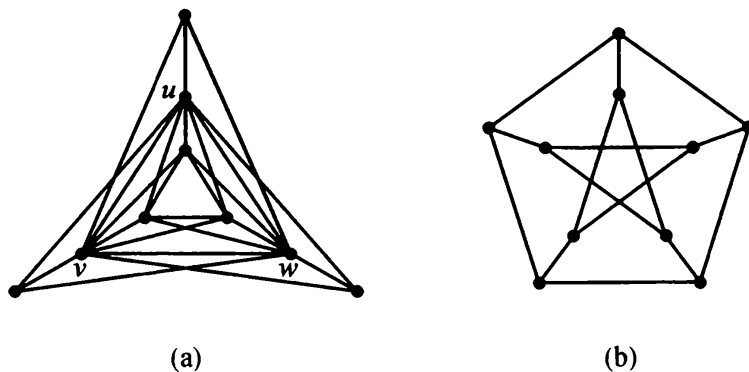
$$W(C - S) \leq |S|.$$

由于  $C - S$  是  $G - S$  的生成子图, 故  $W(G - S) \leq W(C - S) \leq |S|$ . 证毕.

利用定理 4.3.2 可判断下面 (a) 中的图不是  $H$  图. 事实上, 令  $S = \{u, v, w\}$ , 则

$$W(G - S) = 4 > |S|.$$

但无法用该定理给出的必要条件来判断 (b) 中的 Petersen 图不是  $H$  图.



## 二、充分条件

### 1. 度型条件

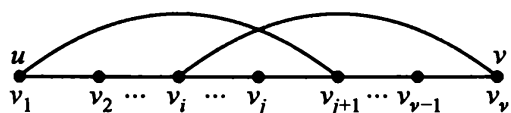
**定理 4.3.3 (Dirac, 1952)**<sup>[16]</sup> 若  $G$  是  $\nu$  阶简单图, 且  $\nu \geq 3, \delta \geq \frac{\nu}{2}$ , 则  $G$  是 Hamilton 图.

**证明** 用反证法: 假设定理不真. 令

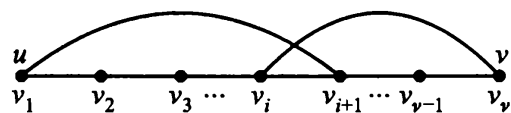
$A = \{G | G \text{ 是 } \nu (\geq 3) \text{ 个顶点的图, 最小度 } \delta(G) \geq \frac{\nu}{2}, \text{ 且 } G \text{ 是非 Hamilton 图}\}.$

则  $A \neq \emptyset$ . 取出  $A$  中边数最多的一个图, 记为  $G$  (因  $\nu$  点图的边数足够多时, 必成为 Hamilton 图, 故这样的  $G$  存在). 因  $\nu \geq 3$ , 故不是完全图 (否则  $G$  是 Hamilton 图). 设  $u$  和  $v$  是  $G$  的不相邻顶点. 由  $G$  的选择,  $G + uv$  是 Hamilton 图. 因  $G$  是非 Hamilton 图, 故  $G + uv$  的  $H$  圈必经过  $e = uv$ . 于是  $G$  中存在以  $u$  为起点  $v$  为终点的 Hamilton 路  $v_1 v_2 \cdots v_\nu$ , 这里  $v_1 = u, v_\nu = v$ . 令

$$S = \{v_j | uv_{j+1} \in E\}, \quad T = \{v_i | v_i v \in E\}.$$



由于  $v_\nu \notin S \cup T$ , 故  $|S \cup T| < \nu$ , 并且  $|S \cap T| = 0$  (因为若  $\exists v_i \in S \cap T$ , 则  $G$  将包含  $H$  圈  $v_1 v_2 \cdots v_i v_\nu v_{\nu-1} \cdots v_{i+1} v_1$ , 如下图所示. 矛盾).



故  $d(u) + d(v) = |S| + |T| = |S \cup T| + |S \cap T| < \nu$ , 这与  $\delta \geq \frac{\nu}{2}$  的前提矛盾. 证毕.

## 2. 闭包型条件

Ore 注意到在上述 Dirac 定理的证明过程中, 条件  $\delta \geq \frac{\nu}{2}$  仅仅用于说明对不相邻顶点  $u, v$ , 有  $d(u) + d(v) \geq \nu$  成立. 因此可以将图存在 Hamilton 圈的条件由  $\delta \geq \frac{\nu}{2}$  弱化为  $d(u) + d(v) \geq \nu$ . 这样便得到下述定理.

**定理 4.3.4 (Ore, 1960)** <sup>[17]</sup> 设  $G$  是简单图,  $u$  和  $v$  是  $G$  中两个不相邻的顶点, 且  $d(u) + d(v) \geq \nu$ , 则  $G$  是 Hamilton 图当且仅当  $G + uv$  是 Hamilton 图.

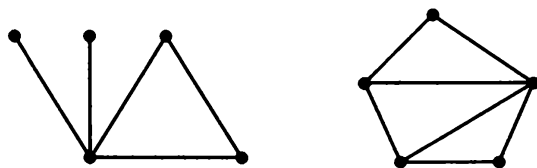
**证明** 必要性是显然的.

**充分性** 若  $G + uv$  是 Hamilton 图而  $G$  不是, 则与定理 4.3.3 一样可推出矛盾. 证毕.

Bondy 和 Chvátal 用更一般的形式表述了 Ore 观察到的实质, 得到了图存在长度为  $k$  的圈和其他子图的充分条件 ( $k = \nu$  时便是 Hamilton 圈)<sup>[18]</sup>. 他们用到了闭包的概念.

**定义 4.3.2** 图  $G$  的**闭包** (closure) 是指由如下方法所得之图: 反复连接  $G$  中度和不小于  $\nu$  的不相邻顶点对, 直到没有这样的顶点对为止. 图  $G$  的闭包通常记为  $C(G)$ .

例如, 对下列两图, 前一个图的闭包是它自己, 后一个图的闭包是完全图  $K_5$ .



**定理 4.3.5**  $G$  的闭包  $C(G)$  是唯一确定的.

**证明** 设  $G_1, G_2$  是按闭包构成方法从  $G$  所得的两个闭包. 用  $e_1, e_2, \cdots, e_m$  和  $f_1, f_2, \cdots, f_n$  分别表示在构作  $G_1, G_2$  过程中给  $G$  添加边的序列. 我们用反证法来证明每个  $e_i$  一定也是  $G_2$  的边, 且每个  $f_j$  一定也是  $G_1$  的边.

假设  $e_1, e_2, \dots, e_m$  中不属于  $G_2$  的边, 取序列中第一条不属于  $G_2$  的边设为  $e_k = uv$ , 令

$$H = G + \{e_1, e_2, \dots, e_{k-1}\}.$$

由  $G_1$  的定义知,  $d_H(u) + d_H(v) \geq \nu$ . 但  $H$  也是  $G_2$  的子图, 因此由  $d_H(u) + d_H(v) \geq \nu$  知  $d_{G_2}(u) + d_{G_2}(v) \geq \nu$ . 而由  $e_k$  的选择,  $u, v$  在  $G_2$  中是不相邻的, 这与  $G_2$  是闭包矛盾. 故每条  $e_i$  都必是  $G_2$  的边.

同理可证, 每条  $f_j$  一定是  $G_1$  的边. 这说明图  $G$  的闭包是唯一的. 证毕.

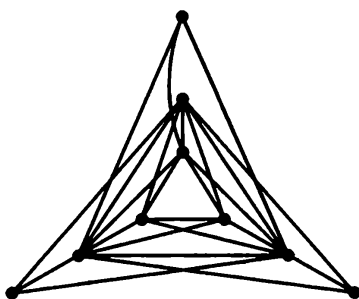
**定理 4.3.6 (Bondy & Chvátal, 1976)** [18] 一个简单图是 Hamilton 图当且仅当它的闭包是 Hamilton 图.

**证明** 在构作闭包过程中, 反复运用定理 4.3.4 即可. 证毕.

根据该定理, 下一个结论是显然的.

**推论 4.3.1** 设  $G$  是  $\nu \geq 3$  的简单图. 若闭包  $C(G)$  是完全图, 则  $G$  是 Hamilton 图.

定理 4.3.2 后的例子改动一条边后所得的如下图, 可以检验它的闭包是完全图, 因而是 Hamilton 图.



### 3. 度序列条件

设简单图  $G$  的顶点度为  $d_1 \leq d_2 \leq \dots \leq d_\nu$ , 则称序列  $(d_1, d_2, \dots, d_\nu)$  为  $G$  的度序列.

Chvátal 发现, 如果图  $G$  满足: 若有某些顶点的度比较小, 则就有相应另一些顶点的度足够大, 那么  $G$  就会成为 Hamilton 图. 具体来说, 设简单图  $G$  的度序列为  $(d_1, d_2, \dots, d_\nu)$ , 如果对任意正整数  $m < \frac{\nu}{2}$ , 要么  $d_m > m$ , 要么  $d_{\nu-m} \geq \nu - m$ , 则  $G$  是 Hamilton 图. 这个结论正式叙述如下.

**定理 4.3.7 (Chvátal, 1972)** [19] 设  $G$  是至少含有 3 个顶点的简单图, 其度序列为  $(d_1, d_2, \dots, d_\nu)$ , 这里  $d_1 \leq d_2 \leq \dots \leq d_\nu$ . 若不存在小于  $\frac{\nu}{2}$  的  $m$  使得  $d_m \leq m$  和  $d_{\nu-m} < \nu - m$  同时成立, 则  $G$  是 Hamilton 图.

**证明** 设  $G$  满足定理条件. 我们来证明  $G$  的闭包  $C(G)$  是完全图. 用反证法.

假设  $C(G)$  不是完全图. 设  $u, v$  是  $C(G)$  中两个不相邻顶点, 满足  $\bar{d}(u) \leq \bar{d}(v)$ , 且  $\bar{d}(u) + \bar{d}(v)$  尽可能大. ( $\bar{d}(v)$  表示  $v$  在  $C(G)$  中的度). 由  $C(G)$  的定义,  $\bar{d}(u) + \bar{d}(v) < \nu$ . 令

$$\bar{N}(u) = \{w \in V - \{u\} | w \text{ 在 } C(G) \text{ 中与 } u \text{ 不相邻}\};$$

$$\bar{N}(v) = \{w \in V - \{v\} | w \text{ 在 } C(G) \text{ 中与 } v \text{ 不相邻}\};$$

则  $|\bar{N}(u)| = \nu - 1 - \bar{d}(u)$ ,  $|\bar{N}(v)| = \nu - 1 - \bar{d}(v)$ . 记  $\bar{d}(u) = m$ , 则  $\bar{d}(v) < \nu - \bar{d}(u) = \nu - m$ , 且

$$|\bar{N}(u)| = \nu - 1 - m, |\bar{N}(v)| = \nu - 1 - \bar{d}(v) > \nu - 1 - (\nu - \bar{d}(u)) = m - 1 \quad (*)$$

另外, 由  $u, v$  的选择 ( $\bar{d}(u) + \bar{d}(v)$  尽可能大) 知,  $\bar{N}(u) \cup \{u\}$  中每个顶点在  $C(G)$  中的度不超过  $\bar{d}(v)$  (否则, 设  $v' \in \bar{N}(u)$ ,  $\bar{d}(v') > \bar{d}(v)$ , 则  $\bar{d}(u) + \bar{d}(v') > \bar{d}(u) + \bar{d}(v)$ , 当初应当选择  $u$  和  $v'$ ). 同理,  $\bar{N}(v)$  中每个顶点在  $C(G)$  中的度不超过  $\bar{d}(u)$ . 结合 (\*) 式可知:

$C(G)$  中至少有  $\nu - m$  个顶点, 其度  $\leq \bar{d}(v) < \nu - m$ , (比如  $\bar{N}(u) \cup \{u\}$  中的顶点);

$C(G)$  中至少有  $m$  个顶点, 其度  $\leq m$ , (比如  $\bar{N}(v)$  中的顶点).

由于  $G$  是  $C(G)$  的子图, 故上述结论对  $G$  也成立. 按照度序列的排列方式, 便有  $d_m \leq m$  和  $d_{\nu-m} < \nu - m$ , 并且  $m < \frac{\nu}{2}$  (因  $m = \bar{d}(u) \leq \bar{d}(v)$  及  $\bar{d}(u) + \bar{d}(v) < \nu$ ). 这与定理的条件矛盾. 该矛盾说明  $C(G)$  确实是完全图. 由推论 4.3.1,  $G$  是 Hamilton 图. 证毕.

关于 Hamilton 图的更多内容可参看文献 [20]~[22].

## §4.4 旅行商问题 (Traveling Salesman Problem, TSP)

**旅行商问题:** 有  $n$  个城镇, 各城镇间的路程不尽相等. 一个旅行商从这  $n$  个城镇中的某一城镇出发, 欲经过其余  $n - 1$  个城镇一次且仅一次, 最后回到出发点. 他应该如何设计旅行路线, 才能使所走路程最短?

**图论描述:** 将城镇作为顶点, 两顶点连边当且仅当对应的两城镇能直达, 每条边上以相应城镇间道路的里程作为权. 从而获得一个赋权图  $G$ . 旅行商问题现在可以叙述为:

给定赋权图  $G$ , 求  $G$  的一个权最小的 Hamilton 圈.

这一问题看似简单, 实际上含有两个困难的问题:

(1) 如何判定  $G$  是否有 Hamilton 圈;

(2) 在已知  $G$  是 Hamilton 图的情况下, 如何求出  $G$  的一个权最小的 Hamilton 圈来.



这两个问题目前尚未找到有效算法, 甚至不知道这样的有效算法是否存在. 事实上它们是 NPC 问题.

**转化:** 在  $G$  中所有不相邻的顶点间添加权为  $\infty$  的边, 将  $G$  化为赋权完全图. 问题转化为: 对赋权完全图  $K_n$ , 求其最小权 Hamilton 圈.

在  $K_n$  中, 共有  $\frac{1}{2}(n-1)!$  个不同的 Hamilton 圈 (每个点  $v$  到其余顶点有  $n-1$  条边). 如果一一计算各 Hamilton 圈的长度, 再逐个比较出权最小的一个, 则计算量太大. 当  $n$  较大时这种枚举算法无法实现.

对这样的 NPC 问题, 一般解决方法是设计较好的近似算法, 求其近似最优解. 下面介绍三种求最小权 Hamilton 圈的近似算法.

### 近似算法 1— 邻近点法

**算法基本思想:** 从任一顶点出发遍历  $K_n$  的所有顶点. 在每次到达的顶点处, 总是选择尚未经过的邻点中最近的一个前行, 直至遍历  $K_n$  的所有顶点后回到出发点.

这种在每一步都追求最优的算法称为贪婪算法. 贪婪算法的结果未必是最优的.

邻近点算法的步骤如下.

**邻近点算法:** 求  $K_n$  中最小权 Hamilton 圈.

**输入:** 赋权完全图  $K_n$ .

**输出:** 近似的最小权 Hamilton 圈.

第 1 步: 任选一点  $v_1 \in V$ , 令  $P_1 = v_1$ ,  $i := 1$ .

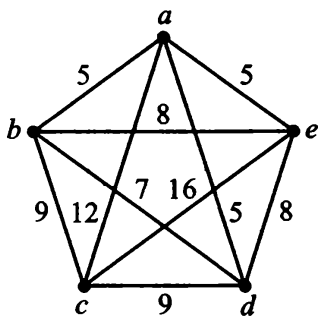
第 2 步: 设已得到  $P_i = v_1 v_2 \cdots v_i$ , 选取  $v_{i+1} \in V \setminus \{v_1, v_2, \cdots, v_i\}$  使得权  $w(v_i v_{i+1})$  最小.

第 3 步: 若  $i = n$ , 则停止,  $C = P_n + v_n v_1 = v_1 v_2 \cdots v_n v_1$  是一条近似的最小权 Hamilton 圈; 否则  $i := i + 1$ , 转第 2 步.

邻近点算法实质上是一个深度优先搜索算法. 在行进到第  $i$  个点处时, 需要在其  $n-i$  个未访问的邻点中选出一个最近的 (比较连边上的权值), 这需要  $n-i-1$  次比较.

因此邻近点算法的计算复杂度为  $\sum_{i=1}^{n-1} (n-i-1) = O(n^2)$ .

**例 4.4.1** 求如下赋权完全图的最小权 Hamilton 圈.



解 因  $n = 5$ , 故存在  $\frac{1}{2} \times 4! = 12$  个不同的 Hamilton 圈. 由计算可知  $abcdea$ ,  $adcbea$  是最优解, 权为 36; 而  $abdeca$  和  $acebda$  是最差解, 权为 48.

按邻近点法来求:

比如从  $a$  点出发, 可得四个近似解:  $abdeca$ , 权为 48;  $adbeca$ , 权为 48;  $aebdca$ , 权为 41;  $aedbca$ , 权为 41.

如果从  $b$  点出发, 可得 2 个近似解:  $badecb$ , 权为 43;  $baedcb$ , 权为 36.

可见, 邻近点法求得的近似解精度不高, 且结果的精度与初始点选择有关.

评判一个近似算法好坏的主要指标是它的求解精度. 衡量近似算法的精度通常有两种方法, 一种称为绝对性能 (absolute performance), 它是算法对所有问题实例的近似值与最优值绝对误差的最大值, 即  $\max_I |A(I) - \text{Opt}(I)|$ , 其中  $I$  是算法适用的所有实例,  $\text{Opt}(I)$  表示实例  $I$  的最优值,  $A(I)$  表示对实例  $I$  运行算法得到的最坏情况近似值. 另一种称为相对性能 (relative performance), 也称为近似比 (approximate performance ratio). 对最小化问题而言, 算法的近似比是算法对所有问题实例的近似值与最优值之比的最大值, 即  $\max_I \left| \frac{A(I)}{\text{Opt}(I)} \right|$ . 近似比越接近 1, 说明算法精度越高. 近似算法的绝对性能和相对性能往往难以精确算得, 一般只能估计它们的尽可能小的上界值.

对邻近点算法, 可以证明<sup>[12]</sup>, 若赋权完全图  $K_n$  各边的权均为正数, 且权满足三角不等式: 对于任意的  $v_i, v_j, v_k \in V, w(v_i v_j) + w(v_j v_k) \geq w(v_i v_k)$ , 则

$$\frac{1}{3} \log_2(n+1) + \frac{4}{9} < \max_I \left| \frac{A(I)}{\text{Opt}(I)} \right| \leq \frac{1}{2} (\lceil \log_2 n \rceil + 1).$$

可见, 邻近点算法的近似比与  $n$  有关, 当  $n \rightarrow \infty$  时, 近似比  $\max_I \left| \frac{A(I)}{\text{Opt}(I)} \right| \rightarrow \infty$ . 因此, 邻近点算法是一个性能很差的近似算法.

### 近似算法 2—最小生成树法

**算法基本思想:** 先求出赋权完全图  $K_n$  的一棵最小生成树  $T$ , 给  $T$  的每条边添加一条重边, 使其成为 Euler 图. 然后沿 Euler 图的一条 Euler 闭迹遍历  $K_n$  的所有顶点. 在每个顶点处, 若沿 Euler 闭迹前行时遇到的顶点已被访问过, 则跳过它直接访问 Euler 闭迹的下一个顶点. 直至遍历  $K_n$  的所有顶点后回到出发点. 注意我们的对象是一个完全图, 故上述跳跃方式的访问总是可行的, 且最终能得到一个 Hamilton 圈.

最小生成树法的步骤如下.

**最小生成树法:** 求  $K_n$  中最小权 Hamilton 圈.

输入: 赋权完全图  $K_n$ .

输出: 近似的最小权 Hamilton 圈.

第 1 步: 求赋权完全图  $K_n$  的一棵最小生成树  $T$ ;

第 2 步: 给  $T$  中各边均添加一条重边, 设所得图为  $G^*$ ;

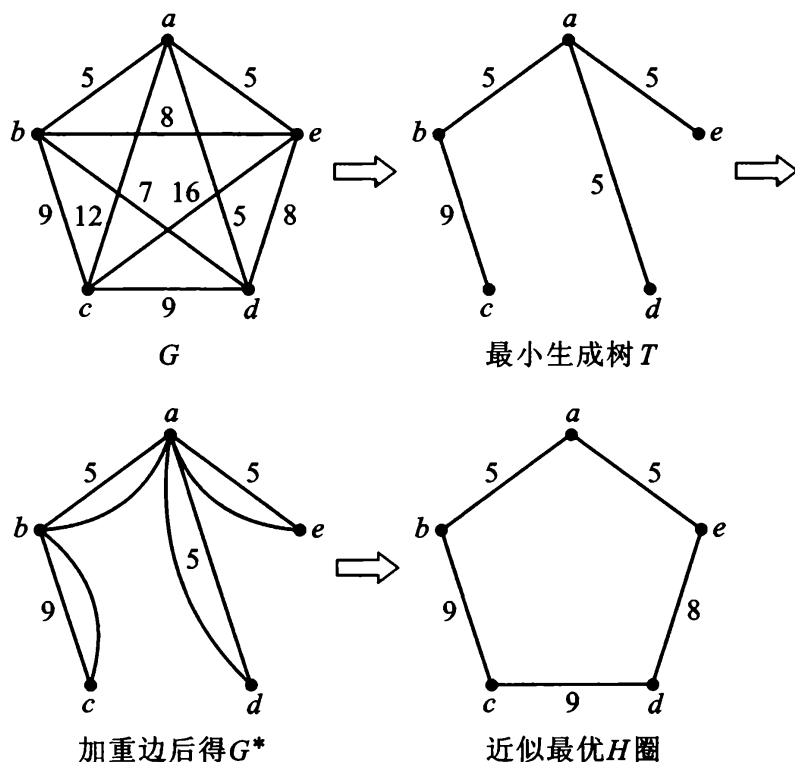
第 3 步: 求  $G^*$  的从某点  $v$  出发的一条 Euler 闭迹  $E_v$ ;

第 4 步: 按如下方法求出  $G$  的一条 Hamilton 路: 从顶点  $v$  出发, 沿  $E_v$  访问  $G^*$  中各顶点. 在此过程中, 一旦遇到已访问过的顶点, 就跳过它直接走到 Euler 环游上的下一个顶点. 直到访问完所有顶点为止. 输出这样的访问所形成的 Hamilton 圈, 算法停止.

最小生成树法的计算量主要在第 1 步和第 3 步. 根据已知的最小生成树算法和求 Euler 闭迹的 Fleury 算法, 可知最小生成树法的计算复杂度为  $O(n^2)$ .

**例 4.4.2** 用最小生成树法求如下赋权完全图的最小权 Hamilton 圈.

**解** 求解过程图示如下. 先求出一棵最小生成树  $T$ ,  $T$  的各边加重边后得  $G^*$ ,  $G^*$  从  $a$  点出发的一条 Euler 闭迹为  $C = ae ad ab cb a$ , 从  $a$  点出发沿  $C$  依次访问各点, 遇到已访问过的点时直接跳到  $C$  的下一个点, 这样便得到近似的 minimum 权 Hamilton 圈  $aedcba$ .



**注** (1) 若  $K_n$  有不只一棵最小生成树, 则最小生成树的选择会影响最终的结果 (所求出的  $H$  圈的权).

(2) Euler 闭迹的行走路线也会影响最终结果.

(3) 选择 Euler 闭迹的不同起点也会影响最终的结果.

**定理 4.4.1** 设赋权完全图  $k_n$  的各边之权均为正数, 且对  $\forall v_i, v_j, v_k \in V$ , 都有  $w(v_i v_j) + w(v_j v_k) \geq w(v_i v_k)$ , 则上述最小生成树法的近似比  $\frac{c_0}{c^*} < 2$ . 这里  $c_0$  是最小生成树法所得的 Hamilton 圈  $H$  的权 (近似最优值), 而  $c^*$  是最小权  $k_n$  的 Hamilton 圈  $H^*$  的权.

**证明** 设  $T$  和  $G^*$  如算法所述. 设  $E$  是  $G^*$  中任一条 Euler 闭迹. 由于对  $\forall v_i, v_j, v_k \in V$ , 都有  $w(v_i v_j) + w(v_j v_k) \geq w(v_i v_k)$ , 故  $w(H) = c_0 \leq w(E) = 2w(T)$ . 此外, 设  $T'$  是  $G$  的最小权 Hamilton 圈  $H^*$  删去任一边后所得的生成树, 则  $w(T) \leq w(T') < c^*$ . 因而  $c_0 < 2c^*$ . 证毕.

### 近似算法 3— 最小权匹配法 (Christofides 算法)<sup>[23]</sup>

**算法基本思想:** 先求出赋权完全图  $K_n$  的一棵最小生成树  $T$ , 对  $T$  中的所有奇度顶点, 求出它们在  $K_n$  中的最小权完美匹配, 将所得匹配边添加到  $T$  上 (可能会出现重边), 使其成为 Euler 图. 然后沿 Euler 图的一条 Euler 闭迹遍历  $K_n$  的所有顶点. 在每个顶点处, 若沿 Euler 闭迹前行时遇到的顶点已被访问过, 则跳过它直接访问 Euler 闭迹的下一个顶点. 直至遍历  $K_n$  的所有顶点后回到出发点.

最小权匹配法的步骤如下:

**最小权匹配法 (Christofides 算法):** 求  $K_n$  中最小权 Hamilton 圈.

**输入:** 赋权完全图  $K_n$ .

**输出:** 近似的 minimum 权 Hamilton 圈.

**第 1 步:** 求赋权完全图  $K_n$  的一棵最小生成树  $T$ .

**第 2 步:** 设  $T$  中奇度顶点的集合为  $V' = \{v_1, v_2, \dots, v_{2k}\}$ . 求  $V'$  在  $K_n$  中的导出子图  $G[V'] = K_{2k}$  的最小权完美匹配, 将得到的  $k$  条边添加到  $T$  上, 得 Euler 图  $G^*$ .

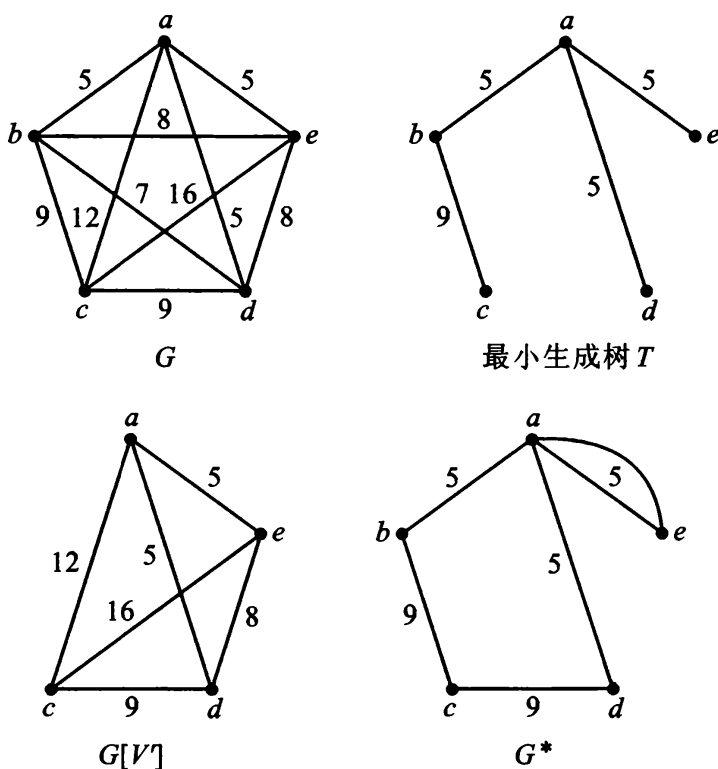
**第 3 步:** 求出  $G^*$  的从某顶点  $v$  出发的一条 Euler 闭迹  $E_v$ .

**第 4 步:** 从  $v$  出发沿  $E_v$  访问  $G^*$  的各个顶点. 在此过程中, 一旦遇到已访问过的顶点, 就跳过它直接走到 Euler 闭迹上的下一个顶点. 直到访问完所有顶点为止. 输出这样的访问所形成的 Hamilton 圈, 算法停止.

最小权匹配法的计算量主要在第 1 步、第 2 步和第 3 步. 执行第 1 步求最小生成树的计算量为  $O(n^2)$ ; 执行第 2 步时若使用最快的匹配算法, 计算量为  $O(n^3)$ ; 执行第 3 步使用 Fleury 算法, 计算量为  $O(\epsilon)$ . 因此, 最小权匹配法的计算复杂度为  $O(n^3)$ .

**例 4.4.3** 用最小权匹配法求如下图所示的赋权完全图  $G$  的最小权 Hamilton 圈.

**解** 求解过程图示如下. 先求出一棵最小生成树  $T$ , 再求出  $T$  中奇度顶点集合  $V' = \{a, c, d, e\}$  在  $G = K_5$  中导出的子图  $G[V']$ , 并求出  $G[V']$  的最小权完美匹配  $\{ae, cd\}$ . 将这两条匹配边添加到  $T$  上, 得 Euler 图  $G^*$ .  $G^*$  从  $a$  点出发的一条 Euler 闭迹为  $C = aeadcba$ , 从  $a$  点出发沿  $C$  依次访问各点, 遇到已访问过的点时直接跳到  $C$  的下一个点, 这样便得到近似的 minimum 权 Hamilton 圈  $aedcba$ .



**定理 4.4.2** 设赋权完全图  $K_n$  的各边之权均为正数, 且对  $\forall v_i, v_j, v_k \in V$ , 都有

$$w(v_i v_j) + w(v_j v_k) \geq w(v_i v_k),$$

则上述最小权匹配法的近似比  $\frac{c_0}{c^*} < \frac{3}{2}$ . 这里  $c_0$  是最小权匹配法所得的 Hamilton 圈  $H$  的权 (近似最优值), 而  $c^*$  是  $K_n$  的最小权 Hamilton 圈  $H^*$  的权.

**证明** 先证四条事实:

(1) 设  $T$  是算法第 1 步中获得的最小生成树, 因赋权完全图  $K_n$  的最小权 Hamilton 圈  $H^*$  去掉一条边后也得到  $K_n$  的一棵生成树, 故  $w(T) < c^*$ .

(2) 设  $E_v$  是算法第 3 步中得出的 Euler 闭迹. 由于对  $\forall v_i, v_j, v_k \in V$ , 都有  $w(v_i v_j) + w(v_j v_k) \geq w(v_i v_k)$ , 故算法最终获得的 Hamilton 圈的权  $c_0 \leq w(E_v)$ .

(3) 设  $G[V'] = K_{2k}$  中最小权 Hamilton 圈的权为  $c'$ . 因  $G[V']$  是完全图  $K_n$  的子图,  $G[V']$  的 Hamilton 圈可由  $K_n$  的 Hamilton 圈通过“去二添一”手续得到, 故  $c' \leq c^*$ .

(4) 因  $G[V']$  中最小权 Hamilton 圈上交替换取边可得  $G[V']$  的一个完美匹配, 故  $G[V']$  中最小权完美匹配  $M$  中各边权之和  $w(M) \leq \frac{c'}{2}$ .

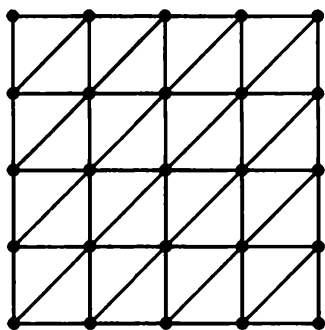
由以上四条, 得  $c_0 \leq w(E_v) = w(T) + w(M) < c^* + \frac{c^*}{2} = \frac{3}{2}c^*$ . 证毕.

TSP 问题是一个内容十分丰富而又非常活跃的研究方向. 对 TSP 问题研究情况的进一步了解可参看文献 [10]~[12], [21] 及 [23]~[43], 其中 [26] 收集了 TSP 方面的大量文献. 特别要指出的是, 作为 TSP 问题研究中的一个重大进展, S. Arora 对欧氏空间中的 TSP 问题找到了多项式时间近似方案 (PTAS, Polynomial time approximation

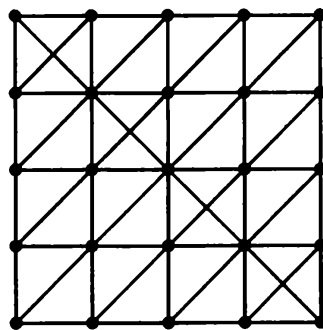
schemes)<sup>[33][34]</sup>.

### 习题四

- 4.1 证明: 一个连通图可不超过  $k$  笔画完当且仅当它最多有  $2k$  个奇度顶点.
- 4.2 确定  $m$  和  $n$  的值使得完全二部图  $K_{m,n}$  是 Euler 图.
- 4.3 证明: 若二部图  $G$  是 Euler 图, 则  $G$  有偶数条边.
- 4.4 证明: 图的每个顶点的度都是偶数当且仅当  $G$  的每个块都是 Euler 图.
- 4.5 是否存在偶数个顶点、奇数条边的 Euler 简单图? 若不存在, 给出理由; 若存在, 举出例子.
- 4.6 证明或反驳:
  - (1) 顶点的度全为偶数的图必定是 Euler 图;
  - (2) Euler 图必定是连通图.
- 4.7 证明: 连通图  $G$  是 Euler 图的充分必要条件是  $G$  可表成无公共边的圈的并.
- 4.8 下列两图是否分别可以一笔画? 若可以, 找出画法; 若不可以, 说明理由.

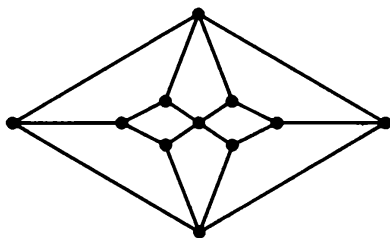


$G_1$

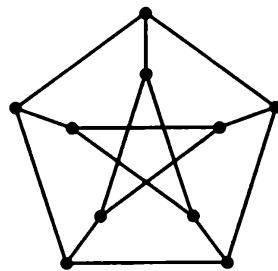


$G_2$

- 4.9 判断 Herschel 图和 Petersen 图分别能几笔画成.



Herschel 图



Petersen 图

- 4.10 给出一个算法, 用以在有 Euler 迹的图中求一条 Euler 迹.
- 4.11 判断下图 1 是否 Euler 图. 若是, 用 Fleury 算法求其一个 Euler 环游.

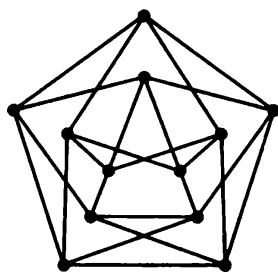


图 1

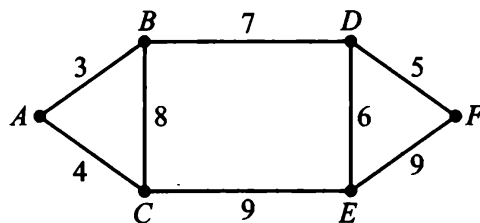
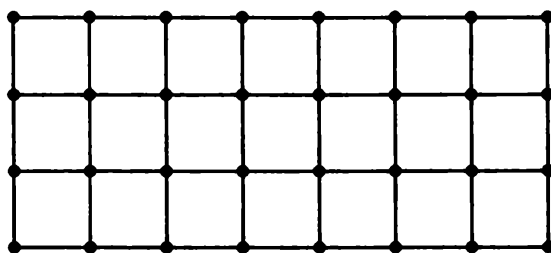
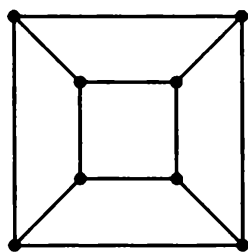


图 2

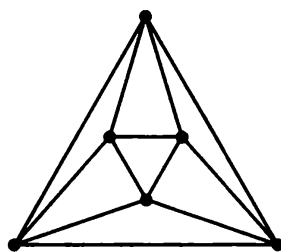
- 4.12 求图 2 的一条最优邮路.
- 4.13 多米诺骨牌每一块上有 0,1,2,3,4,5,6 七个数字之一 (0 用  $O$  表示, 其他数字用点数表示). 将标有不同数字的多米诺骨牌组成骨牌对子, 如  $(0,1)$ ,  $(2,4)$  等 (无序对), 共可组成  $\frac{7 \times 6}{2} = 21$  个不同的对子. 现要将这 21 个对子首尾相连接起来, 使得对任何两个相邻的对子, 前一个对子的尾数与后一个对子的首数相等. 使用图论方法考虑: 是否能按上述规则将这 21 个骨牌对子连成一个圆圈?
- 4.14 用丝线将  $m \times n$  颗珍珠串成一个网 (如图), 现在想将一些丝线段剪断, 得到一个由这些珍珠做成的项链, 问  $m$  和  $n$  应满足什么条件?



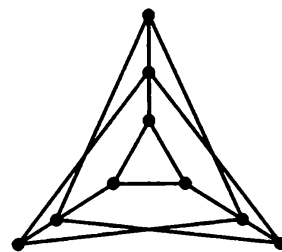
- 4.15 判断下列三个图是否 Hamilton 图. 若是, 找出一条 Hamilton 圈; 若不是, 说明理由.



$G_1$



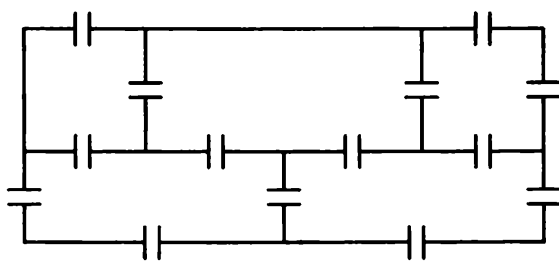
$G_2$



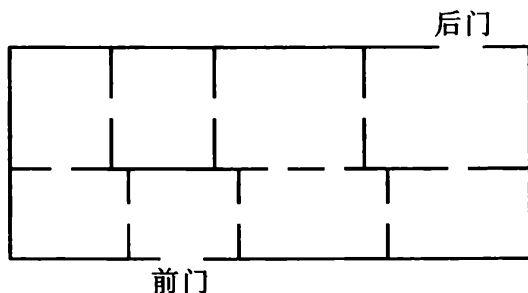
$G_3$

- 4.16 设  $G$  是二部图  $G = (X, Y)$ . 证明: 若  $|X| \neq |Y|$ , 则  $G$  不是 Hamilton 图.
- 4.17 证明 Petersen 图不是 Hamilton 图 (图见题 9).
- 4.18 试给出一种方法, 将完全二部图  $K_{n,n}$  分解为  $\lfloor \frac{n}{2} \rfloor$  个两两无公共边的 Hamilton 圈的并.
- 4.19 证明: 若  $G$  不是 2- 连通图, 则  $G$  不是 Hamilton 图.

- 4.20 设  $k \geq 2$ , 证明顶点数为  $2k - 1$  的  $k$  正则图是 Hamilton 图.
- 4.21 若图  $G$  的任二顶点间有 Hamilton 路, 则称  $G$  是 Hamilton 连通图. 证明:  $G$  是 Hamilton 连通图且  $\nu(G) \geq 4$ , 则  $\varepsilon(G) \geq \left\lfloor \frac{3\nu + 1}{2} \right\rfloor$ .
- 4.22 证明: 若  $G$  有 Hamilton 路, 则对  $V(G)$  的每个真子集  $S$ , 有  $W(G - S) \leq |S| + 1$ .
- 4.23 证明: 简单图  $G$  有 Hamilton 路当且仅当联图  $G \vee K_1$  有 Hamilton 圈.
- 4.24 设简单图  $G$  的度序列为  $(d_1, d_2, \dots, d_\nu)$ , 其中  $d_1 \leq d_2 \leq \dots \leq d_\nu$ . 证明: 若对任意正整数  $m < \frac{\nu + 1}{2}$ , 要么  $d_m > m$ , 要么  $d_{\nu - m + 1} \geq \nu - m$ , 则  $G$  有 Hamilton 路.
- 4.25 教室里有 6 排椅子, 每排 5 张椅子, 每张椅子坐一名学生. 现在班主任老师考虑每周调整一次学生们的座位, 使得每个学生都调到本次调整前与他相邻的某个座位上 (前后左右视为相邻), 一个学期内每个学生都不能调回到他已经坐过的座位上. 请帮老师制定一个调整方案. 如果老师现在改变了主意, 希望每次将每个学生调到本次调整前与他不相邻的某个座位上, 且一个学期内每个学生都不能调回到他已经坐过的座位上. 问是否可行? 应怎样调整?
- 4.26 若围一张圆桌坐着至少 6 个人, 那么一定可以调整他们的位置, 使得每个人两侧都挨着新邻居.
- 4.27 下图是某超市的平面示意图, 一名保安人员在一次巡逻中欲穿过每道门一次且仅一次并回到出发点, 问他是否能够做到? 试用图论方法加以解释.



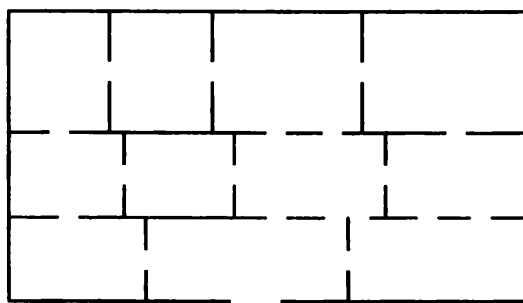
- 4.28 某楼层的平面结构示意图如下. 试说明从前门进去穿过每个门恰好一次且从后门出去是不可能的.



- 4.29 一个博物馆称为设计良好的, 如果存在经过每个房间恰好一次的参观路线. 某博



物馆的楼层平面设计如下图, 试证明这个博物馆的设计不是良好的.



## 参考文献

- [1] L. Euler, Solutio problematic ad geometriam situs pertinentis. *Comment. Academiae Sci. I. Petropolitanae*, 8(1736), 128-140. (appeared in 1741).
- [2] S. Toida, Properties of an Euler graph. *J. Franklin Inst.*, 295(1973), 343-345.
- [3] T.A. Mckee, Recharacterizing Eulerian: intimations of new duality. *Discrete Mathematics*, 51(1984), 327-342.
- [4] J. Edmonds and E. L. Johnson, Matching, Euler tours and the Chinese postman, *Mathematical Programming*, 5(1973), 88-124.
- [5] H. Fleischner, Eulerian graphs, in *Selected Topics in Graph Theory II* (L. W. Beineke, and R.J. Wilson eds.), Academic Press, London, (1983), 17-54.
- [6] R. Arratia, B. Bollobás, D. Coppersmith and G. B. Sorkin, Euler circuits and DNA sequencing by hybridization, *Discrete Applied Mathematics*, 104(2000), 63-96.
- [7] D. Marx, Eulerian disjoint paths problem in grid graphs is NP-complete, *Discrete Applied Mathematics*, 143(2004), 336-341.
- [8] Dengxin Li, Hong-Jian Lai and Mingquan Zhan, Eulerian subgraphs and Hamilton-connected line graphs, *Discrete Applied Mathematics*, 145(2005), 422-428.
- [9] Weiqi Luo, Zhi-Hong Chen, Wei-Guo Chen, Spanning trails containing given edges. *Discrete Mathematics*, 306(2006), 87-98.
- [10] N. Christofides, *Graph Theory-An Algorithmic Approach*, Academic Press, Inc., New York, 1990.
- [11] G. Chartrand, and O.R. Oellermann, *Applied and Algorithmic Graph Theory*, McGraw-Hill, Inc., New York, 1993.
- [12] 谢政. 网络算法与复杂性理论. 2 版. 北京: 国防科技大学出版社, 2003.
- [13] H. Hamers, P. Borm, R. Van de Leensel, and S. Tijs, Cost allocation in the Chinese postman problem, *Eur. J. Oper. Res.*, 118(1999) 153-163.
- [14] D. Granot, H. Hamers, On the equivalence between some local and global Chinese postman and traveling salesman graphs, *Discrete Applied Mathematics*, 134(2004), 67-76.

- [15] B. Raghavachari, J. Veerasamy, A 3/2-approximation algorithm for the mixed postman problem, *SIAM Journal on Discrete Mathematics*, 12(1999), 425-433.
- [16] G. A. Dirac, Some theorems on abstract graphs. *Proc. Lond. Math. Soc.*, 2(1952), 69-81.
- [17] O. Ore, Note on Hamilton circuits, *Amer. Math. Monthly*, 67(1960), 55.
- [18] J. A. Bondy, and V. Chvátal, A method in graph theory, *Discrete Mathematics*, 15(1976), 111-136.
- [19] V. Chvátal, On Hamilton's ideals, *J. Combinatorial Theory(B)*, 12(1972), 163-168.
- [20] V. Chvátal, Hamiltonian cycles, in *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization* (ed. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, D.B. Shmoys), Wiley, (1985), 403-429.
- [21] J.-C. Bermond, Hamiltonian graph, in *Selected Topics in Graph Theory* (L. W. Beineke, and R. J. Wilson eds.), Academic Press, London, (1978) 127-168.
- [22] R. J. Faudree, R. H. Schelp, A. Saito, I. Schiermeyer, Degree conditions for Hamiltonicity: Counting the number of missing edges. *Discrete Mathematics*, 307(2007), 873-877.
- [23] N. Christofides, Worst-case analysis of a new heuristic for the traveling salesman problem, *Technical Report, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA*(1976).
- [24] B. Golden, W. Stewart, Empiric analysis of heuristics, in *The Traveling Salesman Problem*, (editors: E. L. Lawler, J.K. Lenstra, A.H.G. Rinnooy-Kan and D.B. Shmoys), Wiley, New York, 1985, 207-250.
- [25] E.L. Lawler, J.K. Lenstrar, A.H.G. Rinnooy and D.B. Shmoys, *The Traveling Salesman Problem*, New York, Wiley-Interscience, 1986.
- [26] G. Reinelt, TSPLIB-A traveling salesman problem library, *ORSA J. Computing*, 3(1991) 376-384.
- [27] J. Potters, I. Curiel, S. Tijs, Traveling salesman games, *Math. Programming*, 53(1992) 199-211.
- [28] J. Bentley, Fast algorithms for geometric traveling salesman problem, *ORSA J. on Computing*, 4(1992) 387-411.
- [29] C.H. Papadimitriou, M. Yannakakis, The traveling salesman problem with distances one and two, *Mathematics of Operatons Research*, 18(1993), 1-12.
- [30] D. Fogel, Appling evolutionary programming to selected traveling salesman problems, *Cybernetics and Systems: An International Journal*, 24(1993), 27-36.
- [31] G. Reinelt, *The Travaling Salesman: Computational Solutions for TSP Applications*, Springer-Verlag, Berlin, 1994.

- 
- [32] C.S. Mata and J. Mitchell, Approximation algorithms for Geometric tour and network problems, in *Proceeding of 11<sup>th</sup> ACM Symposium on Computing Geomrtry*, 1995, 360-369.
- [33] S. Arora, Nearly linear time approximation schemes for Euclidean TSP and other geometric problems, *Proceedings of 38<sup>th</sup> Annual Symposium Foundation of Computer Science*, Oct. 1997.
- [34] S. Arora, Polynomial time approximation schemes for Euclidean TSP and other geometric problems, *Proceedings of 37<sup>th</sup> Annual Symposium Foundation of Computer Science*, Oct. 1996. Also see *J. ACM*, 45(1998),753-782.
- [35] D. Applegate, R. Bixby, On the Solution of traveling salesman problems, *Documenta Mathematica J. DMV, Extra Volume ICM 1998 III*, (1998) 645-656.
- [36] B. Chandra, H. Karloff, C.Tovey, New results on the old  $k$ -opt algorithm for the traveling salesman problem, *SIAM J. Computing*, 28(1999), 1998-2029.
- [37] S. N. Kabadi, New polynomially solvable classes and a new heuristic for the traveling salesman problem and its generalization, *Discrete Applied Mathematics*, 119(2002), 149-167.
- [38] D. Blokh and E. Levner, An approximation algorithm with performance guarantees for the maximum traveling salesman problem on special matrices, *Discrete Applied Mathematics*, 119(2002), 139-148.
- [39] M. Lewenstein and M. Sviridenko, 5/8 Approximation Algorithm for the Maximum Asymmetric TSP, *SIAM Journal on Discrete Mathematics*, 17(2003-2004), 237-248.
- [40] H. Hernández-Pérez and J.-J. Salazar-González, A branch-and-cut algorithm for a traveling salesman problem with pickup and delivery, *Discrete Applied Mathematics*, 145 (2004), 126-139.
- [41] H. Nagamochi and K. Okada, A faster 2-approximation algorithm for the minmax  $p$ -traveling salesmen problem on a tree, *Discrete Applied Mathematics*, 140(2004), 103-114.
- [42] L. Gouveia, P. Pesneau, On extended formulations for the precedence constrained asymmetric traveling salesman problem, *Networks*, 48(2)(2006), 77-89.
- [43] M. Oswald, G. Reinelt, D.O. Theis, On the graphical relaxation of the symmetric traveling salesman polytope, *Mathematical Programming*, 110(2007), 175-193.

# 第五章 支配集、独立集、覆盖集 和 Ramsey 数

---

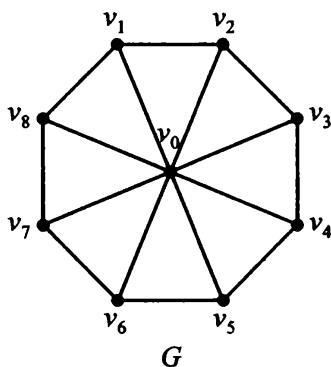
本章讨论图中具有某种特性的顶点子集和边子集, 以及它们之间的关系. 本章所讨论的图均为简单图.

## §5.1 支配集、点独立集、点覆盖集

### 一、支配集 (domination set)

**定义 5.1.1** 设  $D \subseteq V(G)$ , 若对  $\forall u \in V(G)$ , 要么  $u \in D$ , 要么  $u$  与  $D$  中的某些顶点相邻, 则称  $D$  为图  $G$  的一个支配集. 如果一个支配集的任何真子集都不是支配集, 则称它为极小支配集. 图  $G$  的含顶点最少的支配集称为最小支配集. 最小支配集的顶点个数称为  $G$  的支配数, 记为  $\gamma(G)$  或  $\gamma$ .

例如, 在下图中,  $D_0 = \{v_0\}$ ,  $D_1 = \{v_1, v_4, v_7\}$ ,  $D_2 = \{v_1, v_3, v_5, v_6\}$  都是  $G$  的支配集, 前两个是极小支配集,  $D_0$  是最小支配集,  $\gamma(G) = 1$ .



**注** (1) 最小支配集必是一个极小支配集, 反之不然.

(2) 任一支配集必含有一个极小支配集.

(3) 对于非空图, 极小支配集不唯一, 最小支配集往往也不唯一.

(4) 对二部图  $G = (X, Y)$ ,  $X$  和  $Y$  都是支配集.

(5) 若图  $G$  有完美匹配  $M^*$ , 则从  $M^*$  中每边取一个端点构成的顶点集是一个支配集.

(6) 在图  $G$  中, 如果顶点  $u$  与  $v$  相邻或  $u = v$ , 则称  $u$  支配  $v$ .

**定理 5.1.1** 设图  $G$  中无孤立顶点, 则存在支配集  $D$ , 使得  $\bar{D} = V(G) - D$  也是一个支配集.

**证明** 无妨设  $G$  是连通图. 于是  $G$  有生成树  $T$ . 任取  $v_0 \in V(G)$ . 令

$$D = \{v | v \in V(G) \text{ 且 } d_T(v_0, v) = \text{偶数或} 0\},$$

则  $\bar{D} = V(G) - D = \{v | v \in V(G) \text{ 且 } d_T(v_0, v) = \text{奇数}\}$ , 且  $D$  和  $\bar{D}$  都是支配集. 证毕.

**定理 5.1.2** 设图  $G$  无孤立顶点,  $D_1$  是一个极小支配集, 则  $\bar{D}_1 = V(G) - D_1$  也是一个支配集.

**证明** 用反证法. 若不然, 存在  $v_0 \in D_1$ , 它与  $\bar{D}_1$  中所有顶点都无边相连, 但它又不是孤立顶点, 故必与  $D_1$  中顶点连边, 因此  $D_1 - v_0$  仍是支配集, 但这与  $D_1$  是极小支配集矛盾. 证毕.

**推论 5.1.1** 设图  $G$  中无孤立顶点. 对  $G$  的任一个极小支配集  $D_1$ , 必存在另一个极小支配集  $D_2$ , 使得  $D_1 \cap D_2 = \emptyset$ .

**证明** 由定理 5.1.2,  $\bar{D}_1 = V(G) - D_1$  也是一个支配集, 且  $D_1 \cap \bar{D}_1 = \emptyset$ .  $\bar{D}_1$  中必含有一个极小支配集  $D_2$ , 且显然  $D_1 \cap D_2 = \emptyset$ . 证毕.

**定理 5.1.3** 图  $G$  的支配集  $D$  是一个极小支配集当且仅当  $D$  中每个顶点  $v$  满足下列条件之一:

- (1)  $N(v) \cap D = \emptyset$ ;
- (2) 存在  $u \in V(G) - D$  使得  $N(u) \cap D = \{v\}$ .

**证明 充分性** 设  $D$  是  $G$  的一个支配集. 对  $D$  中任一个顶点  $v$ , 因  $v$  满足上述条件之一, 故要么与  $v$  相邻的某个顶点不能被  $D - \{v\}$  支配, 要么  $v$  自己不能被  $D - \{v\}$  支配, 可见,  $D - \{v\}$  不再是支配集. 这表明  $D$  是极小支配集.

**必要性** 设  $D$  是  $G$  的一个极小支配集, 则对  $D$  中任一个顶点  $v$ ,  $D - \{v\}$  不再是支配集. 因此在  $D - \{v\}$  之外存在顶点  $u$ , 它不与  $D - \{v\}$  中任何点相邻. 如果  $u = v$ , 则说明  $v$  不与  $D$  中任何点相邻, 即  $N(v) \cap D = \emptyset$ ; 如果  $u \neq v$ , 则因  $D$  是支配集, 故  $u$  必与  $v$  相邻, 但不与  $D$  中其他点相邻, 即  $N(u) \cap D = \{v\}$ . 证毕.

**定理 5.1.4** 设  $G$  是无孤立顶点的图, 则  $G$  必有最小支配集  $D$  满足: 对  $\forall v \in D, \exists u \in V(G) - D$  使得  $N(u) \cap D = \{v\}$ .

**证明** 用反证法. 在  $G$  的全部最小支配集中, 设  $D$  是使得点导出子图  $G[D]$  含边数最多的一个. 假定该  $D$  不满足定理结论, 即

$$\exists v \in D, \text{使得对 } \forall u \in V(G) - D \text{ 都有 } N(u) \cap D \neq \{v\}.$$

上式表明,  $V(G) - D$  中一个点要么不与  $v$  相邻, 要么既与  $v$  相邻也与  $D$  中另外某些点相邻. 将此结论记为 (\*). 此外, 由定理 5.1.3,  $N(v) \cap D = \emptyset$ , 即  $D$  中所有点都不与  $v$  相邻. 因  $G$  无孤立点, 故  $v$  在  $V(G) - D$  中必有某个邻点  $w$ . 令  $D_1 = (D - \{v\}) \cup \{w\}$ , 则因  $D$  是支配集,  $V(G) - D$  中与  $v$  不相邻的点必与  $D - \{v\}$  中某些点相邻, 且由结论 (\*),  $V(G) - D$  中与  $v$  相邻的点也必与  $D - \{v\}$  中某些点相邻, 而  $v$  被点  $w$  所支配, 故  $D_1 = (D - \{v\}) \cup \{w\}$  是  $G$  的一个支配集. 又因  $|D_1| = |D|$ , 故  $D_1$  也是  $G$  的一个最小支配集, 但其导出子图  $G[D_1]$  含的边数比  $G[D]$  多 (因  $v$  与  $D - \{v\}$  无边相连, 但  $w$  与  $D - \{v\}$  至少有一条边相连). 这与  $D$  的取法矛盾. 证毕.

以下是关于支配数的几个估计式.

**定理 5.1.5** 如果图  $G$  无孤立顶点, 则  $\gamma(G) \leq \frac{\nu}{2}$ .

**证明** 设  $D$  是  $G$  的一个极小支配集, 则由定理 5.1.2,  $V(G) - D$  也是  $G$  的支配集. 因此,

$$\gamma(G) \leq \min\{|D|, |V(G) - D|\} \leq \frac{\nu}{2}. \quad \text{证毕.}$$

**定理 5.1.6 (Arnautov 1974, Payan 1975)** 设  $G$  是一个最小度为  $\delta$  的图, 则

$$\gamma(G) \leq \frac{1 + \ln(\delta + 1)}{1 + \delta} \nu.$$

**证明** 对  $G$  的任一非空顶点子集  $S \subseteq V(G)$ , 用  $U$  表示未被  $S$  中顶点支配 (即不属于  $S$  且不与  $S$  中顶点相邻) 的所有顶点之集. 对  $\forall v \in V(G)$ , 用  $N^*(v)$  表示顶点  $v$  及其所有邻点组成的集合, 即  $N^*(v) = N(v) \cup \{v\}$ . 由于  $U$  中每个顶点至少有  $\delta$  个邻点, 故  $\sum_{v \in U} |N^*(v)| \geq |U|(\delta + 1)$ . 从而在  $V(G) - S$  中至少有一个顶点  $x$ , 它在求和

过程中至少被重复计算  $\frac{|U|(\delta + 1)}{\nu}$  次. (否则, 若  $V(G) - S$  中每个点被重复计算都少于  $\frac{|U|(\delta + 1)}{\nu}$  次, 则  $\sum_{v \in U} |N^*(v)| < (\nu - |S|) \cdot \frac{|U|}{\nu} (\delta + 1) < |U|(\delta + 1)$ , 与前式矛盾). 这意味

着在  $V(G) - S$  中存在某个顶点  $x$ , 它支配  $U$  中至少  $\frac{|U|(\delta + 1)}{\nu}$  个顶点.

现在我们通过迭代来生成  $G$  的一个支配集, 用  $S$  表示在迭代过程中形成的支配集的一部分, 初始时可取最大度顶点放入  $S$ . 在迭代的每一步选择一个顶点放入  $S$ , 所

选择的顶点应能够尽可能多地支配当前的  $S$  尚未支配的顶点. 由前一部分的结论, 如果当前的  $S$  尚未支配的顶点有  $|U| = k$  个, 则在  $V(G) - S$  中存在某个顶点  $x$ , 它支配  $U$  中至少  $\frac{k(\delta+1)}{\nu}$  个顶点. 因此按照我们的选点规则, 再选择一个点放入  $S$  后, 剩下未被支配的顶点不超过  $k(1 - \frac{\delta+1}{\nu})$  个. 在初始时,  $|U| = \nu$ . 选第一个点进入  $S$  后, 未被支配的顶点数至多为  $\nu(1 - \frac{\delta+1}{\nu})$  个, 即  $|U| = \nu(1 - \frac{\delta+1}{\nu})$ ; 选第二个点进入  $S$  后, 剩余未被支配的顶点数至多为  $|U| \cdot (1 - \frac{\delta+1}{\nu}) = \nu(1 - \frac{\delta+1}{\nu})^2$  个;  $\dots$ ; 在  $\frac{\ln(\delta+1)}{\delta+1}\nu$  步后, 未被支配的顶点个数至多为

$$\nu \left(1 - \frac{\delta+1}{\nu}\right)^{\frac{\ln(\delta+1)}{\delta+1}\nu} < \nu e^{-\ln(\delta+1)} = \frac{\nu}{\delta+1}.$$

(上式推导中用到不等式  $1 - p < e^{-p}$ ). 将当前  $S$  中的点和未被  $S$  支配的点放在一起, 便组成  $G$  的一个支配集, 它至多含有  $\frac{\ln(\delta+1)}{\delta+1}\nu + \frac{\nu}{\delta+1} = \frac{1 + \ln(\delta+1)}{\delta+1}\nu$  个顶点. 结论得证. 证毕.

**定理 5.1.7** 对任何图  $G$ , 有  $\left\lceil \frac{\nu}{1 + \Delta(G)} \right\rceil \leq \gamma(G) \leq \nu - \Delta(G)$ .

**证明** 设  $D$  是  $G$  的一个最小支配集, 则  $V(G) - D \subseteq \bigcup_{v \in D} N(v)$  (习题 5.1), 因此

$$|V(G) - D| \leq |D| \cdot \Delta(G).$$

而  $|V(G) - D| = \nu - |D|$ ,  $|D| = \gamma(G)$ , 故由上式知

$$\nu - \gamma(G) \leq \gamma(G)\Delta(G),$$

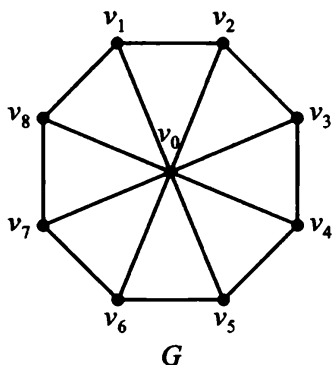
于是  $\gamma(G) \geq \left\lceil \frac{\nu}{1 + \Delta(G)} \right\rceil$ . 证毕.

图的支配集是内容相当丰富的一个研究专题, 它在许多学科领域有重要的应用, 是目前研究的一个热点方向 [1]~[34]. 文献 [1] 是支配集方面的一本很好的专著. [2]~[17] 是支配集理论研究的新进展, [18]~[23] 与支配集的算法有关. 支配集在无线网络中的应用可参阅文献 [24]~[33], 在电力网络中的应用可参见文献 [34].

## 二、点独立集 (vertex independent set)

**定义 5.1.2** 设  $I \subseteq V(G)$ , 若  $I$  中任二顶点均不相邻, 则称  $I$  为图  $G$  的一个点独立集 (简称独立集); 若对  $\forall u \in V(G) \setminus I, I \cup \{u\}$  都不再是  $G$  的独立集, 则称独立集  $I$  为图  $G$  的一个极大点独立集.  $G$  的含点数最多的点独立集称为最大点独立集, 它含点的个数称为  $G$  的独立数, 记为  $\alpha(G)$  或  $\alpha$ .

例如, 在下图中,  $I_1 = \{v_0\}, I_2 = \{v_1, v_4, v_7\}, I_3 = \{v_1, v_3, v_5, v_7\}$  都是  $G$  的独立集, 且都是极大独立集, 其中  $I_3$  是最大独立集,  $\alpha(G) = 4$ .



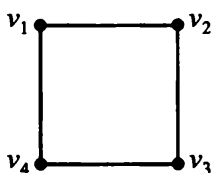
一些文献中将独立集称为稳定集 (stable set), 相应地将独立数称为稳定数.

• 独立集与支配集的关系

**定理 5.1.8** 图  $G$  的极大独立集必是  $G$  的极小支配集.

**证明** 设  $I$  是  $G$  的一个极大独立集. 对  $\forall u \in V(G) \setminus I, u$  必与  $I$  中某顶点相邻 (否则与  $I$  的极大性矛盾). 可见  $I$  是一个支配集. 又对  $\forall v \in I, v$  必与  $I \setminus \{v\}$  中的顶点不相邻, 可见  $I$  是极小支配集. 证毕.

**注** 该定理的逆不真. 例如在下图中,  $\{v_1, v_2\}$  是极小支配集, 但却不是极大独立集. 实际上它根本不是独立集.



但我们有如下结论.

**定理 5.1.9** 若  $I$  是独立集, 则它是极大独立集当且仅当它是支配集.

**证明** 必要性由定理 5.1.8 显然.

**充分性** 设  $I$  是独立集又是支配集, 则对  $\forall v \in V(G) \setminus I$ , 因  $I$  是支配集,  $v$  必与  $I$  中某顶点相邻, 故  $I \cup \{v\}$  不是独立集. 可见  $I$  是极大独立集. 证毕.

**注** (1) 由定理 5.1.8 和定理 5.1.9, 虽然  $G$  的一个独立集未必是  $G$  的支配集, 但一个极大独立集必是  $G$  的极小支配集; 反过来,  $G$  的一个支配集要么不是独立集, 要么是极大独立集;

(2) 一个支配集若不是极小支配集, 则必不是  $G$  的独立集.

**定理 5.1.10** 对任何图  $G, \alpha(G) \geq \gamma(G)$ .



**证明** 设  $I$  是  $G$  的一个最大独立集, 则它必是一个极大独立集, 由定理 5.1.8,  $I$  是  $G$  的一个 (极小) 支配集, 因此  $\gamma(G) \leq |I| = \alpha(G)$ . 证毕.

**注** 定理 5.1.10 中的等号未必总成立, 也就是说, 虽然  $G$  的一个最大独立集必是  $G$  的极小支配集, 但它未必是最小支配集. 例如完全二部图  $K_{1,3}$ , 最大独立集含三个点, 而最小支配集含一个点.

### • 点独立集与连通度

**定理 5.1.11 (Bondy, 1978)** 设  $\nu(G) \geq 2$ . 若图  $G$  中任二不相邻顶点  $x$  与  $y$  均有  $d_G(x) + d_G(y) \geq \nu(G)$ , 则  $\alpha(G) \leq \kappa(G)$ , ( $\kappa(G)$  表示  $G$  的连通度).

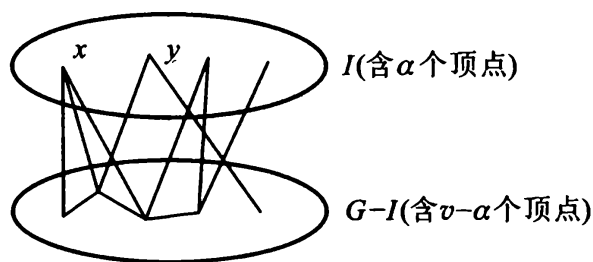
**证明** 首先易知  $G$  是连通的 (若  $G$  不连通, 从不同的连通分支  $G_1, G_2$  中取两点  $x, y$ , 则

$$d_G(x) \leq |G_1| - 1, d_G(y) \leq |G_2| - 1,$$

从而  $d_G(x) + d_G(y) \leq |G_1| + |G_2| - 2 < \nu$ ).

若  $G$  为完全图  $K_\nu$ , 则  $\alpha(K_\nu) = 1 \leq \nu - 1 = \kappa(K_\nu)$ , 结论成立. 下设  $G$  不是完全图. 用反证法证明结论.

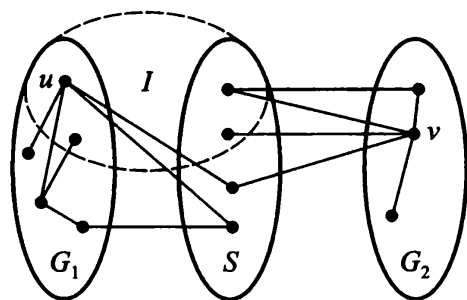
假定  $\alpha(G) \geq \kappa(G) + 1$ , 记  $\kappa(G) = k$ . 设  $I$  和  $S$  分别是  $G$  中的最大点独立集和最小点割集. 则  $|I| = \alpha(G) \geq 2, |S| = \kappa(G) = k$ . 设  $G \setminus S$  的连通分支为  $G_1, G_2, \dots, G_l$  ( $l \geq 2$ ). 由于对  $\forall x, y \in I, |N_G(x) \cup N_G(y)| \leq \nu - \alpha$ .



故由容斥原理,

$$\begin{aligned} |N_G(x) \cap N_G(y)| &= |N_G(x)| + |N_G(y)| - |N_G(x) \cup N_G(y)| \\ &= d_G(x) + d_G(y) - |N_G(x) \cup N_G(y)| \geq \nu - (\nu - \alpha) = \alpha \geq k + 1. \end{aligned}$$

这表明  $I \setminus S$  含在  $G \setminus S$  的同一个连通分支中 (因  $|S| = k$ , 上式说明  $I \setminus S$  的任二点  $x$  与  $y$  在  $G \setminus S$  中有公共的邻点, 故连通). 无妨设  $I \setminus S \subseteq V(G_1)$ , 即  $I \subseteq V(G_1) \cup S$ . 因  $\alpha \geq k + 1$ , 故  $\exists u \in I \cap V(G_1)$ .



取  $v \in V(G_2)$ , 则

$$|N_G(u) \cup N_G(v)| \leq \nu - 2 - (|I \cap V(G_1)| - 1) = \nu - |I \cap V(G_1)| - 1 = \nu - (\alpha - |I \cap S|) - 1.$$

又因  $N_G(u) \cap N_G(v) \subseteq S \setminus I$ , 故  $|N_G(u) \cap N_G(v)| \leq k - |I \cap S|$ . 由此二式可得

$$\begin{aligned} d_G(u) + d_G(v) &= |N_G(u) \cup N_G(v)| + |N_G(u) \cap N_G(v)| \\ &\leq (\nu - \alpha + |I \cap S| - 1) + (k - |I \cap S|) \\ &= \nu - \alpha + k - 1 \leq \nu - (k + 1) + k - 1 = \nu - 2. \end{aligned}$$

这与定理条件矛盾. 因此  $\alpha(G) \leq \kappa(G)$ . 证毕.

**推论 5.1.2** 设  $G$  是  $\nu$  ( $\nu \geq 2$ ) 阶简单图. 若  $\delta(G) \geq \frac{\nu}{2}$ , 则  $\alpha(G) \geq \kappa(G)$ .

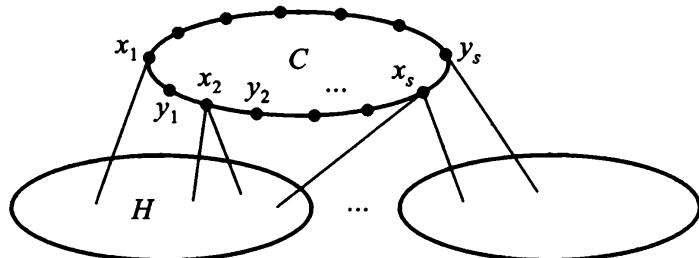
• 独立数与 Hamilton 性:

**定理 5.1.12** (Chvátal & Erdős, 1972) 设  $G$  是  $\nu$  ( $\nu \geq 3$ ) 阶简单图. 若  $\kappa(G) \geq \alpha(G)$ , 则  $G$  是 Hamilton 图.

**证明** 若  $\alpha(G) = 1$ , 则  $G$  是完全图, 从而是 Hamilton 图. 下设  $\alpha(G) \geq 2$ .

由于  $\kappa(G) \geq \alpha(G) \geq 2$ , 故  $G$  含有圈. 设  $C$  是  $G$  的最长圈. 下面用反证法证明  $C$  是 Hamilton 圈.

若  $C$  不是 Hamilton 圈, 则  $V(G) \setminus V(C)$  非空. 令  $H$  是  $G - V(C)$  的任一连通分支, 并令  $\{x_1, x_2, \dots, x_s\}$  是  $C$  上与  $H$  中点相邻的顶点之集. 因  $\kappa(G) \geq 2$ , 故  $s \geq 2$  (否则,  $C$  上只有一个顶点  $v$  与  $H$  相邻,  $G - \{v\}$  不连通). 由  $C$  的最大性及  $H$  的连通性知  $x_1, x_2, \dots, x_s$  在  $C$  上互不相邻 (否则可得更大的圈).



因此  $|V(C)| > s$ , 且  $\{x_1, x_2, \dots, x_s\}$  是  $G$  的点割集 (因去掉  $\{x_1, x_2, \dots, x_s\}$  后  $H$  与  $C$  上其余点不连通). 所以  $\kappa(G) \leq s$  (由  $\kappa(G)$  的定义).

给圈  $C$  定一个方向, 得有向圈  $\vec{C}$ . 令  $Y = \{y_i | x_i y_i \in E(\vec{C}), i = 1, 2, \dots, s\}$ . 则由  $x_i$  在  $C$  上的不相邻性知,  $|Y| = s \geq 2$ . 下面证明,  $Y$  是  $G$  的一个独立集.

事实上, 若  $Y$  不是  $G$  的独立集, 则有边  $y_i y_j \in E(G)$ . 在  $H$  中分别取  $x_i$  和  $x_j$  的一个邻点  $v_i$  和  $v_j$ , 若  $v_i = v_j$ , 则  $C - x_i y_i - x_j y_j + x_i v_i + v_i x_j + y_i y_j$  是  $G$  中一条比  $C$  更长的圈; 若  $v_i \neq v_j$ , 取  $H$  中连接  $v_i$  和  $v_j$  的一条路  $P_{ij}$ , 则  $C - x_i y_i - x_j y_j + x_i v_i + x_j v_j + y_i y_j + P_{ij}$  是  $G$  中一条比  $C$  更长的圈  $v_i = v_j$ . 这与  $C$  是最长圈矛盾, 于是  $Y$  是独立集.

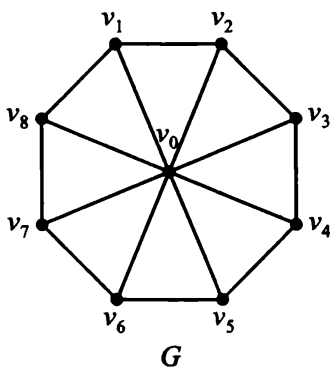
由于  $y_i$  与  $x_i$  相邻, 故  $y_i$  不与  $H$  中任何点相邻 (否则会得到比  $C$  更长的圈),  $i = 1, 2, \dots, s$ . 任取  $y \in V(H)$ , 则  $S = \{y, y_1, \dots, y_s\}$  是  $G$  的独立集, 且  $\alpha(G) \geq |S| = s + 1 \geq \kappa(G) + 1$ . 这与定理的条件矛盾. 因此  $C$  必是 Hamilton 圈. 证毕.

有关独立集和独立数的研究较为活跃, 有兴趣的读者可参阅近期文献 [35]~[72].

### 三、点覆盖 (vertex covering set)

**定义 5.1.3** 设  $F \subset V(G)$ , 若  $G$  的每条边至少有一个端点属于  $F$ , 则称  $F$  是  $G$  的一个点覆盖集, 简称为点覆盖. 若对  $\forall v \in F, F - \{v\}$  都不再是  $G$  的点覆盖, 则称点覆盖  $F$  是一个极小点覆盖. 图  $G$  的含点数最少的点覆盖称为最小点覆盖, 其点数称为  $G$  的点覆盖数, 记为  $\beta(G)$  或  $\beta$ .

例如, 在下图中,  $\{v_0, v_1, v_3, v_5, v_7\}$  和  $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$  都是  $G$  的点覆盖, 且都是极小点覆盖. 前一个点覆盖是最小点覆盖,  $\beta(G) = 5$ .



**注** (1) 最小点覆盖必为极小点覆盖.

(2) 点覆盖集与支配集是容易混淆的两个概念, 它们的本质区别在于, 点覆盖集是用点覆盖边, 而支配集是用点支配点. 在连通图中, 点覆盖集必为支配集, 但支配集未必是点覆盖集. 比如上例中  $\{v_0\}$  及  $\{v_1, v_4, v_7\}$  都是  $G$  的支配集, 但不是点覆盖集.

(3) 极小点覆盖集未必是极小支配集. 例如上例中  $\{v_0, v_1, v_3, v_5, v_7\}$  是  $G$  的极小点覆盖集, 但它不是  $G$  的极小支配集.

• 点覆盖与点独立集的关系:

**定理 5.1.13** 顶点子集  $F$  是图  $G$  的点覆盖集当且仅当  $V(G)\setminus F$  是  $G$  的点独立集.

**证明**  $F$  是图  $G$  的点覆盖集  $\Leftrightarrow G$  的每条边至少有一端在  $F$  中  $\Leftrightarrow$  没有两端都在  $V(G)\setminus F$  中的边  $\Leftrightarrow V(G)\setminus F$  是点独立集. 证毕.

以下两个推论的证明留作习题.

**推论 5.1.3**  $F$  是图  $G$  的极小点覆盖集当且仅当  $V(G)\setminus F$  是  $G$  的极大点独立集.

**推论 5.1.4**  $\alpha(G) + \beta(G) = \nu$ .

## §5.2 边独立集与边覆盖集

### 一、边独立集

**定义 5.2.1** 图  $G$  的一个匹配  $M$  称为  $G$  的一个边独立集.  $G$  的最大匹配所含的边数称为  $G$  的边独立数或匹配数, 记为  $\alpha'(G)$ .

边独立集与点覆盖有密切关系. 由于任意一个顶点不能覆盖边独立集中的两条边, 因此图有大的边独立集, 就有大的点覆盖集. 另一方面, 边独立集中不同的边不能用同一个顶点覆盖, 因此图  $G$  中任何点覆盖集所含的点数不会小于任何边独立集含的边数. 边独立数与点覆盖的下列关系是显然的.

- $\alpha'(K_{2n}) = n < 2n - 1 = \beta(K_{2n});$
- $\alpha'(K_{2n+1}) = n < 2n = \beta(K_{2n+1});$
- $\alpha'(C_{2n}) = n = \beta(C_{2n});$
- $\alpha'(C_{2n+1}) = n < n + 1 = \beta(C_{2n+1});$
- $\alpha'(K_{m,n}) = \min\{m, n\} = \beta(K_{m,n}).$

一般地, 有

**定理 5.2.1** 对任何无环边的图  $G$ ,  $\alpha'(G) \leq \beta(G)$ .

**证明** 设  $S$  是  $G$  中一个最小点覆盖,  $M$  是  $G$  中最大匹配.  $M$  中任一条边  $e$  的两端点至少应有一个属于  $S$ , 因而  $\alpha'(G) \leq \beta(G)$ . 证毕.

**定理 5.2.2 (König, Egerváry, 1931) [73][74]** 对于二部图  $G$ , 有  $\alpha'(G) = \beta(G)$ , 即  $G$  中最大匹配的边数等于最小覆盖的点数.

**证明** 设  $M^*$  是  $G = (X, Y)$  的最大匹配. 不妨设  $|X| \leq |Y|$ .

若  $M^*$  饱和  $X$ , 则  $|X| = |M^*|$ . 因  $X$  显然是  $G$  的一个点覆盖, 故  $\beta \leq |X| = |M^*|$ . 另一方面,  $M^*$  中各条边需要用不同的点来覆盖, 故又有  $\beta \geq |M^*|$ . 从而  $\beta(G) = |M^*| = \alpha'$ .

若  $M^*$  不饱和  $X$ , 设

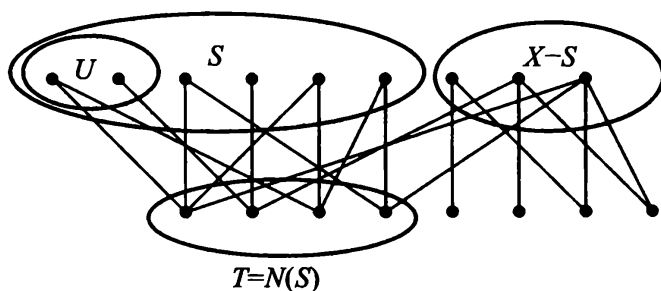
$U = \{x \in X | x \text{ 未被 } M^* \text{ 所饱和}\}$ ,  $A = \{v \in V(G) | v \text{ 到 } U \text{ 有 } M^* \text{ 交错路}\} \cup U$ ,

$$S = A \cap X, T = A \cap Y.$$

则  $N(S) = T$ . (\*)

令  $\bar{K} = (X - S) \cup T$ , 则  $G$  的每条边至少有一端在  $\bar{K}$  中 (否则必存在一条边, 其一端在  $S$  中, 另一端在  $Y \setminus T$  中, 这与  $N(S) = T$  矛盾). 因此  $\bar{K}$  是  $G$  的一个点覆盖, 且

$$|M^*| = |\bar{K}|. (**)$$



故  $\alpha'(G) = |M^*| = |\bar{K}| \geq \beta(G)$ , 而由定理 5.2.1, 又有  $\alpha'(G) \leq \beta(G)$ , 所以  $\alpha'(G) = \beta(G)$ . 证毕.

附 (\*) 式的证明:

首先  $T \subseteq N(S)$ : 对  $\forall y \in T$ ,  $y$  到  $U$  有  $M^*$  交错路  $P$ , 且  $y$  必是  $M^*$  饱和的 (否则  $P$  是  $M^*$  可扩路). 设  $yx \in M^*$ , 则  $x$  到  $U$  有  $M^*$  交错路  $P + yx$ . 这表明  $x \in S$ , 故  $y \in N(S)$ ;

其次  $N(S) \subseteq T$ : 对  $\forall y \in N(S)$ , 设  $y$  在  $S$  中的邻点是  $x$ . 因  $x$  到  $U$  有交错路  $P'$ , 故若  $y \in P'$ , 则  $y$  到  $U$  有交错路; 若  $y \notin P'$ , 则  $P' + xy$  是  $y$  到  $U$  的  $M^*$  交错路. 总之  $y \in A$  故  $y \in T$ . 证毕.

(\*\*) 的证明: 首先  $X - S$  是  $M^*$  饱和的 (因  $X$  中的非饱和点全在  $U$  中),  $T$  也是  $M^*$  饱和的 (否则有  $M^*$  可扩路); 其次,  $X - S$  与  $T$  间不会有  $M^*$  的边 (否则  $X - S$  中的点到  $U$  有  $M^*$  交错路). 因此  $\bar{K} = (X - S) \cup T$  中每点都有一条  $M^*$  匹配边, 且不同点关联的匹配边不同, 故  $|\bar{K}| \leq |M^*|$ ; 另一方面,  $\bar{K}$  是  $G$  的一个点覆盖, 而匹配  $M^*$  中各条边需被  $\bar{K}$  中的不同点所覆盖, 故又有  $|\bar{K}| \geq |M^*|$ . 因而  $|M^*| = |\bar{K}|$ . 证毕.

关于边独立数有如下估计式.

**定理 5.2.3** 设图  $G$  无孤立点, 则  $\left\lceil \frac{\nu}{1 + \Delta(G)} \right\rceil \leq \alpha'(G) \leq \left\lfloor \frac{\nu}{2} \right\rfloor$ .

**证明** 因为每条匹配边匹配图  $G$  的两个顶点, 故上界是显然的.

为证明下界, 对图的边数  $\varepsilon$  作数学归纳法.

$\varepsilon = 1$  时结论显然成立.

假设对任何不超过  $k$  条边的图  $G$ , 下界都成立.

设  $G$  是具有  $k + 1$  条边且无孤立点的图. 若  $G$  中每条边都至少有一个端点是 1 度顶点, 则  $G$  的每个连通分支都是星 (星是一个完全二部图  $K_{1,s}$ ). 星  $K_{1,s}$  的最大匹配只有一条边, 且  $\Delta(K_{1,s}) = s, \nu(K_{1,s}) = s + 1$ , 因此  $\alpha'(K_{1,s}) = 1 = \frac{\nu(K_{1,s})}{1 + \Delta(K_{1,s})}$ .

设  $G$  由  $r$  颗星  $T_1, T_2, \dots, T_r$  组成 ( $r \geq 1$ ), 则

$$\alpha'(G) = \sum_{i=1}^r \alpha'(T_i) = \sum_{i=1}^r \frac{\nu(T_i)}{1 + \Delta(T_i)} \geq \frac{\sum_{i=1}^r \nu(T_i)}{1 + \Delta(G)} = \frac{\nu(G)}{1 + \Delta(G)}.$$

下设  $G$  中存在边  $e$ , 其两 endpoint 都不是 1 度点. 设  $G - e$  有  $m$  个连通分支  $G_1, G_2, \dots, G_m$  ( $m \geq 1$ ), 则每个连通分支都不是孤立点, 且边数都不超过  $k$ . 因此根据归纳假设,

$$\alpha'(G_i) \geq \frac{\nu(G_i)}{1 + \Delta(G_i)}, \quad (i = 1, 2, \dots, m).$$

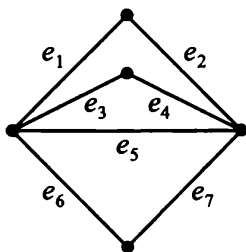
$$\text{从而 } \alpha'(G) \geq \alpha'(G - e) = \sum_{i=1}^m \alpha'(G_i) \geq \sum_{i=1}^m \frac{\nu(G_i)}{1 + \Delta(G_i)} \geq \frac{\sum_{i=1}^m \nu(G_i)}{1 + \Delta(G - e)} \geq \frac{\nu(G)}{1 + \Delta(G)}.$$

归纳完成. 证毕.

## 二、边覆盖

**定义 5.2.2** 设  $L \subseteq E(G)$ . 若  $G$  的每个顶点都与  $L$  中至少一条边关联, 则称  $L$  是  $G$  的边覆盖集, 简称为边覆盖. 若边覆盖  $L$  的任何真子集都不是  $G$  的边覆盖, 则称  $L$  是  $G$  的极小边覆盖.  $G$  的含边数最少的边覆盖称为  $G$  的最小边覆盖, 其所含边的数目称为  $G$  的边覆盖数, 记为  $\beta'(G)$  或  $\beta'$ .

例如, 在下图中,  $L_1 = \{e_2, e_3, e_6\}$  和  $L_2 = \{e_2, e_3, e_7\}$  都是边覆盖, 也是极小边覆盖, 还是最小边覆盖;  $L_3 = \{e_1, e_4, e_5, e_7\}$  是边覆盖, 但不是极小边覆盖.



### • 边覆盖与边独立数 (匹配数) 的关系

**定理 5.2.4 (Gallai, 1959)** 若  $\delta(G) > 0$ , 则  $\alpha'(G) + \beta'(G) = \nu$ .

**证明** 设  $M^*$  是  $G$  的最大匹配. 若  $M^*$  是  $G$  的完美匹配, 则显然  $M^*$  是  $G$  的一个边覆盖, 从而  $\beta' \leq |M^*| = \frac{\nu}{2} = \alpha'$ , 故  $\alpha' + \beta' \leq \frac{\nu}{2} + \frac{\nu}{2} = \nu$ ; 若  $M^*$  不是  $G$  的完美匹配, 设  $U$  是  $M^*$  非饱和点之集, 则  $G[U]$  是无边图 (否则在  $G[U]$  中取一边可使  $M^*$  扩大). 由于  $\delta(G) > 0$ , 所以  $U$  的每个点在  $G$  中都至少与一条边关联. 对  $U$  中每个点取一条关联边, 这些边的集合记为  $E'$ . 显然  $M^* \cup E'$  是  $G$  的边覆盖, 且  $M^* \cap E' = \phi$ . 因而 (注意  $M^*$  中的边数为  $\alpha'$ ,  $E'$  中的边数为  $\nu - 2\alpha'$ )

$$\beta' \leq |M^* \cup E'| = |M^*| + |E'| = \alpha' + (\nu - 2\alpha') = \nu - \alpha', \text{ 即 } \alpha' + \beta' \leq \nu.$$

另一方面, 设  $L$  是  $G$  的一个最小边覆盖, 即  $|L| = \beta'$ . 令  $H = G[L]$ , 则  $V(H) = V(G) = \nu$  (因  $L$  覆盖了  $G$  的所有顶点). 设  $M$  是  $H$  的最大匹配,  $U$  为  $H$  中  $M$  非饱和点集, 则  $H[U]$  是无边图, 从而

$$|L| - |M| = |L - M| \geq |U| = \nu - 2|M|. \quad (*)$$

其中第一个等号用到  $L \supseteq M$ , 此包含关系成立是因为  $M$  是  $L$  的边导出子图  $H = G[L]$  中的匹配; (\*) 式中的不等号是因为  $M$  是  $H$  的最大匹配,  $L - M$  中每条边要么两 endpoint 都  $M$  饱和, 要么只有一 endpoint  $M$  不饱和, 故  $L - M$  中每条边至多有一个 endpoint 在  $U$  中, 而且  $U$  中每个点至少由  $L - M$  中一条边关联.

由 (\*) 式,  $|L| + |M| \geq \nu$ . 又因  $H$  是  $G$  的生成子图, 故  $M$  也是  $G$  的匹配, 因此  $\alpha' \geq |M|$ . 从而  $\alpha' + \beta' \geq |M| + |L| \geq \nu$ . 证毕.

**推论 5.2.1** 设  $\delta(G) > 0$ , 则  $\alpha'(G) \leq \beta'(G)$ , 等号成立当且仅当  $G$  有完美匹配;

**证明** 设  $M^*$  是  $G$  的最大匹配, 则欲覆盖  $M^*$  关联的  $2|M^*|$  个点至少需要  $|M^*|$  条边. 故

$$\alpha'(G) = |M^*| \leq \beta'(G).$$

等号成立时, 由 Gallai 定理  $\alpha'(G) + \beta'(G) = \nu$  知,  $\alpha'(G) = \frac{\nu}{2}$ , 故  $G$  有完美匹配. 证毕.

**推论 5.2.2** 图  $G$  的边覆盖数等于  $G$  的边独立数加上最大匹配非饱和点的个数.

**证明** 由定理 5.2.4,  $\beta'(G) = \nu - \alpha'(G) = \alpha'(G) + (\nu - 2\alpha'(G))$ , 注意到  $\alpha'(G)$  是  $G$  的最大匹配的边数, 故  $\nu - 2\alpha'(G)$  是未被最大匹配饱和的点数. 证毕.

该推论将求  $G$  的最小边覆盖集的问题转化为求  $G$  的最大匹配问题: 求出  $G$  的一个最大匹配  $M^*$ , 对于在  $M^*$  下未饱和的每个点, 任选其一条关联的边, 这些边与  $M^*$  的边放在一起便形成  $G$  的一个最小边覆盖.

### • 边覆盖数与点独立数的关系

**定理 5.2.5** 对任何图  $G$ , 都有  $\alpha(G) \leq \beta'(G)$ .

**证明** 设  $I$  是  $G$  的最大点独立集. 因  $I$  中点彼此不相邻, 故至少要用  $|I|$  条边才能覆盖完  $I$  中的所有顶点, 故  $\alpha(G) = |I| \leq \beta'(G)$ . 证毕.

**定理 5.2.6** 设  $G$  是二部图且  $\delta(G) > 0$ , 则  $\alpha(G) = \beta'(G)$ .

**证明** 由定理 5.2.4 及推论 5.1.4,  $\alpha' + \beta' = \nu = \alpha + \beta$ . 而由定理 5.2.2,  $\alpha' = \beta$ , 故  $\alpha = \beta'$ . 证毕.

### • 其他关系

**定理 5.2.7** 设  $\delta(G) > 0$ , 则

(1) 对  $G$  的任一匹配  $M$  和任一边覆盖  $L$ , 有  $|M| \leq |L|$ . 等号成立时  $M$  为完美匹配且  $L$  为最小边覆盖;

(2) 对  $G$  的任一匹配  $M$  和任一点覆盖  $F$ , 有  $|M| \leq |F|$ . 等号成立时,  $M$  为最大匹配且  $F$  为最小点覆盖;

(3) 对  $G$  中任一点独立集  $I$  和任一边覆盖  $L$ , 有  $|I| \leq |L|$ . 等号成立时,  $I$  为最大点独立集且  $L$  为最小边覆盖.

**证明** (1) 对任一匹配  $M$  和任一边覆盖  $L$ , 由推论 5.2.1, 有  $|M| \leq \alpha'(G) \leq \beta'(G) \leq |L|$ .

当  $|M| = |L|$  时, 上式成为  $|M| = \alpha'(G) = \beta'(G) = |L|$ . 因而  $M$  是最大匹配,  $L$  是最小边覆盖, 且由推论 5.2.1 知, 此时  $M$  是完美匹配.

(2) 对  $G$  的任一匹配  $M$  和任一点覆盖  $F$ , 由定理 5.2.1, 有  $|M| \leq \alpha' \leq \beta \leq |F|$ . 当  $|M| = |F|$  时, 上式成为  $|M| = \alpha' = \beta = |F|$ , 可见  $M$  是最大匹配而  $F$  是最小点覆盖.

(3) 对  $G$  中任一点独立集  $I$  和任一边覆盖  $L$ , 由定理 5.2.5 得:  $|I| \leq \alpha \leq \beta' \leq |L|$ . 当  $|I| = |L|$  时, 上式成为  $|I| = \alpha = \beta' = |L|$ , 可见此时  $I$  是最大点独立集而  $L$  是最小边覆盖. 证毕.

**注** (1) 对完美匹配  $M$  和最小边覆盖  $L$ , 必定  $|M| = |L|$  (习题 5.17);

(2) 对最大匹配  $M$  和最小点覆盖  $F$ , 未必  $|M| = |F|$ . 例如, 对于圈  $C_5$ , 最大匹配含 2 边, 但最小点覆盖含 3 点;

(3) 对最大点独立集  $I$  和最小边覆盖  $L$ , 未必  $|I| = |L|$ . 同样可取圈  $C_5$  检验.

**定理 5.2.8** 设图  $G$  无孤立顶点, 则  $\gamma(G) \leq \min\{\alpha(G), \alpha'(G), \beta(G), \beta'(G)\}$ .

**证明**  $\gamma(G) \leq \alpha(G)$  是定理 5.1.10 的结论.



由于无孤立顶点的图中每个点覆盖都是支配集, 因此  $\gamma(G) \leq \beta(G)$  成立.

设  $E_1$  是  $G$  的一个最小边覆盖, 则  $G$  的每个顶点是  $E_1$  中至少一条边的端点. 对  $E_1$  中每条边取其一个端点, 所形成的顶点集合记为  $D$  (重复的只取一次), 则  $D$  是  $G$  的一个支配集. 因此  $\gamma(G) \leq |D| \leq |E_1| = \beta'(G)$ .

设  $M$  是  $G$  的一个最大匹配, 对  $M$  中任一条边  $e = uv$ , 若端点  $u$  有  $M$  不饱和的邻点, 则另一个端点  $v$  要么没有  $M$  不饱和邻点, 要么它的不饱和邻点也是  $u$  的 (不饱和) 邻点 (否则有  $M$  增广路, 与  $M$  是最大匹配矛盾). 对  $M$  中每条边, 按如下方式选其一个端点: 如果其两个端点都没有  $M$  不饱和邻点或者都有  $M$  不饱和邻点, 则任取其一个端点; 若一个端点有  $M$  不饱和邻点, 而另一个端点没有, 则选有不饱和邻点的那一个. 这样选出的点形成  $G$  的一个支配集 (对  $G$  中任一个  $M$  饱和顶点, 与它关联的  $M$  匹配边必有一个端点在  $D$  中, 故被  $D$  支配; 对  $G$  中每个  $M$  非饱和顶点  $w$ , 其邻点必定是  $M$  饱和的, 按  $D$  的选法,  $w$  必有一个邻点在  $D$  中, 因此  $w$  也被  $D$  支配). 于是  $\gamma(G) \leq |D| = |M| = \alpha'(G)$ . 证毕.

关于点覆盖数的上下界估计, 有下述结果.

**定理 5.2.9** 设图  $G$  无孤立点, 则  $\left\lceil \frac{\nu}{2} \right\rceil \leq \beta'(G) \leq \left\lfloor \frac{\nu \cdot \Delta(G)}{1 + \Delta(G)} \right\rfloor$ .

**证明** 由定理 5.2.4,  $\alpha'(G) + \beta'(G) = \nu$ , 得  $\beta'(G) = \nu - \alpha'(G)$ . 再利用定理 5.2.3,

$$\left\lfloor \frac{\nu}{1 + \Delta(G)} \right\rfloor \leq \alpha'(G) \leq \left\lfloor \frac{\nu}{2} \right\rfloor,$$

由此立即可得结论. 证毕.

有关边覆盖的更多内容可参看文献 [75].

## §5.3 支配集、点独立集、点覆盖集的求法

### 一、应用背景

#### 1. 应急增援中心的选址

欲在  $n$  个地点  $v_1, v_2, \dots, v_n$  设置若干个应急中心, 使得每个地点都与至少一个应急中心相邻 (有直达通路). 同时, 为了减少造价, 应急中心的数目要尽可能少. 问应设多少个? 如何设置?

**图论模型:** 以  $n$  个地点为顶点集, 两点之间有直达通路时, 连边, 得一图  $G$ . 问题是: 求图  $G$  的最小支配集.

## 2. 收款台的设置

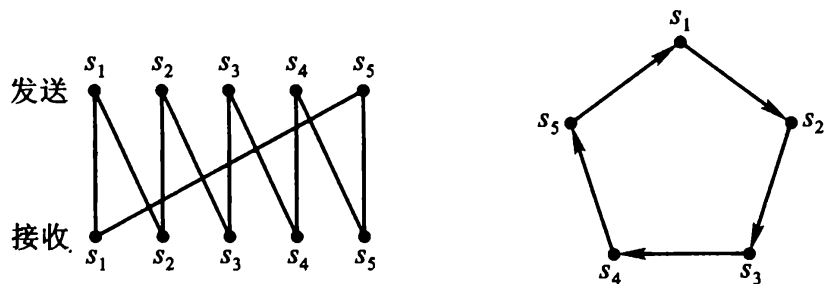
某商场实行收款台收款制度. 为使顾客在任何一个货架通道都能看到至少一个收款台, 问至少应设置多少个收款台? 设在什么位置?

类似的问题有: 执勤岗哨的设置问题、交叉路口交通监控器的设置问题等.

**图论模型:** 以货架通道为边、通道交叉点为顶点构作图  $G$ . 问题是求图  $G$  的最小点覆盖集.

## 3. 通信信号差错控制问题

设某信息传输系统的基本信号集合为  $S = \{s_1, s_2, \dots, s_5\}$ . 由于干扰等原因, 使得接收端收到的信号可能发生错乱. 比如发送  $s_1$ , 可能收到  $s_2$ . 通过统计分析已经知道哪些信号间容易发生错乱. 例如, 假定容易发生错乱的情况如下图所示, 为了能从收到的输出信号辨识出正确的输入信号, 我们不能使用所有 5 个信号来传送信息, 而只能从中选出一部分使用. 问如何选取尽可能多的可用信号?



**图论模型:** 以  $s_1, s_2, \dots, s_5$  为顶点构作图  $G$ , 若信号  $s_i$  容易错收为  $s_j$ , 则从顶点  $s_i$  向  $s_j$  连一条有向边. 问题是求图  $G$  中的最大点独立集.

目前还没有有效算法来求任意给定图的最小点覆盖集、最大点独立集和最小支配集. 下面介绍的方法用到数理逻辑运算.

## 二、逻辑运算及其性质

设  $G$  是一个图, 用  $v_i$  表示事件“包含顶点  $v_i$ ”;  $v_i + v_j$  或  $v_i \vee v_j$  表示事件“要么包含顶点  $v_i$ , 要么包含顶点  $v_j$ ” (逻辑“或”);  $v_i v_j$  或  $v_i \wedge v_j$  表示事件“既包含顶点  $v_i$  又包含顶点  $v_j$ ” (逻辑“与”). 则如下逻辑运算律成立:

- (1) 交换律:  $v_i + v_j = v_j + v_i, v_i v_j = v_j v_i$ .
- (2) 结合律:  $(v_i + v_j) + v_k = v_i + (v_j + v_k), (v_i v_j) v_k = v_i (v_j v_k)$
- (3) 分配律:  $v_i (v_j + v_k) = v_i v_j + v_i v_k, (v_i + v_j) v_k = v_i v_k + v_j v_k$
- (4) 吸收律:  $v_i + v_i = v_i, v_i v_i = v_i, v_i + v_i v_j = v_i, v_i (v_i + v_j) = v_i$ .

为方便起见, 以下将使用记号  $\sum v_i$  和  $\prod v_i$ :

$$\sum_{i=1}^n v_i = v_1 \vee v_2 \vee \cdots \vee v_n = v_1 + v_2 + \cdots + v_n,$$

$$\prod_{i=1}^n v_i = v_1 \wedge v_2 \wedge \cdots \wedge v_n = v_1 v_2 \cdots v_n.$$

### 三、算法

#### 1. 求所有极小点覆盖和极大点独立集

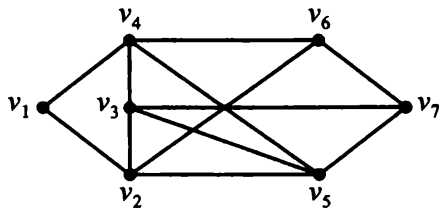
(1) 由点覆盖的概念,  $V(G)$  的子集  $F$  是  $G$  的点覆盖, 当且仅当对每个  $v_i \in V(G)$ , 要么  $v_i \in F$ , 要么  $N_G(v_i) \subseteq F$ .

(2) 由推论 5.1.3,  $V(G)$  的子集  $F$  是  $G$  的极小点覆盖当且仅当  $V(G) - F$  是  $G$  的极大点独立集.

由此可得求所有极小点覆盖和极大点独立集的方法:

利用逻辑运算性质将表达式  $\prod_{i=1}^{\nu} \left( v_i + \prod_{u \in N(v_i)} u \right)$  展开成乘积之和的形式, 则每个乘积项表示一个极小点覆盖,  $G$  的所有极小点覆盖都由这些乘积项给出, 而这些极小点覆盖的补集给出了  $G$  的所有极大点独立集. 进而可以由所有极小点覆盖比较出最小点覆盖, 由所有极大点独立集比较出最大点独立集. 表达式  $\prod_{i=1}^{\nu} \left( v_i + \prod_{u \in N(v_i)} u \right)$  通常记为  $\varphi(v_1, v_2, \cdots, v_{\nu})$ .

**例 5.3.1** 求下图的所有极小点覆盖和极大点独立集.



**解**

$$\begin{aligned} \varphi(v_1, v_2, \cdots, v_7) &= (v_1 + v_2 v_4)(v_2 + v_1 v_3 v_5 v_6)(v_3 + v_2 v_4 v_5 v_7)(v_4 + v_1 v_3 v_5 v_6) \\ &\quad \cdot (v_5 + v_2 v_3 v_4 v_7)(v_6 + v_2 v_4 v_7)(v_7 + v_3 v_5 v_6) \\ &= v_1 v_3 v_5 v_6 + v_2 v_3 v_4 v_5 v_6 + v_2 v_4 v_5 v_7 + v_2 v_3 v_4 v_7. \end{aligned}$$

故极小点覆盖有:

$$\begin{aligned} F_1 &= \{v_1, v_3, v_5, v_6\}, & F_2 &= \{v_2, v_3, v_4, v_5, v_6\}, \\ F_3 &= \{v_2, v_4, v_5, v_7\}, & F_4 &= \{v_2, v_3, v_4, v_7\}. \end{aligned}$$

而极大点独立集有:

$$I_1 = \bar{F}_1 = \{v_2, v_4, v_7\}, \quad I_2 = \bar{F}_2 = \{v_1, v_7\},$$

$$I_3 = \bar{F}_3 = \{v_1, v_3, v_6\}, \quad I_4 = \bar{F}_4 = \{v_1, v_5, v_6\}.$$

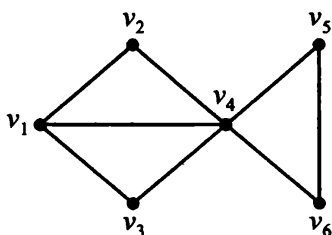
同时, 不难得出最小点覆盖为  $F_1$  和  $F_3$ , 最大点独立集为  $I_1$ 、 $I_2$  和  $I_4$ .

## 2. 求所有极小支配集

由定理 5.1.8, 上述方法得出的极大独立集必是  $G$  的极小支配集. 但一般情况下这并未包含  $G$  的所有极小支配集. 求所有极小支配集的方法为:

利用逻辑运算性质将表达式  $\prod_{i=1}^{\nu} \left( v_i + \sum_{u \in N(v_i)} u \right)$  展开成乘积之和的形式, 则每个乘积项给出一个极小支配集. 所有极小支配集都由这些乘积项给出. 进而可由所有极小支配集比较出最小支配集. 表达式  $\prod_{i=1}^{\nu} \left( v_i + \sum_{u \in N(v_i)} u \right)$  通常记为  $\psi(v_1, v_2, \dots, v_{\nu})$ .

**例 5.3.2** 求下图的所有极小支配集和最小支配集.



**解** 
$$\begin{aligned} \psi(v_1, v_2, \dots, v_6) &= (v_1 + v_2 + v_3 + v_4)(v_2 + v_1 + v_4)(v_3 + v_1 + v_4) \\ &\quad \cdot (v_4 + v_1 + v_2 + v_3 + v_5 + v_6)(v_5 + v_4 + v_6)(v_6 + v_4 + v_5) \\ &= v_1 v_5 + v_1 v_6 + v_4 + v_2 v_3 v_5 + v_2 v_3 v_6. \end{aligned}$$

故  $G$  的所有极小支配集为:  $\{v_1, v_5\}, \{v_1, v_6\}, \{v_4\}, \{v_2, v_3, v_5\}, \{v_2, v_3, v_6\}$ . 显然最小支配集为  $\{v_4\}$ .

上述求最小点覆盖集、最大点独立集和最小支配集的算法是指数阶的. 例如, 对于求最小点覆盖集、最大点独立集的算法, 我们需要处理  $\prod_{i=1}^{\nu} \left( v_i + \prod_{u \in N(v_i)} u \right)$  的展开式中  $2^{\nu}$  个乘积项, 因此计算复杂度至少是  $O(2^{\nu})$ . 事实上, 最小点覆盖集问题、最大点独立集问题和最小支配集问题都是 NP-hard 问题<sup>[49]~[53][62][83][84][86][87]</sup>, 目前尚没有有效的精确型算法求最小点覆盖集、最大点独立集和最小支配集的最优解.

设计求最小点覆盖集、最大点独立集和最小支配集的近似算法, 或在某些约束条件下寻找它们的精确算法, 是目前的一个研究热点. 文献 [49]~[54] 是涉及这些算法的

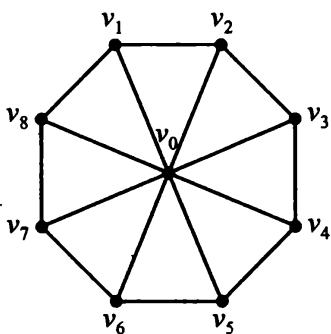
专著和网页. 文献 [55]~[72] 研究点独立集的有关算法, [56]~[57] 及 [75]~[97] 涉及覆盖的算法, [18]~[33] 及 [56]~[57] 给出了支配集和连通支配集的各种算法.

## §5.4 Ramsey 数

### 一、团 (clique)

**定义 5.4.1** 设  $Q \subset V(G)$ , 若导出子图  $G[Q]$  是完全图, 则称  $Q$  是  $G$  的一个团. 若给团  $Q$  添加  $V(G) \setminus Q$  中任何顶点, 导出子图都不再是团, 则称  $Q$  是一个极大团. 图  $G$  的含顶点数最多的团称为  $G$  的最大团, 其顶点数称为  $G$  的团数, 记为  $\omega(G)$ . 含  $k$  个顶点的团称为  $k$ -团.

例如, 在下图中,  $\{v_0, v_1, v_2\}$  和  $\{v_0, v_2, v_3\}$  都是  $G$  的团. 前者是极大团, 而后者不是.  $\{v_0, v_2, v_3, v_4\}$  是  $G$  的最大团.



**注** (1) 任何非平凡的树, 其团数均为 2.

(2)  $Q$  是  $G$  的团当且仅当  $Q$  是  $\bar{G}$  的独立集;  $Q$  是  $G$  的极大 (最大) 团当且仅当  $Q$  是  $\bar{G}$  中极大 (最大) 独立集.

关于团有许多研究专题, 文献 [98]~[103] 分别涉及最大团问题、最大赋权团问题、最大团覆盖问题、边-团覆盖、团独立集、团划分等. 团也与图的染色有密切关系, 这将在下一章讨论.

显然, 团和点独立集是两个互补的概念. 即顶点子集  $Q$  是图  $G$  的团当且仅当  $Q$  是补图  $\bar{G}$  的独立集. 直观上判断, 一个图如果没有大的团则应该有大的点独立集. 这一问题属于 Ramsey 理论的研究范畴.

### 二、Ramsey 数

先来看一个例子.

**例 5.4.1** 任意 6 个人的聚会上, 总有 3 人互相认识或互不认识.

**证明** 用 6 个顶点代表六个人, 构造 6 阶完全图  $K_6$ . 如果两人互相认识, 则将它们连边染红色, 否则染蓝色. 则问题转化为: 证明将  $K_6$  的边进行任意红、蓝染色后,

一定会出现红色三角形或蓝色三角形.

在完全图  $K_6$  中任取一个顶点记为  $v_1$ , 由组合学中的鸽笼原理, 以  $v_1$  为端点的 5 条连边中至少有 3 条是同色的. 不妨设边  $v_1v_2$ 、 $v_1v_3$ 、 $v_1v_4$  都为红色. 现考察连接  $v_2$ 、 $v_3$ 、 $v_4$  的 3 条边, 若这 3 条边全为蓝色, 则  $\Delta v_2v_3v_4$  就是一个同色三角形 (蓝色). 否则, 3 条边中必有红色边, 不妨设为  $v_2v_3$ , 则  $\Delta v_2v_3v_4$  构成一个同色三角形 (红色). 证毕.

上例已表明: 用红蓝两种颜色对完全图  $K_6$  的边染色 (每条边完全染红或染蓝) 后, 要么有红色三角形 (各边全为红色的  $K_3$ ), 要么有蓝色三角形 (各边全为蓝色的  $K_3$ ). 但对  $K_5$  的边进行红、蓝染色后, 不总是能得到同色三角形. 比如, 将  $K_5$  中的边  $v_1v_2$ 、 $v_2v_3$ 、 $v_3v_4$ 、 $v_5v_6$ 、 $v_6v_1$  染红色, 其余边染蓝色, 则既没有红色三角形, 也没有蓝色三角形. 这表明, 用两种颜色给完全图的边染色, 若要求总能出现同色三角形 ( $K_3$ ), 则完全图至少应为 6 阶 ( $K_6$ ). 一个自然的问题是:

用红、蓝两种颜色对完全图的边染色, 要求不论怎么染, 都能要么出现染红色的  $p$  阶完全子图  $K_p$ , 要么出现染蓝色的  $q$  阶完全子图  $K_q$ , 这样的完全图至少应有多少个顶点 (即至少应为多少阶图)?

这个问题称为 **Ramsey 问题**, 所求的完全图的最少顶点数 (即完全图应具有的最小阶数) 称为 **Ramsey 数**, 记为  $r(p, q)$ . 特别地, 定义  $r(1, q) = r(p, 1) = 1$ .

如果将红色边形成的子图记为  $G$ , 则蓝色边形成的子图是  $G$  的补图. 因此上述问题也可描述为:

求  $\min\{r\}$  每个  $r$  阶图  $G$  要么本身含有  $p$ -团, 要么其补图  $\bar{G}$  含有  $q$ -团}.

亦即:

求  $\min\{r\}$  每个  $r$  阶图  $G$  要么有  $p$ -团, 要么有含  $q$  个顶点的独立集}.

Ramsey 问题的上述三种表述经常交替使用.

例 5.4.1 以及其后对  $K_5$  的分析实际上证明了  $r(3, 3) = 6$ .

**定理 5.4.1**  $r(3, 4) = 9$ .

**证明** 先证  $r(3, 4) \leq 9$ .

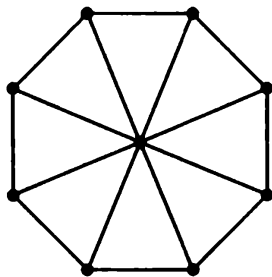
对具有 9 个顶点的任何图  $G$ , 必存在顶点  $v_0$ , 其度  $d(v_0) \neq 3$ . (否则,  $G$  的顶点度之和为奇数).

如果  $d(v_0) \geq 4$ , 设  $v_1$ 、 $v_2$ 、 $v_3$ 、 $v_4$  是其 4 个邻点. 若  $v_1$ 、 $v_2$ 、 $v_3$ 、 $v_4$  间有一条连边, 比如  $v_1v_2$ , 则  $v_0$ 、 $v_1$ 、 $v_2$  构成  $G$  的一个 3-团; 否则,  $v_1$ 、 $v_2$ 、 $v_3$ 、 $v_4$  间没有连边, 形成  $G$  的一个 4 顶点独立集.

如果  $d(v_0) < 3$ , 则  $G$  中至少有 6 个顶点与  $v_0$  不相邻. 由于  $r(3, 3) = 6$ , 故这 6 个顶点在  $G$  中导出的子图要么有 3-团, 要么有含 3 个顶点的独立集. 如果是前者, 则结论已成立; 如果是后者, 则与  $v_0$  一起构成含 4 个顶点的独立集.

综上, 具有 9 个顶点的任何图  $G$ , 要么有 3-团, 要么有 4 个顶点的独立集, 因此  $r(3, 4) \leq 9$ .

另一方面, 如下所示的 8 阶图既无 3-团, 又无 4 个顶点的独立集, 这说明  $r(3, 4) \geq 9$ . 于是  $r(3, 4) = 9$ . 证毕.



**定理 5.4.2**  $r(4, 4) = 18$ .

**证明** 对具有 18 个顶点的任何图  $G$ , 任取其中一个顶点  $v_0$ .

如果  $d(v_0) \geq 9$ , 取其 9 个邻点, 将这 9 个邻点在  $G$  中的点导出子图记为  $G_1$ , 由定理 5.4.1,  $r(3, 4) = 9$ , 故  $G_1$  中要么有 3-团, 要么有 4 个顶点的独立集. 若是前者, 则这个 3-团与  $v_0$  组成  $G$  的一个 4-团; 若是后者, 则这个独立集也是  $G$  中的 4 个顶点的独立集.

如果  $d(v_0) < 9$ , 则与  $v_0$  不相邻的顶点至少有 9 个, 即在  $G$  的补图  $\bar{G}$  中  $v_0$  至少有 9 个邻点. 与以上类似可得,  $\bar{G}$  中要么有 4-团, 要么有 4 个顶点的独立集. 从而  $G$  中要么有 4-团, 要么有 4 个顶点的独立集.

以上结果表明  $r(4, 4) \leq 18$ .

下证  $r(4, 4) > 17$ . 我们只需找出对  $K_{17}$  的边的一种红蓝染色, 使得不出现同色的  $K_4$  即可. 在一个圆周上画出 17 个等分点, 将其依次编号为  $v_0, v_1, v_2, \dots, v_{16}$ , 把整数  $1, 2, \dots, 16$  分为两组:

$$X = \{1, 2, 4, 8, 9, 13, 15, 16\}, \quad Y = \{3, 5, 6, 7, 10, 11, 12, 14\}.$$

按下列规则对  $K_{17}$  进行红、蓝边染色: 对于边  $v_i v_j$ , ( $0 \leq i, j \leq 16$ ),

当  $|i - j| \in X$  时, 将边  $v_i v_j$  染成红色; 当  $|i - j| \in Y$  时, 将边  $v_i v_j$  染成蓝色.

按此规则, 下列边被染成红色

$$(v_0, v_1), (v_0, v_2), (v_0, v_4), (v_0, v_8), (v_0, v_9), (v_0, v_{13}), (v_0, v_{15}), (v_0, v_{16});$$

$$(v_1, v_2), (v_1, v_3), (v_1, v_5), (v_1, v_9), \dots, (v_1, v_{16});$$

$$(v_{15}, v_{16}).$$

其余边全染蓝色. 可以检验, 在这样的颜色下  $K_{17}$  中没有同色的  $K_4$ . 这表明  $r(4, 4) > 17$ .

由以上两方面便得  $r(4, 4) = 17$ . 证毕.

有时用“边染色”的说法进行论证更为直观和简便. 比如上一定理的前半部分可以用边染色方法论述如下.

设对  $K_{18}$  的边进行了任意的红、蓝染色. 考察  $K_{18}$  中从任一顶点  $v_0$  出发的 17 条边. 由鸽笼原理知, 其中至少有 9 条边是同色的, 不妨设为红色. 这 9 条边异于  $v_0$  的 9 个端点构成完全子图  $K_9$ , 由  $r(3, 4) = 9$  知, 该  $K_9$  中要么有红色  $K_3$ , 要么有蓝色  $K_4$ . 若是前者, 则这个红色  $K_3$  与  $v_0$  一起便构成了红色  $K_4$ ; 若是后者, 则本身就是一个蓝色  $K_4$ . 由此便知  $r(4, 4) \leq 18$ .

### 三、Ramsey 数的性质

**定理 5.4.3** (1)  $r(p, q) = r(q, p)$ ;

(2)  $r(2, q) = q, r(p, 2) = p$ ;

(3) 当  $p, q \geq 2$  时, 有  $r(p, q) \leq r(p, q-1) + r(p-1, q)$ , 且当  $r(p, q-1)$  和  $r(p-1, q)$  都是偶数时, 不等号严格成立.

**证明** (1) 设  $r(p, q) = k$ , 则对任何  $k$  阶图  $G$ , 其补图  $\bar{G}$  也是一个  $k$  阶图, 故  $\bar{G}$  中要么有  $p$ -团, 要么有  $q$  个顶点的独立集, 从而  $G$  中要么有  $q$ -团, 要么有  $p$  个顶点的独立集, 由  $G$  的任意性, 知  $r(q, p) = k$ .

(2) 对任一个  $q$  阶图, 要么至少有一条边, 而任何一条边的两个端点即构成 2-团; 要么没有边, 此时所有  $q$  个顶点构成一个独立集. 故  $r(2, q) = q$ . 同理可证  $r(p, 2) = p$ .

(3) 我们用“边染色”的语言来证明该性质, 读者可尝试用团和独立集的表述改写这个证明.

令  $n = r(p, q-1) + r(p-1, q)$ , 下面证明, 用红蓝两色对  $K_n$  的边任意染色后, 必有红色的  $K_p$  或蓝色的  $K_q$ , 从而  $r(p, q) \leq n$ .

事实上, 任取  $K_n$  的一个顶点  $v$ , 设  $v$  连出的  $n-1$  条边中有  $n_1$  条红色边、 $n_2$  条蓝色边. 则

$$n_1 + n_2 = n - 1 = r(p, q-1) + r(p-1, q) - 1.$$

(i) 若  $n_1 \geq r(p-1, q)$ , 则  $v$  通过红边相连的顶点至少有  $r(p-1, q)$  个. 按照  $r(p-1, q)$  的定义, 这  $r(p-1, q)$  个点构成的完全子图  $K_{r(p-1, q)}$  中要么含有红色的  $K_{p-1}$ , 要么含有蓝色的  $K_q$ . 若是前者, 则该红色的  $K_{p-1}$  与  $v$  便构成  $K_n$  中一个红色的  $K_p$ ; 若是后者, 则自然有蓝色的  $K_q$ .

(ii) 若  $n_1 < r(p-1, q)$ , 则  $n_2 \geq r(p, q-1)$ . 与上述过程同样可证得: 任意红蓝边染色后的  $K_n$  中必有红色的  $K_p$  或蓝色的  $K_q$ .



综上, 总有  $r(p, q) \leq n = r(p, q-1) + r(p-1, q)$ .

下面证明当  $r(p, q-1)$  和  $r(p-1, q)$  都是偶数时, 不等号严格成立. 设  $m = r(p, q-1) + r(p-1, q) - 1$ . 因  $m$  是奇数, 故用红蓝两色对完全图  $K_m$  的边任意染色后, 必有某点  $v_0$  关联的边有偶数条被染红 (若每点处都有奇数条红边, 则所有红色边导数的生成子图中顶点度之和为奇数, 这是不可能的). 因  $d(v_0) = m-1 = r(p, q-1) + r(p-1, q) - 2$ , 如果与  $v_0$  关联的  $m-1$  条边中红色边至少有  $r(p-1, q)$  条, 则与上述 (i) 同样可证,  $K_m$  中必有红色的  $K_p$  或蓝色的  $K_q$ ; 如果与  $v_0$  关联的红色边少于  $r(p-1, q)$  条, 则由  $v_0$  的取法 (与  $v_0$  关联的红色边有偶数条),  $v_0$  至多有  $r(p-1, q) - 2$  条红边, 因此至少有  $r(p, q-1)$  条蓝边, 此时与上述 (ii) 同样可知,  $K_m$  中必有红色的  $K_p$  或蓝色的  $K_q$ .

由以上可见, 对  $r(p, q-1) + r(p-1, q) - 1$  个顶点的完全图  $K_m$  的边任意红蓝染色, 必有红色的  $K_p$  或蓝色的  $K_q$ , 故

$$r(p, q) \leq r(p, q-1) + r(p-1, q) - 1 < r(p, q-1) + r(p-1, q). \quad \text{证毕.}$$

**例 5.4.2** 证明:  $r(3, 5) = 14$ .

**证明** 由定理 5.4.3 中的性质 (3),  $r(3, 5) \leq r(3, 4) + r(2, 5) = 9 + 5 = 14$ .

另一方面, 按如下方法对完全图  $K_{13}$  的边进行红、蓝染色: 设  $K_{13}$  的顶点为  $0, 1, 2, \dots, 12$ , 将这些顶点依次放在一个圆周的 13 等分点上. 当点  $i$  与  $k$  满足

$$k = i + 1, i + 5, i + 8, i + 12, (\text{mod } 13),$$

时, 点  $i$  与  $k$  之间连红色边; 否则, 连蓝色边. 可以检验, 对  $K_{13}$  的边这样进行红蓝染色后, 既无红色的  $K_3$ , 又无蓝色的  $K_5$ . 这表明  $r(3, 5) \geq 14$ .

由以上两方面即知,  $r(3, 5) = 14$ . 证毕.

#### 四、Ramsey 数的上下界

**定理 5.4.4** 设  $p, q \geq 2$ , 则  $r(p, q) \leq \binom{p+q-2}{p-1}$ , 其中  $\binom{p+q-2}{p-1}$  表示从  $p+q-2$  个元素中取  $p-1$  个元素的组合数.

**证明** 用数学归纳法.

由性质,  $r(2, q) = q = \binom{2+q-2}{2-1}$ ,  $r(p, 2) = p = \binom{p+2-2}{p-1}$ , 可见当  $p=2$  或  $q=2$  时, 结论成立. 假定对  $r(s-1, t)$  和  $r(s, t-1)$ , 结论都成立. 则

$$r(s, t) \leq r(s-1, t) + r(s, t-1) \leq \binom{s+t-3}{s-2} + \binom{s+t-3}{s-1} = \binom{s+t-2}{s-1}.$$

这里用到了组合恒等式  $\binom{n-1}{m} + \binom{n-1}{m-1} = \binom{n}{m}$ . 证毕.

**定理 5.4.5** 当  $q \geq 3$  时,  $r(3, q) \leq \frac{q^2 + 3}{2}$ .

**证明** 对  $q$  进行归纳. 当  $q = 3$  时,  $r(3, 3) = 6$ , 且  $\frac{q^2 + 3}{2} = 6$ , 结论不等式成立.

当  $q = 4$  时,  $r(3, 4) = 9$ , 而  $\frac{q^2 + 3}{2} = \frac{19}{2}$ , 结论不等式严格成立.

假设  $q \leq k-1$  时结论成立. 则  $q = k$  时,

$$r(3, k) \leq r(2, k) + r(3, k-1) \leq k + \frac{(k-1)^2 + 3}{2} = \frac{k^2 + 4}{2}.$$

下证上述不等式严格成立, 便证明了  $r(3, k) \leq \frac{k^2 + 3}{2}$ .

事实上, 若  $k$  是奇数, 则  $k^2 + 4$  是奇数, 故此时,  $r(3, k) < \frac{k^2 + 4}{2}$ ; 下设  $k$  是偶数. 此时如果  $r(3, k-1) < \frac{(k-1)^2 + 3}{2}$ , 则所证不等式严格成立; 如果  $r(3, k-1) = \frac{(k-1)^2 + 3}{2}$ , 则因  $k$  是偶数, 从而  $r(2, k) = k$  和  $r(3, k-1) = \frac{(k-1)^2 + 3}{2} = \frac{k^2}{2} - k + 2$  都是偶数, 由定理 5.4.3, 此时  $r(3, k) < r(2, k) + r(3, k-1)$ , 故所证不等式严格成立. 证毕.

当  $p = q$  时, Ramsey 数  $r(p, p)$  称为对角 Ramsey 数. 对角 Ramsey 数的一个下界如下.

**定理 5.4.6** 当  $p \geq 3$  时,  $r(p, p) \geq 2^{\frac{p}{2}}$ .

**证明** 对  $n$  阶完全图  $K_n$  进行任意红蓝边染色时, 由于其每条边都有染红和染蓝两种可能, 而  $K_n$  共有  $\frac{n(n-1)}{2}$  条边, 故不同的染色方案共有  $2^{\frac{n(n-1)}{2}}$  种. 对于  $K_n$  中指定的  $p$  个顶点, 它们之间的连边有  $\frac{p(p-1)}{2}$  条. 在  $K_n$  的所有染色方案中, 使这  $p$  个顶点形成的完全子图  $K_p$  全染红边的方案有  $2^{\frac{n(n-1)}{2} - \frac{p(p-1)}{2}}$  种 (除  $K_p$  的边外, 其余  $\frac{n(n-1)}{2} - \frac{p(p-1)}{2}$  条边每条有 2 种染法). 因  $K_n$  中指定  $p$  个顶点的方式有  $\binom{n}{p}$  种, 故在  $K_n$  的所有红蓝边染色方案中, 出现红色  $K_p$  的染法有  $\binom{n}{p} 2^{\frac{n(n-1)}{2} - \frac{p(p-1)}{2}}$  种. 它占染色方案总数的比例为

$$\frac{\binom{n}{p} 2^{\frac{n(n-1)}{2} - \frac{p(p-1)}{2}}}{2^{\frac{n(n-1)}{2}}} = \binom{n}{p} 2^{-\frac{p(p-1)}{2}} < \frac{n^p 2^{-\frac{p(p-1)}{2}}}{p!}.$$

假定  $n < 2^{\frac{p}{2}}$ , 则  $\frac{\binom{n}{p} 2^{\frac{n(n-1)}{2} - \frac{p(p-1)}{2}}}{2^{\frac{n(n-1)}{2}}} < \frac{2^{\frac{p^2}{2}} 2^{-\frac{p(p-1)}{2}}}{p!} = \frac{2^{\frac{p}{2}}}{p!} = \frac{\sqrt{2}}{1} \cdot \frac{\sqrt{2}}{2} \cdot \frac{\sqrt{2}}{3} \cdots \frac{\sqrt{2}}{p} < \frac{1}{2}$ .

这表明在  $K_n$  的所有红蓝边染色方案中, 出现红色  $K_p$  的染法不到染色方案总数的一半. 由两种颜色的对等性, 出现蓝色  $K_p$  的染法也不到染色方案总数的一半. 由此可知, 只要  $n < 2^{\frac{p}{2}}$ , 则必有  $K_n$  的一种红蓝边染色方案, 既不出现红色  $K_p$  也不出现蓝色  $K_p$ . 故  $r(p, p) \geq 2^{\frac{p}{2}}$ . 证毕.

由定理 5.4.6 容易得到下列推论 (习题 5.23).

**推论 5.4.1** 设  $m = \min\{p, q\}$ , 则  $r(p, q) \geq 2^{\frac{m}{2}}$ .

下表列出了目前已经确定的所有 Ramsey 数和一些上下界, 此表引自 [104] 和 [105].

一些  $r(p, q)$  的值和界

$p \backslash q$	3	4	5	6	7	8	9
3	6	9	14	18	23	28	36
4		18	25	35/41	49/61	55/84	69/115
5			43/49	58/87	80/143	95/216	121/316
6				102/165	109/298	122/495	153/780
7					205/540	?/1031	?/1713
8						?/1870	282/3583
9							?/6625
10							
$p \backslash q$	10	11	12	13	14	15	
3	40/43	46/51	51/60	59/69	66/78	73/89	
4	80/149	96/191	106/238	118/291	129/349	134/417	
5	?/442						
6	?/1171						
7	?/2826						
8	?/6096						
9	565/12715						
10	?/23854	798/?					

表中  $a/b$  表示  $a$  为下界,  $b$  为上界; ? 表示目前尚未确定出的下界或上界.

## 五、Ramsey 数的一般形式

**定义 5.4.2** 设  $p_1, p_2, \dots, p_m$  是一组给定的大于 1 的正整数, 用  $m$  种颜色  $C_1, C_2, \dots, C_m$  对完全图的边染色, 要求不论怎么染, 都能要么出现  $C_1$  色的完全子图  $K_{p_1}$ , 要么出现  $C_2$  色的完全子图  $K_{p_2}, \dots$ , 要么出现  $C_m$  色的完全子图  $K_{p_m}$ , 这样的完全图至少应有多少个顶点 (即至少应为多少阶图)? 这个问题称为一般 Ramsey 问题, 所求的最少顶点数 (即完全图的阶数) 称为一般 Ramsey 数, 记为  $r(p_1, p_2, \dots, p_m)$ .

**例 5.4.3** 有 17 位学者, 每人给其他人各写一封信, 讨论三个问题中的某一个问题, 且两人之间互相通信讨论的是同一问题. 证明至少有三位学者, 他们之间通信讨论的是同一个问题.

**证明** 此问题等价于证明用红黄蓝三种颜色对  $K_{17}$  的边任意染色后, 必出现同色三角形  $K_3$ .

设  $v_0, v_1, v_2, \dots, v_{16}$  是  $K_{17}$  的顶点. 对  $K_{17}$  的边任意进行红黄蓝三染色后,  $v_0$  与其余 16 点的连边中至少有  $\left\lfloor \frac{16}{3} \right\rfloor = 6$  条同色边, 不妨设为  $v_0v_1, v_0v_2, \dots, v_0v_6$  这六条边被染红色. 考虑由  $v_1, v_2, \dots, v_6$  组成的完全子图  $K_6$ . 如果其中有一条红色边, 比如  $v_1v_2$ , 则  $v_0, v_1, v_2$  三点便组成红色三角形; 如果其中没有红色边, 则  $K_6$  的边全为黄蓝两色, 归结为  $K_6$  的边 2 染色问题, 由于  $r(3, 3) = 6$ , 因此其中必含有黄色三角形或蓝色三角形. 证毕.

关于一般 Ramsey 数, 有如下类似于定理 5.4.3 的性质.

(1) 对正整数  $p_1, p_2, \dots, p_m$  的任一种排列  $p_{i_1}, p_{i_2}, \dots, p_{i_m}$ , 有

$$r(p_1, p_2, \dots, p_m) = r(p_{i_1}, p_{i_2}, \dots, p_{i_m});$$

(2)  $r(2, p_1, p_2, \dots, p_m) = r(p_1, p_2, \dots, p_m)$ ;

(3)  $r(p_1, p_2, \dots, p_m) \leq r(p_1 - 1, p_2, \dots, p_m) + r(p_1, p_2 - 1, \dots, p_m) + \dots + r(p_1, p_2, \dots, p_m - 1) - m + 2$ ;

(4)  $r(p_1 + 1, p_2 + 1, \dots, p_m + 1) \leq \frac{(p_1 + p_2 + \dots + p_m)!}{p_1! p_2! \dots p_m!}$ .

其中 (1)~(3) 的证明与定理 5.4.3 类似, (4) 的证明与定理 5.4.4 类似.

Greenwood 和 Gleason 已经证明:  $r(3, 3, 3) = 17$ . 另有一些一般 Ramsey 数已获得其上下界.

Ramsey 理论是一个十分精彩而又相当难的研究方向. Ramsey 数有多种推广形式. 有关更深入的内容可参看专著 [106]、[107] 及文献 [108]~[112].

## 习题五

5.1 设  $D$  是  $G$  的一个支配集, 证明  $V(G) - D \subseteq \bigcup_{v \in D} N(v)$ .

5.2 表述支配数为 1 的图的特征.

5.3  $C_n$  和  $P_n$  分别表示有  $n$  个顶点的圈和路. 给出求  $\gamma(C_n)$  和  $\gamma(P_n)$  的公式.

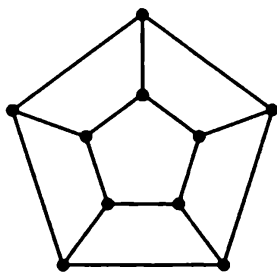
5.4 确定 Petersen 图的支配数.

5.5 设  $G$  是一个简单图, 证明:  $\gamma(G) \leq \frac{\nu}{2}$ . (提示: 利用结论: 若  $S$  是极小支配集, 则  $\bar{S}$  也是支配集.)

5.6 设  $G$  是简单图, 证明:  $\frac{1 + \text{diam}G}{3} \leq \gamma(G) \leq \nu - \left\lfloor \frac{\text{diam}G}{2} \right\rfloor$ .

5.7 证明: 若图  $G$  的直径至少是 3, 则  $\gamma(\bar{G}) \leq 2$ .

5.8 下图是连接 10 个城市的宽带流媒体骨干网络的结构示意图. 在这个网络上, 用户可以进行通信或获取各种网络信息, 也可以召开网络会议、进行远程教学, 还可以实时在线点播音乐、电影和电视剧、玩电子游戏或视屏聊天. 现在网络设计者想要在部分节点设置一些大型数据库存放网络资源供用户下载, 这些数据库完全相同并需要定期更新. 为了节省网络带宽和缩短用户下载时延, 要求每个城市经过至多一跳 (即一条边) 便能访问到数据库. 另一方面, 出于建设成本和更新维护成本的考虑, 设计者又希望设置尽可能少的数据库达到上述要求. 请建立这一问题的图论模型, 并帮设计者找出一个最优的设置方案, 同时要解释你的方案为什么是最优的.



5.9 证明: 图  $G$  是二部图当且仅当对  $G$  的每个子图  $H$  均有  $\alpha(H) \geq \frac{1}{2}\nu(H)$ .

5.10 设  $G$  是一个二部图, 证明  $\alpha(G) = \frac{1}{2}\nu(G)$  当且仅当  $G$  有一个完美匹配.

5.11 设  $G$  是恰含一个圈的非二部图, 证明  $\alpha(G) \geq \left\lfloor \frac{\nu-1}{2} \right\rfloor$ .

5.12 设  $G$  是一个非平凡简单图, 证明: (1)  $\alpha(G) \leq \nu - \frac{\varepsilon}{\Delta}$ ;

(2) 如果  $G$  是正则图, 则  $\alpha(G) \leq \frac{\nu}{2}$ .

5.13 设  $G$  是一个简单图, 其中任意  $k$  个顶点的度之和小于  $n - k$ , 证明图  $G$  的任意极大独立集的顶点数大于  $k$ .

5.14 证明推论 5.1.3:  $F$  是图  $G$  的极小点覆盖集当且仅当  $V(G) \setminus F$  是  $G$  的极大点独立集.

5.15 证明推论 5.1.4:  $\alpha(G) + \beta(G) = \nu$ .

5.16 利用 Konig 定理证明: 任一个二部图  $G$  有一个至少含  $\frac{\varepsilon(G)}{\Delta(G)}$  条边的匹配. 并由此证明: 若  $H$  是完全二部图  $K_{n,n}$  的一个子图, 且  $\varepsilon(H) > (k-1)n$ , 则  $H$  中存在至少含有  $k$  条边的匹配.

5.17 设  $M$  是图  $G$  的完美匹配,  $L$  是  $G$  的最小边覆盖, 证明必定  $|M| = |L|$ . 举例说明: 若  $M$  是图  $G$  的最大匹配,  $L$  是  $G$  的最小边覆盖, 则未必有  $|M| = |L|$ .

5.18 设  $G$  是一个无孤立点的图,  $M$  是  $G$  的一个极大匹配,  $L$  是  $G$  的一个极小边覆盖. 证明:

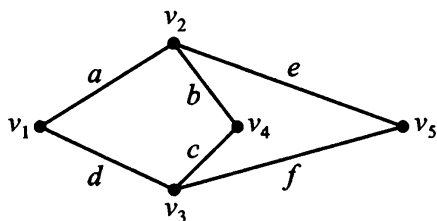
- (1)  $M$  是最大匹配当且仅当  $M$  含于某个最小边覆盖中;
- (2)  $L$  是最小边覆盖当且仅当  $L$  包含一个最大匹配.

5.19 设  $G$  是无孤立点的图, 证明:  $\gamma(G) \leq \nu - \beta'(G) \leq \frac{\nu}{2}$ .

5.20 证明: 图  $G$  是二部图当且仅当对  $G$  的每个适合  $\delta(H) > 0$  的子图  $H$  均有  $\alpha(H) = \beta'(H)$ .

5.21 如图  $G$ , 求:

- (1)  $G$  的所有极小支配集及支配数;
- (2)  $G$  的所有极小点覆盖集及点覆盖数;
- (3)  $G$  的所有极大点独立集及点独立数;
- (4)  $G$  的一个最大边独立集和一个最小边覆盖集.



5.22 用边染色表述证明定理 5.4.1, 即:

- (1) 证明: 对 9 个顶点构成的完全图  $K_9$ , 将它的边进行任意红、蓝染色后, 一定会出现同色的三角形或同色的  $K_4$ .
- (2) 对完全图  $K_8$ , 找出一种对其边进行红蓝染色的方案, 使得既不出现红色的  $K_3$  又不出现蓝色的  $K_4$ .

5.23 证明推论 5.4.1: 设  $m = \min\{p, q\}$ , 则 Ramsey 数  $r(p, q) \geq 2^{\frac{m}{2}}$ .

5.24 证明: 对任意正整数  $p_1, p_2, \dots, p_m$ , 有  $r(p_1, p_2, \dots, p_m) \leq r(p_1, r(p_2, \dots, p_m))$ .

5.25 证明:

(1) 对正整数  $p_1, p_2, \dots, p_m$  的任一种排列  $p_{i_1}, p_{i_2}, \dots, p_{i_m}$ , 有

$$r(p_1, p_2, \dots, p_m) = r(p_{i_1}, p_{i_2}, \dots, p_{i_m}).$$

(2)  $r(2, p_1, p_2, \dots, p_m) = r(p_1, p_2, \dots, p_m)$ .

5.26 证明:  $r(3, 3, 3) \leq 17$ .

5.27 平面上有 6 个点, 其中任意三点都能组成一个不等边三角形. 证明: 在这些点构成的所有三角形中, 必有一个三角形的最短边是另一个三角形的最长边.

5.28 九位科学家在一次国际会议上相遇, 发现他们中的任意 3 个人中, 至少有 2 个人可以用同一种语言交谈. 如果每个科学家最多会说 3 种语言, 证明至少有 3 个科学家可以用同一种语言交谈.

## 参考文献

### 支配集

- [1] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater, *Fundamentals of Domination in Graphs*, Marcel Dekker, Inc., 1998.
- [2] Y. Caro, D. B. West, and R. Yuster, Connected domination and spanning trees with many leaves, *SIAM Journal on Discrete Mathematics*, 13(2000), 202-211.
- [3] F. Harary, T. Haynes, Double domination in graphs, *ARS Combinatoria*, 55(1) (2000), 201-13.
- [4] J. S. Deogun and D. Kratsch, Dominating Pair Graphs, *SIAM Journal on Discrete Mathematics*, 15(2001-2002), 353-366.
- [5] Chin Lung Lu, Ming-Tat Ko and Chuan Yi Tang, Perfect edge domination and efficient edge domination in graphs, *Discrete Applied Mathematics*, 119(2002), 227-250.
- [6] Chin Lung Lu and Chuan Yi Tang, Weighted efficient domination problem on some perfect graphs, *Discrete Applied Mathematics*, 117(2002), 163-182.
- [7] D. Kobler and U. Rotics, Edge dominating set and colorings on graphs with fixed clique-width, *Discrete Applied Mathematics*, 126(2003), 197-221.
- [8] B.G.. Xu, Two classes of edge domination in graphs, *Discrete Applied Mathematics*, 154(2006), 1541-1546.
- [9] T.W. Haynes, M.A. Henning, and J.Howard, Locating and total dominating sets in trees, *Discrete Applied Mathematics*, 154(2006), 1293-1300.
- [10] Jun-Ming Xu, Fang Tian, Jia Huang, Distance irredundance and connected domination numbers of a graph. *Discrete Mathematics*, 306(2006), 2943-2953.

- [11] F. V. Fomin and D. M. Thilikos, Dominating Sets in Planar Graphs: Branch-Width and Exponential Speed-Up, *SIAM Journal on Computing*, 36(2006), 281-309.
- [12] K. I. Kawarabayashi, M. D. Plummer and A. Saito, Domination in a graph with a 2-factor, *Journal of Graph Theory*, 52(2006), 1-6.
- [13] P. Haxell, B. Seamone, J. Verstraete, Independent dominating sets and Hamiltonian cycles, *Journal of Graph Theory*, 54(2007), 233-244.
- [14] M. A. Henning, Restricted domination in graphs with minimum degree 2. *Discrete Mathematics*, 307(2007), 1356-1366.
- [15] E. J. Cockayne, Irredundance, secure domination and maximum degree in trees. *Discrete Mathematics*, 307(2007), 12-17.
- [16] J. Raczek, J. Cyman, Total restrained domination numbers of trees. *Discrete Mathematics*, 308(2008), 44-50.
- [17] O. Favaron, A. Hansberg, L. Volkmann, On  $k$ -domination and minimum degree in graphs, *Journal of Graph Theory*, 57(2008), 33-40.

#### 支配集算法

- [18] R. Raz, and S. Safra, A sub-constant error-probability low-degree test, and sub-constant error-probability PCP characterization of NP, *Proc. 29th Ann. ACM Symp. on Theory of Comp.*, ACM, (1997), 475-484.
- [19] U. Feige, A threshold of  $\ln n$  for approximating set cover, *J. ACM* 45(1998), 634-652.
- [20] S. Khanna, R. Motwani, M. Sudan, and U. Vazirani, On syntactic versus computational views of approximability, *SIAM J. Comp.*, 28(1999), 164-191.
- [21] Toshihiro Fujito and Hiroshi Nagamochi, A 2-approximation algorithm for the minimum weight edge dominating set problem, *Discrete Applied Mathematics*, 118(2002), 199-207.
- [22] Lucia D. Penso and V.C. Valmir C. Barbosa, A distributed algorithm to find  $k$ -dominating sets, *Discrete Applied Mathematics*, 141(2004), 243-253.
- [23] A. Berger, O. Parekh, Linear time algorithms for generalized edge dominating set problems, *Algorithmica*, 50(2)(2008), 244-254.

#### 支配集应用

- [24] B. Das, and V. Bharghavan, Routing in ad hoc networks using minimum connected dominating sets, *ICC'97*, Montreal, Canada, June, 1997.
- [25] S. Guha, and S. Khuller, Approximation algorithms for connected dominating sets, *Algorithmica*, 20(4)(1998), 374-387. also see *Algorithmica*, 49(1)(2007), 79.
- [26] J. Wu, and H.L.Li, On calculating connected dominating set for efficient routing in ad hoc wireless networks, *Proceedings of the 3<sup>rd</sup> ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, 1999, 7-14.



- [27] I. Stojmenovic, M. Seddigh, and J. Zunic, Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks, *Proc. IEEE Hawaii Int. Conf. On System Sciences*, Jan. 2001.
- [28] K. M. Alzoubi, P.-J. Wan, O. Frieder, Distributed heuristics for connected dominating sets in wireless ad hoc networks, *Journal of Communications and Networks*, 4(1)(2002), 22-29.
- [29] R. Klasing, C. Laforest, Hardness results and approximation algorithms of  $k$ -tuple domination in graphs, *Information Processing Letters*, 89(2) (2004), 75-83.
- [30] P.-J. Wan, K. M. Alzoubi, O. Frieder, Distributed construction of connected dominating set in wireless ad hoc networks, *Mobile Networks and Applications*, 9(2) (2004), 141-149.
- [31] Y. S. Li, M. T. Thai, F. Wang, C.-W. Yi, P.-J. Wan, and D.-Z. Du, On greedy construction of connected dominating sets in wireless networks, *Wiley Journal on Wireless Communications and Mobile Computing*, 5(8) (2005), 927-932.
- [32] W. Wu, H. Du, X. Jia, Y. Li, C.-H. Huang, Minimum connected dominating sets and maximal independent sets in unit disk graphs, *Theoretical Computer Science*, 352(1) (2006), 1-7.
- [33] F. Dai and J. Wu, On constructing  $k$ -connected  $k$ -dominating set in wireless ad hoc and sensor networks, *Journal of Parallel and Distributed Computing*, 66(7)(2006), 947-958.
- [34] T.W. Haynes, S.M. Hedetniemi, S.T. Hedetniemi, and M.A. Henning, Domination in graphs applied to electric power networks, *SIAM J. Discrete Mathematics*, 15: 4(2001-2002), 519-529.

### 独立集

- [35] B. Toft. Colouring, stable sets and perfect graphs. In *Handbook of Combinatorics* (R. Graham, M. Grötschel, and L. Lovász, editors), volume 1, chapter 4, North-Holland, 1995, pages 233–288.
- [36] A. Jagota, G. Narasimhan and L'ubomír Šoltés, A Generalization of maximal independent sets, *Discrete Applied Mathematics*, 117(2002), 223-235.
- [37] U. Feige and R. Krauthgamer, The Probable Value of the Lovász–Schrijver Relaxations for Maximum Independent Set, *SIAM Journal on Computing*, 32(2002-2003), 345-370.
- [38] R. J. Faudree, Z. Ryjáček and R. H. Schelp, On local and global independence numbers of a graph, *Discrete Applied Mathematics*, 132(2003), 79-84.
- [39] A. S. Pedersen and P. D. Vestergaard, The number of independent sets in unicyclic graphs, *Discrete Applied Mathematics*, 152(2005), 246-256.
- [40] M. Fischermann, L. Volkmann and D. Rautenbach, A note on the number of matchings and independent sets in trees, *Discrete Applied Mathematics*, 145(2005), 483-489.
- [41] G. C. Ying, K. K. Meng, B. E. Sagan, V. R. Vatter, Maximal independent sets in graphs with at most  $r$  cycles, *Journal of Graph Theory*, 53(2006), 270-282.

- [42] B. E. Sagan, V. R. Vatter, Maximal and maximum independent sets in graphs with at most  $r$  cycles, *Journal of Graph Theory*, 53(2006), 283-314.
- [43] J. Lauer, N. Wormald, Large independent sets in regular graphs of large girth, *Journal of Combinatorial Theory*, (Ser.B), 97(2007), 999-1009.
- [44] N. Alon, B. Sudakov, On graphs with subgraphs having large independence numbers, *Journal of Graph Theory*, 56(2007), 149-157.
- [45] K.-I. Kawarabayashi, Zi-Xia Song, Independence number and clique minors, *Journal of Graph Theory*, 56(2007), 219-226.
- [46] S. S. Kamath, R. S. Bhat, On strong (weak) independent sets and vertex coverings of a graph. *Discrete Mathematics*, 307(2007), 1136-1145.
- [47] Y. Egawa, H. Enomoto, S. Jendrol, K. Ota, I. Schiermeyer, Independence number and vertex-disjoint cycles. *Discrete Mathematics*, 307(2007), 1493-1498.
- [48] Guantao Chen, Xuechao Li, Zhengsheng Wu, Xingping Xu, The neighborhood union of independent sets and Hamiltonicity of graphs. *Discrete Mathematics*, 307(2007), 2226-2234.

#### 点独立集算法

- [49] D. S. Johnson, Approximation algorithms for combinatorial problems, *J. Comput. System Sci.* 9(1974), 256-278.
- [50] M.R. Garey, and D.S. Johnson, *Computers and Intractability: A Guide to The Theory of NP-completeness*, Freeman, New York, 1979.
- [51] D.S. Hochbaum, *Approximation Algorithms for NP-hard Problems*, International Thomson Publishing Inc., 1997 (影印本: *NP 难解问题的近似算法*. 世界图书出版公司).
- [52] V. Kann, *On the Approximability of NP-complete Optimization Problems*, PhD thesis, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, 1992.
- [53] D. S. Hochbaum, Efficient bounds for the stable set, vertex cover and set packing problems, *Disc. Appl. Math.* 6(1983), 243-254.
- [54] <http://www.nada.kth.se/~viggo/problemlist/>
- [55] K. Nakajima, S. Masuda, An optimal algorithm for finding a maximum independent set in circular-arc graphs, *SIAM Journal of Computing*, 17(1988), 41-52.
- [56] B. S. Baker, Approximation algorithms for NP-complete problems on planar graphs, *J. ACM*, 41 (1994), 153-180.
- [57] C. H. Papadimitriou, and M. Yannakakis, Optimization, approximation, and complexity classes, *J. Comput. System Sci.*, 43 (1991), 425-440.
- [58] M. M. Halldórsson and J. Radhakrishnan, Greedy is good: approximating independent sets in sparse and bounded-degree graphs, Proceedings of 26<sup>th</sup> ACM Symposium on Theory of Computing, 1994, 439-448

- [59] P. Berman and T. Fujito, Approximating independent sets in degree 3 graphs, *Proc. 4th Workshop on Algorithms and Data Structures*, Lecture Notes in Comput. Sci. 955, Springer-Verlag, (1995), 449-460.
- [60] M. M. Halldórsson, Approximating discrete collections via local improvements, *Proc. 6th Ann. ACM-SIAM Symp. on Discrete Algorithms*, ACM-SIAM, (1995), 160-169.
- [61] N. Alon, U. Feige, A. Wigderson, and D. Zuckerman, Derandomized graph products, *Computational Complexity*, **5** (1995), 60-75.
- [62] O. Verbitsky, On the hardness of approximating some optimization problems that are supposedly easier than Max Clique, *Combinatorics, Probability and Computing*, **4**(1995), 167-180.
- [63] M. M. Halldórsson, and H. C. Lau, Low-degree graph partitioning via local search with applications to constraint satisfaction, max cut, and 3-coloring, *J. Graph Algorithms and Applications*, **1**(1997), 1-13.
- [64] H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, S. S.Ravi, D. J. Rosenkrantz, and R. E. Stearns, Nc-approximation schemes for NP- and PSPACE-hard problems for geometric graphs, *J. Algorithms*, **26**(1998), 238-274.
- [65] P. Berman, and M. Karpinski, On some tighter inapproximability results, Technical Report TR98-065, ECCC (1998). <ftp://ftp.eccc.uni-trier.de/pub/eccc/reports/1998/TR98-065/index.html>
- [66] D. Karger, R. Motwani, and M. Sudan, Approximate graph coloring by semidefinite programming, *J. ACM*, **45**(1998), 246-265.
- [67] M. M. Halldórsson, Approximations of weighted independent set and hereditary subset problems, *Proc. 5th Ann. Int. Conf. on Computing and Combinatorics*, Lecture Notes in Comput. Sci., Springer-Verlag, (1999), 261-270.
- [68] S. Sakai, M. Togasaki and K. Yamazaki, A note on greedy algorithms for the maximum weighted independent set problem, *Discrete Applied Mathematics*, **126**(2003), 313-322.
- [69] V. E. Alekseev, Polynomial algorithm for finding the largest independent sets in graphs without forks, *Discrete Applied Mathematics*, **135**(2004), 3-16.
- [70] T. Erlebach and K. Jansen, Conversion of coloring algorithms into maximum weight independent set algorithms, *Discrete Applied Mathematics*, **148**(2005), 107-125.
- [71] U. Faigle and G. Frahling, A combinatorial algorithm for weighted stable sets in bipartite graphs, *Discrete Applied Mathematics*, **154**(2006), 1380-1391.
- [72] J. Snoeyink, Maximum independent set for intervals by divide and conquer with pruning, *Networks*, **49**(2)(2007), 158-159.

#### 点覆盖及其算法

- [73] D. König, Graphen und Matrizen, *Math. Lapok*, **38**(1931), 116-119.

- [74] E. Egerváry, On combinatorial properties of matrices, *Math. Lapok*, 38(1931), 16-28.
- [75] A. Schrijver, *Combinatorial Optimization Polyhedra and Efficiency*, New York, NY: Springer-Verlag, 2003.
- [76] D.S. Hochbaum, Approximation algorithms for set covering and vertex cover problems, *SIAM Journal on Computing*, 11(1982) 555-556.
- [77] C. Savage, Depth-first search and the vertex cover problem, *Inform. Process. Lett.*, 14(1982), 233-237.
- [78] K.L. Clarkson, A modification of the greedy algorithm for vertex cover, *Information Processing Letters*, 17(1983) 23-25.
- [79] R. Bar-Yehuda, and S. Moran, On approximation problems related to the independent set and vertex cover problems, *Discrete Applied Mathematics*, (1984), 1-10.
- [80] R. Bar-Yehuda and S. Even, A local-ratio theorem for approximating the weighted vertex cover problem, *Annals of Discrete Mathematics*, 25(1985) 27-45.
- [81] B. Monien, and E. Speckenmeyer, Ramsey numbers and an approximation algorithm for the vertex cover problem, *Acta Inf.*, 22, (1985), 115-123.
- [82] E. M. Arkin, M. M. Halldórsson, and R. Hassin, Approximating the tree and tour covers of a graph, *Inform. Process. Lett.*, 47 (1993), 275-282.
- [83] D. Zuckerman, NP-complete problems have a version that's hard to approximate, *Proc. Eight Ann. Structure in Complexity Theory Conf.*, IEEE Computer Society, (1993), 305-312.
- [84] E. Petrank, The hardness of approximation: gap location, *Computational Complexity* 4(1994), 133-157.
- [85] A. Clementi, and L. Trevisan, Improved non-approximability results for vertex cover problems with density constraints, *Proc. 2nd Ann. Int. Conf. on Computing and Combinatorics*, Lecture Notes in Comput. Sci. 1090, Springer-Verlag, (1996), 333-342.
- [86] P. Alimonti, and V. Kann, Hardness of approximating problems on cubic graphs, *Proc. 3rd Italian Conf. on Algorithms and Complexity*, Lecture Notes in Comput. Sci. 1203, Springer-Verlag, 1997, pp288-298.
- [87] M. Karpinski, Polynomial time approximation schemes for some dense instances of NP-hard optimization problems, *Proc. 1st Symp. on Randomization and Approximation Techniques in Comput. Sci.*, Lecture Notes in Comput. Sci. 1269, Springer-Verlag, (1997), 1-14.
- [88] M. Karpinski, and A. Zelikovsky, Approximating dense cases of covering problems, Technical Report TR97-004, ECCO. 1997, <ftp://ftp.eccc.uni-trier.de/pub/eccc/reports/1997/TR97-004/index.html>
- [89] J. Håstad, Some optimal inapproximability results, *Proc. 29th Ann. ACM Symp. on Theory of Comp.*, ACM, (1997), 1-10.

- [90] N. Bshouty, and L. Burroughs, Massaging a linear programming solution to give a 2-approximation for a generalization of the vertex cover problem, *Proc. 15th Ann. Symp. on Theoretical Aspects of Comput. Sci.*, Lecture Notes in Comput. Sci., Springer-Verlag, (1998), 298-308.
- [91] H. Nagamochi, and T. Ibaraki, An approximation of the minimum vertex cover in a graph, *Japan J. Indust. Appl. Math.*, 16(1999), 369-375.
- [92] E. Halperin, Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs, *Proc. 11th Ann. ACM-SIAM Symp. on Discrete Algorithms*, ACM-SIAM., (2000).
- [93] J. Chen, I. Kanj, On constrained minimum vertex covers of bipartite graphs: improved algorithms, *Lecture notes in Computer Science* 2204, 2001, 55-65.
- [94] J. Chen, I. Kanj, W. Jia, Vertex cover: further observations and further improvements, *Journal of Algorithms*, 41(2001), 280-301.
- [95] S. Yang, F. Dai, M. Cardei, J. Wu, On connected multiple point coverage in wireless sensor networks, *Proceedings of the 2<sup>nd</sup> IEEE International Conference on mobile Ad-hoc and Sensor Systems*, 2005. also see *International Journal of Wireless Information Networks*, 34(4) (2006), 289-301.
- [96] Xiaole Bai, S. Kumar, Dong Xuan, Ziqiu Yun, Ten H. Lai, Deploying wireless sensors to achieve both coverage and connectivity, *Proceedings of the 7<sup>th</sup> International Symposium on Mobile Ad-hoc Networking and Computing*, Florence, Italy, 2006, 131-142.
- [97] Xiaole Bai, Ziqiu Yun, Dong Xuan, Ten H. Lai, Weijia Jia, Deploying four-connectivity and full-coverage wireless sensor networks, *Proceedings of the 27<sup>th</sup> Conference on Computer and Communications (INFOCOM)*, 2008.
- 团
- [98] Patric R. J. Östergård, A fast algorithm for the maximum clique problem, *Discrete Applied Mathematics*, 120(2002), 197-207.
- [99] S. Busygin, A new trust region technique for the maximum weight clique problem, *Discrete Applied Mathematics*, 154(2006), 2080-2096.
- [100] J.M.Keil, and L.Stewart, Approximating the minimum clique cover and other hard problems in subtree filament, *Discrete Applied Mathematics*, 154(2006), 1983-1995.
- [101] G.Durán, M.C.Lin, S.Mera and J.L.Sawarcfiter, Algorithms for clique-independent sets on subclasses of circular-arc graphs, *Discrete Applied Mathematics*, 154(2006), 1783-1790.
- [102] T.S. Michael, and T.Quint, Sphericity, cubicity, and edge clique covers of graphs, *Discrete Applied Mathematics*, 154(2006), 1309-1313.
- [103] I. Charon, and O. Hudry, Noising methods for a clique partitioning problem, *Discrete Applied Mathematics*, 154(2006), 754-769.

## Ramsey 数

- [104] D.B. West, *Introduction to Graph Theory* (second edition), Prentice-Hall, Inc., 2001. (中译本: 图论导引. 李建中, 骆吉周译. 北京: 机械工业出版社, 2006).
- [105] 张先迪, 李正良. 图论及其应用. 北京: 高等教育出版社, 2005.
- [106] T.D. Parsons, *Ramsey graph theory*, Selected Topics in Graph Theory (L. W. Beineke, and R.J. Wilson eds.), Academic Press, London, (1978) 361-384.
- [107] R.L. Graham, B.L. Rothschild, and J.E. Spencer, *Ramsey Theory*, Wiley, New York, 1990.
- [108] O. Pikhurko, Asymptotic Size Ramsey Results for Bipartite Graphs, *SIAM Journal on Discrete Mathematics*, 16(2002-2003), 99-113.
- [109] E. T. Baskoro, Hasmawati, H. Assiyatun, The Ramsey numbers for disjoint unions of trees. *Discrete Mathematics*, 306(2006), 3297-3301.
- [110] Surahmat, E. T. Baskoro, I. Tomescu, The Ramsey numbers of large cycles versus wheels. *Discrete Mathematics*, 306(2006), 3334-3337.
- [111] H. Bielak, Ramsey and 2-local Ramsey numbers for disjoint unions of cycles. *Discrete Mathematics*, 307(2007), 319-330.
- [112] A. N. M. Salman, H. Broersma, On Ramsey numbers for paths versus wheels. *Discrete Mathematics*, 307(2007), 975-982.

# 第六章 染色理论

很多实际问题可以归结为求图的匹配或者独立集. 此外, 在许多应用中, 人们希望知道: 一个给定的图, 它的边集至少能划分成多少个边不重的匹配? 或它的顶点集至少能划分成多少个无公共顶点的独立集? 这便是图的边染色和顶点染色问题.

## §6.1 边染色

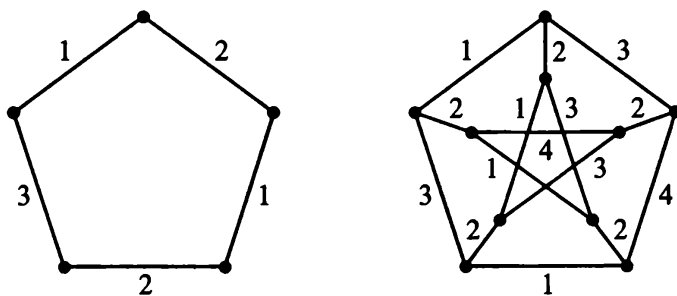
**定义 6.1.1** 非空无环图  $G$  的边正常  $k$  染色 (proper edge  $k$ -colouring) 是指  $k$  种颜色  $1, 2, \dots, k$  对  $E(G)$  中元素的一种分配, 使得相邻两条边所染颜色不同. 换句话说,  $G$  中边正常  $k$  染色是映射

$$c: E(G) \rightarrow \{1, 2, \dots, k\},$$

使得对每个  $i (i = 1, 2, \dots, k), c^{-1}(i)$  是匹配或者空集.

**注** 若令  $E_i = c^{-1}(i) = \{e \in E(G) | c(e) = i\}$ , ( $i = 1, 2, \dots, k$ ), 则  $G$  的一个边正常  $k$  染色可看成是边集的一种划分  $c = (E_1, E_2, \dots, E_k)$ , 其中每个  $E_i$  是匹配或空集.

例如, 下面给出了 5-圈的一种边正常 3 染色和 Petersen 图的一种边正常 4 染色.



**定义 6.1.2** 若存在  $G$  的一种边正常  $k$  染色, 则称  $G$  是边  $k$  色可染的 (edge  $k$ -colourable).

注 (1) 每个无环非空图的边必  $\varepsilon$  色可染.

(2) 若  $G$  是边  $k$  色可染的, 则对  $\forall l \geq k, G$  也是  $l$  色可染的.

定义 6.1.3 正整数  $\chi'(G) = \min\{k | G \text{ 是边 } k \text{ 色可染的}\}$  称为  $G$  的边色数 (edge chromatic number).

注 (1) 若  $\chi'(G) = k$ , 则  $G$  中边的任何  $k$  染色  $c = (E_1, E_2, \dots, E_k)$  中每个  $E_i$  都是非空的匹配.

(2) 若将  $G$  的边集划分为尽可能少的若干边不重的匹配, 则所能划分出的匹配的最少个数恰等于  $\chi'(G)$ .

(3)  $\chi'(K_{2n}) = 2n - 1 = \Delta(K_{2n})$  (因完全图  $K_{2n}$  的边可划分为  $2n - 1$  个边不重的匹配).

(4)  $\chi'(G) \geq \Delta(G)$ . (设  $d(v) = \Delta(G)$ , 则与  $v$  关联的  $\Delta(G)$  条边至少需  $\Delta(G)$  种色才能正常染色).

设  $v$  是图  $G$  的一个顶点, 用一些颜色对  $G$  的边进行染色 (未必是正常染色) 后, 如果  $v$  关联的某条边被染了颜色  $i$ , 则称色  $i$  在顶点  $v$  处出现.

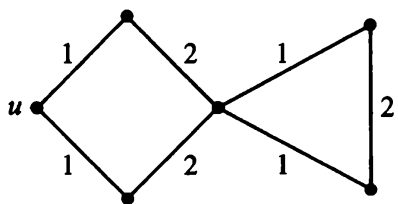
引理 6.1.1 设  $G$  是无环边的非空连通图, 且不是奇圈, 则存在  $G$  的边 2-染色, 使得所用的两种色在每个度  $\geq 2$  的顶点处都出现.

证明 (1) 设  $G$  是 Euler 图, 则  $G$  中无奇度点.

若  $G$  本身是一个偶长度圈, 则命题显然. 若  $G$  不是一个偶长度圈, 则  $G$  至少有一个顶点  $v_0$  满足  $d(v_0) \geq 4$  (否则,  $G$  中所有顶点都是 2 度的, 由于  $G$  连通, 从而  $G$  是圈, 由引理条件,  $G$  不是奇圈, 故为偶圈, 矛盾).

设  $v_0 e_1 v_1 e_2 \dots e_\varepsilon v_0$  是  $G$  的一条 Euler 闭迹. 令  $E_1 = \{e_i | i \text{ 为奇数}\}$ ,  $E_2 = \{e_i | i \text{ 为偶数}\}$ . 于是  $c = (E_1, E_2)$  即为所求的边 2-染色.

需要说明的是, Euler 闭迹从度  $\geq 4$  的顶点出发是必需的. 例如在下图中, 若从 2 度顶点  $u$  处出发沿 Euler 闭迹交替地对边进行 2 染色, 则  $u$  点可能仅能获得一种色 (如图, 1、2 表示两种颜色).



(2) 设  $G$  不是 Euler 图. 此时给  $G$  增添一个新顶点  $v_0$ , 将  $v_0$  与  $G$  的每个奇度顶点连一条边, 得到一个新图  $G^*$ . 显然  $G^*$  的所有顶点都是偶数度的, 因而是 Euler 图. 设  $v_0 e_1 v_1 e_2 \dots e_\varepsilon(G^*) v_0$  是  $G^*$  的一个 Euler 闭迹, 令  $E_1^* = \{e_i | i \text{ 为奇数}\}$ ,  $E_2^* = \{e_i | i \text{ 为偶数}\}$ , 这样可得  $G^*$  的一个边 2-染色  $c^* = (E_1^*, E_2^*)$ , (当  $d_{G^*}(v_0) = 2$  时,  $v_0$  点



的关联边有可能染同一种色). 按这种办法给  $G^*$  的边染色后, 去掉  $v_0$  及其关联的边, 便得到  $G$  的一个边 2-染色. 对于  $G$  中偶度点, 它关联的边及其颜色与  $G^*$  中相同; 对  $G$  的任何奇度点  $v$ , 在  $G$  中比在  $G^*$  中少关联一条边, 但只要  $d_G(v) > 1$ , 便有  $d_G(v) \geq 3$ , 故由染色的方法知, 与  $v$  点关联的边中两种颜色的都有. 这说明  $G$  的边 2-染色  $c = (E_1^* \cap E(G), E_2^* \cap E(G))$  即为所求的边 2-染色. 证毕.

**定义 6.1.4** 对于  $G$  的一个边  $k$ -染色  $c$ ,  $c(v)$  表示顶点  $v$  处出现的不同颜色的数目. 设  $c$  与  $c'$  都是  $G$  的边  $k$ -染色 (未必是正常染色). 若相应的  $c(v)$  与  $c'(v)$  满足:

$$\sum_{v \in V(G)} c'(v) > \sum_{v \in V(G)} c(v),$$

则称  $c'$  是对  $c$  的一个改进. 不能改进的边  $k$  染色称为最佳边  $k$  染色.

**引理 6.1.2** 设  $c = (E_1, E_2, \dots, E_k)$  是  $G$  的一个最佳边  $k$  染色, 且存在一个顶点  $u$  及两种颜色  $i$  和  $j$ , 色  $i$  不在  $u$  处出现, 而色  $j$  在  $u$  处出现了至少两次, 则  $G[E_i \cup E_j]$  中含  $u$  的连通分支必是奇圈.

**证明** 设  $G_1$  是  $G[E_i \cup E_j]$  中含  $u$  的连通分支. 若  $G_1$  不是奇圈, 则由引理 6.1.1,  $G_1$  有一个边 2-染色, 其两种色在  $G_1$  中度  $\geq 2$  的每个顶点处都出现. 按这种染色办法用色  $i$  和  $j$  给  $E_i \cup E_j$  中的边重新染色后, 得到  $G$  的一个新的边  $k$ -染色  $c' = (E'_1, E'_2, \dots, E'_k)$ , 其中  $i, j$  两色都在  $u$  处出现, 故  $c'(u) = c(u) + 1$ , 而对  $u$  之外的其他顶点  $v$ , 都有  $c'(v) \geq c(v)$ . 于是  $\sum_{v \in V(G)} c'(v) > \sum_{v \in V(G)} c(v)$ . 这与  $c$  是最佳边  $k$ -染色矛盾. 证毕.

**定理 6.1.1 (König<sup>[1]</sup>, 1916)** 设  $G$  是二部图, 则  $\chi'(G) = \Delta(G)$ .

**证明** (反证法): 假设  $\chi'(G) \neq \Delta(G)$ . 则由定义 6.1.3 的注 (4),  $\chi'(G) > \Delta(G)$ . 设  $c = (E_1, E_2, \dots, E_\Delta)$  是  $G$  的一个最佳边  $\Delta$  染色, 则  $c$  必定不是正常染色. 故存在顶点  $u$  满足  $c(u) < d(u)$ , 因而必有某两条在  $u$  点相邻的边染了同一种色. 而且, 因  $d(u) \leq \Delta(G)$ , 故  $\Delta$  种色中必有某种色不在  $u$  上出现. 显然  $u$  满足引理 6.1.2 的条件, 因此  $G$  中有奇圈. 这与  $G$  是二部图矛盾. 证毕.

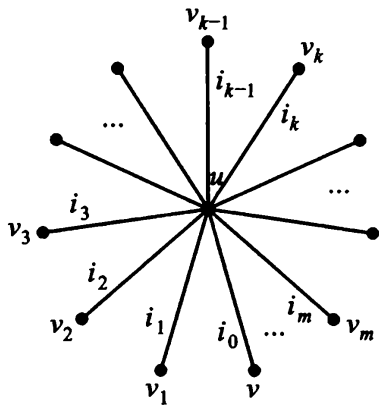
**注** 定理 6.1.1 也可利用推论 3.3.3 来证明: 因二部图的边集可分解为  $\Delta(G)$  个边不重的匹配, 故  $\chi'(G) = \Delta(G)$ .

**定理 6.1.2 (Vizing 定理, 1964)** 设  $G$  是无环边的非空简单图, 则

$$\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1.$$

**证明** 显然  $\chi'(G) \geq \Delta(G)$  (定义 6.1.3 的注 (4)). 下面用反证法证明  $\chi'(G) \leq \Delta(G) + 1$ .

假如  $\chi'(G) > \Delta(G) + 1$ , 令  $c = (E_1, E_2, \dots, E_{\Delta+1})$  是  $G$  的最佳  $\Delta + 1$  边染色. 因  $\chi' > \Delta + 1$ , 故  $c$  必不是正常染色. 设  $u$  是使  $c(u) < d(u)$  的顶点. 则必存在色  $i_1$ ,  $i_1$  至少在  $u$  处出现两次, 同时也存在色  $i_0$ ,  $i_0$  不在  $u$  处出现 (因  $d(u) < \Delta + 1$ ). 设  $uv$  和  $uv_1$  染有颜色  $i_1$  (如图).

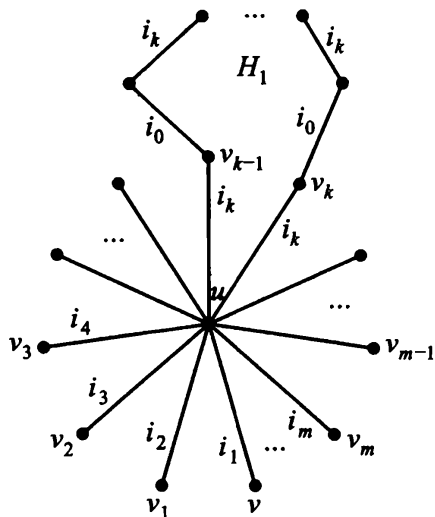


因  $d(v_1) < \Delta + 1$ , 故必有某种色  $i_2$  不在  $v_1$  处出现. 这样  $i_2$  必然在  $u$  处出现 (否则, 可用  $i_2$  给  $uv_1$  重新染色, 得到一个改进的  $\Delta + 1$  边染色, 与  $c$  是最佳染色矛盾). 因此存在一条边  $uv_2$  染有色  $i_2$ .

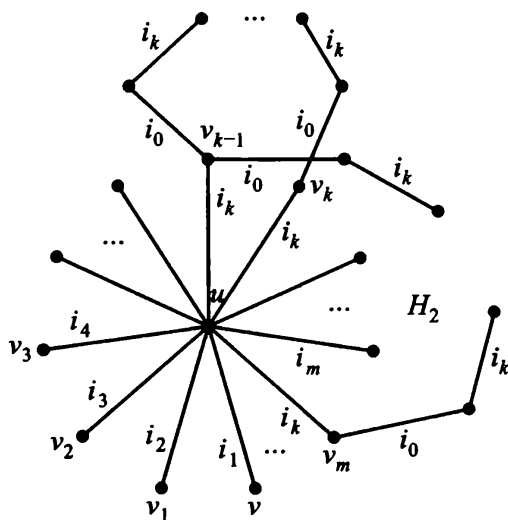
又因  $d(v_2) < \Delta + 1$ , 必有某种色  $i_3$  不在  $v_2$  处出现.  $i_3$  必然在  $u$  处出现 (理由同上). 因此存在一边  $uv_3$  染有色  $i_3$ .

继续这个过程, 可找出一个顶点序列  $v_1, v_2, \dots$ , 以及一个颜色序列  $i_1, i_2, \dots$ , 使得边  $uv_j$  染有颜色  $i_j$ , 且色  $i_{j+1}$  不在点  $v_j$  处出现, ( $j = 1, 2, \dots$ ). 而且, 因  $d(u)$  是有限数, 故存在一个最小整数  $m$ , 使得对某个  $k < m$ , 有  $i_{m+1} = i_k$ .

现在, 对  $1 \leq j \leq k-1$ , 用颜色  $i_{j+1}$  给边  $uv_j$  重新染色. 这样产生一个新的  $(\Delta + 1)$  边染色  $C' = (E'_1, E'_2, \dots, E'_{\Delta+1})$ . 显然对所有  $v \in V$ ,  $c'(v) \geq c(v)$ . 因此  $c'$  也是  $G$  的一个最佳  $(\Delta + 1)$  边染色. 由引理 6.1.2,  $G[E'_{i_0} \cup E'_{i_k}]$  中含有  $u$  的那个分支  $H_1$  是个奇圈.



而对  $k \leq j \leq m-1$ , 用颜色  $i_{j+1}$  给  $uv_j$  重新染色, 而用颜色  $i_k$  给  $uv_m$  重新染色, 又得到一个  $(\Delta+1)$  边染色  $c'' = (E''_1, E''_2, \dots, E''_{\Delta+1})$ . 同理,  $c''(v) \geq c(v)$  对所有  $v \in V$  成立. 故由引理 6.1.2,  $G[E''_{i_0} \cup E''_{i_k}]$  中含有  $u$  的分支  $H_2$  是个奇圈.



但是, 因  $v_k$  在  $H_1$  中的度为 2 (恰与一条  $i_0$  色边和一条  $i_k$  色边相关联), 故它在  $H_2$  中的度为 1 (仅与一条  $i_0$  色边相关联). 这与  $H_2$  是奇圈矛盾. [注意  $v_k$  必在分支  $H_2$  中, 因它与  $v_{k-1}$  有  $i_0, i_k$  交错路 ( $H_1$  的一段) 相连]. 由此可知反证法假设不能成立. 证毕.

对于有重边的图  $G$ , 设  $\mu(G)$  表示  $G$  中边的最大重数, Vizing 实际上证明了一个更一般的结论:  $\Delta(G) \leq \chi'(G) \leq \Delta(G) + \mu(G)$ .

Vizing 定理提出了一个分类问题: 使  $\chi'(G) = \Delta(G)$  的简单图称为**第一类图**, 使  $\chi'(G) = \Delta(G) + 1$  的简单图  $G$  称为**第二类图**. 确定一个图属于第一类还是第二类是很困难的, 目前仅对少量的图类判明了它们所属的类. 路、树、二部图、偶数阶完全图、轮图都是第一类图 (轮图  $W_{1,n}$  是由一个点与长为  $n$  的圈上  $n$  个顶点相连形成的图); 奇圈、奇数阶完全图都是第二类图 (见有关定理和习题). 一般情况下判别一个图属于第几类图, 尚没有好的充分必要判别条件, 一些充分性判别条件见习题.

第二类图相对较稀少. 在  $\nu \leq 6$  的 143 个连通简单图中仅有 8 个属于第二类; 而 Erdős 和 Wilson (1977) 已证明:  $\lim_{\nu \rightarrow \infty} \frac{|c_1(\nu)|}{|c_1(\nu) \cup c_2(\nu)|} = 1$ , 其中  $c_i(\nu)$  表示  $\nu$  阶第  $i$  类图的集合. 这表明当顶点数充分大时, 几乎所有非空简单图都是第一类图.

已经知道存在最大度为 2, 3, 4, 5 的第二类平面图, 但 Vizing (1965) 已证明: 不存在最大度  $\geq 8$  的第二类平面图. 目前尚不知道是否存在最大度为 6 或 7 的第二类平面图.

由 Vizing 定理, 每个无环边的非空简单图  $G$  都是可  $(\Delta+1)$  边可染色的. 实际上, 可以根据 Vizing 定理的上述证明方法给出求图  $G$  的正常  $(\Delta+1)$  边染色的多项式时间算法. 但是, 确定一般图  $G$  的边色数  $\chi'(G)$  及其相应的正常边染色尚无多项式时间

算法. 事实上, 已经证明这是一个 NPC 问题<sup>[2][3]</sup>. 求二部图的边色数的一个多项式时间算法将在 §6.5 中介绍. 有关边染色的更多内容可参看文献 [4]~[8], 其中 [4] 是边染色问题的专著.

## §6.2 点染色

### 一、点染色的基本概念

**定义 6.2.1** 设  $G$  是一个无环边的图.  $G$  的顶点正常  $k$  染色 (proper vertex  $k$ -colouring)  $\pi$  是指  $k$  种颜色  $1, 2, \dots, k$  对于  $G$  的各顶点的一种分配, 使得任二相邻的顶点被染上不同的颜色. 换句话说,  $G$  的顶点正常  $k$  染色是一个映射

$$\pi: V(G) \rightarrow \{1, 2, \dots, k\},$$

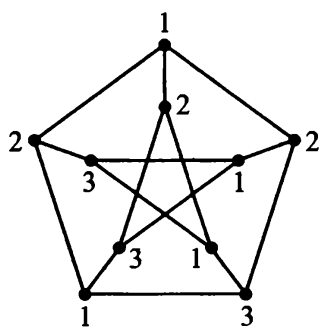
使得  $\pi^{-1}(i)$  是独立集或空集 ( $i = 1, 2, \dots, k$ ).

**注** 设  $\pi$  是  $G$  的一个顶点正常  $k$  染色. 令

$$V_i = \pi^{-1}(i) = \{x \in V(G) | \pi(x) = i\}, \quad (i = 1, 2, \dots, k).$$

则  $\pi$  实际上是对顶点集  $V(G)$  的一种划分:  $\pi = (V_1, V_2, \dots, V_k)$ , 其中  $V_i \cap V_j = \phi$ ,  $\bigcup_{i=1}^k V_i = V(G)$ , 且每个  $V_i$  是独立集或空集 ( $i = 1, 2, \dots, k$ ).

例如, 下图给出了 Peterson 图的一种顶点正常 3 染色.



**定义 6.2.2** 若存在  $G$  的一种顶点正常  $k$  染色, 则称图  $G$  是点  $k$  色可染的 (vertex  $k$ -colourable), 有时简称为  $k$  色可染的 或 可  $k$  染色的.

**注** (1) 每个图  $G$  一定是  $v(G)$  色可染的.

(2) 若图  $G$  是  $k$  色可染的, 则对任何正整数  $m \geq k$ ,  $G$  也  $m$  色可染.

**定义 6.2.3** 设  $G$  是无环边的图, 令

$$\chi(G) = \min\{k | G \text{ 是 } k \text{ 色可染的}\},$$

称  $\chi(G)$  为  $G$  的点色数, 有时简称为色数 (chromatic number). 若  $\chi(G) = k$ , 则称  $G$  为  $k$  色图 ( $k$ -chromatic graph). 特别地, 零图 ( $V(G) = \phi$ ) 的色数定义为 0.

注 (1) 若  $\chi(G) = k$  (即  $G$  是  $k$  色图), 则  $G$  中任何点  $k$  染色  $\pi = (V_1, V_2, \dots, V_k)$  中每个  $V_i$  都是非空的独立集. 换言之,  $G$  的色数是将  $V(G)$  划分为尽可能少的彼此无公共点的非空独立集时, 所得独立集的最小数目.

(2) 由色数的定义容易证明如下结论 (习题 6.21):

- $\chi(G) = 1$  当且仅当  $G$  是空图但非零图, 即  $G = \overline{K}_\nu$  且  $\nu \geq 1$ .
- $\chi(G) = 2$  当且仅当  $G$  是非空二部图.
- $\chi(G) = \nu(G)$  当且仅当  $G \supseteq K_\nu$ .
- $\chi(C_{2n+1}) = 3$ .
- $\chi(G) \geq 3$  当且仅当  $G$  含有奇圈.

点染色理论的基本问题: 给定图  $G$ , 确定  $\chi(G)$  的值.

目前, 人们仅对某些特殊图类, 确定出了计算  $\chi(G)$  的公式. 对一般图  $G$ , 得出了  $\chi(G)$  的各种上下界. 为了介绍有关结果, 需要引入色临界图的概念.

## 二、色临界图

定义 6.2.4 设  $\chi(G) = k, (k \geq 1)$ . 若对  $G$  的任何真子图  $H$ , 均有  $\chi(H) < k$ , 则称  $G$  是临界  $k$  色图 (critical  $k$ -chromatic graph). 临界  $k$  色图 ( $k = 1, 2, \dots$ ) 统称为色临界图.

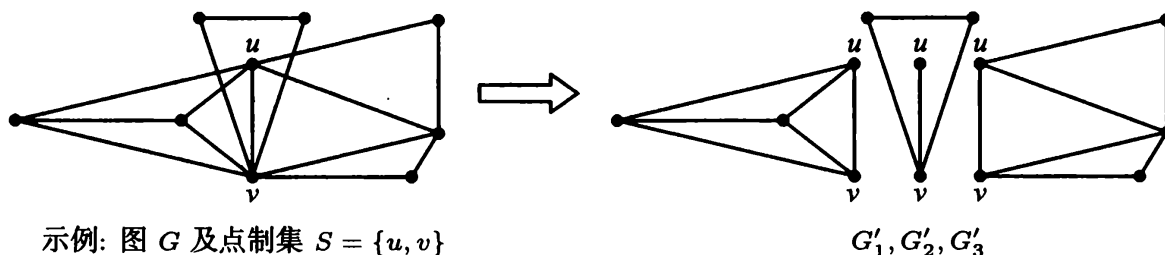
注 临界  $k$  色图实际上就是  $k$  色极小子图, 也就是说, 它本身是  $k$  色的, 但任意删去一个顶点或一条边后色数就会减少. 按上述色临界图的概念容易知道下列结论成立 (习题 6.22).

- $G$  是临界 1 色图当且仅当  $G$  是  $K_1$ .
- $G$  是临界 2 色图当且仅当  $G$  是  $K_2$ .
- $G$  是临界 3 色图当且仅当  $G$  是奇圈.

此外, 容易证明: (1) 任何  $k$  色图都包含临界  $k$  色子图; (2) 每个色临界图都是连通的简单图. (习题 6.22).

定理 6.2.1 色临界图的顶点割集不是团.

证明 (反证法) 假设图  $G$  是一个临界  $k$  色图, 但有一个点割集  $S$  是团. 记  $G \setminus S$  的连通分支为  $G_1, G_2, \dots, G_n$ . 将  $G[S]$  按  $G$  中的连接方式分别与  $G_1, G_2, \dots, G_n$  相连, 得到子图  $G'_1, G'_2, \dots, G'_n$ .



示例: 图  $G$  及点割集  $S = \{u, v\}$

$G'_1, G'_2, G'_3$

因  $G$  是临界  $k$  色的, 故每个  $G'_i$  是  $k-1$  色可染的. 由于  $S$  是团, 所以  $S$  中各个顶点在  $G'_i$  的任何  $k-1$  正常点染色中必染到相异的色. 我们总可以调整各  $G'_i$  中颜色的编号, 使得  $S$  中每一个顶点在各  $G'_i$  中染相同的色. 这些  $G'_i$  的染色合在一起便形成  $G$  的一个  $k-1$  染色. 这与  $G$  是临界  $k$  色图矛盾. 证毕.

**推论 6.2.1** 每个色临界图都是块 (即不含割点).

**证明** 假如某临界图不是块, 则它有割点. 这个割点构成一个团, 与定理 6.2.1 矛盾. 证毕.

由这个推论知, 顶点数至少为 3 的色临界图必定 2 连通.

下一个推论是显然的.

**推论 6.2.2** 若临界  $k$  色图  $G$  有 2 顶点割集  $\{u, v\}$ , 则  $u$  与  $v$  不相邻.

**定理 6.2.2 (Dirac, 1952)** 设  $G$  是临界  $k$  色图 ( $k \geq 2$ ), 则边连通度  $\kappa'(G) \geq k-1$ .

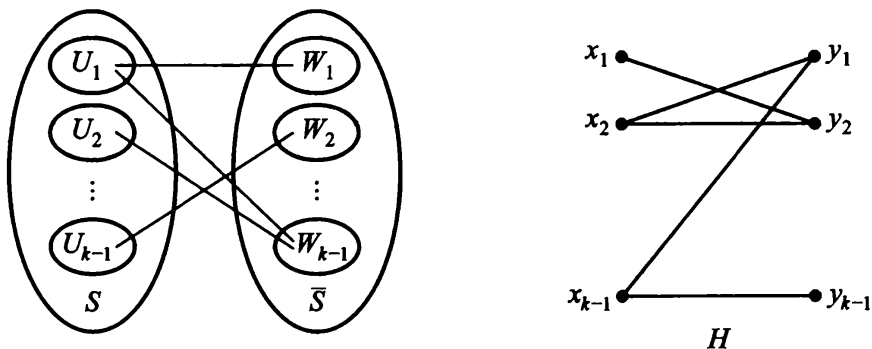
**证明** 若  $k=2$ , 则  $G$  是  $K_2$ , 故  $\kappa'(G) = 1$ .

下设  $k \geq 3$ , 用反证法.

假如  $\kappa'(G) < k-1$ , 则存在  $S \subset V(G)$  使得  $|(S, \bar{S})| = \kappa'(G) < k-1$ . 因  $G$  是临界  $k$  色图, 故  $G_1 = G[S]$  与  $G_2 = G[\bar{S}]$  中的顶点都  $k-1$  色可染. 设  $\pi_1 = (U_1, U_2, \dots, U_{k-1})$  和  $\pi_2 = (W_1, W_2, \dots, W_{k-1})$  分别是  $G_1$  和  $G_2$  中点的  $k-1$  染色. 构造二部图

$$H = (X, Y), \quad X = \{x_1, x_2, \dots, x_{k-1}\}, \quad Y = \{y_1, y_2, \dots, y_{k-1}\},$$

其中  $x_i y_j \in E(H)$  当且仅当  $U_i$  的点与  $W_j$  的点在  $G$  中无连边.



因  $|(S, \bar{S})| = \kappa'(G) < k-1$ , 故  $\varepsilon(H) > (k-1)^2 - (k-1) = (k-1)(k-2)$ . (注意  $X$  与  $Y$  间共可连出  $(k-1)^2$  条可能的边, 但因在  $G$  中  $S$  与  $\bar{S}$  间连边数  $|(S, \bar{S})| < k-1$ ,

故  $X$  与  $Y$  间在  $H$  中不能连的边数  $< k - 1$ ).

我们来证明  $H$  必有完美匹配.

事实上,  $H$  的点覆盖数  $\beta(H) \geq k - 1$  (否则,  $\beta(H) \leq k - 2$ . 设  $\{v_1, v_2, \dots, v_\beta\}$  是  $H$  的一个最小点覆盖集, 则因  $\Delta(H) \leq k - 1$ , 故

$$\varepsilon(H) \leq d_H(v_1) + d_H(v_2) + \dots + d_H(v_\beta) \leq \beta(H) \cdot (k - 1) \leq (k - 2)(k - 1),$$

与  $\varepsilon(H) > (k - 1)(k - 2)$  矛盾. 由 König 定理 (第五章定理 5.2.2),  $\alpha'(H) = \beta(H) \geq k - 1$ . 这表明  $H$  有完美匹配 (因对于  $H$ ,  $X = Y = k - 1$ ).

设  $H$  的一个完美匹配为  $M^*$ :  $M^* = \{x_i y_{j_i} \mid i = 1, 2, \dots, k - 1\}$ , 相应地令  $V_i = U_i \cup W_{j_i}$ , 则  $V_i$  是  $G$  的独立集 ( $i = 1, 2, \dots, k - 1$ ), 因此  $\pi = (V_1, V_2, \dots, V_{k-1})$  是  $G$  的一个点  $k - 1$  染色. 但这与  $\chi(G) = k$  矛盾. 故  $\kappa'(G) \geq k - 1$ . 证毕.

**推论 6.2.3** 设  $G$  是临界  $k$ -色图, 则  $\delta(G) \geq k - 1$ .

**证明** 由定理  $\delta(G) \geq \kappa'(G) \geq \kappa(G)$  及定理 6.2.2, 有  $\delta(G) \geq \kappa'(G) \geq k - 1$ . 证毕.

**推论 6.2.4** 任何  $k$  色图至少有  $k$  个顶点的度  $\geq k - 1$ .

**证明** 设  $G$  是  $k$  色图,  $H$  是其一个临界  $k$  色子图, 由推论 6.2.3,  $H$  的每个顶点在  $H$  中的度  $\geq k - 1$ , 故在  $G$  中的度也  $\geq k - 1$ . 由于  $H$  是  $k$  色临界的, 因此它至少有  $k$  个顶点. 证毕.

### 三、色数的上下界

**定理 6.2.3** 对任何简单图  $G$ ,  $\chi(G) \leq \Delta(G) + 1$ .

**证明** 设  $\chi(G) = k$ , 且  $H$  是  $G$  的临界  $k$  色子图, 由推论 6.2.3,  $\delta(H) \geq k - 1$ . 故  $\Delta(G) \geq \Delta(H) \geq \delta(H) \geq k - 1 = \chi(G) - 1$ . 证毕.

定理 6.2.3 给出的上界是可以达到的, 例如奇圈和完全图的色数恰好等于它们的最大度加 1. 下面来证明, 达到这个上界的连通简单图只有这两类图.

**定理 6.2.4 (Brooks, 1941)** 设  $G$  是连通的简单图, 且  $G$  既不是奇圈又不是完全图, 则

$$\chi(G) \leq \Delta(G).$$

**证明** 设  $G$  是满足定理条件的  $k$  色图. 下面分两种情况考虑.

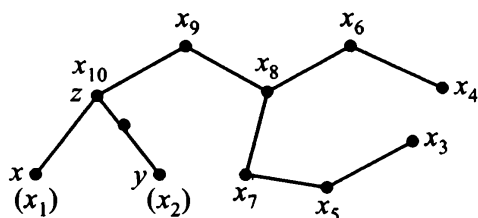
**情况 1**  $G$  本身是临界  $k$  色图.

由推论 6.2.1,  $G$  是一个块. 又由于临界 1 色图和临界 2 色图是完全图, 而临界 3 色图是奇圈, 故  $k \geq 4$ . 由推论 6.2.3,  $\delta(G) \geq 3$ .

如果  $G$  是 3 连通的, 则因  $G$  不是完全图, 故必存在顶点  $x, y, z \in V(G)$ , 使得  $xy \notin E(G)$ , 而  $xz, yz \in E(G)$  (习题 2.1), 并且  $G - \{x, y\}$  连通.

如果  $G$  的连通度为 2, 则  $G$  中存在一点  $z$ , 使得  $G - \{z\}$  连通且有割点, 此时  $G - \{z\}$  至少有两个连通块只含 1 个割点 (见第二章习题 2.33), 设  $B_1, B_2$  为这样两个块. 因  $G$  本身 2 连通 (无割点), 故  $B_1, B_2$  均有  $G$  的非割点在  $G$  中与  $z$  相邻. 设  $x \in B_1$  和  $y \in B_2$  是两个与  $z$  相邻的非割点. 由于  $x$  和  $y$  在不同的块中, 因此它们在  $G - \{z\}$  中不相邻, 因而在  $G$  中也不相邻, 且因  $d_G(z) \geq \delta(G) \geq 3, G - \{x, y\}$  连通.

总之, 必存在顶点  $x, y, z \in V(G)$ , 使得  $xy \notin E(G), xz, yz \in E(G)$  且  $G - \{x, y\}$  连通. 给  $G$  的顶点重新编号: 首先  $x_1 = x, x_2 = y$ , 然后对  $G' = G \setminus \{x, y\}$  中的顶点, 按  $G'$  中到  $z$  的距离由远及近的次序依次用  $x_3, x_4, \dots, x_\nu$  编号, 即:  $d_{G'}(x_i, z) \geq d_{G'}(x_{i+1}, z)$ , ( $i = 3, 4, \dots, \nu$ ), (注意  $G'$  仍连通).



因此  $x_\nu = z$ , 且对每个  $i = 1, 2, \dots, \nu - 1$ , 必存在  $j > i$  使得  $x_i x_j \in E(G)$ . 由此可知, 对每个  $i = 1, 2, \dots, \nu - 1, x_i$  与  $\{x_1, x_2, \dots, x_{i-1}\}$  中最多  $\Delta(G) - 1$  个顶点相邻 (因与  $x_i$  相邻的至少有一个顶点的下标  $> i$ ), 且因  $d(z) \geq \delta(G) \geq 3$ , 故  $x_{\nu-1} x_\nu \in E(G)$ .

这样一来, 可用  $\Delta(G)$  种颜色给顶点进行染色: 将  $x_1$  和  $x_2$  染色 1, 然后按颜色  $1, 2, \dots, \Delta$  的顺序依次给  $x_3, x_4, \dots, x_\nu$  进行染色, 使相邻两点染不同的色. 染法为:

设  $x_1, \dots, x_{i-1}$  已染好, 考虑  $x_i$ , ( $3 \leq i \leq \nu - 1$ ). 由于  $x_i$  仅与  $x_1, \dots, x_{i-1}$  中最多  $\Delta(G) - 1$  个顶点相邻, 所以  $1, 2, \dots, \Delta$  种颜色中至少有一种颜色  $\alpha$  在  $N(x_i) \cap \{x_1, x_2, \dots, x_{i-1}\}$  中未曾用过, 因此可给点  $x_i$  染以颜色  $\alpha$ . 最后, 因  $x_\nu (= z)$  既与  $x_1$  又与  $x_2$  相邻, 且  $x_1$  和  $x_2$  染的色相同, 所以  $\Delta$  种色中同样也有一种色  $\beta$  在  $N(x_\nu)$  中未曾用过, (因  $d(x_\nu) \leq \Delta$ , 而  $x_\nu$  的邻点中已有两点染同一种色, 故  $\Delta$  种色不会全在  $N(x_\nu)$  中出现). 因此可给  $x_\nu$  染以色  $\beta$ .

如此便得图  $G$  的一个正常  $\Delta$ -染色, 即有:  $\chi(G) \leq \Delta(G)$ .

**情况 2**  $G$  是  $k$  色非临界的. 此时取  $G$  的一个临界  $k$  色子图  $H$ .

(1) 若  $H$  是完全图, 则  $H = K_k$ , 从而  $\chi(G) = \chi(H) = k = (k-1) + 1 = \Delta(H) + 1 \leq \Delta(G)$  ( $G$  中有不属于完全子图  $H$  的边和点);

(2) 若  $H$  是一个奇圈, 则  $\chi(G) = \chi(H) = 3 \leq \Delta(G)$  ( $G$  中至少有一条不属于奇圈  $H$  的边).

(3) 若  $H$  既不是完全图, 也不是奇圈, 则由情况 1 的证明可知  $\chi(H) \leq \Delta(H) \leq \Delta(G)$ , 从而  $\chi(G) = k = \chi(H) \leq \Delta(G)$ . 证毕.

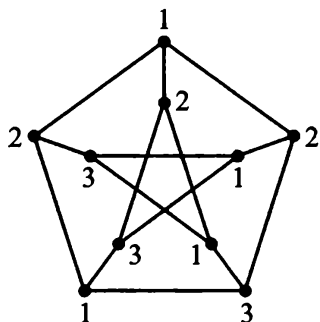


**例 6.2.1** 求 Peterson 图的色数.

**解** 因 Peterson 图  $G$  含有奇圈, 故不是二部图, 因而  $\chi(G) \geq 3$ ;

另一方面,  $G$  既不是奇圈又不是完全图, 且  $\Delta(G) = 3$ ,

故  $\chi(G) \leq \Delta(G) = 3$ . 因此,  $\chi(G) = 3$ .



虽然奇圈和完全图可以达到定理 6.2.3 提供的色数上界  $\Delta(G) + 1$ , 但某些图类的色数与这个上界相差很远, 比如星图  $K_{1,n}$  的色数为 2, 而  $\Delta(K_{1,n}) + 1 = n + 1$ . 下面给出的几个上界在某种程度上是对定理 6.2.3 的改进.

**定理 6.2.5** 设连通简单图  $G$  的度序列为  $d_1 \geq d_2 \geq \cdots \geq d_\nu$ , 则

$$\chi(G) \leq 1 + \max_i \min\{d_i, i - 1\}$$

**证明** 将  $G$  的所有顶点按度的递减 (不减) 顺序排序, 设为  $v_1, v_2, \cdots, v_\nu$ . 从  $v_1$  开始依次给顶点染色. 在第  $i$  次染色时, 用  $v_i$  的邻点尚未使用的颜色中色号最小的颜色给  $v_i$  染色 (事先对使用的颜色编号), ( $i = 1, 2, \cdots, \nu$ ). 由于对  $v_i$  染色时, 下标比  $i$  大的顶点尚未染色, 而下标小于  $i$  的顶点中与  $v_i$  相邻的顶点不超过  $\min\{d_i, i - 1\}$  个, 因此这些顶点已使用的颜色数也不会超过  $\min\{d_i, i - 1\}$ , 从而按染色规则, 对  $v_i$  染色使用的颜色编号不会超过  $\min\{d_i, i - 1\} + 1$ . 这个结论对所有  $i = 1, 2, \cdots, \nu$  都成立. 因此将  $G$  的所有顶点全部染色使用的颜色不会超过  $\max_i \min\{d_i, i - 1\} + 1$  种. 证毕.

**定理 6.2.6** 对任何图  $G, \chi(G) \leq 1 + \max_{H \subseteq G} \delta(H)$ .

**证明** 对于空图, 结论显然成立. 因此可设  $\chi(G) = k \geq 2$ .

取  $G$  的一个临界  $k$  色子图  $H_0$ , 则  $\delta(H_0) \leq \max_{H \subseteq G} \delta(H)$ . 由推论 6.2.3,

$$\chi(G) = \chi(H_0) = k \leq 1 + \delta(H_0) \leq 1 + \max_{H \subseteq G} \delta(H). \quad \text{证毕.}$$

**定理 6.2.7** 用  $l(G)$  表示图  $G$  中最长路的长度, 则  $\chi(G) \leq 1 + l(G)$ .

**证明** 结论对于空图显然成立. 因此可设  $\chi(G) = k \geq 2$ .

取  $G$  的一个临界  $k$  色子图  $H$ , 由推论 6.2.3, 则  $\delta(H) \geq k - 1$ . 用最短路方法不难证明,  $H$  中有长为  $\delta(H)$  的路. 因此  $l(G) \geq \delta(H) \geq k - 1$ . 从而  $\chi(G) = k \leq 1 + l(G)$ . 证毕.

关于色数的下界有如下定理.

**定理 6.2.8** 对任何图  $G$ ,  $\chi(G) \geq \frac{\nu^2}{\nu^2 - 2\varepsilon}$

**证明** 设  $G$  的色数是  $k$ , 则有  $V(G)$  的一种划分:  $\pi = (V_1, V_2, \dots, V_k)$ , 其中  $V_i \cap V_j = \emptyset$ ,  $\bigcup_{i=1}^k V_i = V(G)$ , 且每个  $V_i$  是非空的独立集 ( $i = 1, 2, \dots, k$ ). 按分组  $V_1, V_2, \dots, V_k$  的次序将  $G$  的所有顶点依次编号后, 其邻接矩阵可写为如下分块形式

$$A(G) = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ A_{21} & A_{22} & \cdots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1} & A_{k2} & \cdots & A_{kk} \end{pmatrix}$$

其中主对角线上每个  $A_{ii}$  都是零矩阵.

设  $|V_i| = p_i$ , 则矩阵  $A(G)$  中至少有  $\sum_{i=1}^k p_i^2$  个零元素. 另一方面,  $A(G)$  共有  $\nu^2$  个元素但  $G$  只有  $\varepsilon$  条边. 由邻接矩阵的对称性,  $\varepsilon$  条边在  $A(G)$  中形成  $2\varepsilon$  个非零元素, 故  $A(G)$  中零元素的个数应为  $\nu^2 - 2\varepsilon$  个. 由以上两方面, 有  $\nu^2 - 2\varepsilon \geq \sum_{i=1}^k p_i^2$ . 利用

柯西不等式  $\left(\sum_{i=1}^k a_i^2\right) \left(\sum_{i=1}^k b_i^2\right) \geq \left(\sum_{i=1}^k a_i b_i\right)^2$ , 有  $k \cdot \sum_{i=1}^k p_i^2 \geq \left(\sum_{i=1}^k p_i\right)^2 = \nu^2$ . 从而  $\nu^2 - 2\varepsilon \geq \frac{\nu^2}{k}$ . 由此解得  $k \geq \frac{\nu^2}{\nu^2 - 2\varepsilon}$ , 即  $\chi(G) \geq \frac{\nu^2}{\nu^2 - 2\varepsilon}$ . 证毕.

图的色数与独立集和团有关. 利用一个图的独立数和团数, 可以获得其色数的上下界.

**定理 6.2.9** 对任何图  $G$ , 有

- (1)  $\chi(G) + \alpha(G) \leq \nu + 1$ ;
- (2)  $\chi(G) \cdot \alpha(G) \geq \nu$ . ( $\alpha(G)$  是  $G$  的点独立数)

**证明** (1) 设  $I$  是  $G$  的一个最大点独立集, 则  $|I| = \alpha(G)$ . 给  $I$  中的点染上一种颜色,  $G$  中其余  $\nu - \alpha(G)$  个顶点每点染一种不同的颜色, 这样得到  $G$  的一个  $\nu - \alpha(G) + 1$  种颜色的正常染色, 因此  $\chi(G) \leq \nu - \alpha(G) + 1$ .

(2) 设  $\chi(G) = k$ . 按定义,  $V(G)$  可划分为  $k$  个非空独立集  $V_1, V_2, \dots, V_k$ , 则

$$\alpha(G) \geq |V_i|, \quad i = 1, 2, \dots, k.$$

从而  $k\alpha(G) \geq \sum_{i=1}^k |V_i| = \nu$ , 即  $\chi(G) \cdot \alpha(G) \geq \nu$ . 证毕.

**定理 6.2.10** 对任何图  $G$ , 有  $\chi(G) \geq \omega(G)$ ,  $\omega(G)$  是  $G$  的团数.

**证明** 设  $W$  是  $G$  的一个最大团, 因  $W$  中所有点都在  $G$  中彼此相邻, 因此对  $G$  染色时  $W$  的每个点需要一种不同的颜色, 故  $\chi(G) \geq \omega(G)$ . 证毕.

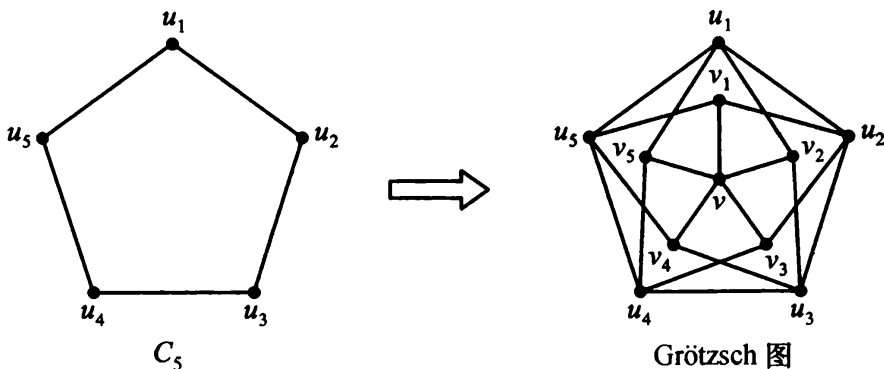
Reed 在 1998 年提出一个猜想<sup>[9]</sup>: 图的色数的上界  $\Delta(G) + 1$  和下界  $\omega(G)$  的平均值也是色数的一个上界, 即  $\chi(G) \leq \left\lceil \frac{\Delta(G) + 1 + \omega(G)}{2} \right\rceil$ . 这个猜想至今尚未得到证实.

Borodin 和 Kostochka 曾猜想<sup>[10]</sup>: 当  $\Delta(G) \geq 9$  时, 如果  $\omega(G) < \Delta(G)$ , 则  $\chi(G) < \Delta(G)$ . 1999 年 Reed 证明了在  $\Delta(G) \geq 10^{14}$  时, 上述结论是正确的<sup>[11]</sup>.

按照定理 6.2.10, 一个图有大的团则必有大的色数. 直觉上看, 反过来也应该成立, 即  $G$  的色数较大则它应该也含有大的团, 或者说团数  $\omega(G)$  较小时色数  $\chi(G)$  也会较小. 已被部分证实的上述 Borodin-Kostochka 猜想似乎也支持这一说法. 但令人惊奇的是, 事实并非如此. 下一个定理表明, 存在  $\omega(G) = 2$  但  $\chi(G)$  可以任意大的图.

**定理 6.2.11**<sup>[12]</sup> 对任意的正整数  $k$ , 存在不含三角形的  $k$  色图.

**证明** 对  $k = 1, 2$ , 定理的结论显然成立. 下设  $k \geq 3$ . 一个不含三角形的 3 色图是长为 5 的圈  $C_5$ . 下面我们从  $C_5$  出发递推地由某个不含三角形的  $k$  色图  $G_k$  构造不含三角形的  $k+1$  色图  $G_{k+1}$ . 构造方法如下: 设  $G_k$  的顶点集为  $\{u_1, u_2, \dots, u_n\}$ , 给  $G_k$  添加  $n+1$  个新顶点  $v_1, v_2, \dots, v_n, v$ , 每个  $v_i$  与  $u_i$  在  $G_k$  中的所有邻点连边, 同时也与  $v$  连边,  $i = 1, 2, \dots, n$ . 例如由  $C_5$  按上述方法构造出的图如下, 该图称为 Grötzsch 图. 容易检验它是一个无三角形的 4 色图 (习题 6.24). 下面证明按这种方法构造出的  $G_{k+1}$  都是无三角形的  $k+1$  色图.



令  $V^* = \{v_1, v_2, \dots, v_n\}$ . 假如  $G_{k+1}$  中有三角形  $C_3$ , 则  $C_3$  中必含有  $V^*$  的点. 另一方面, 因  $V^*$  是独立集, 故  $C_3$  中至多含  $V^*$  中的一个点, 由于  $v$  仅与  $V^*$  中点有连边, 因此不可能出现在  $C_3$  上. 无妨设  $C_3$  的顶点是  $u_i, u_j, v_t$ . 按  $G_{k+1}$  的构造方法, 与除  $v_t$  相邻的旧顶点一定与  $u_t$  相邻, 这表明  $u_i, u_j, u_t$  是  $G_k$  中的一个三角形. 这与  $G_k$  中不含三角形矛盾, 因此  $G_{k+1}$  中不可能出现三角形.

设  $\chi(G_k) = k$ , 我们来证明  $\chi(G_{k+1}) = k + 1$ .

首先在  $G_k$  的正常  $k$  染色基础上, 可得到  $G_{k+1}$  的正常  $k+1$  染色. 事实上, 由于在  $G_{k+1}$  中每个  $v_i$  与相应的  $u_i$  不相邻, 因此只要给  $v_i$  染  $u_i$  原有的颜色 ( $i = 1, 2, \dots, n$ ), 而给  $v$  染第  $k+1$  种颜色, 即可将  $G_k$  中的正常  $k$  染色扩充为  $G_{k+1}$  中的正常  $k+1$  染色. 由此可知  $\chi(G_{k+1}) \leq k + 1$ .

下面来证明  $\chi(G_{k+1}) \geq k + 1$ , 用反证法. 假定  $\chi(G_{k+1}) \leq k$ , 则因  $G_{k+1}$  的子图  $G_k$  的色数为  $k$ , 因此必有  $\chi(G_{k+1}) = k$ . 设  $\pi$  是  $G_{k+1}$  的一个正常  $k$  染色, 通过对颜色重新命名, 总可假定  $v$  染第  $k$  种色. 由于  $v$  与所有  $v_i$  相邻, 因此  $v_1, v_2, \dots, v_n$  的颜色全在  $\{1, 2, \dots, k-1\}$  中. 设在当前染色下, 顶点  $u_1, u_2, \dots, u_n$  中染第  $k$  种颜色的顶点的集合为  $V_k$ . 对  $\forall u_i \in V_k$ , 将其染色改为  $\pi(v_i)$ . 我们来证明经过这种改动后, 得到  $G_k$  的正常  $k-1$  染色.

事实上, (1) 由于在原来的染色下  $V_k$  中的顶点染相同的色, 因此  $V_k$  中的顶点互不相邻.

(2)  $V(G_k) - V_k$  中的所有顶点染色未作改变, 因此其中相邻的顶点保持不同的色.

(3) 对  $\forall u_i \in V_k$  和  $\forall u_j \in V(G_k) - V_k$ , 如果它们相邻, 则由  $G_{k+1}$  的构造知,  $u_i$  的相应顶点  $v_i$  也与  $u_j$  相邻, 因此在原有染色中  $u_j$  与  $v_i$  染有不同的色, 而染色改动后  $u_i$  的颜色正是  $v_i$  的颜色,  $u_j$  的颜色未变, 可见颜色改动后  $u_i$  与  $u_j$  异色. 由以上三种情况可知改动后的染色的确是  $G_k$  的正常  $k-1$  染色. 但这与  $G_k$  是  $k$  色图矛盾.

至此, 我们已证明了  $G_{k+1}$  是无三角形的  $k+1$  色图. 证毕.

该定理中由某个不含三角形的  $k$  色图  $G_k$  构造不含三角形的  $k+1$  色图  $G_{k+1}$  的构造方法称为 Mycielski 构造.

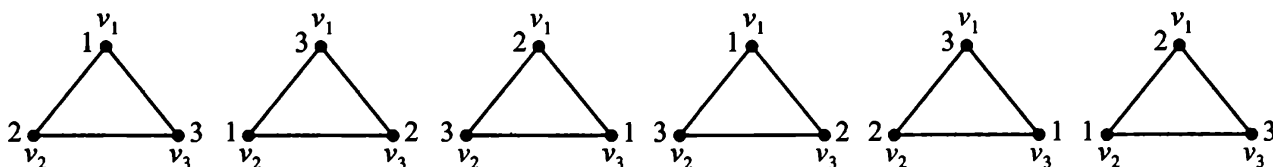
Hadwiger 曾提出一个猜想<sup>[13]</sup>: 若  $G$  为  $k$ -色图, 则  $G$  可收缩为一个包含  $K_k$  的图. Wagner 已证明  $k = 5$  时的 Hadwiger 猜想等价于著名的四色猜想<sup>[14]</sup>. Hajós 提出另一个有关的猜想<sup>[15]</sup>: 若  $G$  为  $k$ -色图, 则  $G$  包含完全图  $K_k$  的一个剖分. (收缩和剖分的概念见第七章 §7.3). 当  $k = 3, 4$  时 Hajós 猜想被证明是成立的 (见 [16][17]);  $k \geq 7$  时, 文献 [18] 举出反例否定了这一猜想, 文献 [19] 甚至证明 Hajós 猜想几乎对所有图都不成立. 但  $k = 5, 6$  时 Hajós 猜想是否成立, 仍然是一个悬而未决的问题. 另一方面, Ballobás 等人已经证明, Hadwiger 猜想对大多数图都成立<sup>[25]</sup>. 有关这两个猜想, 可进一步参看文献 [16]~[41], 其中 [16]~[23] 与 Hajós 猜想有关, [24]~[41] 与 Hadwiger 猜想有关.

关于点染色和边染色的更多内容,可参阅图论有关专著和网页 [42]~[59].

### §6.3 色多项式

本节要考虑的问题是: 对于给定的 (标定) 图  $G$ , 用  $k$  种颜色 ( $k \geq 1$ ) 对  $G$  进行正常顶点染色, 共有多少种不同的染色方式? 这里两种染色方式不同, 是指至少有一个顶点在这两种染色方式下染有不同的色.

例如, 用三种方式给  $K_3$  染色, 不同的染色方式有 6 种.



我们用  $P(G, k)$  表示使用  $k$  种色对图  $G$  的顶点正常染色的不同方式数. 对给定的图  $G$ ,  $P(G, k)$  是  $k$  的一个函数, 定义域为自然数集. 以下将会看到,  $P(G, k)$  是一个关于  $k$  的多项式. 这个多项式称为图  $G$  的色多项式(chromatic polynomial).

显然, 对图  $G$  添加或删除重边不会改变其色多项式.

- 定理 6.3.1** (1)  $P(G, k) > 0$  的充分必要条件是  $\chi(G) \leq k$ ;  
 (2)  $\varepsilon(G) = 0$  (即  $G$  是空图) 时,  $P(G, k) = k^\nu$ ;  
 (3) 若  $G = K_\nu$  (即  $G$  是  $\nu$  阶完全图), 则  $P(G, k) = k(k-1)\cdots(k-\nu+1)$ .

**证明** (1)  $P(G, k) > 0$  当且仅当  $G$  至少有一种  $k$  正常染色, 即  $\chi(G) \leq k$ .

(2)  $G$  是空图则  $G$  的每个顶点都可以用  $k$  种颜色中的任一种来染色, 故  $P(G, k) = k^\nu$ .

(3)  $G$  是完全图  $K_\nu$ , 则  $G$  中第一个顶点有  $k$  种色可选, 第二个顶点有  $k-1$  种色可选,  $\dots$ , 第  $\nu$  个顶点有  $k-\nu+1$  种色可选. 故  $P(G, k) = k(k-1)\cdots(k-\nu+1)$ . 证毕.

设  $e$  是图  $G$  的一条边, 我们用  $G \cdot e$  表示边的收缩, 即从  $G$  中删去  $e$  并将  $e$  的两个端点粘合在一起且删去重边后, 所得的图.

**定理 6.3.2** 设  $G$  是简单图, 对任意  $e \in E(G)$ , 有  $P(G, k) = P(G-e, k) - P(G \cdot e, k)$ .

**证明** 设  $e = uv$ , 考虑用  $k$  种颜色对  $G-e$  染色的方式.

- (1) 对  $G-e$  染色且使  $u$  与  $v$  染同色的方式数恰为  $P(G \cdot e, k)$ ;  
 (2) 对  $G-e$  染色且使  $u$  与  $v$  染异色的方式数恰为  $P(G, k)$ .

因此,  $P(G, k) = P(G-e, k) - P(G \cdot e, k)$ . 证毕.

利用定理 6.3.2 给出的公式, 可以通过减边或增边及边的收缩运算求得一些小阶图的色多项式.

**例 6.3.1** 求  $K_{1,3}$  和  $C_4$  的色多项式.

**解** 通过减边和边的收缩运算求  $K_{1,3}$  的色多项式, 过程如下所示:

$$\begin{aligned}
 \text{星形图} &= \text{V形图} - \text{V形图} = (\text{星形图} - \text{星形图}) - (\text{星形图} - \text{星形图}) = \dots \\
 &= k^4 - 3k^3 + 3k^2 - k.
 \end{aligned}$$

通过增边和边的收缩运算求  $C_4$  的色多项式, 过程如下所示;

$$\begin{aligned}
 \text{正方形} &= \text{正方形} + \text{带环的边} = (\text{正方形} + \text{带环的边}) + (\text{带环的边} + \text{带环的边}) = \dots \\
 &= k(k-1)(k-2)(k-3) + 2k(k-1)(k-2) + k(k-1) = k(k-1)(k^2 - 3k + 3).
 \end{aligned}$$

**定理 6.3.3** 设  $h_r(G)$  表示将  $G$  的顶点集划分为  $r$  个非空独立集的划分个数, 则

$$P(G, k) = \sum_{r=1}^{\nu} [h_r(G)k(k-1)(k-2)\cdots(k-r+1)].$$

**证明** 对每个正整数  $r$ , ( $1 \leq r \leq k$ ), 如果  $G$  不存在正常  $r$  点染色, 则  $h_r(G) = 0$ ; 如果  $G$  存在正常  $r$  点染色 (恰使用  $r$  种颜色), 则  $G$  的顶点集可划分为  $r$  个非空独立集  $V_1, V_2, \dots, V_r$ . 现用  $k$  种颜色给这些独立集染色 ( $k \geq r$ ), 使得每个独立集的顶点染一种色, 不同独立集的顶点染不同的色, 共有  $k(k-1)(k-2)\cdots(k-r+1)$  中染法. 每一种染法都给出  $G$  的一个正常  $k$  点染色. 由于将  $G$  的顶点集划分为  $r$  个非空独立集的划分有  $h_r(G)$  种方式, 因此对应于  $r$  个独立集划分的各种可能方式共能得到图  $G$  的  $h_r(G) \cdot k(k-1)(k-2)\cdots(k-r+1)$  种正常  $k$  点染色. 当  $r$  从 1 取至  $k$  时, 得

到  $\sum_{r=1}^k [h_r(G)k(k-1)(k-2)\cdots(k-r+1)]$  种正常  $k$  点染色, 这些点染色显然是不同的正常  $k$  点染色. 由于  $G$  的每种正常  $k$  点染色都可看作将  $G$  的顶点集划分为  $k$  个独立集 (可以有空集), 因此都在上述计数之内. 这说明  $G$  的正常  $k$  点染色的方式数恰为  $\sum_{r=1}^k [h_r(G)k(k-1)(k-2)\cdots(k-r+1)]$ , 即

$$P(G, k) = \sum_{r=1}^k [h_r(G)k(k-1)(k-2)\cdots(k-r+1)].$$

当  $k < \nu$  且  $r > k$  时,  $k(k-1)(k-2)\cdots(k-r+1) = 0$ ; 当  $k > \nu$  且  $r > \nu$  时,  $h_r(G) = 0$ . 因此上式与  $P(G, k) = \sum_{r=1}^{\nu} [h_r(G)k(k-1)(k-2)\cdots(k-r+1)]$  等价. 证毕.

我们利用定理 6.3.3 来求  $C_4$  的色多项式. 将  $C_4$  划分为两个点独立集只有一种方式, 即相间隔的两点作为一个独立集, 因此  $h_2(C_4) = 1$ ; 将  $C_4$  划分为三个点独立集有两种方式: 相间隔的两点作为一个独立集, 另两点各自作为独立集, 因此  $h_3(C_4) = 2$ . 此外, 显然  $h_1(C_4) = 0, h_4(C_4) = 1$ . 按定理 6.3.3,

$$P(C_4, k) = 1 \cdot k(k-1) + 2 \cdot k(k-1)(k-2) + 1 \cdot k(k-1)(k-2)(k-3) = k(k-1)(k^2 - 3k + 3).$$

这与前面计算的结果一致.

**定理 6.3.4** 对任何简单无环图  $G$ ,  $P(G, k)$  是  $k$  的整系数  $\nu$  次多项式, 且

(1) 首项为  $k^\nu$ ; (2) 第二项为  $-\varepsilon k^{\nu-1}$ ; (3) 常数项为零; (4) 系数正负交替出现.

**证明** 对  $\varepsilon$  作归纳. 不妨设  $G$  是简单图.

$\varepsilon = 0$  时,  $P(G, k) = k^\nu$ , 结论成立.

假设  $\varepsilon \leq m-1$  时结论成立. 考虑  $\varepsilon = m$  的情形. 由归纳假设,

$$P(G - e, k) = k^\nu - (\varepsilon - 1)k^{\nu-1} + \sum_{i=1}^{\nu-2} (-1)^{\nu-i} a_i k^i,$$

$$P(G \cdot e, k) = k^{\nu-1} - \varepsilon' k^{\nu-2} + \sum_{i=1}^{\nu-3} (-1)^{\nu-i-1} b_i k^i,$$

其中  $\varepsilon'$  是  $G \cdot e$  的边数,  $a_i, b_i$  是非负整数. 由定理 6.3.2,

$$\begin{aligned} P(G, k) &= P(G - e, k) - P(G \cdot e, k) \\ &= k^\nu - (\varepsilon - 1)k^{\nu-1} + \sum_{i=1}^{\nu-2} (-1)^{\nu-i} a_i k^i - [k^{\nu-1} - \varepsilon' k^{\nu-2} + \sum_{i=1}^{\nu-3} (-1)^{\nu-i-1} b_i k^i] \\ &= k^\nu - \varepsilon k^{\nu-1} + (\varepsilon' + a_{\nu-2})k^{\nu-2} + \sum_{i=1}^{\nu-3} (-1)^{\nu-i} (a_i + b_i) k^i. \end{aligned}$$

下一个定理是显然的. 证毕.

**定理 6.3.5** (1) 若图  $G$  有  $w$  个连通分支  $G_1, G_2, \dots, G_w$ , 则  $P(G, k) = \prod_{i=1}^w P(G_i, k)$ .

(2)  $P(G, k)$  中系数非零的项的最低次幂不低于  $w$  次幂  $k^w$ , 其中  $w$  是  $G$  的连通分支数.

**定理 6.3.6**  $G$  是树的充分必要条件是  $P(G, k) = k(k-1)^{\nu-1}$ .

**证明 充分性** 设  $P(G, k) = k(k-1)^{\nu-1}$ , 则色多项式的最低次幂是  $k$  的 1 次项. 由定理 6.3.5 知,  $w = 1$ , 即  $G$  是连通图; 又因  $k(k-1)^{\nu-1}$  的  $\nu-1$  次项的系数是  $-(\nu-1)$ , 故  $\varepsilon = \nu-1$ . 因此  $G$  是树.

**必要性** 设  $G$  是树. 下面对  $G$  的顶点数  $\nu$  用归纳法来证明  $P(G, k) = k(k-1)^{\nu-1}$ . 当  $\nu = 1$  时, 显然  $P(G, k) = k$ , 结论成立.

假设对  $\nu \leq n-1$  的所有树, 结论成立.

对  $\nu = n$  的任何树  $G$ , 取  $G$  的一个叶顶点  $v_0$ , 令  $G' = G - v_0$ , 则由归纳假设,

$$P(G', k) = k(k-1)^{\nu(G')-1} = k(k-1)^{\nu(G)-2}.$$

注意对  $G'$  的每一个点正常  $k$  染色,  $v_0$  有  $k-1$  种染色法使  $G$  正常染色, 故

$$P(G, k) = P(G', k)(k-1) = k(k-1)^{\nu(G)-1}. \quad \text{证毕.}$$

**定理 6.3.7** 设  $G$  是圈, 则  $P(G, k) = (k-1)^\nu + (-1)^\nu(k-1)$ .

**证明** 对  $G$  的顶点数  $\nu$  作归纳.

$\nu = 3$  时,  $G$  是完全图  $K_3$ , 故  $P(G, k) = k(k-1)(k-2) = (k-1)^3 + (-1)^3(k-1)$ .

假设对  $\nu = n$  的圈, 定理结论成立. 考虑顶点数  $\nu(G) = n+1$  的圈  $G$ .

从  $G$  中任取一条边  $e$ , 显然  $G - e$  是一个  $n+1$  阶树. 由定理 6.3.6,

$$P(G - e, k) = k(k-1)^{n+1-1} = k(k-1)^n.$$

此外,  $G \cdot e$  是  $n$  阶圈, 由归纳假设,  $P(G \cdot e, k) = (k-1)^n + (-1)^n(k-1)$ . 再由定理 6.3.2,

$$\begin{aligned} P(G, k) &= P(G - e, k) - P(G \cdot e, k) = k(k-1)^n - [(k-1)^n + (-1)^n(k-1)] \\ &= (k-1)^{n+1} + (-1)^{n+1}(k-1) = (k-1)^\nu + (-1)^\nu(k-1). \quad \text{证毕.} \end{aligned}$$

本节最后, 我们列举关于色多项式的几个未解决的问题.

(1) 色多项式与图不是一一对应的. 虽然同构的图有相同的色多项式, 但色多项式相同的两个图未必同构. 例如, 所有  $\nu$  个顶点的树都有相同的色多项式, 但并非所有  $\nu$  阶树都是同构的.

**问题 1.** 有相同色多项式的图有什么共同特点?

(2) 满足定理 6.3.4 的多项式未必就是色多项式. 例如, 多项式  $k^4 - 3k^3 + 3k^2$  满足定理 6.3.4, 但它不是任何图的色多项式. 事实上, 假如它是某个图  $G$  的色多项式, 则

①若  $G$  连通, 因  $\varepsilon = 3$ ,  $\nu = 4$ , 故  $G$  应是一个树, 其色多项式应为

$$k(k-1)^{\nu-1} = k(k-1)^3 \neq k^4 - 3k^3 + 3k^2.$$



②若  $G$  不连通, 则因它有 4 个顶点 3 条边, 故只能是  $G = K_3 \cup K_1$ . 而此时

$$P(G, k) = P(K_3, k) \cdot P(K_1, k) = k(k-1)(k-2) \cdot k \neq k^4 - 3k^3 + 3k^2.$$

**问题 2.** 如何判断一个多项式是色多项式?

3. Read 猜想: 色多项式  $P(G, k)$  的系数的绝对值总是先严格单调递增, 然后严格单调递减.

**问题 3.** 证明或否定这一猜想.

**问题 4.** 设计求色多项式的更好的算法.

**注** 求图的色多项式的有效算法将给出求图的点色数的有效算法, 因此, 确定图的色多项式的问题是一个 NPC 问题.

图的色多项式概念是 1912 年由美国数学家 Birkhoff 为研究四色猜想提出的. 利用色多项式, 四色猜想可叙述为: 若图  $G$  是平面图, 则  $P(G, 4) > 0$ . 尽管到目前用此方法尚未能解决四色猜想, 但色多项式的研究丰富了染色理论, 引出了许多令人感兴趣的问题. 有关色多项式的更多内容可参看文献 [54]~[57].

## §6.4 完美图

在 6.2 节, 我们得到色数的一个下界:  $\chi(G) \geq \omega(G)$ , 其中  $\omega(G)$  表示图  $G$  的团数. 有一些图的色数能达到这个下界, 比如完全图和二部图. 如果一个图的任何导出子图的色数都能达到这种下界, 则称这个图为完美图. 完美图的正式定义如下.

**定义 6.4.1** 若图  $G$  的每个点导出子图  $H$  都有  $\chi(H) = \omega(H)$ , 则称  $G$  为完美图.

容易验证, 完全图及完全图的补图 (空图) 都是完美图.

**定理 6.4.1** (1)  $n \geq 2$  时, 偶圈  $C_{2n}$  及其补图都是完美图;

(2)  $n \geq 2$  时, 奇圈  $C_{2n+1}$  及其补图都不是完美图;

**证明** (1)  $n \geq 2$  时, 偶圈  $C_{2n}$  的色数和团数都是 2. 而且,  $C_{2n}$  除自身外的任何导出子图  $H$  是一些路的不交并 (可能会有孤立顶点). 如果  $H$  不全是孤立顶点, 则其色数和团数都是 2; 如果全是孤立顶点, 则色数和团数都是 1. 因此  $C_{2n}$  是完美图.

对  $C_{2n}$  的补图  $\overline{C_{2n}}$ , 因其中任何 3 点都不是独立集, 对其顶点正常染色时每个色类 (染同一色的顶点集合) 不超过 2 个顶点, 因此  $\overline{C_{2n}}$  的顶点正常染色至少需要  $n$  种色. 另一方面, 设沿着圈  $C_{2n}$  顶点依次为  $v_1, v_2, \dots, v_{2n}$ ,  $\overline{C_{2n}}$  中对  $v_1, v_2$  染同一种色,  $v_3, v_4$  染同一种色,  $\dots, v_{2n-1}, v_{2n}$  染同一种色, 则获得  $\overline{C_{2n}}$  的一种顶点正常  $n$  染色, 可见  $\chi(\overline{C_{2n}}) = n$ . 其次, 所有奇数下标的点在  $C_{2n}$  中彼此互不相邻, 从而构成  $\overline{C_{2n}}$  的一个  $n$  点团. 因此  $\omega(\overline{C_{2n}}) \geq n = \chi(\overline{C_{2n}})$ . 另一方面, 由色数的下界  $\chi(\overline{C_{2n}}) \geq \omega(\overline{C_{2n}})$ , 便知  $\chi(\overline{C_{2n}}) = \omega(\overline{C_{2n}})$ .

对  $\overline{C}_{2n}$  的任一异于  $C_{2n}$  的导出子图  $H$ , 设其顶点下标 (按  $C_{2n}$  上的标号) 由小到大依次为  $v_{i_1}, v_{i_2}, \dots, v_{i_k}$ , ( $k < 2n$ ), 这些顶点在  $C_{2n}$  上导出的子图  $H'$  是一些路及一些孤立顶点的不交并. 设这些路中有  $r_1$  条含有偶数个顶点,  $r_2$  条含有奇数个顶点, 另有  $r_3$  个孤立顶点. 对每条路, 取其一个端点作为起始点, 依次对路上的点交替标上奇偶标号. 并从起始边开始间隔取边作为匹配边, 这样可得到  $H'$  的一个最大匹配. 设该最大匹配有  $r$  对点 (即  $r$  条边), 则这  $r$  对顶点的每一对在  $H$  中互不相邻, 可染同一种色. 又因  $H$  中任何三点都不构成独立集, 故  $\chi(H) = r + (k - 2r) = k - r$ . 另一方面, 取上述各条路上所有奇标号的顶点, 与  $H'$  的所有孤立点组成顶点子集  $A$ , 则  $A$  是  $H'$  的最大独立集, 从而是  $\overline{H}' = H$  的最大团. 因为在含有偶数个顶点的路上, 奇标号顶点的个数与匹配边的个数相等, 而在每条含有奇数个顶点的路上, 奇标号顶点的个数比匹配边的个数多 1, 因此  $|A| = r + r_2 + r_3$ . 因  $r_2 + r_3$  是各条含有奇数个顶点的路上未被匹配的点数与  $H'$  中孤立顶点个数之和, 故  $r_2 + r_3 = k - 2r$ . 从而  $|A| = r + (k - 2r) = k - r$ . 由此可见  $\omega(H) = |A| = k - r = \chi(H)$ . 因此  $\overline{C}_{2n}$  是完美图.

(2)  $n \geq 2$  时, 奇圈  $C_{2n+1}$  的色数是 3, 团数是 2, 因此不是完美图.

对  $C_{2n+1}$  的补图  $\overline{C}_{2n+1}$ , 因其中任何 3 点都不是独立集, 对其顶点正常染色时每个色类 (染同一色的顶点集合) 不超过 2 个顶点, 因此  $\overline{C}_{2n+1}$  的顶点正常染色至少需要  $n + 1$  种色. 另一方面, 设沿着圈  $C_{2n+1}$  顶点依次为  $v_1, v_2, \dots, v_{2n+1}$ ,  $\overline{C}_{2n+1}$  中对  $v_1, v_2$  染同一种色,  $v_3, v_4$  染同一种色,  $\dots, v_{2n-1}, v_{2n}$  染同一种色, 最后给  $v_{2n+1}$  染一种色, 则获得  $\overline{C}_{2n+1}$  的一种顶点正常  $n + 1$  染色, 可见  $\chi(\overline{C}_{2n+1}) = n + 1$ . 其次, 考虑  $\overline{C}_{2n+1}$  的最大团. 因  $v_i$  与  $v_{i+1}$  在  $\overline{C}_{2n+1}$  中不相邻 ( $i = 1, 2, \dots, 2n$ ), 且  $v_{2n+1}$  与  $v_1$  也不相邻, 故从  $v_1, v_2, \dots, v_{2n+1}$  中最多能取出  $n$  个在  $\overline{C}_{2n+1}$  中彼此相邻的点, 即  $\overline{C}_{2n+1}$  的最大团含点数不超过  $n$  个. 因此  $\omega(\overline{C}_{2n+1}) \leq n < n + 1 = \chi(\overline{C}_{2n+1})$ . 这表明  $\overline{C}_{2n+1}$  不是完美图. 证毕.

**定理 6.4.2** 任何二部图及二部图的补图都是完美图.

**证明** 因为二部图的色数和团数都为 2, 且其任何一个导出子图仍为二部图, 故二部图都是完美图. 下证二部图的补图是完美图.

设  $G$  是一个二部图, 对补图  $\overline{G}$  的任一个子图  $G'$ , 它必定是  $G$  的某个子图  $H$  的补图, 即  $\exists H \subseteq G$ , 使  $G' = \overline{H}$ . 下面只需证明  $\chi(\overline{H}) = \omega(\overline{H})$  即可.

用  $\alpha(H)$  和  $\beta'(H)$  分别表示  $H$  的独立数和边覆盖数, 则  $\omega(\overline{H}) = \alpha(H)$ , 且由推论 5.2.2,  $\alpha(H) = \beta'(H)$ . 另一方面, 设  $\chi(\overline{H}) = k$ ,  $\{V_1, V_2, \dots, V_k\}$  是  $\overline{H}$  的一个顶点正常  $k$  染色, 则每个  $V_i$  是  $H$  的一个团. 由于二部图  $H$  中无三角形,  $H$  的团要么只有 1 个点, 要么只有两个点. 因此, 各个  $V_i$  在  $H$  中的导出子图合在一起形成了  $H$  的一个匹配  $M$  和一些  $M$  未饱和点. 由于  $\{V_1, V_2, \dots, V_k\}$  是  $\overline{H}$  的顶点正常染色的最小色类划分, 故  $M$  是  $H$  的最大匹配. 由推论 5.2.1,  $\beta'(H) = |M| + M$  未饱和点的个数  $= k$ .

因此,  $\omega(\overline{H}) = \alpha(H) = \beta'(H) = k = \chi(\overline{H})$ . 证毕.

设  $G$  是一个图.  $G$  的线图 (line graph)  $L(G)$  是以  $G$  的边集作为顶点集的图,  $L(G)$  中两顶点相邻当且仅当相应的两条边在  $G$  中相邻.

线图的性质已得到较好的研究, 读者可参看 [59] 或 [60]

值得注意的是,  $G$  的线图  $L(G)$  中的一个团与  $G$  中的一个顶点及其关联边对应, 该团含的顶点数等于  $G$  中对应顶点的度.  $L(G)$  中的一个独立集与  $G$  中的一个匹配相对应,  $L(G)$  的独立数等于  $G$  的边覆盖数 (即最大匹配所含的边数).  $L(G)$  的点正常染色对应于  $G$  的边正常染色, 故  $\chi(L(G)) = \chi'(G)$ .

**定理 6.4.3** 设  $G$  是二部图, 则  $G$  的线图  $L(G)$  及其补图  $\overline{L(G)}$  都是完美图.

**证明** 因  $\omega(L(G))$  等于  $G$  的顶点最大度  $\Delta(G)$ , 而  $\chi(L(G)) = \chi'(G)$ .  $G$  是二部图, 由定理 6.1.1,  $\chi'(G) = \Delta(G)$ , 从而  $\chi(L(G)) = \omega(L(G))$ . 由于  $L(G)$  的点导出子图对应于  $G$  的边导出子图, 而二部图的边导出子图仍是二部图, 故上述推导对  $L(G)$  的所有点导出子图都成立, 因此  $L(G)$  是完美图.

下面考虑线图  $L(G)$  的补图  $\overline{L(G)}$  的完美性. 设  $\chi(\overline{L(G)}) = k, \{V_1, V_2, \dots, V_k\}$  是  $\overline{L(G)}$  的一个顶点  $k$  正常染色. 每个  $V_i$  是  $\overline{L(G)}$  的一个独立集, 即  $L(G)$  的一个团, 它对应于  $G$  中某个顶点 (记为  $v_i$ ) 关联的所有边  $E_i, (i = 1, 2, \dots, k)$ . 因  $V_1 \cup V_2 \cup \dots \cup V_k = V(L(G))$ , 故  $E_1 \cup E_2 \cup \dots \cup E_k = E(G)$ , 这表明  $\{v_1, v_2, \dots, v_k\}$  是  $G$  的一个点覆盖集, 而且  $\chi(\overline{L(G)}) = k$  保证了它是  $G$  的一个最小点覆盖, 因此,  $\chi(\overline{L(G)}) = k = \beta(G)$ . 另一方面,  $\omega(\overline{L(G)})$  等于  $L(G)$  中最大点独立集含的顶点数, 进而等于  $G$  中最大匹配所含的边数, 即  $\omega(\overline{L(G)}) = \alpha'(G)$ . 由 König-Egerváry 定理 (定理 5.2.2), 对于二部图  $G$ , 有  $\alpha'(G) = \beta(G)$ , 因此  $\chi(\overline{L(G)}) = \omega(\overline{L(G)})$ . 因  $\overline{L(G)}$  的点导出子图必是  $L(G)$  的相应点导出子图的补图, 它对应于  $G$  的一个边导出子图, 而二部图的边导出子图仍是二部图, 故上述推导对  $\overline{L(G)}$  的所有点导出子图都成立, 因此  $\overline{L(G)}$  是完美图. 证毕.

以上完美图和非完美图的个例都支持一个普遍性的命题: 一个图是完美图当且仅当它的补图是完美图. 这个命题称为完美图猜想, 由 Berge 于 1961 年提出. 许多人认为要证明这个猜想是很困难的. 1971 年, 22 岁的 Lovász 证明了这个著名的猜想, 在当时引起了组合数学界的轰动. 下面我们来介绍对完美图猜想的证明.

**引理 6.4.1** 图  $G$  是完美图当且仅当对  $G$  的每个点导出子图  $H$ , 存在一个独立集  $I$ , 使得

$$\omega(H - I) < \omega(H).$$

即  $G$  是完美图当且仅当  $G$  的每个点导出子图  $H$  都有一个独立集  $I$ , 它与  $H$  的所有最大团都有公共顶点.

**证明 必要性** 设  $G$  是完美图,  $H$  是  $G$  的一个点导出子图, 则  $\chi(H) = \omega(H)$ . 设  $\chi(H) = k$ . 在  $H$  的任何一种顶点  $k$  正常染色下, 设  $I$  是一个色类 (染某种色的所有顶点之集). 则有

$$\omega(H - I) \leq \chi(H - I) = k - 1 = \chi(H) - 1 = \omega(H) - 1 < \omega(H).$$

**充分性** 对  $\omega(G)$  作归纳法.

当  $\omega(G) = 1$  时,  $G$  显然是完美图.

假定对满足定理条件且  $\omega(G) < k$  的任何图  $G$ ,  $G$  都是完美图.

设  $G$  是一个使得  $\omega(G) = k$  的图, 且对  $G$  的每个点导出子图  $H$ , 存在一个独立集  $I$ , 使得  $\omega(H - I) < \omega(H)$ . 由归纳假设,  $H - I$  是完美图, 于是  $\chi(H - I) = \omega(H - I)$ . 设  $\chi(H - I) = r$  且  $\{V_1, V_2, \dots, V_r\}$  是  $H - I$  的一个顶点正常  $r$  染色. 则  $\{V_1, V_2, \dots, V_r, I\}$  是  $H$  的一个顶点正常  $r + 1$  染色. 故

$$\chi(H) \leq r + 1 = \chi(H - I) + 1 = \omega(H - I) + 1 \leq \omega(H).$$

另一方面,  $\chi(H) \geq \omega(H)$ . 因此  $\chi(H) = \omega(H)$ . 这表明  $G$  是完美图. 归纳完成. 证毕.

设  $G$  是一个图, 其顶点集为  $\{v_1, v_2, \dots, v_n\}$ ,  $H_1, H_2, \dots, H_n$  是一些无公共顶点的图. 将  $G$  中每个顶点  $v_i$  用  $H_i$  替换 ( $i = 1, 2, \dots, n$ ), 并按  $G$  中顶点的相邻关系连接  $H_1, H_2, \dots, H_n$ : 如果  $v_i$  与  $v_j$  在  $G$  中相邻, 则将  $H_i$  的顶点与  $H_j$  的顶点之间建立完全连接. 这样得到的图称为  $H_1, H_2, \dots, H_n$  在  $G$  上的合成图, 记为  $G\{H_1, H_2, \dots, H_n\}$ . 确切地说,  $G\{H_1, H_2, \dots, H_n\}$  的顶点集为  $V(H_1) \cup V(H_2) \cup \dots \cup V(H_n)$ , 对  $G\{H_1, H_2, \dots, H_n\}$  中任意两顶点  $u, v$ , 当且仅当  $u$  和  $v$  在某个  $H_i$  中相邻, 或者  $u$  和  $v$  分别属于  $H_i$  和  $H_j$  且相应的  $v_i$  和  $v_j$  在  $G$  中相邻时,  $u, v$  在  $G\{H_1, H_2, \dots, H_n\}$  中相邻.

**引理 6.4.2** 如果  $G$  和  $H_1, H_2, \dots, H_n$  都是完美图, 则合成图  $G\{H_1, H_2, \dots, H_n\}$  也是完美图.

**证明** 由于合成  $G\{H_1, H_2, \dots, H_n\}$  可以看成是用  $H_1, H_2, \dots, H_n$  逐次替换  $v_1, v_2, \dots, v_n$  得到的, 因此只需证明用一个完美图  $H$  替换  $G$  的某一个顶点  $v$  得到的图  $G\{H\}$  是完美图即可. 按照引理 1, 完美图  $H$  有一个点独立集  $I$  使得  $\omega(H - I) < \omega(H)$ . 用  $\chi(G)$  种颜色对  $G$  的顶点进行正常染色, 将  $v$  及与  $v$  同色的顶点的集合记为  $A$ , 则  $B = (I \cup A) - \{v\}$  是  $G\{H\}$  的一个独立集. 我们断言  $B$  与  $G\{H\}$  的每个最大团都有公共顶点.

事实上, 设  $Q$  是  $G\{H\}$  的一个最大团. 如果  $Q$  含有  $H$  中的顶点, 则因在  $G\{H\}$  中  $H$  的各个顶点对  $H$  外的相邻关系是相同的, 故此时  $H$  中每个顶点都与  $Q - H$  的所有顶点相邻. 因此  $Q$  必定含有  $H$  的一个最大团. 又因  $\omega(H - I) < \omega(H)$ ,  $H$  的每个

最大团都与  $I$  有公共顶点, 从而  $Q$  必定与  $I$  有公共顶点, 于是  $B = (I \cup A) - \{v\}$  与最大团  $Q$  有公共顶点.

如果  $G\{H\}$  的最大团  $Q$  不含有  $H$  中的顶点, 则  $Q \subseteq V(G) - \{v\}$  且在  $G$  中  $v$  不与  $Q$  的顶点相邻, 因此  $Q$  是  $G$  的一个最大团. 因  $G$  是完美图,  $\chi(G) = \omega(G)$ , 且对  $G$  的任何一种  $\chi(G)$  正常染色, 每个色类 (同色顶点之集) 至多含最大团  $Q$  中一个点, 而  $|Q| = \omega(G) = \chi(G)$ , 于是每个色类恰含  $Q$  中一个点. 因此在  $G$  的任何一种  $\chi(G)$  正常染色下,  $Q$  与每个色类都有公共点. 特别地,  $Q$  与前述色类  $A$  有公共点, 从而与  $I \cup A$  有公共点, 又因为  $Q$  不含顶点  $v$ , 故  $Q$  与集合  $B = (I \cup A) - \{v\}$  有公共点.

对  $G\{H\}$  的任一点导出子图  $G'$ , 如果  $G' \subseteq G$  或  $G' \subseteq H$ , 则因完美图的子图仍是完美图, 故由引理 1,  $G'$  有一个独立集, 它与  $G'$  的所有最大团都有公共顶点. 否则,  $G'$  可视为由  $G$  的某个含  $v$  的点导出子图与  $H$  的某个点导出子图在  $v$  点合成的图, 按上述推导,  $G'$  也存在独立集, 它与  $G'$  的所有最大团都有公共顶点.

由以上结果可见, 图  $G\{H\}$  满足引理 1 的条件, 故是完美图. 证毕.

**定理 6.4.4 (完美图定理, Lovász<sup>[61][62]</sup>, Fulkerson, 1971<sup>[63]</sup>)** 一个图是完美图当且仅当它的补图是完美图.

**证明 必要性** 设  $G$  是完美图. 为证其补图  $\bar{G}$  也是完美图, 由引理 1, 只需证明对  $\bar{G}$  的任何点导出子图  $\bar{H}$ , 都有一个独立集  $I$ , 使得  $\omega(\bar{H} - I) < \omega(\bar{H})$ . 将  $\bar{H}$  的补图记为  $H$ , 它是  $\bar{H}$  的顶点在  $G$  中的导出子图. 从  $H$  的角度来看, 需要证明的就是, 存在  $H$  的一个团  $Q$ , 使得  $\alpha(H - Q) < \alpha(H)$ . 我们用反证法来证明这一结论.

设  $H$  的顶点集为  $\{v_1, v_2, \dots, v_\nu\}$ ,  $Q_1, Q_2, \dots, Q_r$  是  $H$  的全部团. 假定对每个  $Q_i$ , 都存在  $H - Q_i$  的某个点独立集  $A_i$ , 使得  $|A_i| = \alpha(H)$ , ( $i = 1, 2, \dots, r$ ). 对  $H$  的任一顶点  $v_k$ , 设  $A_1, A_2, \dots, A_r$  中有  $n_k$  个含有  $v_k$ , 用  $n_k$  阶完全图  $K_{n_k}$  替代  $H$  中的顶点  $v_k$ , ( $k = 1, 2, \dots, \nu$ ), 则得到合成图  $G' = H\{K_{n_1}, K_{n_2}, \dots, K_{n_\nu}\}$ . 由于完美图  $G$  的子图  $H$  是完美图, 完全图也是完美图, 按照引理 2,  $G'$  是完美图. 下面来计算  $\omega(G')$  和  $\chi(G')$ .

$G'$  的最大团必可由  $H$  的某个团  $Q_j$  的顶点用相应的完全图替代后获得, 因此,  $\omega(G') = \sum_{v_k \in Q_j} n_k$ . 注意  $n_k$  是  $r$  个独立集  $A_1, A_2, \dots, A_r$  含有顶点  $v_k$  的独立集的个数. 对  $i = 1, 2, \dots, r$ , 因  $A_i$  是独立集而  $Q_j$  是团, 故每个  $A_i$  至多含有  $Q_j$  中一个顶点. 也就是说, 对  $Q_j$  中任何两个顶点  $v_{k'}$  和  $v_{k''}$ , 这  $r$  个独立集中含有  $v_{k'}$  的独立集和含有  $v_{k''}$  的独立集互不相同. 此外, 与  $Q_j$  对应的独立集  $A_j$  不含有  $Q_j$  中任何顶点. 因此, 含有  $Q_j$  中各个顶点的独立集个数之和不会大于  $r - 1$ , 即

$$\omega(G') = \sum_{v_k \in Q_j} n_k \leq r - 1. \quad (1)$$

另一方面, 设  $\chi(G') = t, V_1, V_2, \dots, V_t$  是  $G'$  的一个顶点  $t$  正常染色, 则

$$\nu(G') = |V_1| + |V_2| + \dots + |V_t| \leq \chi(G') \cdot \alpha(G'). \quad (2)$$

由  $G'$  的构造可知,

$$\alpha(G') = \alpha(G). \quad (3)$$

且由  $A_1, A_2, \dots, A_r$  的性质, 有

$$\sum_{i=1}^r |A_i| = r \cdot \alpha(G). \quad (4)$$

构造矩阵  $M = (m_{ij})$ , 其中  $m_{ij}$  当  $A_i$  含有  $H$  的顶点  $v_j$  时取值为 1, 否则取值为 0, ( $i = 1, 2, \dots, r; j = 1, 2, \dots, \nu(H)$ ). 注意到  $\sum_{i=1}^r |A_i|$  等于将矩阵  $M$  的元素按行相加,

而将  $M$  的元素按列相加的结果等于  $\sum_{j=1}^{\nu(H)} n_j = \nu(G')$ . 因此,  $\sum_{i=1}^r |A_i| = \nu(G')$ . 结合 (2)、(3)、(4) 式有:

$$\chi(G') \cdot \alpha(G') \geq \nu(G') = \sum_{i=1}^r |A_i| = r \cdot \alpha(G) = r \cdot \alpha(G').$$

即  $\chi(G') \geq r$ , 与 (1) 式结合, 得  $\chi(G') > \omega(G')$ . 这与  $G'$  的完美性矛盾. 定理的必要性得证.

**充分性** 设  $\bar{G}$  是完美图, 则因  $G$  是  $\bar{G}$  的补图, 故由必要性的证明, 知  $G$  也是完美图. 证毕.

在历史上, 人们曾定义了两种完美图. 为了给出第二种完美图的定义, 我们需要引入团覆盖集的概念.

设  $G$  是一个图,  $F = \{Q_1, Q_2, \dots, Q_r\}$  是  $G$  的一些团的集合, 如果  $Q_1 \cup Q_2 \cup \dots \cup Q_r = V(G)$ , 则称  $F$  是  $G$  的一个团覆盖集.  $G$  的含团数最少的团覆盖集称为最小团覆盖集, 它所含的团的个数称为  $G$  的团覆盖数, 记为  $\theta(G)$ .

设  $F = \{Q_1, Q_2, \dots, Q_r\}$  是简单图  $G$  的任一个团覆盖,  $I$  是  $G$  的任一个点独立集. 因  $F$  是团覆盖, 对  $I$  的任意一点  $v$ , 必存在某个团  $Q_i$ , 使  $v \in Q_i$ , 而  $I$  中任意两点不会属于同一个团, 因此  $|I| \leq r = |F|$ , 由此可知  $\theta(G) \geq \alpha(G)$ .

**定义 6.4.2** 若图  $G$  的每个点导出子图  $H$  都有  $\alpha(H) = \theta(H)$ , 则称  $G$  为关于点独立数的完美图, 简称为  $\alpha$ -完美图.

相应地, 我们将定义 6.4.1 中的完美图称为关于色数的完美图, 简称为  $\chi$ -完美图. 下面的定理表明这两种完美图的定义本质上是相同的.

**定理 6.4.5** 图  $G$  是  $\alpha$ -完美图当且仅当它是  $\chi$ -完美图.

**证明** 首先证明  $G$  是  $\alpha$ -完美图当且仅当其补图  $\bar{G}$  是  $\chi$ -完美图.

事实上,  $\bar{G}$  的每个点导出子图  $\bar{H}$  必是  $G$  的某个点导出子图  $H$  的补图. 若  $G$  是  $\alpha$ -完美图, 则  $G$  有  $\alpha(H) = \theta(H)$ . 设  $\chi(\bar{H}) = k, V_1, V_2, \dots, V_k$  是  $\bar{H}$  的一个顶点  $k$  正常染色, 则  $V_1, V_2, \dots, V_k$  构成  $H$  的一个最小团覆盖. 因此  $\chi(\bar{H}) = \theta(H)$ . 此外,  $H$  的最大独立集必为  $\bar{H}$  的最大团, 故  $\alpha(H) = \omega(\bar{H})$ . 从而  $\chi(\bar{H}) = \theta(H) = \alpha(H) = \omega(\bar{H})$ . 这表明  $\bar{G}$  是  $\chi$ -完美图.

反之, 若  $G$  的补图  $\bar{G}$  是  $\chi$ -完美图, 则对  $G$  的任一个点导出子图  $H, \chi(\bar{H}) = \omega(\bar{H})$ . 与前面完全同样的道理, 得  $\chi(\bar{H}) = \theta(H)$  及  $\alpha(H) = \omega(\bar{H})$ . 因此,

$$\alpha(H) = \omega(\bar{H}) = \chi(\bar{H}) = \theta(H).$$

这表明  $G$  是  $\alpha$ -完美图.

其次, 由完美图定理 (定理 6.4.4),  $\bar{G}$  是  $\chi$ -完美图当且仅当  $G$  是  $\chi$ -完美图. 从而定理得证. 证毕.

目前, 人们已经发现近百种完美图<sup>[64]</sup>. 其中除了前面介绍的一些外, 还有区间图和弦图也是较常见的两种.

设  $J_1, J_2, \dots, J_n$  是  $n$  个实数区间, 以  $\{J_1, J_2, \dots, J_n\}$  为顶点集构造图  $G, G$  中顶点  $J_i$  与  $J_j$  相邻当且仅当区间  $J_i$  与  $J_j$  相交. 这样得到的图称为区间图 (interval graph).

设  $C$  是一个圈,  $C$  的一条弦是本身不在  $C$  上但两端点在  $C$  上的边. 如果图  $G$  中每个长至少为 4 的圈都有弦, 则称  $G$  为弦图 (chordal graph).

**定理 6.4.6** 区间图是完美图.

**证明** 设  $G$  是一个区间图, 按  $G$  的顶点对应区间的左端点值由小到大对  $G$  的所有顶点排序, 设顺序为  $v_1, v_2, \dots, v_\nu$ . 依次对  $G$  的顶点染色: 对顶点  $v_i$ , 用此前  $v_i$  的邻点尚未使用的最小色号的颜色给  $v_i$  染色, ( $i = 1, 2, \dots, \nu$ ). 这样可得到  $G$  的一个顶点正常染色. 设这样染色后用到的最大色号为  $k$ , 假定顶点  $v_j$  染有颜色  $k$ , 这说明在给  $v_j$  染色时, 已经至少有  $v_j$  的  $k-1$  个下标小于  $j$  的邻点分别染有颜色 1 到  $k-1$ . 设  $v_j$  对应的区间  $J_j$  的左端点为  $x$ , 则上述  $k-1$  个顶点对应的区间都与  $J_j$  相交且左端点小于和等于  $x$ , 因而这些区间必定彼此相交. 由此可知这  $k$  个顶点构成  $G$  的一个  $k$ -团. 从而  $\chi(G) \leq k \leq \omega(G)$ . 另一方面, 对任何图都有  $\chi(G) \geq \omega(G)$ , 故  $\chi(G) = \omega(G)$ .

由于区间图  $G$  的点导出子图  $H$  仍是区间图, 故对  $H$  上述推导仍成立. 因此  $G$  是完美图. 证毕.

与定理 6.4.6 类似可证, 弦图也是完美图 (见 [54]).

含有奇圈的无三角形图, 由于其团数为 2, 色数至少是 3, 故不是完美图. 按照完美图定理, 这样的图的补图也不是完美图. 这就是说, 若  $G$  是完美图, 则  $G$  和  $\bar{G}$  的任何导出子图必定不含有长至少是 5 的奇圈. Berge 猜想, 这个命题的逆命题也是成立的, 这便是著名的强完美图猜想.

**强完美图猜想** (Berge<sup>[65]</sup>, 1961)  $G$  是完美图当且仅当  $G$  和  $\bar{G}$  的任何导出子图都不是长至少为 5 的奇圈.

经过人们四十年的努力, 强完美图猜想终于在 2002 年被 Chudnovsky 等人所证明<sup>[64][66][67]</sup>. 此外, 判别一个图是否为完美图的多项式时间算法也已被得到<sup>[68][69]</sup>. 这是完美图理论的重大进展. 通过围绕完美图猜想和强完美图猜想的研究, 加深了人们对图的结构性质的认识. 目前完美图理论已经形成了十分丰富的内容, 这一领域的研究仍很活跃.

有关完美图的更多内容可参看文献 [54]、[58]、[64] 或 [70]~[74].

## §6.5 图的边染色算法和点染色算法

### 一、排课表问题 — 二部图的边染色算法

#### 1. 问题

有  $m$  位教师  $x_1, x_2, \dots, x_m$ ,  $n$  个班级  $y_1, y_2, \dots, y_n$ . 教师  $x_i$  每周需要给班级  $y_j$  上  $p_{ij}$  次 (节) 课. 要求制订一张周课时尽可能少的课程表.

#### 2. 图论模型

构造二部图  $G = (X, Y)$ , 其中  $X = \{x_1, x_2, \dots, x_m\}$ ,  $Y = \{y_1, y_2, \dots, y_n\}$ , 顶点  $x_i$  与  $y_j$  之间连  $p_{ij}$  条边.

一个课时的安排方案对应于二部图  $G$  的一个匹配. 排课表问题等价于: 将  $E(G)$  划分成一些匹配, 使得匹配的数目尽可能地少.

按  $\chi'(G)$  的定义, 这个最小的数目便是  $\chi'(G)$ . 由定理 6.1.1,  $\chi'(G) = \Delta(G)$ . 因此, 排课表问题等价于: 求二部图  $G$  的边正常  $\Delta(G)$  染色.

如§6.1 中所述, 虽然求简单图的正常  $(\Delta + 1)$  边染色存在多项式时间算法, 但求简单图  $G$  的边色数  $\chi'(G)$  及其相应的正常边染色是一个 NPC 问题<sup>[2]</sup>. 尽管如此, 求二部图的边正常  $\chi'(G)$  ( $=\Delta(G)$ ) 染色却有多项式时间算法. 求二部图和一般图的边色数算法方面的研究可参考文献 [75]~[99].

#### 3. 求二部图 $G = (X, Y)$ 的边正常 $\Delta(G)$ 染色的算法

• **算法思想:** 给  $G$  添加必要的顶点使得  $|X| = |Y|$ , 再添加必要的边使得  $G$  成为  $\Delta(G)$  正则二部图, 所得图记为  $G^*$ , 然后反复运用匈牙利算法求  $G^*$  的完美匹配. 由第



3 章 König 定理,  $G^*$  存在完美匹配. 每求出  $G^*$  的一个完美匹配, 便可用一种颜色给这个完美匹配的边染色. 因为共可求得  $G^*$  的  $\Delta(G)$  个彼此无公共边的完美匹配, 故可得  $G^*$  (从而  $G$ ) 的边正常  $\Delta(G)$  染色.

• **二部图边染色算法:** 求二部图的边正常  $\Delta(G)$  染色 (求二部图的  $\Delta(G)$  个彼此无公共边的匹配).

输入: 二部图  $G = (X, Y)$

输出:  $G$  的边正常  $\Delta(G)$  染色 ( $\Delta(G)$  个彼此无公共边的匹配)

第 0 步: 添加顶点使得  $|X| = |Y|$ , 所得图记为  $G^*$ .

第 1 步: 若  $\Delta(G^*) = \delta(G^*)$ , 令  $k = 1$ , 转第 3 步; 否则, 转第 2 步.

第 2 步: 取  $x_0 \in X, y_0 \in Y$ , 使得  $d_{G^*}(x_0) = \min_{x_i \in X} d_{G^*}(x_i), d_{G^*}(y_0) = \min_{y_i \in Y} d_{G^*}(y_i)$ ,

令  $G^* := G^* + x_0y_0$ , 转第 1 步.

第 3 步: 任取  $G^*$  的一个匹配  $M$ .

第 4 步: 若  $X$  已  $M$  饱和, 则输出当前的完美匹配, 转第 7 步; 否则取  $X$  中一个  $M$  不饱和点  $u$ , 置  $S := \{u\}, T := \phi$ .

第 5 步: 在  $N(S) \setminus T$  中取一点  $y$ .

第 6 步: 若  $y$  是  $M$  饱和的, 则存在  $yz \in M$ , 置  $S := S \cup \{z\}, T := T \cup \{y\}$ , 转第 5 步; 否则, 存在一条  $M$  可扩路  $P(u, y)$ , 置  $M := M \oplus E(P)$ , 转第 4 步.

第 7 步: 若  $k = \Delta$ , 则停止; 否则, 令  $k := k + 1, G^* := G^* \setminus M$ , 转第 3 步.

算法的核心是循环执行第 3 步 ~ 第 6 步使用匈牙利算法反复求完美匹配 ( $\Delta$  个). 设  $|X| \geq |Y|$ , 因匈牙利算法的时间复杂度为  $O(\nu^3)$ , 而第 1 步和第 2 步的加边循环不超过  $|X| \cdot \Delta$  次, 故该算法的计算复杂度为  $O(\Delta \cdot \nu^3)$ .

#### 4. 带有约束的排课表问题

设学校每周有  $l$  节课, 安排在一张有  $p$  节课时的课表中 (前面介绍的方法求得一张  $\Delta$  节课时的课表). 这样, 平均每一课时要上  $\left\lceil \frac{l}{p} \right\rceil$  节课, 因此需要  $\left\lceil \frac{l}{p} \right\rceil$  间教室. 比如,  $l = 510, p = 20$ , 则需要  $\left\lceil \frac{l}{p} \right\rceil = \left\lceil \frac{510}{20} \right\rceil = 26$  间教室.

**问题:** 设学校已确定各班级的任课教师及各科每周课时. 全校每周有  $l$  节课 (每节课 1 课时), 问可否在一张有  $p$  节课时的课表里安排这  $l$  节课, 使得在每节课时内至多使用  $\left\lceil \frac{l}{p} \right\rceil$  间教室?

**问题的图论描述:** 构造二部图  $G = (X, Y)$ , 其中  $X = \{x_1, x_2, \dots, x_m\}$  对应于教师集合,  $Y = \{y_1, y_2, \dots, y_n\}$  对应于班级集合, 若教师  $x_i$  每周给班级  $y_j$  上  $p_{ij}$  节课, 则在顶点  $x_i$  与  $y_j$  之间连  $p_{ij}$  条边. 因全校每周有  $l$  节课, 故  $G$  是有  $l$  条边的二部图 (可能有重边), 问题是判断这样一个二部图其边集合能否划分成  $p$  个匹配, 使得每个匹配

含边数不超过  $\left\lceil \frac{l}{p} \right\rceil$ ? 用图染色的术语来说, 就是判断一个  $l$  条边的二部图 (允许有重边), 能否用  $p$  种色进行边正常染色, 使每种色染边不超过  $\left\lceil \frac{l}{p} \right\rceil$  条 (每种色使用不超过  $\left\lceil \frac{l}{p} \right\rceil$  次), 如果能, 求出一种满足要求的边正常  $p$  染色.

下面一个引理见第 3 章习题 3.6.

**引理 6.5.1** 设  $M$  和  $N$  是图  $G$  的两个无公共边的匹配, 并且  $|M| > |N|$ , 则存在  $G$  的无公共边的匹配  $M'$  和  $N'$ , 使得  $|M'| = |M| - 1$ ,  $|N'| = |N| + 1$ , 且  $M' \cup N' = M \cup N$ .

**定理 6.5.1** 若图  $G$  是一个二部图, 且  $p \geq \Delta(G)$ , 则  $G$  中存在  $p$  个无公共边的匹配  $M_1, M_2, \dots, M_p$ , 使得  $E = M_1 \cup M_2 \cup \dots \cup M_p$ , 且对每个  $i$  ( $i = 1, 2, \dots, p$ ) 均有

$$\left\lfloor \frac{\varepsilon(G)}{p} \right\rfloor \leq |M_i| \leq \left\lceil \frac{\varepsilon(G)}{p} \right\rceil.$$

**证明** 因图  $G$  是二部图, 故由本章定理 6.1.1, 边集  $E(G)$  可划分为  $\Delta$  个匹配  $M'_1, M'_2, \dots, M'_\Delta$ , 因而对任何  $p \geq \Delta(G)$ ,  $G$  中存在  $p$  个无公共边的匹配  $M'_1, M'_2, \dots, M'_\Delta, M'_{\Delta+1}, \dots, M'_p$  (其中  $M'_{\Delta+1} = \dots = M'_p = \phi$ ), 使得  $E(G) = \bigcup_{i=1}^p M'_i$ . 对这些匹配反复运用引理 6.5.1, 便可得到满足定理要求的匹配. 证毕.

**注** 若  $p < \Delta(G)$ , 则二部图  $G$  的边显然无法划分为  $p$  个无公共边的匹配 (因最大度顶点关联的  $\Delta$  条边无法全部划分到  $p$  个无公共边的匹配中). 因此定理 6.5.1 实际上给出了判断二部图的边集能否划分为  $p$  个无公共边的匹配的充分必要条件, 它也是判断二部图能否进行边正常  $p$  染色的一个充分必要条件.

定理 6.5.1 不仅给出了前述带约束排课表问题中满足要求的课表是否存在的判断条件, 同时其证明过程也给出了所需教室数最少的  $p$  课时课表的求法: 先按二部图边染色算法求出相应二部图的一个正常  $\Delta(G)$  边染色, 然后反复运用引理 6.5.1 对其进行调整, 最终使得染不同颜色的两个边集所含边数至多相差 1.

**例 6.5.1** 设有 4 个教师  $x_1, x_2, x_3, x_4$  和 5 个班级  $y_1, y_2, y_3, y_4, y_5$ , 周课时教学矩阵  $T = (t_{ij})$  表示如下:

$$T = \begin{matrix} & y_1 & y_2 & y_3 & y_4 & y_5 \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix} & \begin{pmatrix} 2 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{matrix}$$

问: (1) 课表至少需要几课时?

(2) 按二部图边染色算法给出一个课时数最少的课表.

(3) 在课时数最少的前提下, 给出需教室数最少的课表方案.

解 构造二部图如下图 1:

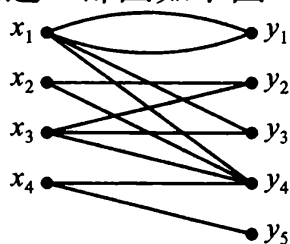


图1

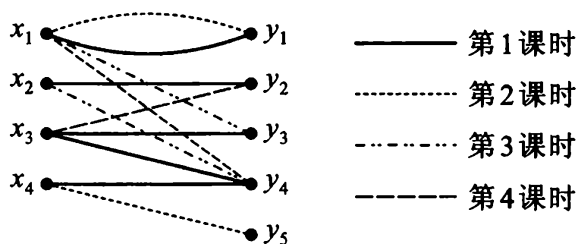


图2

由于  $\Delta(G) = 4$ , 故课表至少需 4 课时.

执行二部图边染色算法得  $G$  的 4 个边不交的匹配 (如图 2). 相应的一个 4 课时课表如下:

教师 \ 课时	1	2	3	4
$x_1$	$y_1$	$y_1$	$y_3$	$y_4$
$x_2$	$y_2$		$y_4$	
$x_3$	$y_3$	$y_4$		$y_2$
$x_4$	$y_4$	$y_5$		

按这张课表安排, 需 4 间教室. 但因  $\varepsilon(G) = 11$ ,  $\left\lceil \frac{\varepsilon}{p} \right\rceil = \left\lceil \frac{11}{4} \right\rceil = 3$ , 由定理 6.5.1, 可排出一张只需 3 个教室的 4 课时课表. 事实上, 将教师  $x_4$  在第 1 课时的课调到第 3 课时而将教师  $x_2$  在第 3 课时的课与第 1 课时对调即可. 从二部图的匹配上来看, 是将第 1 课时和第 3 课时对应的匹配施行了一次引理 6.5.1 的操作. 一张只需 3 间教室的 4 课时课表如下:

教师 \ 课时	1	2	3	4
$x_1$	$y_1$	$y_1$	$y_3$	$y_4$
$x_2$	$y_4$		$y_2$	
$x_3$	$y_3$	$y_4$		$y_2$
$x_4$		$y_5$	$y_4$	

排课表问题可推广为更一般的时间表问题, 有关研究进展可参看文献 [100]~[107].

## 二、图的点染色算法

### 1. 点染色应用背景举例

#### (1) 考试安排问题

某校有  $n$  门选修课  $L_1, L_2, \dots, L_n$  需进行期末考试, 同一个学生不能在同一天里参加两门或两门以上课程的考试. 试问: 该校的期末考试至少需要几天? 如何安排?

**图论模型:** 构造图  $G = (V, E)$  如下:  $V(G) = \{L_1, L_2, \dots, L_n\}, L_i L_j \in E(G)$  当且仅当课程  $L_i$  和  $L_j$  被同一学生选修.

将同一天的考试课程在  $G$  中对应的顶点染同一种色, 则考试安排相当于对  $G$  进行顶点正常染色. 所求最少天数即为  $G$  的点色数  $\chi(G)$ . 问题化为: 求图  $G$  的正常  $\chi(G)$  点染色.

### (2) 电视频道分配问题

某地区有  $n$  家无线电视发射台  $T_1, T_2, \dots, T_n$ , 主管部门给每家电视台分配一个发射频道. 为排除同频干扰, 使用相同频道的发射台之间相距必须大于一定的距离  $d$ . 试问: 该地区至少需要多少个频道? 如何分配?

**图论模型:** 构造图  $G = (V, E)$  如下:  $V(G) = \{T_1, T_2, \dots, T_n\}, T_i T_j \in E(G)$  当且仅当发射台  $T_i$  和  $T_j$  的距离不超过  $d$ .

考虑对图  $G$  进行点正常染色. 按要求分配一个频道, 相当于给  $G$  中一些不相邻的顶点染同一种色; 反之, 染同一种色的顶点可分配给同一频道. 因此, 频道分配相当于对  $G$  进行点正常染色. 所求最少频道数即为点色数  $\chi(G)$ . 问题化为: 求图  $G$  的正常  $\chi(G)$  点染色.

### (3) 储藏问题

某公司生产  $n$  种化学品  $C_1, C_2, \dots, C_n$ , 其中某些制品不能存放在同一个仓库中. 问至少需要多少个仓库? 如何安排存放?

**图论模型:** 构造图  $G = (V, E)$  如下:  $V(G) = \{C_1, C_2, \dots, C_n\}, C_i C_j \in E(G)$  当且仅当制品  $C_i$  和  $C_j$  不能放在同一仓库中.

若干制品可放在同一仓库当且仅当它们在  $G$  中对应的顶点可染同一种色. 可见给这些制品分配仓库相当于对  $G$  进行正常点染色. 所需的最少仓库数即为  $G$  的点色数  $\chi(G)$ . 问题化为: 求图  $G$  的正常  $\chi(G)$  点染色.

## 2. 图的点染色算法

图的点染色算法方面的研究相当活跃, 有关研究内容可参见著作 [42]~[53] 或文献 [108]~[164], 其中文献 [108]~[121] 研究图的点染色问题的各种精确算法, [122]~[136] 研究图的点染色问题的近似算法及其性能, 在不同条件下给出了较低近似比的染色算法, [137]~[160] 将邻域方法、禁忌搜索和局部搜索方法、模拟退火法、人工神经网络、遗传算法、蚁群算法等方法以及其他一些启发式方法用于图的点染色问题, [161]~[164] 讨论点染色的相关计算复杂性问题. 对既不是完全图又不是奇圈的图  $G$ , 利用 Brooks 定理的证明可以得到求  $G$  的顶点  $\Delta(G)$ - 正常染色的一個有效算法<sup>[123]</sup>. 但是, 在一般情况下,  $\Delta(G)$  与  $\chi(G)$  并不相等. 求任意图的正常  $\chi(G)$  点染色是一个 NPC 问题<sup>[53][54][123]</sup>, 目前没有多项式时间精确算法, 仅有一些适用于小规模问题的非多项式时间算法和一些多项式时间近似算法. 遗憾的是, 迄今为止找到的多项式时间近似算

法其近似比都不等于常数. 可见, 从计算复杂性的角度来看, 图的点染色问题是如此困难, 以至于连近似比等于常数的多项式时间近似算法都难以找到. 实际上, 已经证明, 如果存在多项式时间染色算法使得对每个图  $G$  都可使用不多于  $2\chi(G)$  种色进行正常顶点染色, 则必定存在  $G$  的  $\chi(G)$  顶点染色多项式时间算法<sup>[124][125]</sup>. 这表明, 对图的点染色问题, 找近似比等于常数的多项式时间近似算法与找  $\chi(G)$  顶点染色的多项式时间精确算法一样困难.

下面介绍图的几种点染色算法, 其中第一种是求色数  $\chi(G)$  的非多项式时间精确算法, 后三种是多项式时间近似算法.

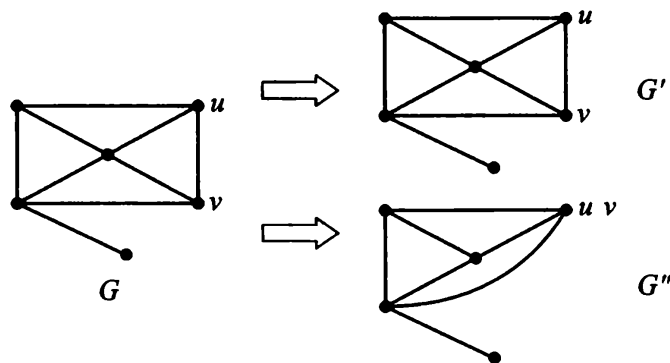
### 算法 1 添边粘合法

(1) 添边、粘合操作: 给定图  $G=(V, E)$ , 设  $u, v \in V(G)$ , 且  $u, v$  在  $G$  中互不相邻, 则

(2) 添边操作: 给  $G$  添加边  $uv$ , 得图  $G'$ ;

(3) 粘合操作: 将  $u, v$  两点粘合为一点, 并去掉所得的重边, 得图  $G''$ .

例如, 下图中图  $G$  通过在  $u, v$  两点间添边得图  $G'$ , 通过粘合  $u, v$  两点得图  $G''$ .



**定理 6.5.2**  $\chi(G) = \min\{\chi(G'), \chi(G'')\}$ , 其中  $G'$  和  $G''$  分别是对图  $G$  中两个不相邻顶点进行上述添边和粘合操作所得的图.

**证明** 考虑图  $G$  的所有可能的正常点染色方案. 设  $u, v$  是  $G$  中互不相邻的两点, 则在  $G$  的正常点染色下  $u, v$  的染色有两种可能:  $u$  与  $v$  同色, 或  $u$  与  $v$  异色.  $G$  的使  $u$  与  $v$  染同色的正常点染色方案与  $G''$  的正常点染色方案一一对应, 而  $G$  的使  $u$  与  $v$  染异色的正常点染色方案与  $G'$  的正常点染色方案一一对应. 因此  $\chi(G) = \min\{\chi(G'), \chi(G'')\}$ . 证毕.

**添边粘合算法基本思想:** 对于给定的图  $G$ , 若  $G$  是一个  $\nu$  阶完全图, 则  $\chi(G) = \nu$ . 此时给  $G$  的每个顶点一个不同的色即可. 若  $G$  不是完全图, 则可取两个不相邻顶点  $u, v$ , 对  $G$  进行添边操作和粘合操作, 并删去出现的重边. 反复进行这个过程, 直至得到的图都成为完全图为止. 设最终得到的阶数最小的完全图为  $K_m$ , 则由定理 6.5.2,  $\chi(G) = m$ . 该最小阶完全图各顶点所对应的粘合点集合, 便是图  $G$  的顶点正常  $\chi(G)$  染色划分  $V_1, V_2, \dots, V_{\chi(G)}$ .

### 添边粘合算法步骤

输入: 图  $G$ .

输出:  $G$  的色数  $\chi(G)$ , 即正常  $\chi(G)$  点染色方案:  $V_1, V_2, \dots, V_{\chi(G)}$ .

第 0 步:  $A := \{G\}, K := \emptyset$

第 1 步: 任取  $A$  中一个图  $G$ , 任取  $G$  中两个不相邻顶点  $u, v$ , 转下步.

第 2 步: 给  $G$  添加边  $uv$ , 得图  $G'$ . 若  $G'$  是完全图, 则令  $K := K \cup \{G'\}$ , 转下步; 否则令  $A := A \cup \{G'\}$ , 转下步.

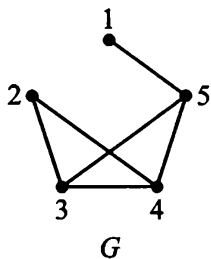
第 3 步: 将  $G$  中  $u, v$  两点粘合为一点, 并去掉所得的重边, 得图  $G''$ . 若  $G''$  是完全图, 则令  $K := K \cup \{G''\}$ , 转下步; 否则令  $A := A \cup \{G''\}$ , 转下步.

第 4 步: 令  $A := A - \{G\}$ . 若  $A \neq \emptyset$ , 转第 1 步. 否则, 转下步.

第 5 步: 求出  $K$  中各完全图的最小阶, 即为  $\chi(G)$ . 在  $K$  中取达到该最小阶的一个完全图, 其各顶点对应的粘合点集合  $V_1, V_2, \dots, V_{\chi(G)}$ , 便是  $G$  的正常  $\chi(G)$  点染色划分.

根据定理 6.5.2, 添边粘合算法的运行结果是顶点染色问题的精确解. 但添边粘合法的执行过程是一棵二叉树, 因此其计算复杂性是指数阶的.

**例 6.5.2** 求下图  $G$  的色数, 及顶点正常  $\chi(G)$  染色.



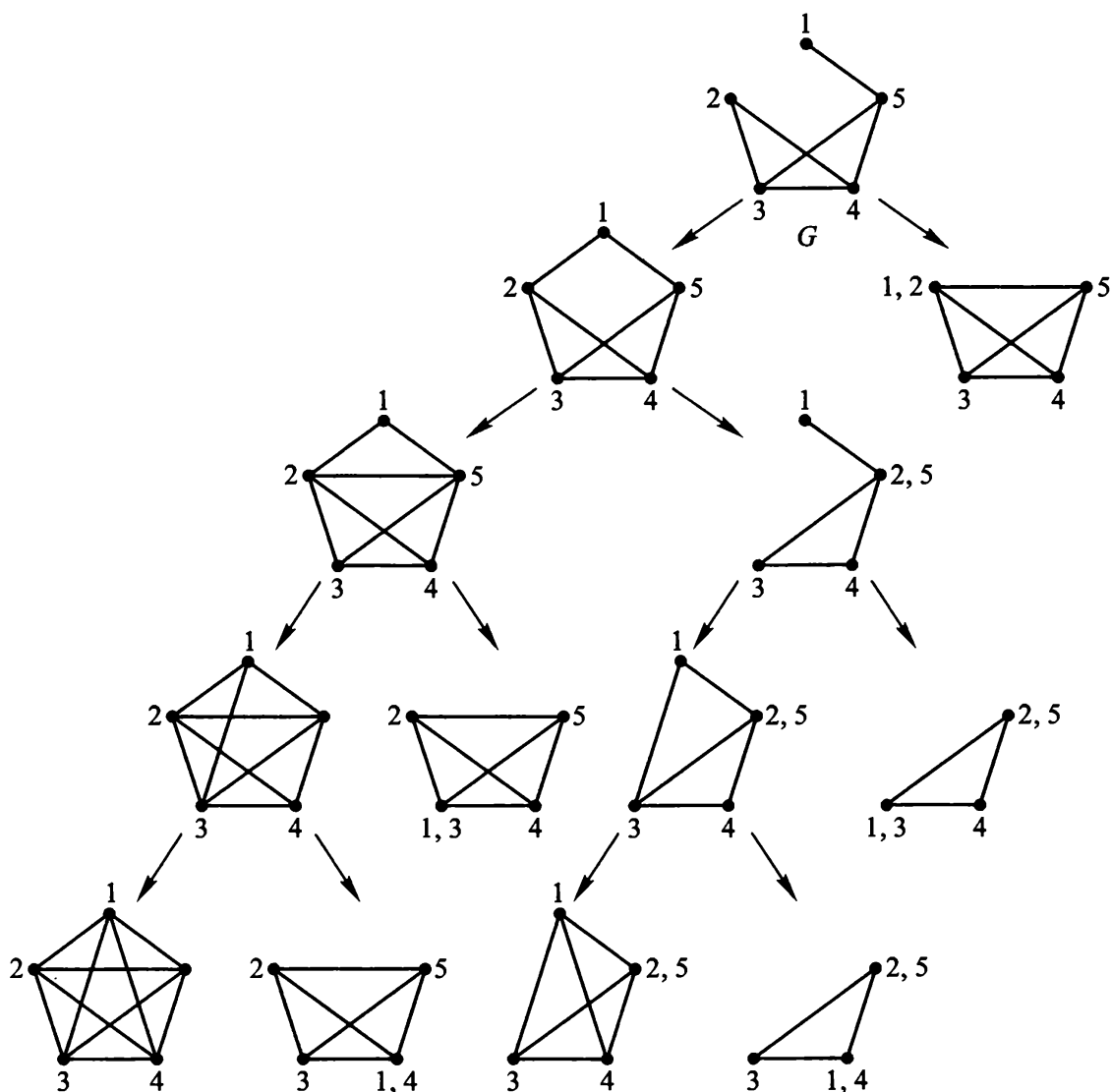
**解** 添边和粘合运算过程如下页图形所示.

在计算过程中, 原图 1、2 顶点连边和粘合, 得到第一层的两个图; 其中第一个图的 2、5 顶点连边和粘合, 得到第二层的两个图; 对它们的 1、3 顶点连边和粘合, 得到第三层的四个图; 其中第一个图和第三个图都对 1、4 顶点连边和粘合, 得到最后一层图. 这棵二叉树的每个叶顶点都是完全图, 故添边粘合过程结束. 得到的最小阶完全图为 3 阶; 因此原图  $G$  的色数为  $\chi(G) = 3$ .

由第三层和第四层所得的两个 3 阶完全图的粘合点情况, 得到原图  $G$  的两种正常 3 染色划分:  $V_1 = \{1, 3\}, V_2 = \{2, 5\}, V_3 = \{4\}$  和  $V_1 = \{1, 4\}, V_2 = \{2, 5\}, V_3 = \{3\}$ .

### 算法 2 规范染色法 (极大独立集法)

设  $\varphi = \{V_1, V_2, \dots, V_k\}$  是图  $G$  的点  $k$  染色. 若  $V_1$  是  $G$  的极大独立集,  $V_2$  是  $G \setminus V_1$  的极大独立集,  $V_3$  是  $G \setminus (V_1 \cup V_2)$  的极大独立集,  $\dots$ , 一般地,  $V_i$  是  $G \setminus \bigcup_{j=1}^{i-1} V_j$  的极大独立集 ( $i = 2, 3, \dots, k$ ), 则称  $\varphi$  是图  $G$  的规范  $k$  染色 (canonical  $k$ -colouring).



**定理 6.5.3** 图  $G$  是顶点可  $k$  正常染色的当且仅当  $G$  存在规范  $k$  染色.

**证明** 充分性是显然的.

**必要性** 设  $\{V_1, V_2, \dots, V_k\}$  是图  $G$  的一个顶点正常  $k$  染色, 则  $V_1, V_2, \dots, V_k$  都是  $G$  的独立集. 若  $V_1$  不是  $G$  的极大独立集, 则可从  $V \setminus V_1$  中调一些顶点进入  $V_1$ , 使  $V_1$  扩大成  $G$  的极大独立集  $V_1^{(1)}$ ,  $V_2, V_3, \dots, V_k$  变成  $V_2^{(1)}, V_3^{(1)}, \dots, V_k^{(1)}$ . 然后考虑  $V_2^{(1)}$ . 若它不是  $G \setminus V_1^{(1)}$  的极大独立集, 则可从  $V \setminus (V_1^{(1)} \cup V_2^{(1)})$  中调一些顶点进入  $V_2^{(1)}$ , 将其扩充成  $G \setminus V_1^{(1)}$  的极大独立集  $V_2^{(2)}, V_2^{(1)}, V_3^{(1)}, \dots, V_k^{(1)}$  变成  $V_2^{(2)}, V_3^{(2)}, \dots, V_k^{(2)}$ . 如此类推, 最后可得图  $G$  的规范  $k$  染色  $V_1^{(1)}, V_2^{(2)}, \dots, V_k^{(k)}$ . 证毕.

**规范染色法基本思想:** 求  $G$  的极大点独立集  $V_1$ , 将其顶点都染上色 1; 再求  $G \setminus V_1$  的极大点独立集  $V_2$ , 将其顶点都染上色 2; 如此类推, 直至染完所有顶点.

**规范染色法步骤**

输入: 图  $G$ .

输出:  $G$  的顶点正常色划分 (独立集划分):  $V_1, V_2, \dots, V_k$ .

第 0 步:  $S = V, i := 1$ .

第 1 步: 任取  $x \in S, V_i := \{x\}, T := (S - \{x\}) \cap \bar{N}(x)$ , 转下步.

第 2 步: 若  $T = \phi$ , 输出  $V_i$ , 并令  $S = S - V_i$ , 转第 4 步; 否则, 转第 3 步.

第 3 步: 任取  $y \in T, V_i = V_i \cup \{y\}, T := (T - \{y\}) \cap \bar{N}(y)$ , 转第 2 步.

第 4 步: 若  $S = \phi$ , 停止; 否则, 令  $i := i + 1$ , 转第 1 步.

算法第 1 步至第 4 步反复循环, 第  $i$  次循环得到极大独立集  $V_i$ . 在第  $i$  次循环中, 第 2 步到第 3 步反复循环向  $V_i$  中添加不相邻顶点, 使其成为极大独立集. 在算法中,  $\bar{N}(x)$  表示顶点  $x$  的不相邻顶点集. 集合  $S$  表示除了已得到的极大独立集之外, 剩余的顶点集.  $T$  记录在形成极大独立集  $V_i$  的过程中, 尚未进入  $V_1, V_2, \dots, V_i$  且与已进入  $V_i$  的顶点不相邻的顶点之集. 算法结束时, 得到图  $G$  的顶点正常色划分 (独立集划分):  $V_1, V_2, \dots, V_k$ .

注 (1) 这是一种贪婪算法.

(2) 规范  $k$  染色一般不唯一.

(3) 运行一次这个算法能得到图  $G$  的一种正常顶点染色, 但它所用的颜色数未必就是  $\chi(G)$ . 因此该算法仅是一个近似算法. 如下图, 其色数是 3 (图 1), 但按规范染色法得到的一种染色需使用 4 种颜色 (图 2), 图 2 中顶点的标号既表示颜色编号, 也表示算法产生的极大独立集的次序.

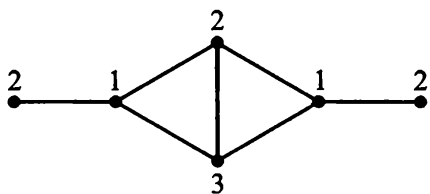


图 1

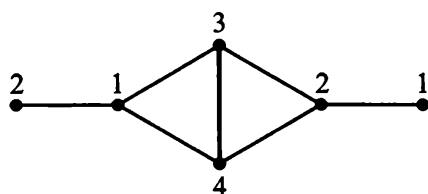


图 2

由于算法每执行一轮第 1 步至第 4 步大循环, 便产生一个极大独立集, 尚未染色的每个顶点要么在该独立集中, 要么至少与该独立集中一个点相邻. 因此, 每产生一个极大独立集, 顶点剩余度数便至少减 1 (顶点剩余度数是指从  $G$  中去掉所产生的极大独立集及其关联的边后, 剩余图中顶点的度). 可见, 大循环至多执行  $\Delta(G)$  次后, 尚未染色顶点的剩余度数便成为 0, 即剩余的顶点互不相邻, 可用同一种颜色染色. 这表明规范染色法最坏情况下需使用  $\Delta(G) + 1$  种色.

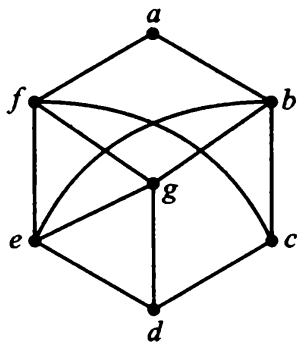
在第  $i$  轮大循环中, 至多有  $\nu - i$  个顶点参与运算 (即  $|S| \leq \nu - i$ ), 对这些点反复执行第 2 步至第 3 步小循环, 每次小循环的主要计算量在于计算集合  $T := (T - \{y\}) \cap \bar{N}(y)$ , 即在  $T - \{y\}$  中检查  $y$  的不相邻点, 这至多需要  $\nu - i - 1$  次判断. 因此该近似算法的计算复杂度为  $O\left(\sum_{i=1}^{\nu-1} (\nu - i - 1)\right) = O(\nu^2)$ .

由定理 6.5.3,  $G$  可用  $\chi(G)$  种颜色正常顶点染色, 则必存在  $\chi(G)$  规范染色. 因此只需从一切规范染色中选出颜色数最少者, 即可求得  $\chi(G)$ . 但由于需要枚举所有规范

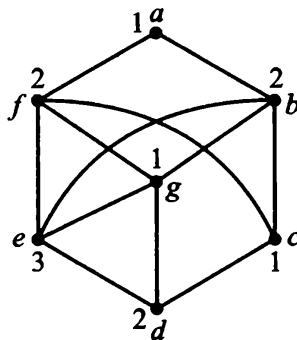


染色才能确定出  $\chi(G)$ , 故用该方法求  $\chi(G)$  是不实际的, 它只适用于阶数较小的图.

**例 6.5.3** 利用规范染色法对下图  $G$  进行顶点正常染色.



**解**  $G$  的一个极大独立集为  $V_1 = \{a, g, c\}$ ,  $G \setminus V_1$  的一个极大独立集为  $V_2 = \{b, d, f\}$ ,  $G \setminus (V_1 \cup V_2)$  的一个极大独立集为  $V_3 = \{e\}$ . 故  $G$  可正常 3 顶点染色,  $\chi(G) \leq 3$ . 相应的顶点 3 正常染色如下所示:



**注** 对本题所给的图  $G$ , 上述过程恰好得到  $\chi(G)$ . 事实上, 因  $G$  含有  $K_3$ , 因此正常顶点染色至少需要 3 种颜色. 从而  $\chi(G) = 3$ .

### 算法 3 顺序染色法

**顺序染色法基本思想:** 将图  $G$  的顶点排一个顺序  $v_1, v_2, \dots, v_\nu$ . 先给顶点  $v_1$  染上色  $c_1$ . 对顶点  $v_2$ , 从颜色集合  $\{c_1, c_2, c_3, \dots\}$  中选取未被  $v_2$  的邻点使用的编号最小的色来染它. 一般地, 对顶点  $v_i$ , 从颜色集合  $\{c_1, c_2, c_3, \dots\}$  中选取未被  $v_i$  的邻点使用的编号最小的色给其染色,  $i = 1, 2, \dots, n$ .

**顺序染色法步骤:**

**输入:** 图  $G$ .

**输出:**  $G$  的顶点正常色划分 (独立集划分):  $V_1, V_2, \dots, V_k$ .

**第 0 步:** 将  $G$  的所有顶点排序  $V = \{v_1, v_2, \dots, v_\nu\}$ .

**第 1 步:** 令  $i := 1$ .

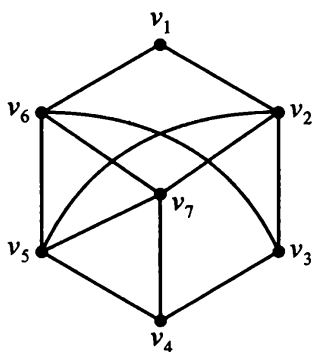
**第 2 步:** 令  $c := 1$ .

**第 3 步:** 若  $v_i$  的所有邻点都未使用颜色  $c$ , 则用  $c$  染  $v_i$ , 转第 5 步; 否则, 转下步.

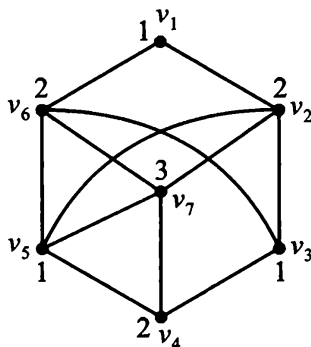
**第 4 步:**  $c := c + 1$ , 转第 3 步.

**第 5 步:** 若  $i < \nu$ , 则  $i := i + 1$ , 转第 2 步; 否则, 输出染色结果, 停止.

例 6.5.4 对下图  $G$ , 利用顺序染色法对其进行顶点正常染色.



解 任取顶点的一种顺序如图所示:  $v_1, v_2, \dots, v_7$ . 先考虑  $v_1$ , 由于它的所有邻点尚未染任何色, 故按算法, 用色 1 给  $v_1$  染色. 再考虑  $v_2$ , 其所有邻点尚未使用的最小色为颜色 2, 故  $v_2$  应染色 2. 对于  $v_3$ , 其邻点尚未使用的最小色号为颜色 1, 故应染色 1; 同样, 按照算法规则,  $v_4$  应染色 2,  $v_5$  应染色 1,  $v_6$  应染色 2. 最后,  $v_7$  既有染 1 色邻点又有染 2 色的邻点, 但无染颜色 3 的点, 因此染色 3. 最终得到原图的一个顶点 3 正常染色如下所示.



注 (1) 顺序染色法是多项式时间算法, 计算复杂度为  $O(v^2)$ .

(2) 顺序染色法是一种近似算法, 其算法结果 (所用颜色数) 与顶点的标号顺序有密切关系. 比如, 对完全二部图  $K_{n,n} = (X, Y)$ , 若顶点标号顺序为  $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n$ , 则算法结果只用两种颜色, 是最优解. 但若顶点顺序为  $x_1, y_1, x_2, y_2, \dots, x_n, y_n$ , 则算法结果需  $n$  种颜色. 此时算法近似比  $\frac{\tilde{c}}{c^*} = \frac{n}{2}$ , 当  $n \rightarrow \infty$  时, 近似比趋于无穷大!

#### 算法 4 最大度优先算法

**最大度优先算法思想:** 将图的所有顶点按度由大到小排序, 将颜色标号. 按颜色标号由小到大依次使用颜色给顶点染色. 每使用一种色时, 用这种色给尽可能多的不相邻顶点染色. 每次染色时若有多个顶点可选, 优先选择度较大的顶点. 可以看出, 这个算法综合了规范染色法和顺序染色法两方面的考虑.

**最大度优先算法步骤:**

输入: 图  $G$ .

输出:  $G$  的顶点正常色划分 (独立集划分):  $V_1, V_2, \dots, V_k$ .

第 1 步: 将图  $G$  的顶点按度数递减的顺序排列, 设为  $v_1, v_2, \dots, v_\nu$ . 令  $S := \phi$

第 2 步: 用第 1 种颜色给  $v_1$  染色, 令  $V_1 := \{v_1\}, i := 1$ .

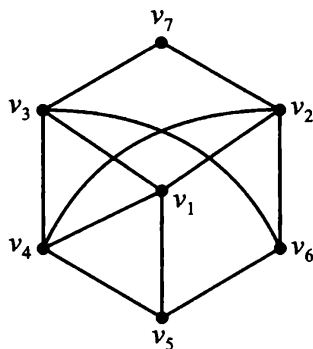
第 3 步: 若存在与  $V_i$  中所有点都不相邻的未染色顶点, 则按顺序找出第一个与  $V_i$  中所有点都不相邻的未染色顶点  $v$ , 转下步; 否则, 输出  $V_i$ , 并令  $S := S \cup V_i$ , 转第 5 步.

第 4 步: 给  $v$  染上第  $i$  种色, 令  $V_i := V_i \cup \{v\}$ , 转第 3 步.

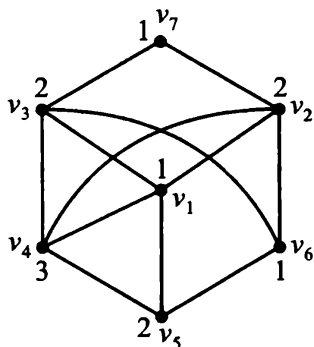
第 5 步: 若  $|S| = \nu$ , 停止; 否则  $i := i + 1$ , 转下步.

第 6 步: 取尚未染色的顶点中编号最小的顶点  $v$ , 给  $v$  染上色  $i$ . 令  $V_i := \{v\}$ , 转第 3 步.

例 6.5.5 利用最优先算法对下图  $G$  进行顶点正常染色.



解 顶点按度由大到小排序为  $v_1, v_2, v_3, v_4, v_5, v_6, v_7$ . 先用色 1 给  $v_1$  染色,  $V_1 := \{v_1\}$ .  $v_6, v_7$  与  $v_1$  不相邻且尚未染色, 按算法, 选择下标较小的  $v_6$ , 给其染颜色 1,  $V_1 := \{v_1, v_6\}$ .  $v_7$  仍未染色且与  $V_1$  中各点不相邻, 用颜色 1 给  $v_7$  染色,  $V_1 := \{v_1, v_6, v_7\}$ . 目前所有未染色点都与  $V_1$  相邻, 按算法, 再考虑使用颜色 2 来染色, 选择尚未染色的点中下标最小的, 当前为  $v_2$ , 染上色 2,  $V_2 := \{v_2\}$ , 点  $v_3, v_5$  尚未染色且与  $v_2$  不相邻, 按算法, 选择  $v_3$ , 给其染颜色 2,  $V_2 := \{v_2, v_3\}$ .  $v_5$  仍未染色且与  $V_2$  中各点不相邻, 用颜色 2 给  $v_5$  染色,  $V_2 := \{v_2, v_3, v_5\}$ . 目前未染色顶点只有  $v_4$ , 它与  $V_2$  相邻, 按算法, 再使用颜色 3 染色给  $v_4$  染色,  $V_3 := \{v_4\}$ . 此时所有顶点都得到染色, 其中  $V_1 = \{v_1, v_6, v_7\}, V_2 = \{v_2, v_3, v_5\}, V_3 = \{v_4\}$ . 染色结果如下图所示.



最大度优先算法是在规范染色法和顺序染色法基础上的一种改进, 因此, 该算法的性能一般优于前两种方法. 但是最大度优先算法也只是一近似算法 (习题 6.52), 最坏情况下需使用  $\Delta(G) + 1$  种颜色. 与规范染色法类似可知, 最大度优先算法的计算复杂度为  $O(\nu^2)$ .

对最大度优先算法的一种修改是: 在每启用一种新的颜色时 (即上述算法开始新一轮第 6 步至第 3 步循环时), 删去已染色的顶点, 在剩余图中对顶点按度的大小重新排序, 然后继续算法第 6 步. 基于最大度优先算法, 文献 [59] 提出了一种最大色度优先算法, 其中一个顶点的色度是指该点的邻点已使用过的颜色数目.

图的染色是内容十分丰富而又应用广泛的一个图论专题. 关于染色问题的其他选题, 可进一步参阅文献 [165]~[229]. 其中 [165]~[167] 研究有约束的图染色问题, [168]~[169] 研究赋权染色问题, [170]~[171] 涉及图的色幂 (coloring powers), [172]~[176] 与随机图染色和随机算法有关, [177]~[181] 研究全染色和全色数 (total-coloring), [182]~[185] 研究表色数 (list-coloring), [186]~[193] 涉及图的和染色 (sum coloring) 及相关算法的性能, [194]~[215] 研究其他与图的染色有关的问题, [216]~[229] 给出了染色问题在通信网络优化、地理信息系统、寄存器配置等方面的应用.

## 习题六

- 6.1 证明  $n \neq 0$  时, (1)  $\chi'(K_{2n+1}) = 2n + 1 = \Delta(K_{2n+1})$ ;  
(2)  $\chi'(K_{2n}) = n - 1 = \Delta(K_{2n})$ .
- 6.2 找出一个适当的边染色以证明  $\chi'(K_{m,n}) = \Delta(K_{m,n})$ .
- 6.3 轮图  $W_{1,n}$  是由一个点与长为  $n$  的圈上所有顶点相连形成的图, 证明: 若  $n \geq 4$ , 则  $\chi'(W_{1,n}) = \Delta(W_{1,n}) = n$ .
- 6.4 证明: 对 Petersen 图  $G$ ,  $\chi'(G) = 4$ .
- 6.5 (1) 证明: 对于最大度为  $\Delta$  的任何二部图  $G$ , 存在着包含  $G$  的  $\Delta$  正则简单二部图.  
(2) 若  $G$  是最大度为  $\Delta$  的无环图, 则存在  $\Delta$  正则图以  $G$  为子图.
- 6.6 试用  $\varepsilon$  和  $\alpha'(G)$  给出  $\chi'(G)$  的一个下界.
- 6.7 设  $G$  是一个无环边的图,  $\Delta(G) = 3$  且  $G$  包含一个生成二部子图  $H$ , 使得  $\forall v \in V(G)$ ,  $d_H(v) \geq \frac{1}{2}d_G(v)$ , 试证明 (不使用 Vizing 定理):  $\chi'(G) \leq 4$ .
- 6.8 设  $G$  是二部图, 且  $\delta(G) > 0$ , 证明: 图  $G$  有一种  $\delta(G)$  边染色, 使得所有的色都在  $G$  的每个顶点上出现.
- 6.9 证明: 每个 3 正则 Hamilton 图都有正常 3 边着色.
- 6.10 证明:

(1) 若  $G$  是简单图, 且  $\nu(G) = 2n + 1, \varepsilon(G) > n\Delta(G)$ , 则  $\chi'(G) = \Delta(G) + 1$ .

(2) 利用上述结果证明:

(a) 若  $G$  是一个由顶点数为偶数的  $k$ -正则简单图 ( $k > 1$ ) 剖分一边后所得之图 (剖分一条边是指在该边上加入一个新顶点), 则  $\chi'(G) = \Delta(G) + 1$ ;

(b) 若  $G$  是一个由顶点数为奇数的  $k$  正则简单图去掉少于  $\frac{k}{2}$  条边后得到的图 ( $k > 1$ ), 则  $\chi'(G) = \Delta(G) + 1$ .

6.11 证明: 若  $G$  是非空正则简单图, 且  $\nu(G)$  为奇数, 则  $\chi'(G) = \Delta(G) + 1$ .

6.12 证明: 若  $G$  是有割点的正则简单图, 则  $\chi'(G) = \Delta(G) + 1$ .

6.13 设图  $G$  的边独立数是  $\alpha'(G)$ . 证明: 如果  $\varepsilon(G) > \Delta(G)\alpha'(G)$ , 则  $G$  是第二类图. 利用此结果证明: 如果  $\varepsilon(G) > \Delta(G) \cdot \lfloor \frac{\nu}{2} \rfloor$ , 则图  $G$  是第二类图.

6.14 证明: 对不是奇圈的连通图, 如果它的所有圈有相同的奇偶性, 则  $\chi'(G) = \Delta(G)$ .

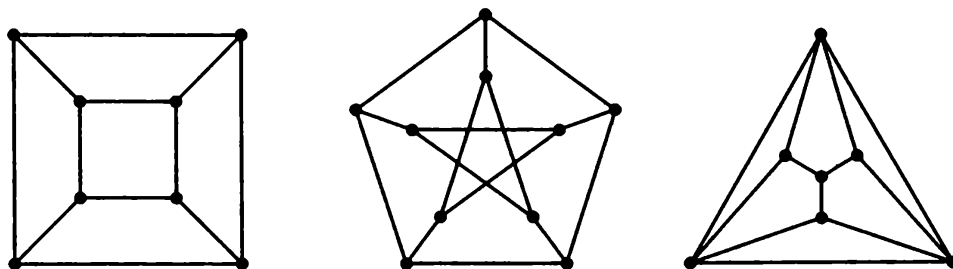
6.15 设  $G$  为  $\nu$  阶简单图, 证明: 若  $G$  中恰有一个点的度达到  $\nu - 1$ , 则  $\chi'(G) = \Delta(G) = \nu - 1$ .

6.16 证明或否定: 如果  $G_1$  和  $G_2$  是第一类图, 且  $G_1 \subseteq H \subseteq G_2$ , 则  $H$  是第一类图.

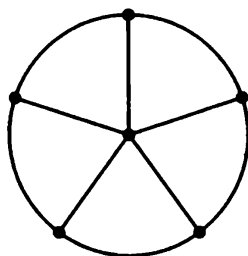
6.17 设  $G$  是简单图,  $x$  与  $y$  是  $G$  中两个不相邻的顶点,  $\pi$  是  $G$  的一个正常  $k$  边染色. 证明: 若在该染色下  $x, y$  以及  $x$  的所有邻点均有颜色未在其上体现, 则  $G + xy$  也可  $k$  边正常染色.

6.18 利用上题的结果证明 Vizing 定理. 并证明: 若无孤立点的简单图  $G$  恰有一个度为  $\Delta(G)$  的顶点, 或  $G$  恰有两个度为  $\Delta(G)$  的顶点但这两个顶点相邻, 则  $G$  是第一类图, 即  $\chi'(G) = \Delta(G)$ .

6.19 求下列图的色数.



6.20 轮图  $W_{1,n}$  是由一个点与长为  $n$  的圈上所有顶点相连形成的图,  $W_{1,5}$  如下所示. 证明  $\chi(W_{1,5}) = 4$ .



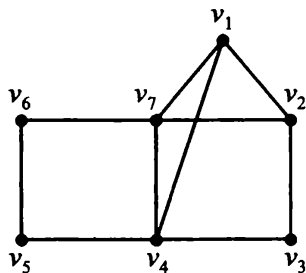
6.21 证明:

- (1)  $\chi(G) = 1$  当且仅当  $G$  是空图但非零图, 即  $G = \overline{K}_\nu$  且  $\nu \geq 1$ ;
- (2)  $\chi(G) = 2$  当且仅当  $G$  是非空二部图;
- (3)  $\chi(G) = \nu(G)$  当且仅当  $G \supseteq K_\nu$ ;
- (4)  $\chi(C_{2n+1}) = 3$ ;
- (5)  $\chi(G) \geq 3$  当且仅当  $G$  含有奇圈.

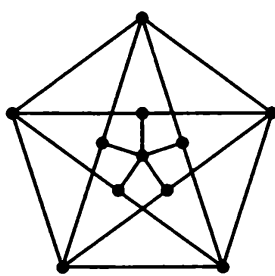
6.22 证明:

- (1)  $G$  是临界 1 色图当且仅当  $G$  是  $K_1$ ;
- (2)  $G$  是临界 2 色图当且仅当  $G$  是  $K_2$ ;
- (3)  $G$  是临界 3 色图当且仅当  $G$  是奇圈;
- (4) 任何  $k$  色图都包含临界  $k$  色子图;
- (5) 每个色临界图都是连通的简单图.

6.23 下图是不是临界 3 色图? 如果是, 给出证明: 如果不是, 求出它的一个临界 3 色子图.



6.24 证明: 下列 Grötzsch 图是临界 4 色图.



Grötzsch 图

6.25 设  $H$  是图  $G$  的子图, 证明  $\chi(G) \geq \chi(H)$ .

6.26 设  $G_1, G_2, \dots, G_k$  是图  $G$  的所有块, 证明:  $\chi(G) = \max_i \{\chi(G_i)\}$ .

6.27 给出一个图  $G$  使得它有顶点  $v$  满足  $\chi(G - v) < \chi(G)$ , 且  $\chi(\overline{G} - v) < \chi(\overline{G})$ .

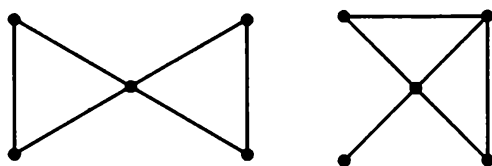
6.28 证明或否定: 任何  $k$  色图  $G$  有一个正常  $k$  染色使得某个色类含  $\alpha(G)$  个顶点.

6.29 证明或否定: 对任何连通图  $G$ , 有  $\chi(G) \leq 1 + \frac{2\varepsilon}{\nu}$ . ( $\frac{2\varepsilon}{\nu}$  是图  $G$  的顶点平均度).

6.30 两个图  $G$  与  $H$  的并  $G \cup H$  是以图  $G$  和  $H$  的顶点集的并作为顶点集, 且以  $G$  和  $H$  的边集的并作为边集形成的图. 证明或否定:  $\chi(G \cup H) \leq \chi(G) + \chi(H)$ .

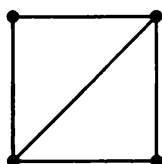
- 6.31 设  $G$  和  $H$  不相交 (即无公共点),  $G + H$  表示  $G$  和  $H$  的不交并 (即将  $G$  和  $H$  作为分支放在一起组成的图). 证明  $\chi(G + H) = \max\{\chi(G), \chi(H)\}$ .
- 6.32 图  $G$  和  $H$  的联  $G \vee H$  表示将  $G$  的每个顶点与  $H$  的每个顶点间都连边后形成的图. 证明  $\chi(G \vee H) = \chi(G) + \chi(H)$ .
- 6.33 设  $G$  是完全图  $K_r$  与长为 5 的圈  $C_5$  的联:  $G = C_5 \vee K_r$ , 证明:  $\chi(G) > \omega(G)$ . 并证明  $G$  是一个色临界图.
- 6.34 设  $I$  是色临界图  $G$  的任一个点独立集, 证明:  $\chi(G - I) = \chi(G) - 1$ .
- 6.35 证明: (1) 若  $G$  的任二奇圈都有公共顶点, 则  $\chi(G) \leq 5$ ;  
 (2) 若  $G$  的任二圈都没有公共顶点, 则  $\chi(G) \leq 3$ ;  
 (3) 若  $G$  中的奇圈个数不超过 2 个, 则  $\chi(G) \leq 3$ .
- 6.36 证明: 若  $G$  是二部图, 则  $\chi(\overline{G}) = \omega(\overline{G})$ .
- 6.37 证明: 若  $G$  中最长奇圈的长度为  $m \geq 3$ , 则  $\chi(G) \leq m + 1$ .
- 6.38 (1) 设简单图  $G$  的顶点集为  $\{v_1, v_2, \dots, v_\nu\}$ , 证明: 如果每个  $v_j$  最多只与  $\{v_1, v_2, \dots, v_{j-1}\}$  中的  $k$  个顶点相邻 ( $2 \leq j \leq \nu$ ), 则有  $\chi(G) \leq k + 1$ ;  
 (2) 利用 (1) 证明: 对任何简单图  $G$  都有  $\chi(G) \leq \Delta(G) + 1$ ;  
 (3) 利用 (1) 证明: 对任何简单图  $G$  都有  $\chi(G) \leq \max_H \delta(H) + 1$ , 其中  $H$  取遍  $G$  的所有点导出子图;  
 (4) 利用 (3) 证明: 若  $G$  是连通简单图但不是正则图, 则  $\chi(G) \leq \Delta(G)$ ;  
 (5) 利用 (4) 证明: 若  $G$  是连通的正则简单图且含有 1 顶点割, 则  $\chi(G) \leq \Delta(G)$ ;  
 (6) 利用 (4) 证明: 若  $G$  是连通的  $k$  正则简单图 ( $k \geq 3$ ), 且  $G$  有 2 顶点割, 则  $\chi(G) \leq \Delta(G)$ ;  
 (7) 利用上述结论证明 Brooks 定理: 设  $G$  是连通简单图, 且既不是奇圈也不是完全图, 则  $\chi(G) \leq \Delta(G)$ .
- 6.39 设  $C = (V_1, V_2, \dots, V_\chi)$  为简单图  $G$  的一个顶点正常  $\chi$  染色, 其中  $\chi$  是  $G$  的色数.  
 (1) 证明: 每个  $V_i$  中至少存在一个顶点  $v_i$ , 它与其他每个  $V_j$  中至少一个顶点相邻.  
 (2) 利用 (1) 的结论证明: 图  $G$  中至少有  $\chi$  个度大于或等于  $\chi - 1$  的顶点, 并由此证明  $\chi \leq \Delta(G) + 1$ .
- 6.40 将简单图  $G$  的顶点任意排序:  $v_1, v_2, \dots, v_\nu$ , 记  $d(v_i) = d_i$ ,  $i = 1, 2, \dots, \nu$ . 证明:  $\chi(G) \leq \max_i \min\{i, d_i + 1\}$ .
- 6.41 设  $G$  是一个简单图, 证明:  
 (1)  $2\sqrt{\nu} \leq \chi(G) + \chi(\overline{G}) \leq \nu + 1$ ; (2)  $\nu \leq \chi(G) \cdot \chi(\overline{G}) \leq \left(\frac{\nu + 1}{2}\right)^2$ .

6.42 求下列两图的色多项式.



6.43 证明: 若  $H = G \vee K_1$ , 则  $P(H, k) = k \cdot P(G, k - 1)$ . 利用这一结果证明: 对有  $n$  根辐条的轮图  $W_{1,n}$ ,  $P(W_{1,n}, k) = k(k - 2)^n + (-1)^n k(k - 2)$ .

6.44 利用树和  $C_4$  的色多项式结果, 求下图的色多项式.



6.45 证明:

(1) 若图  $G$  含有  $r$  个孤立顶点, 则色多项式  $P(G, k) = k^r \cdot P(G', k)$ , 其中  $G'$  是  $G$  去掉  $r$  个孤立顶点后所得的图;

(2) 若图  $G$  有重边, 用  $G_1$  表示  $G$  去掉重边后所得的图, 则  $P(G, k) = P(G_1, k)$ .

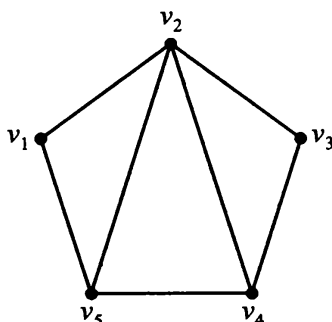
6.46 设  $v$  是图  $G$  的一个 1 度顶点, 利用上题结论证明:  $P(G, k) = (k - 1)P(G - v, k)$ .

6.47 设  $G$  是连通图, 证明:  $P(G, k) \leq k(k - 1)^{v-1}$ .

6.48 证明: 如果  $G$  至少含有一条边, 则其色多项式的系数之和等于零.

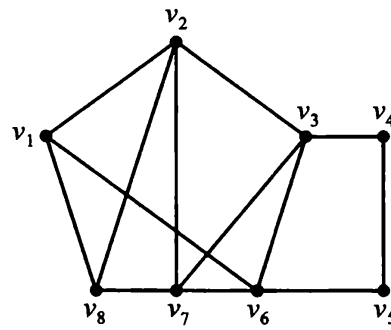
6.49 证明: Petersen 图不是完美图.

6.50 用添边粘合法求下图的色数, 并给出相应的点染色方案.



6.51 试分别用规范染色法和顺序染色法对下图进行正常点染色, 并用最大度优先法给出一种必须要用 4 种颜色的染法. 同时, 指出一种 3 正常点染色.





## 参考文献

- [1] D. König, Über Graphen und ihre Anwendung auf Determinantentheorie und mengenlehre, *Math. Ann.*, 77(1916), 453–465.
- [2] I. Holyer, The NP-completeness of edge-coloring, *SIAM J. Computing*, 10: 4(1981), 718–720.
- [3] A. Gibbons, *Algorithmic Graph Theory*, Cambridge University Press, Melbourne Sydney, 1985.
- [4] S. Fiorini, and R. J. Wilson, *Edge-Colorings of Graphs*, Pitman, New York, 1995. [Also see *Selected Topics in Graph Theory* (eds. L. W. Beineke, and R. J. Wilson), Academic Press, London, (1978) 103–126.]
- [5] L. D. Andersen, On edge-colourings of graphs, *Math. Scand.* 40(1977), 161–175.
- [6] N. Robertson, P. D. Seymour and R. Thomas, Tutte's edge-coloring conjecture, *J. Combin. Theory (Ser. B)*, 70(1997), 166–183.
- [7] A. Nadolski, The circular chromatic index of some class 2 graphs. *Discrete Mathematics*, 307(2007), 1447–1454.
- [8] D. R. Woodall, The average degree of an edge-chromatic critical graph. *Discrete Mathematics*, 308(2008), 803–819.
- [9] B. Reed,  $\omega$ ,  $\Delta$ , and  $\chi$ , *J. Graph Theory*, 27(1998), 177–212.
- [10] O. V. Borodin, and A. V. Kostochka, On an upper bound of the graph's chromatic number depending on the graph's degree and density, *J. Comb. Theory (B)*, 23(1977) 247–250.
- [11] B. Reed, A strengthening of Brooks' theorem, *J. Comb. Theory (B)*, 76(1999), 136–149.
- [12] J. Mycielski, Sur le coloriage des graphes, *Coll. Math.*, 3(1955), 161–162.

## Hajos 猜想和 Hadwigers 猜想

- [13] H. Hadwiger, Über eine Klassifikation der Streckenkomplexe, *Vierteljschr. Naturforsch. Gesellsch. Zürich*, 88(1943), 133–142.
- [14] K. Wagner, Beweis einer Abschwächung der Hadwiger-Vermutung, *Math. Ann.*, 153 (1964), 139–141.

- [15] G. Hajós, Über eine Konstruktion nicht  $n$ -färbbarer Graphen, *Wiss. Z. Martin-Luther-  
Univ. Halle-Wittenberg, Math. - Nat. Reihe.*, 10(1961), 116–117.
- [16] G. A. Dirac, A property of 4-chromatic graphs and some remarks on critical graphs,  
*Journal of London Mathematical Society*, 27(1952), 85–92.
- [17] J. A. Bondy, and U. S. Murty, *Graph theory with applications*, 1976, 有中译本 (图论及  
其应用. 吴望名等译. 北京: 科学出版社, 1987).
- [18] P. Catlin, Hajós' graph-coloring conjecture: variations and counterexamples, *Journal of  
Combinatorial Theory*, (Ser.B), 26(1979), 268–274.
- [19] P. Erdős, S. Fajtlowicz, On the conjecture of Hajós, *Combinatorica*, 1(1981), 141–143.
- [20] C. Thomassen, Some remarks on Hajós conjecture, *Journal of Combinatorial Theory*,  
(Ser.B), 93(2005), 95–105.
- [21] B. Mohar, Triangulations and the Hajós' conjecture, *The Electronic Journal of Combi-  
natorics*, 12(2005), #N15.
- [22] Xingxing Xu, F. Zickfeld, Reducing Hajós' coloring conjecture to 4-connected graphs,  
*Journal of Combinatorial Theory*, (Ser.B), 96(2006), 482–492.
- [23] C. Thomassen, Hajós' conjecture for line graphs, *Journal of Combinatorial Theory*,  
(Ser.B), 97(2007), 156–157.
- [24] I. T. Jakobsen, A. homomorphism theorem with applications to the conjecture of Had-  
wiger, *Studia Sci. Math. Hung.*, 6(1971)151–160.
- [25] B. Bollobás, P. Catlin, P. Erdős, Hadwiger's conjecture is true for almost every graph,  
*European Journal of Combinatorics*, 1 (1980) 195–199.
- [26] J. Mayer, Hadwiger's conjecture ( $k=6$ ): neighbour configurations of 6-vertices in  
contraction-critical graphs, *Discrete Mathematics*, 74(1989), 137–148.
- [27] A. E. Kezdy, P. J. McGuinness, Do  $3n-5$  edges suffice for a subdivision of  $K_5$ ? *Journal of  
Graph Theory*, 15(1991), 389–406.
- [28] N. Robertson, P. D. Seymour, R. Thomas, Hadwiger's conjecture for  $K_6$ -free graphs,  
*Combinatorica*, 13(1993), 279–361.
- [29] B. Toft, A. survey of Hadwiger's conjecture, *Congr. Numer.*, 115(1996), 249–283.
- [30] W. Mader,  $3n-5$  edges do force a subdivision of  $K_5$ , *Combinatorica*, 18(1998), 569–595.
- [31] A. Kotlov, Matchings and Hadwiger's conjecture, *Discrete Mathematics*, 244(2002), 241–  
252.
- [32] F. N. Abu-Khzam, M. A. Langston, Graph coloring and the immersion order, in *Pro-  
ceedings of COCOON 2003, Computin and Combinatorics* (Lecture Notes in Computer  
Science, Vol. 2697), Springer Berlin, 2003, pp394–403.
- [33] B. Reed, P. Seymour, Hadwiger's conjecture for line graphs, *European Journal of Com-  
binatorics*, 24(6)(2004), 73–76.

- [34] K. I. Kawarabayashi, B. Toft, Any 7-chromatic graph has  $K_7$  and  $K_{4,4}$  as a minor, *Combinatorica*, 25(2005), 327–353.
- [35] R. Naserasr, Y. Nigussie, On a new reformulation of Hadwiger's conjecture. *Discrete Mathematics*, 306(2006), 3136–3139
- [36] N. Belkale, S. Chandran, Hadwiger's conjecture for proper circular arc graphs, *arXiv.org*, <http://arxiv.org/abs/math/0605503v2>, Cornell University Library, 2006.
- [37] K. I. Kawarabayashi, On the connectivity of minimum and minimal counterexamples to Hadwiger's conjecture, *Journal of Combinatorial Theory*, (Ser.B), 97(2007), 144–150.
- [38] Deming Li, Mingju Liu, Hadwiger's conjecture for powers of cycles and their complements, *European Journal of Combinatorics*, 28(4)(2007), 1152–1155.
- [39] J. Blasiak, A. special case of Hadwiger's conjecture, *Journal of Combinatorial Theory*, (Ser.B), 97(2007), 1056–1073.
- [40] G. Fijavz, Hadwiger's conjecture for circular colorings of edge-weighted graphs, *Discrete Mathematics*, 307(2007), 402–408.
- [41] L. S. Chandran, N. Sivadasan, On the Hadwiger's conjecture for graph products. *Discrete Mathematics*, 307(2007), 266–273.

#### 与边染色、点染色及色多项式有关的著作和综述

- [42] N. Christofides, *Graph Theory: An Algorithmic Approach*, Academic Press, New York, (1990), 58–78.
- [43] R. Gould, *Graph Theory*, Benjamin Cummings, Menlo, Park, CA(1988).
- [44] G. Chartrand, O. R. Oellermann, *Applied and Algorithmic Graph Theory*, McGraw-Hill, Inc., New York, (1993), 283–310.
- [45] M. R. Garey, and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, New York, 1979.
- [46] D. S. Hochbaum, *Approximation Algorithms for NP-hard Problems*, International Thomson Publishing Inc., 1997 (影印本: NP 难解问题的近似算法, 世界图书出版公司).
- [47] S. Even, *Algorithmic Combinatorics*, Macmillan, New York, 1973.
- [48] D. W. Matula, G. Marble, and J. D. Isaacson, Graph colouring algorithms, in *Graph Theory and Computing* (Read Ed.), Academic Press, New York, (1972) 109–122.
- [49] Reinhard Diestel, *Graph Theory*, Springer, 1997.
- [50] Tommy R. Jensen and Bjarne Toft, *Graph Coloring Problems*, Wiley-Interscience, New York, 1995.
- [51] B. Toft, 75 graph-colouring problems, in *Graph Colourings* (R. Nelson and R. J. Wilson, Eds.), Wiley, New York, (1990), pp9–35.
- [52] <http://www.nada.kth.se/~viggo/problemlist/>
- [53] <http://web.cs.ualberta.ca/~joe/Coloring/#Links.other>

- [54] D. B. West, *Introduction to Graph Theory* (second edition), Prentice-Hall, Inc., 2001. (中译本: 图论导引. 李建中, 骆吉周译. 北京: 机械工业出版社, 2006).
- [55] W. T. Tutte, *Graph Theory*, Cambridge University Press, 2001.
- [56] R. C. Read and W. T. Tutte, *Chromatic polynomials*, Selected Topics in Graph Theory 3, (eds. L. W. Beineke, and R. J. Wilson), Academic Press, New York, (1988) 15–42.
- [57] D. Reinhard, *Graph Theory* (second edition), Springer-Verlag New York, 2000.

#### 完美图

- [58] A. Schrijver, *Combinatorial Optimization Polyhedra and Efficiency*, New York, NY: Springer-Verlag, 2003.
- [59] F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969. (中译本: 图论. 李慰萱译. 上海: 上海科技出版社, 1980).
- [60] R. L. Hemminger, and L. W. Beineke, Line graphs and line digraphs, *Selected Topics in Graph Theory* (L. W. Beineke, and R. J. Wilson editors), Academic Press, London, (1983), 271–306.
- [61] L. Lovász, Normal hypergraphs and the perfect graph conjecture, *Discrete Mathematics*, 2(1972), 253–267.
- [62] L. Lovász, A characterization of perfect graphs, *J. Comb. Theory (B)*, 13(1972), 95–98.
- [63] D. R. Fulkerson, Blocking and anti-blocking pairs of polyhedra, *Math. Programming*, 1(1971), 168–194.
- [64] <http://www.cs.rytgers.edu/~chvatal/perfect/spgt.html>
- [65] C. Berge, *Färbung von Graphen deren sämtliche bzw. Deren ungerade Kreise starr sind* (Zusammenfassung), *Wissenschaftliche Zeitschrift, Martin Luther Universität Halle-Wittenberg, Mathematisch-Naturwissenschaftliche Reihe*, (1961), 114–115.
- [66] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas, Progress on perfect graphs, *Mathematical Programming Ser. B*, 97(2003), 405–422.
- [67] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas, The strong perfect graph theorem, *Annals of Mathematics*, 164:1(2006), 51–229.
- [68] M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour, K. Vušković, Recognizing Berge graphs, *Combinatorica*, 25(2005), 143–186.
- [69] G. Cornuéjols, X. Liu, and K. Vušković, A polynomial algorithm for recognizing perfect graphs, *Proceeding of 44<sup>th</sup> FOCS* (2003), 20–27.
- [70] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, 1980
- [71] L. Lovász, Perfect graphs, in *Selected Topics in Graph Theory II* (L. W. Beineke, and R. J. Wilson eds.), Academic Press, London, (1983), 55–87.
- [72] C. Berge, and V. Chvátal, Topics on Perfect Graphs, *Ann. Discr. Math.* 21, North Holland, 1984.

- [73] J. L. Ramírez Alfonsín, and B. A. Reed, *Perfect graphs*, J. Wiley and Sons, Chichester, 2001.
- [74] G. Cornuéjols, The strong perfect graph conjecture, in *Proceedings of the ICM*, Beijing, China, 2002.

### 边染色算法

- [75] O. Terada, and T. Nishizeki, Approximate algorithms for the edge-coloring of graphs, *Trans. Inst. Electron. Commun. Engr. Japan J65-D*, 11(1982), 1382–1389.
- [76] D. Leven and Z. Galil, NP completeness of finding the chromatic index of regular graphs, *J. Algorithms*, 4(1983) 35–44.
- [77] M. K. Goldberg, Edge-colorings of multigraphs: recoloring technique, *J. Graph Theory*, 8(1984), 123–127.
- [78] D. S. Hochbaum, T. Nishizeki and D. B. Shmoys, Better than “Best Possible” algorithm to edge color multi graphs, *Journal of Algorithms*, 7(1986), 79–104.
- [79] H. J. Karloff and D. B. Shmoys, Efficient parallel algorithms for edge coloring problems. *J. Algorithms* 8 (1987), 39–52.
- [80] M. Chrobak, and T. Nishizeki, Improved edge-coloring algorithms for planar graphs, *J. Algorithms*, 11(1990), 102–116.
- [81] T. Nishizeki and K. Kashiwagi, On the 1.1 edge-coloring of multigraphs, *SIAM J. Disc. Math.* , 3(1990), 391–410.
- [82] B. Berger and J. Rompel, Simulating  $(\log^c n)$ -wise independence in NC. *J. ACM* 38 (1991), 1026–1046.
- [83] J. Misra and D. Gries, A constructive proof of Vizing’s theorem. *Inform. Process. Lett.* 41 (1992), 131–133.
- [84] E. Petrank, The hardness of approximation: gap location, *Computational Complexity*, 4 (1994), 133–157.
- [85] R. Motwani, J. Naor and M. Naor, The probabilistic method yields deterministic parallel algorithms. *J. Comput. System Sci.* 49 (1994), 478–516.
- [86] W. Liang, Fast parallel algorithms for the approximate edge-coloring problem. *Inform. Process. Lett.* 56 (1995), 333–338.
- [87] W. Liang, X. Shen and Q. Hu, Parallel algorithms for the edge-coloring and edge-coloring update problems. *J. Parallel Distrib. Comput.* 32 (1996), 66–73.
- [88] D. Bertsimas, C. P. Teo, and R. Vohra, On dependent randomized rounding algorithms, *Proc. 5th Int. Conf. on Integer Prog. and Combinatorial Optimization*, Lecture Notes in Comput. Sci. 1084, Springer-Verlag, (1996), 330–344.
- [89] M. Fürer and B. Raghavachari, Parallel edge coloring approximation, *Parallel Process. Lett.*, 6 (1996), 321–329.

- [90] J. Kahn, Asymptotics of the chromatic index for multigraphs, *Journal of Combinatorial Theory (Ser. B)*, 68(1996), 233–254.
- [91] X. Zhou H. Susuki, and T. Nishizeki, A linear algorithm for edge-coloring series-parallel multigraphs, *J. Algorithms*, 20(1996), 174–201.
- [92] X. Zhou H. Susuki, and T. Nishizeki, An NC parallel algorithm for edge-coloring series-parallel multigraphs, *J. Algorithms*, 23(1997), 359–374.
- [93] D. A. Grable, A. Panconesi, Nearly optimal distributed edge coloring in  $O(\log \log n)$  rounds, *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, January, (1997), 278–285.
- [94] P. Crescenzi, V. Kann, R. Silvestri, and L. Trevisan, Structure in approximation classes, *SIAM J. Comp.*, 28 (1999), 1759–1782.
- [95] A. Schrijver, Bipartite edge-colouring in  $O(\Delta m)$  time, *SIAM J. Computing*, 28(1999), 841–846.
- [96] Yijie Han, Weifa Liang and Xiaojun Shen, Very fast parallel algorithms for approximate edge coloring, *Discrete Applied Mathematics*, 108(2001), 227–238.
- [97] I. Caragiannis, A. Ferreira, C. Kaklamanis, S. Perennes, P. Persiano and H. Rivano, Approximate constrained bipartite edge coloring, *Discrete Applied Mathematics*, 143(2004), 54–61
- [98] M. R. Salavatipour, A polynomial time algorithm for strong edge coloring of partial  $k$ -trees, *Discrete Applied Mathematics*, 143(2004), 285–291.
- [99] R. Cole, L. Kowalik, New linear-time algorithms for edge-coloring planar graphs, *Algorithmica*, 50(3)(2008), 351–368.

### 时间表问题

- [100] J. Aubin, J. A. Ferland, A large scale timetabling problem, *Computer and Operations Research*, 16(1998), 67–77.
- [101] A. Hertz, V. Robert, Constructing a course schedule by solving a series of assignment type problems, *European Journal of Operational Research*, 108(1998), 585–603.
- [102] R. J. Willems, School timetable construction: algorithms and complexity, *PhD thesis, Technische Universiteit Eindhoven, The Netherlands*, 2002.
- [103] E. K. Burke, S. Petrovic, Recent research trends in automated timetabling, *European Journal of Operational Research*, 140(2002), 266–280.
- [104] R. Alvarez, E. Crespo, J. M. Tamarit, Design and implementation of a course scheduling system using tabu search, *European Journal of Operational Research*, 137(2002), 512–523.
- [105] L. T. G. Merlot, Techniques for academic timetabling, PhD thesis, University of Melbourne, Australia, 2005.

- [106] A. Caprara, M. Monaci, P. Toth and P. L. Guida, A Lagrangian heuristic algorithm for a real-world train timetabling problem, *Discrete Applied Mathematics*, 154(2006), 738–753.
- [107] N. Boland, B. D. Hughes, L. T. G. Merlot, P. J. Stuckey, New integer linear programming approaches for course timetabling, *Computer and Operations Research*, 35(2008), 2209–2233.

### 点染色的精确算法

- [108] N. Christofides, An algorithm for the chromatic number of a graph, *The Computer Journal*, 14(1971), 38–39.
- [109] D. G. Corneil and B. Graham, An algorithm for determining the chromatic number of a graph, *SIAM Journal on Computing*, 2(1973), 211–218.
- [110] D. Brelaz, New methods to color the vertices of a graph, *Communications of the Association for Computing Machinery*, 22(1979), 251–256.
- [111] M. Kubale and B. Jackowski. A generalized implicit enumeration algorithm for graph coloring. *Communications of the ACM*, 28 (1985), 412–418.
- [112] A. Mehrotra and M. Trick, A column generation approach for graph coloring. *INFORMS Journal On Computing*, 8(4) (1996), 344–354.
- [113] E. C. Sewell. An Improved algorithm for exact graph coloring, in “*Cliques, Coloring and Satisfiability*”. (DIMACS Series in Discrete Mathematics and Theoretical Computer Science), vol. 26 (1996), 359–373. American Mathematical Society, Providence, RI, USA.
- [114] D. Kirovski and M. Potkonjak, Efficient coloring of a large spectrum of graphs. In *DAC '98: Proceedings of the 35th annual conference on Design automation*, (1998), 427–432, ACM Press, New York, NY, USA.
- [115] O. Johansson, Simple distributed  $\Delta + 1$ -coloring of graphs, *Information Processing Letters*, 70(1999), 229–232.
- [116] S. R. Vegdahl, Using node merging to enhance graph coloring. In *PLDI '99: Proceedings of the ACM SIGPLAN 1999 conference on Programming language design and implementation*, (1999), 150–154, ACM Press, New York, NY, USA.
- [117] F. Herrmann and A. Hertz, Finding the chromatic number by means of critical graphs. *Journal of Experimental Algorithmics*, 7(2002), p10.
- [118] I. M. Diaz and P. Zabala, A branch-and-cut algorithm for graph coloring, (2002), 55–62, Ithaca, New York, USA.
- [119] D. Eppstein, Small maximal independent sets and faster exact graph coloring, *Journal of Graph Algorithms and Applications*, 7(2) (2003), 131–140.
- [120] M. Caramia and P. Dell'Olmo. Bounding vertex coloring by truncated multistage branch and bound, *Networks*, 44(4) (2004), 231–242.

- [121] G. A. Kochenberger, F. Glover, B. Alidaee and C. Rego, An unconstrained quadratic binary approach to the vertex coloring problem. *Annals of Operations Research*, 139(1) (2005), 229–241.

#### 点染色的近似算法

- [122] A Wigderson, improving the performance guarantee for approximate graph coloring, *Journal of the ACM*, 30(1983), 729–735.
- [123] B Berger and J Rompel. A better performance guarantee for approximate graph coloring. *Algorithmica*, 5(1990), 459–466.
- [124] A. Blum. Some tools for approximate 3-coloring. In *Proceedings 31st IEEE Symposium on Foundations of Computer Science*, pages 554–562, Los Angeles, CA, 1990. IEEE Computer Society.
- [125] S. Khanna, N. Linial, and S. Safra, On the hardness of approximating the chromatic number, *Proc. 2nd Israel Symp. on Theory of Computing and Systems*, IEEE Computer Society, (1993), 250–260.
- [126] M. M. Halldórsson, A still better performance guarantee for approximate graph coloring, *Information Processing Letters*, 45(1993), 19–23.
- [127] C. Lund, and M. Yannakakis, On the hardness of approximating minimization problems, *J. ACM* 41(1994), 960–981.
- [128] A. Blum, New Approximation algorithms for graph coloring, *Journal of the ACM*, 41(1994), 470–516.
- [129] A. Blum, and D. Karger, An  $\tilde{O}(n^{\frac{3}{14}})$ -coloring algorithm for 3-colorable graphs, *Information Processing Letters*, 61(1997), 49–53.
- [130] L. J. Cowen, W. Goddard, and C. E. Jesurum, Coloring with defect, *Proc. 8th Ann. ACM-SIAM Symp. on Discrete Algorithms*, ACM-SIAM, (1997), 548–557.
- [131] R. Duh, and M. Fürer, Approximation of  $k$ -set cover by semi-local optimization, *Proc. 29th Ann. ACM Symp. on Theory of Comp.*, ACM, (1997), 256–265.
- [132] M Bellare, O Goldreich, and M. Sudan, Free bits, PCPs and non-approximability - towards tight results, *SIAM J. Comp.* 27(1998), 804–915.
- [133] D Karger, R. Motwani, and M. Sudan, Approximate graph coloring by semidefinite programming, *J. ACM*, 45(1998), 246–265. 或见 *Proceedings of the 35<sup>th</sup> Annual IEEE Symposium on Foundations of Computer Science*, (1994), 2–13.
- [134] M. Krivelevich, and B. Sudakov, Approximate coloring of uniform hypergraphs, *Proc. 6th Ann. European Symp. on Algorithms*, Lecture Notes in Comput. Sci., Springer-Verlag, (1998), 477–489.
- [135] T. Hofmeister, and H. Lefmann, Approximating maximum independent sets in uniform hypergraphs, *Proc. 23rd International Symp. on Mathematical Foundations of Comput. Sci.*, Lecture Notes in Comput. Sci. 1450, Springer-Verlag, (1998), 562–570.



- [136] V. Kumar, Approximating circular arc colouring and bandwidth allocation in all-optical ring networks, *Proc. 1st Int. Workshop on Approximation Algorithms for Combinatorial Problems*, Lecture Notes in Comput. Sci., Springer-Verlag, (1998), 147–158.

### 点染色的启发式算法

- [137] A. Hertz and D. de Werra, Using tabu search techniques for graph coloring. *Computing*, 39(4) (1987), 345–351.
- [138] C. Morgenstern, and H. Shapiro, Coloration neighborhood structures for general graph coloring. In *Proceedings of the first annual ACM-SIAM Symposium on Discrete algorithms*, (1990), 226–235. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.
- [139] D. de Werra, Lausanne, Heuristics for graph coloring. *Computing Supplementum*, 7(1990) 191–208.
- [140] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon. Optimization by simulated annealing: An experimental evaluation; part II, graph coloring and number partitioning. *Operations Research*, 39(3):378–406, 1991.
- [141] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz, Simple heuristics for unit disk graphs, *Networks*, 25(1995), 59–68.
- [142] A Jagota, An adaptive, multiple restarts neural network algorithm for graph coloring. *European Journal of Operational Research*, 93(2) (1996), 257–270.
- [143] C. Fleurent and J. Ferland, Object-oriented implementation of heuristics search methods for graph coloring, maximum clique, and satisfiability. vol. 26 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, (1996), 619–652, American Mathematical Society, Providence, RI, USA.
- [144] G. Lewandowski and A. Condon, Experiments with parallel graph coloring heuristics and applications of graph coloring. vol. 26 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, (1996), 309–334. American Mathematical Society, Providence, RI, USA.
- [145] D. C. and A. Hertz, Ants can colour graphs, *Journal of the Operational Research Society*, 48(1997), 295–305.
- [146] Eiben A. E., Hauw J. K. V. D., and Hemert J. I. V., Graph coloring with adaptive evolutionary algorithms, *Journal of Heuristics*, 4(1) (1998), 25–46.
- [147] Galinier P. and Hao J., Hybrid evolutionary algorithms for graph coloring. *Journal of Combinatorial Optimization*, 3(4) (1999), 379–397.
- [148] N. Alon, M. Krivelevich, and B. Sudakov, Coloring graphs with sparse neighborhoods. *Journal of Combinatorial Theory, Ser. B*, 77(1999), 73–82.
- [149] M. Laguna, and R. Martí, A GRASP for coloring sparse graphs. *Computational optimization and applications*, 19(2) (2001), 165–178.

- [150] J. González-Velarde and M. Laguna, Tabu search with simple ejection chains for coloring graphs. *Annals of Operations Research*, 117(1–4) (2002), 165–174.
- [151] M. Chiarandini and T. Stützle, An application of iterated local search to graph coloring. In *Proceedings of the Computational Symposium on Graph Coloring and its Generalizations*, edited by D.S. Johnson, A. Mehrotra, and M. Trick, (2002), 112–125. Ithaca, New York, USA.
- [152] C. Croitoru, H. Luchian, O. Gheorghies, and A. Apetrei, A new genetic graph coloring heuristic. (2002), 63–74. Ithaca, New York, USA.
- [153] T. N. Bui and C. M. Patel, An ant system algorithm for coloring graphs, (2002), 83–91. Ithaca, New York, USA.
- [154] A. Fabrikant and T. Hogg, Graph coloring with quantum heuristics. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence*, Edmonton, Alberta, Canada. AAAI Press, (2002), 22–27.
- [155] C. Avanthay, A. Hertz, and N. Zufferey, A variable neighborhood search for graph coloring. *European Journal of Operational Research*, (2003).
- [156] A. D. Blas, A. Jagota, and R. Hughey, A range-compaction heuristic for graph coloring. *Journal of Heuristics*, 9(3) (2003), 489–506.
- [157] V. Cutello, G. Nicosia, and M. Pavone, A hybrid immune algorithm with information gain for the graph coloring problem. vol. 2723 of *Lecture Notes in Computer Science*, (2003), 171–182. Springer Verlag, Berlin, Germany.
- [158] C. A. Glass and A. Prügel-Bennett, Genetic algorithms for graph colouring: Exploration of Galinier and Hao’s algorithm. *Journal of Combinatorial Optimization*, 7(3) (2003), 229–236.
- [159] I. Finocch, A. Panconesi, R. Silvestri, An experimental analysis of simple, distributed vertex coloring algorithms, *Algorithmica*, 41(1)(2004), 1–23.
- [160] C. A. Glass and A. Prügel-Bennett, A polynomially searchable exponential neighbourhood for graph colouring. *Journal of the Operational Research Society*, 56(3) (2005), 324–330.

#### 点染色问题的计算复杂性

- [161] J. S. Turner. Almost all  $K$ -colorable graphs are easy to color. *Journal of Algorithms*, 9(1988), 63–82.
- [162] Ludek Kucera, Graphs with small chromatic numbers are easy to color, *Information Processing Letters*, 30(1989), 233–236.
- [163] Ludek Kucera, The greedy coloring is a bad probabilistic algorithm, *Journal of Algorithms*, 12(1991), 674–684.

- [164] V. T. Paschos, Polynomial approximation and graph-coloring. *Computing*, 70(1) (2003), 41-86.

#### 有约束的图染色问题

- [165] H. R. Hind, *Restricted Edge-Colourings*, Ph.D. thesis, University of Cambridge, 1988.
- [166] N. Alon, Restricted colorings of graphs, in *Surveys in Combinatorics 1993* (Proc. 14<sup>th</sup> British Combinatorial Conference), Cambridge Univ. Press, Cambridge, (1993), 1-33.
- [167] M. Caramia and P. Dell'Olmo, Constraint propagation in graph coloring. *Journal of Heuristics*, 8(1) (2002), 83-107.

#### 赋权染色

- [168] Egon Balas and Jue Xue, Minimum weighted coloring of triangulated graphs, with application to maximum weight vertex packing and clique finding in arbitrary graphs, *SIAM Journal on Computing*, 20(2) (1991), 209-221.
- [169] M. Caramia and P. Dell'Olmo, Solving the minimum-weighted coloring problem. *Networks*, 38(2) (2001), 88-101.

#### 图的色幂

- [170] G. Agnarsson and M. M. Halldórsson, Coloring Powers of Planar Graphs, *SIAM J. Discrete Math.*, 16(2002-2003), 651-662.
- [171] D. Král', Coloring Powers of Chordal Graphs, *SIAM J. Discrete Math.*, 18(2004-2005), 451-461.

#### 随机图染色和随机算法

- [172] T. Luczak, The chromatic number of random graphs, *Combinatorica*, 11(1) (1991), 45-54.
- [173] M. Fürer and C. R. Subramanian. Coloring random graphs. In *Proceedings 3rd Scandinavian Workshop on Algorithmic Theory*, Berlin, 1992. Springer Lecture Notes in Computer Science.
- [174] R. Motwani and P. Raghavan, *Randomized Algorithms*, Cambridge University Press, New York, 1995.
- [175] U. Feige, Randomized graph products, chromatic numbers, and the Lovász-function, in *Proceedings of the 27<sup>th</sup> Annual ACM Symposium on Theory of Computing*, (1995), 635-640.
- [176] Cécile Murat, Vangelis Th Paschos, On the probabilistic minimum coloring and minimum  $k$ -coloring, *Discrete Applied Mathematics*, 154(2006), 564-586.

#### 图的全染色

- [177] O. V. Borodin, On the total coloring of planar graphs, *J. Reine Angew. Math.*, 394(1989), 180-185.

- [178] D. L. Chen and J. L. Wu, The total coloring of some graphs, in “*Combinatorics: Graph Theory, Algorithms, and Applications*, Beijing, 1993”, World Science, River Edge, NY, (1994), 17–20.
- [179] Celina M. H. de Figueiredo, J. Meidanis and C. P. De Mello, Total-Chromatic number and chromatic index of dually chordal graphs, *Information Processing Letters*, 70(1999), 147–152.
- [180] D. Sanders and Y. Zhao, On the entire coloring conjecture, *Canad. Math. Bull.*, 43(2000), 108–114.
- [181] V. A. Bojarshinov, Edge and total coloring of interval graphs, *Discrete Applied Mathematics*, 114(2001), 23–28.

#### 图的表色数

- [182] F. Galvin, The list chromatic index of a bipartite multigraph, *Journal of Combinatorial Theory (Ser. B)*, 63(1995), 153–158.
- [183] J. Kahn, Asymptotically good list-colorings, *J. Combin. Theory (Ser. A)*, 73(1996), 1–59.
- [184] O. V. Borodin and A. V. Kostochka, List edge and list total colorings of multigraphs, *Journal of Combinatorial Theory (Ser. B)*, 71(1997), 184–204.
- [185] A. Daneshgar and H. Hajiabolhassan, Unique list-colourability and the fixing chromatic number of graphs, *Discrete Applied Mathematics*, 152(2005), 123–138.

#### 图的和染色

- [186] A. Bar-Noy, M. Bellare, M. M. Halldórsson, H. Shachnai, and T. Tamir, On chromatic sums and distributed resource allocation, *Inform. and Comput.* 140(1998), 183–202.
- [187] A. Bar-Noy, and G. Kortsarz, The minimum color sum of bipartite graphs, *J. Algorithms* 28 (1998), 339–365.
- [188] A. Bar-Noy, M. M. Halldórsson, G. Kortsarz, H. Shachnai, and R. Salman, Sum multicoloring of graphs, *Proc. 7th Ann. European Symp. on Algorithms*, Lecture Notes in Comput. Sci., Springer-Verlag, 1999.
- [189] M. M. Halldórsson, and G. Kortsarz, Multicoloring planar graphs and partial  $k$ -trees, *Proc. 2nd Int. Workshop on Approximation Algorithms for Combinatorial Problems*, Lecture Notes in Comput. Sci., Springer-Verlag, 1999.
- [190] M. M. Halldórsson, G. Kortsarz, A. Proskurowski, R. Salman, H. Shachnai and J. A. Telle, Multi-coloring trees, *Proc. 5th Ann. Int. Conf. on Computing and Combinatorics*, Lecture Notes in Comput. Sci., Springer-Verlag, (1999), 271–280.
- [191] S. Nicoloso, X. Song, M. Sarrafzadeh, On the sum coloring problem on interval graphs, *Algorithmica*, 23(1999), 109–126.
- [192] K. Jansen, Approximation results for the optimum cost chromatic partition problem, *J. Algorithms*, 34(2000), 54–89.

- [193] M. R. Salavatipour, On sum coloring of graphs, *Discrete Applied Mathematics*, 127(2003), 477–488.

#### 其他有关染色问题

- [194] A. Hertz, B. Jaumard, and M. P. de Aragao, Local optima topology for the  $k$ -coloring problem, *Discrete Applied Mathematics*, 49(1-3) (1994), 257–280.
- [195] M. Larsen, J. Propp, and D. Ullman, The fractional chromatic number of Mycielski's graphs, *J. Graph. Theory*, 19(1995), 411–416.
- [196] A. Chaudhary, and S. Vishwanathan, Approximation algorithms for the achromatic number, *Proc. 8th Ann. ACM-SIAM Symp. on Discrete Algorithms*, ACM-SIAM, (1997), 558–563.
- [197] U. Feige, and J. Kilian, Zero knowledge and the chromatic number, *Journal of Computer and System Sciences*, 57(2)(1998), 187–199.
- [198] J. Janssen, and K. Kilakos, Bounded Stable Sets: Polytopes and Colorings, *SIAM J. Discrete Math.*, 12(1999), 262–275.
- [199] A. Gyárfás, Z. Király, and J. Lehel, On-Line 3-Chromatic Graphs I. Triangle-Free Graphs, *SIAM J. Discrete Math.*, 12(1999), 385–411.
- [200] H. Enomoto, M. Hornák, and S. Jendrol', Cyclic chromatic number of 3-connected plane graphs, *SIAM J. Discrete Math.*, 14(2001), 121–137.
- [201] J. Culberson and I. P. Gent, Frozen development in graph coloring. *Theoretical Computer Science*, 265(1–2) (2001).
- [202] N. Alon and M. Krivelevich, Testing  $k$ -colorability, *SIAM J. Discrete Math.*, 15(2001-2002), 211–227.
- [203] O. V. Borodin and D. R. Woodall, Cyclic colorings of 3-polytopes with large maximum face size, *SIAM J. Discrete Math.*, 15(2001-2002), 143–154.
- [204] A. V. Gelder, Another look at graph coloring via propositional satisfiability, *Ithaca*, (2002), 48–54, New York.
- [205] J. Fiala, K. Jansen, V. B. Le, and E. Seidel, Graph Subcolorings: Complexity and Algorithms, *SIAM J. Discrete Math.*, 16(2002-2003), 635–650.
- [206] U. Feige, M. Langberg, and G. Schechtman, Graphs with Tiny Vector Chromatic Numbers and Huge Chromatic Numbers, *SIAM J. Computing*, 33(2003–2004), 1338–1368
- [207] G. Bacsó, S. Gravier, A. Gyárfás, M. Preissmann, and A. Sebo, Coloring the Maximal Cliques of Graphs, *SIAM J. Discrete Math.*, 17(2003–2004), 361–376.
- [208] M. Vasquez, New results on the queens- $n^2$  graph coloring problem, *Journal of Heuristics*, 10(4) (2004), 407–413.
- [209] M. O. Albertson, A. V. Kostochka, and D. B. West, Precoloring Extensions of Brooks' Theorem, *SIAM J. Discrete Math.*, 18(2004–2005), 542–553.

- [210] V. Guruswami, and S. Khanna, On the Hardness of 4-Coloring a 3-Colorable Graph, *SIAM J. Discrete Math.*, 18(2004–2005), 30–40.
- [211] G. Chartrand, L. Nebeský and P. Zhang, Hamiltonian colorings of graphs, *Discrete Applied Mathematics*, 146(2005), 257–272
- [212] H. Broersma, F. V. Fomin, J. Kratochvíl, G. J. Woeginger, Planar graph coloring avoiding monochromatic subgraphs: trees and paths make it difficult, *Algorithmica*, 44(4)(2006), 343–361.
- [213] S. Kenkre, S. Vishwanathan, A bound on the chromatic number using the longest odd cycle length, *Journal of Graph Theory*, 54(2007), 267–276.
- [214] F. Kramer, H. Kramer, A survey on the distance-colouring of graphs. *Discrete Mathematics*, 308(2008), 422–426.
- [215] Béla Bollobás, András Gyárfás, Highly connected monochromatic subgraphs. *Discrete Mathematics*, 308(2008), 1722–1725.

#### 染色的应用

- [216] M. R. Garey, D. S. Johnson, and H. C. So, An application of graph coloring to printed circuit testing. *IEEE Trans. on Circuits and Systems*, CAS-23 (1976), 591–599.
- [217] Andreas Gamst, Some lower bounds for a class of frequency assignment problems, *IEEE Transactions of Vehicular Echnology*, 35(1) (1986), 8–14.
- [218] Fred C. Chow and John L. Hennessy, The priority-based coloring approach to register allocation. *ACM Transactions on Programming Languages and Systems*, 12(4) (1990), 501–536.
- [219] G. Lewandowski, Practical implementations and applications of graph coloring, *PhD thesis, University of Wisconsin-Madison, USA, 1994.*
- [220] S. Gerke, *Weighted colouring and channel assignment*, DPhil thesis, University of Oxford, 2000.
- [221] C. Mcdiarmid, and B. Reed Channel assignment and weighted colouring, *Networks*, 36(2000), 114–117.
- [222] R. Freimer, Generalized map coloring for use in geographical information systems, In *ACM-GIS 2000, Proceedings of the Eighth ACM Symposium on Advances in Geographic Information Systems*, edited by K.J. Li, K. Makki, N. Pissinou, and S. Ravada, (2000), 167–173. ACM press, New York, NY, USA.
- [223] C. Nomikos, A. Pagourtzis and S. Zachos, Routing and path multicoloring, *Information Processing Letters*, 80(2001), 249–256.
- [224] N. Barnier and P. Brisset, Graph coloring for air traffic flow management, In *CPAIOR'02: Fourth International Workshop on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimisation Problems*, (2002), 133–147, Le Croisic, France.

- 
- [225] M. Allen, G. Kumaran and T. Liu, A combined algorithm for graph-coloring in register allocation. *Ithaca*, (2002), 100 –111, New York.
- [226] Hung Q. Ngo and Van H. Vu, Multirate Rearrangeable Clos Networks and a Generalized Edge-Coloring Problem on Bipartite Graphs, *SIAM J. Computing*, 32(2002–2003), 1040–1049.
- [227] A. Zymolka, A. M. C. A. Koster, and R. Wessäly, Transparent optical network design with sparse wavelength conversion, In *Proceedings of the 7th IFIP Working Conference on Optical Network Design & Modelling*, (2003), 61-80, Budapest, Hungary.
- [228] G. Chaitin, Register allocation and spilling via graph coloring, *SIGPLAN Not.*, 39(4) (2004), 66–74.
- [229] M. D. Smith, N. Ramsey and G. Holloway, A generalized algorithm for graph-coloring register allocation, In *PLDI '04: Proceedings of the ACM SIGPLAN 2004 Conference on Programming Language Design and Implementation*, (2004), 277–288. ACM Press, New York.

# 第七章 平面图

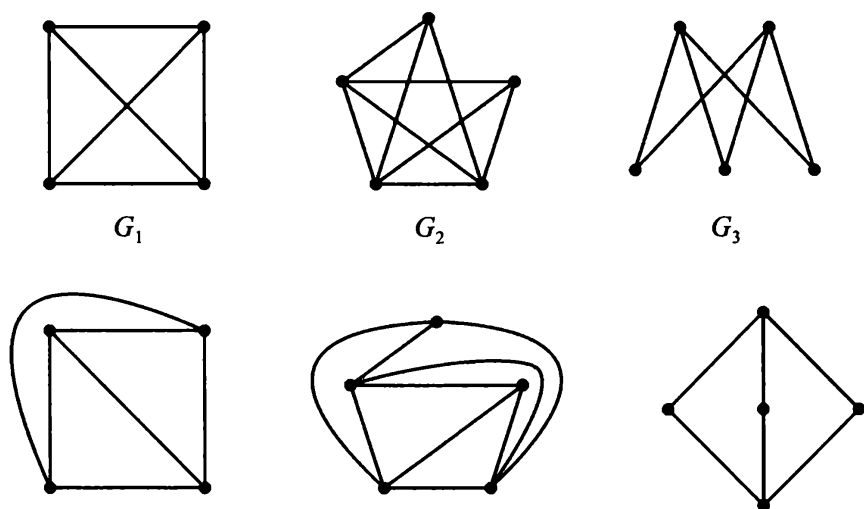
---

## §7.1 平面图的概念

**定义 7.1.1** 如果图  $G$  能画在曲面  $S$  上, 使得任意两边互不交叉, 则称  $G$  可嵌入曲面  $S$ . 若图  $G$  可嵌入平面, 则称  $G$  是可平面图 (planar graph), 否则称为不可平面图 (nonplanar graph).

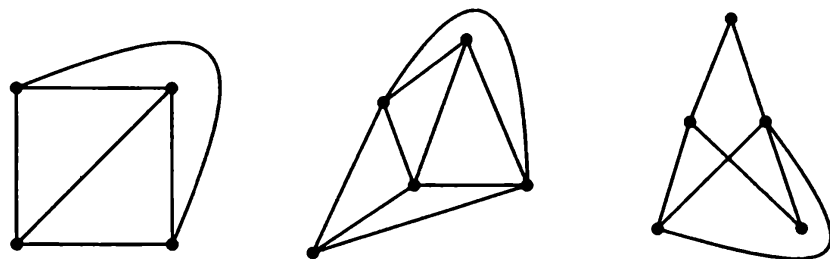
**定义 7.1.2** 可平面图  $G$  在平面上画出的无交叉边的图示称为它的平面嵌入 (planar embedding). 可平面图的任何一个平面嵌入都称为一个平面图 (plane graph).

例如, 下面是三个可平面图及其平面嵌入.



由于一个图的图示一般是不唯一的, 因此一个可平面图的平面嵌入 (从而对应的平面图) 一般也是不唯一的. 例如下列各图依次是上述  $G_1$ 、 $G_2$ 、 $G_3$  的另外一种平面嵌入, 它们与上一组平面嵌入是不同的平面图.





根据定义, 下列定理是显然的.

**定理 7.1.1** 若图  $G$  是可平面图, 则  $G$  的任何子图都是可平面图.

**定理 7.1.2** 若图  $G$  是不可平面图, 则  $G$  的任何母图都是不可平面图.

**定理 7.1.3** 若图  $G$  是可平面图, 则在  $G$  中添加重边或环边后所得之图仍是可平面图.

**注** 由定义和以上定理易知

(1)  $K_n$  ( $n \leq 4$ ) 和  $K_{1,n}$  ( $n \geq 1$ ) 及其所有子图都是可平面图.

(2) 环边和重边不影响图的可平面性.

因此, 以下讨论平面性时若无特别说明, 总假定图是简单图.

**定义 7.1.3** 设图  $G$  是平面图,  $G$  的边将平面划分出的区域称为图  $G$  的**面** (face). 其中面积无限的面称为**无限面**或**外部面** (exterior face), 面积有限的面称为**有限面**或**内部面** (interior face). 包围一个面的所有边称为该面的**边界** (boundary). 一个面的边界上的边数称为该面的**度数** (degree of face, 割边按两次计), 面  $f$  的度数记为  $\deg(f)$ .

**定理 7.1.4** 平面图  $G$  中所有面的度数之和等于  $G$  的边数的两倍, 即

$$\sum_{i=1}^{\varphi} \deg(f_i) = 2\varepsilon(G),$$

其中  $f_1, f_2, \dots, f_{\varphi}$  是  $G$  的所有面.

**证明** 对  $G$  的任何一条边  $e$ , 若  $e$  是两个面  $f_i$  和  $f_j$  的公共边界, 则在计算  $f_i$  和  $f_j$  的度数时,  $e$  各提供了 1; 若  $e$  只是某一个面的边界 (割边), 则在计算该面的度数时,  $e$  提供了 2. 可见在计算面的度数之和时, 每条边都贡献了 2. 因而结论成立. 证毕.

**定义 7.1.4** 设  $G$  是简单图且为可平面图, 若在  $G$  的任意不相邻顶点  $u, v$  之间加边  $uv$  都会使所得之图成为不可平面图, 则称  $G$  是**极大可平面图** (maximal planar graph). 极大可平面图的任一个平面嵌入都称为**极大平面图** (maximal plane graph).

$K_1, K_2, K_3, K_4, K_5 - e$  都是极大可平面图.

容易证明下列定理 7.1.5 和定理 7.1.6.

**定理 7.1.5** 极大可平面图是连通的.

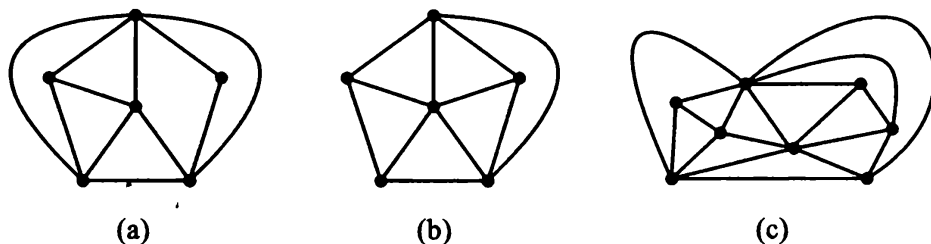
**定理 7.1.6** 至少含 3 个顶点的极大可平面图中没有割边和割点.

**定理 7.1.7** 对于至少含 3 个顶点的极大平面图, 其每个面的度数必定都是 3.

**证明** 设  $G$  是一个极大平面图, 则  $G$  是简单、连通的. 由于  $G$  中无环边和重边, 于是  $G$  中各面的度数至少为 3. 以下用反证法证明  $G$  中各面的度数不会大于 3.

事实上, 假如  $G$  的某个面  $f_i$  的度数  $\deg(f_i) = s \geq 4$ , 则  $f_i$  的边界上至少有 4 个顶点, 设其中 4 个依次相邻的顶点为  $v_1, v_2, v_3, v_4$ . 若  $v_1$  与  $v_3$  在  $G$  中不相邻, 则在  $f_i$  内添加边  $v_1v_3$  不破坏平面性, 这与  $G$  是极大平面图矛盾. 因而  $v_1$  与  $v_3$  在  $G$  中必定相邻. 但因面  $f_i$  的存在, 边  $v_1v_3$  必在  $f_i$  的外边. 类似地,  $v_2$  与  $v_4$  在  $G$  中也必定相邻且边  $v_2v_4$  必在  $f_i$  的外边. 于是边  $v_1v_3$  与边  $v_2v_4$  必在  $f_i$  的外部相交. 这与  $G$  是平面图矛盾. 从而  $G$  中各面的度数不会大于 3. 证毕.

由定理 7.1.7 可知, 下列平面图中, 只有 (c) 是极大平面图.



## §7.2 Euler 公式及其应用

**定理 7.2.1 (Euler 公式, 1753)** 设  $G$  是连通的平面图,  $\varepsilon, \nu, \varphi$  分别是其顶点数、边数和面数, 则  $\nu - \varepsilon + \varphi = 2$ .

**证明** 对边数  $\varepsilon$  作数学归纳法.

当  $\varepsilon=0$  时, 因  $G$  是连通图, 所以  $G$  只能是平凡图, 结论显然成立.

假设当  $\varepsilon = k$  时, 结论成立. 下面证明  $\varepsilon = k + 1$  的情况.

若  $G$  不含圈, 则  $G$  是树, 至少有两片树叶. 设  $v$  是  $G$  的一片树叶. 令  $G' = G - v$ , 则  $G'$  仍是连通图, 且  $G'$  的边数  $\varepsilon' = \varepsilon - 1 = k$ , 由归纳假设知,  $\nu' - \varepsilon' + \varphi' = 2$ , 而  $\nu' = \nu - 1, \varphi' = \varphi$ , 于是

$$\nu - \varepsilon + \varphi = (\nu' + 1) - (\varepsilon' + 1) + \varphi' = \nu' - \varepsilon' + \varphi' = 2.$$

若  $G$  中含有圈. 设边  $e$  在  $G$  的某个圈上. 令  $G' = G - e$ , 则  $G'$  仍是连通图, 且  $G'$  的边数  $\varepsilon' = \varepsilon - 1 = k$ , 由归纳假设知

$$\nu' - \varepsilon' + \varphi' = 2, \text{ 而 } \nu' = \nu, \varphi' = \varphi - 1,$$

于是  $\nu - \varepsilon + \varphi = \nu' - (\varepsilon' + 1) + (\varphi' + 1) = \nu' - \varepsilon' + \varphi' = 2$ . 证毕.

**定理 7.2.2 (Euler 公式的推广形式)** 对于具有  $w(w \geq 1)$  个连通分支的平面图  $G$ , 有

$$\nu - \varepsilon + \varphi = w + 1.$$

其中  $\nu, \varepsilon, \varphi$  分别是其顶点数、边数和面数.

**证明** 设  $G$  的连通分支分别为  $G_1, G_2, \dots, G_w$ , 并设  $G_i$  的顶点数、边数和面数分别为  $\nu_i, \varepsilon_i, \varphi_i$ . 由 Euler 公式可知

$$\nu_i - \varepsilon_i + \varphi_i = 2, \quad (i = 1, 2, \dots, w).$$

注意到  $\varepsilon = \sum_{i=1}^w \varepsilon_i, \nu = \sum_{i=1}^w \nu_i$ . 此外, 由于  $G$  的每个分支有一个外部面, 而  $G$  只有一个外部面, 故  $G$  的面数  $\varphi = \sum_{i=1}^w \varphi_i - w + 1$ . 于是将上列等式两端对应相加, 得

$$2w = \sum_{i=1}^w (\nu_i - \varepsilon_i + \varphi_i) = \sum_{i=1}^w \nu_i - \sum_{i=1}^w \varepsilon_i + \sum_{i=1}^w \varphi_i = \nu - \varepsilon + \varphi + w - 1$$

整理即得结论. 证毕.

**定理 7.2.3** 设  $G$  是连通的平面图, 且每个面的度数至少为  $l(l \geq 3)$ , 则  $G$  的边数  $\varepsilon$  与顶点数  $\nu$  有如下关系:

$$\varepsilon \leq \frac{l}{l-2}(\nu - 2).$$

**证明** 设  $G$  的面为  $f_1, f_2, \dots, f_\varphi$ , 由定理 7.1.4 可知

$$2\varepsilon = \sum_{i=1}^{\varphi} \deg(F_i) \geq l \cdot \varphi.$$

由 Euler 公式可知  $\varphi = 2 + \varepsilon - \nu$ . 将此式代入上式, 得  $2\varepsilon \geq l(2 + \varepsilon - \nu)$ , 整理便得结论. 证毕.

**推论 7.2.1**  $K_5$  和  $K_{3,3}$  都是不可平面图.

**证明** 若  $K_5$  是可平面图,  $G$  是其任一个平面嵌入, 由于  $K_5$  中无环边和重边, 故  $G$  的每个面的度数至少为 3, 而  $G$  的边数为 10. 由定理 7.2.3, 应有

$$10 \leq \frac{3}{3-2}(5-2) = 9,$$

这是不可能的. 因此  $K_5$  是不可平面图.

若  $K_{3,3}$  是可平面图,  $G$  是其任一个平面嵌入, 由于  $K_{3,3}$  中最短圈的长度为 4, 故  $G$  中每个面的度数至少为 4, 而  $G$  的边数为 9. 由定理 7.2.3, 应有

$$9 \leq \frac{4}{4-2}(6-2) = 8,$$

这是不可能的. 因此  $K_{3,3}$  是不可平面图. 证毕.

**推论 7.2.2**  $K_n(n \geq 5)$  和  $K_{3,n}(n \geq 3)$  都是不可平面图.

**证明** 由定理 7.1.2 和推论 7.2.1 立即可知.

**定义 7.2.1** 若不可平面图  $G$  任意删除一条边后, 所得之图都是可平面图, 则称  $G$  为极小不可平面图 (minimal nonplanar graph).

**推论 7.2.3**  $K_5$  和  $K_{3,3}$  都是极小不可平面图.

**证明** 由推论 7.2.1 和极小不可平面图的定义容易验证.

**定理 7.2.4** 设  $G$  是具有  $w(w \geq 1)$  个连通分支的平面图, 各面的度数至少为  $l(l \geq 3)$ , 则  $G$  的边数  $\varepsilon$  与顶点数  $\nu$  有如下关系:

$$\varepsilon \leq \frac{l}{l-2}(\nu - w - 1).$$

**证明** 利用 Euler 公式的推广形式 (定理 7.2.2), 与上一定理类似可证.

**定理 7.2.5** 设简单图  $G$  是具有  $\varepsilon$  条边的  $\nu$  ( $\nu \geq 3$ ) 阶平面图, 则  $\varepsilon \leq 3\nu - 6$ .

**证明** 设  $G$  有  $w(w \geq 1)$  个连通分支. 若  $G$  中无圈, 则  $G$  为树或森林,  $\varepsilon = \nu - w \leq 3\nu - 6$ . 若  $G$  中有圈, 则因  $G$  是简单图, 所以各圈的长度至少是 3, 从而  $G$  中面的最小度数  $l \geq 3$ . 又因函数

$$\frac{l}{l-2} = 1 + \frac{2}{l-2}$$

在  $l = 3$  时达到最大值 3, 于是由定理 7.2.4,

$$\varepsilon \leq \frac{l}{l-2}(\nu - w - 1) \leq 3(\nu - 2) = 3\nu - 6. \quad \text{证毕.}$$

**定理 7.2.6** 设  $G$  是具有  $\varepsilon$  条边的  $\nu$  阶极大平面图 ( $\nu \geq 3$ ), 则  $\varepsilon = 3\nu - 6$ , 且  $\varphi = 2\nu - 4$ .

**证明** 由于极大平面图是连通图, 由 Euler 公式,  $\varphi = 2 + \varepsilon - \nu$ . 又因  $G$  是极大平面图, 由定理 7.1.7 可知,  $G$  的每个面的度数均为 3, 故

$$2\varepsilon = \sum_{i=1}^{\varphi} \deg(f_i) = 3\varphi.$$

由以上两式整理即得:  $\varepsilon = 3\nu - 6$ , 且  $\varphi = 2\nu - 4$ . 证毕.

**定理 7.2.7** 设  $G$  是一个有  $\varepsilon$  条边、 $\nu$  个顶点的平面图 ( $\nu \geq 3$ ), 且为连通的简单图, 则  $G$  是极大平面图当且仅当  $G$  的每个面的度数均为 3.

**证明** 由定理 7.1.7 知必要性成立. 下证充分性.

设  $G$  是连通图, 且每个面的度数均为 3. 由定理 7.2.6 的证明知,  $\varepsilon = 3\nu - 6$ . 假如  $G$  不是极大平面图, 则  $G$  中一定存在不相邻的顶点  $u$  和  $v$ , 使得  $G' = G + uv$  仍是平面图, 且为简单图, 而  $G'$  的边数  $\varepsilon' = \varepsilon + 1, \nu' = \nu$ , 结合  $\varepsilon = 3\nu - 6$  得:  $\varepsilon' = 3\nu' - 5 > 3\nu' - 6$ . 这与定理 7.2.5 矛盾. 证毕.

**定理 7.2.8** 设简单图  $G$  是具有  $\varepsilon$  条边的  $\nu$  阶平面图 ( $\nu \geq 3$ ), 则  $G$  的最小度  $\delta \leq 5$ .

**证明** 若  $G$  的阶数  $\nu \leq 6$ , 则结论显然成立.

下设  $\nu \geq 7$ . 假若  $\delta \geq 6$ , 则

$$2\varepsilon = \sum_{i=1}^{\nu} d(v_i) \geq 6\nu.$$

因而  $\varepsilon \geq 3\nu$ , 这与定理 7.2.5 矛盾. 证毕.

### §7.3 可平面图判断

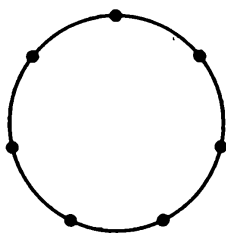
**定义 7.3.1** 设  $e = uv$  是图  $G$  的一条边. 在边  $e$  上加入一个新顶点  $w$ , 将其分为两条新边  $uw$  和  $wv$ , (而  $w$  成为新图的 2 度顶点), 这个过程称为对边  $e$  的剖分 (edge subdivision); 对图  $G$  的边进行一系列剖分后所得之图称为  $G$  的剖分图 (subdivision).

为后面叙述方便, 我们约定图  $G$  本身也看作  $G$  的一个剖分图.

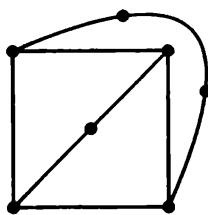
**定义 7.3.2** 设  $e$  是图  $G$  的一条边, 从  $G$  中删去  $e$  并将  $e$  的两个端点粘合在一起, 删去由此得到的环边和重边, 这个过程称为边  $e$  的收缩 (edge contraction). 若图  $G_1$  可通过一系列边的收缩得到与  $G_2$  同构的图, 则称  $G_1$  可收缩到  $G_2$ .

我们约定, 图  $G$  本身也是可收缩为  $G$  的一个图.

例如, 下列 (a) 是  $K_3$  的一个剖分图, (b) 是  $K_4$  的一个剖分图. 反过来, 它们分别可收缩到  $K_3$  和  $K_4$ .



(a)

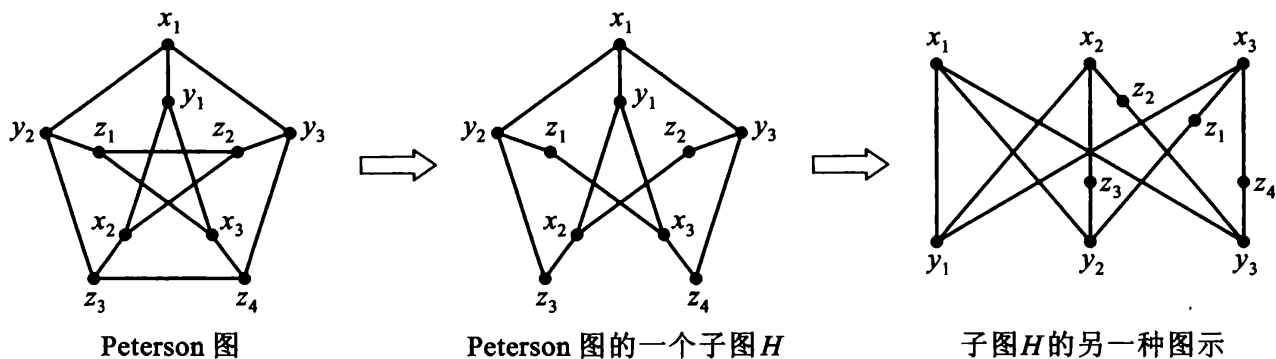


(b)

下列两个定理是可平面图判定定理, 证明从略, 有兴趣的读者可参看文献 [1]~[2]、或 [3]~[11].

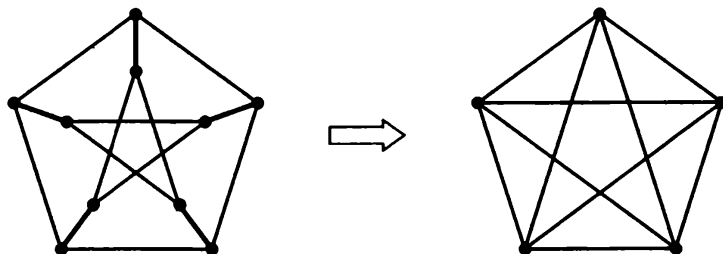
**定理 7.3.1 (Kuratowski, 1930)** 图  $G$  是可平面图当且仅当  $G$  中既不含  $K_5$  的剖分图, 也不含  $K_{3,3}$  的剖分图.

Peterson 图是不可平面图, 因为它含有  $K_{3,3}$  的剖分图 (如下所示).



**定理 7.3.2 (Wagner, 1937)** 图  $G$  是可平面图当且仅当  $G$  中既没有可收缩到  $K_5$  的子图, 也没有可收缩到  $K_{3,3}$  的子图.

Peterson 图可收缩到  $K_5$  (如下图所示, 收缩粗边), 由此再一次看到 Peterson 图是不可平面图.



关于平面图的综合参考见 [1]~[18]; 关于图的可平面性的进一步刻画见文献 [19]~[32]; 判断一个给定的图是否可平面图有线性时间算法, 有关图的可平面性的判断算法见 [33]~[45].

## §7.4 平面图的对偶图

**定义 7.4.1** 设  $G$  是一个平面图,  $G$  的对偶图 (dual graph)  $G^*$  是以  $G$  的面作为顶点的图: 将  $G$  的各个面用平面上的点来表示, 若两个面有公共边界, 且公共边界上有  $k$  条边, 则对应的两个顶点间连  $k$  条边 (越过公共边界上每一边连出一条边). 若某个面上含有割边, 则该面在  $G^*$  中对应的点上连一条环边.

从定义不难看出平面图  $G$  的对偶图  $G^*$  有以下性质:

(1)  $G^*$  是平面图.

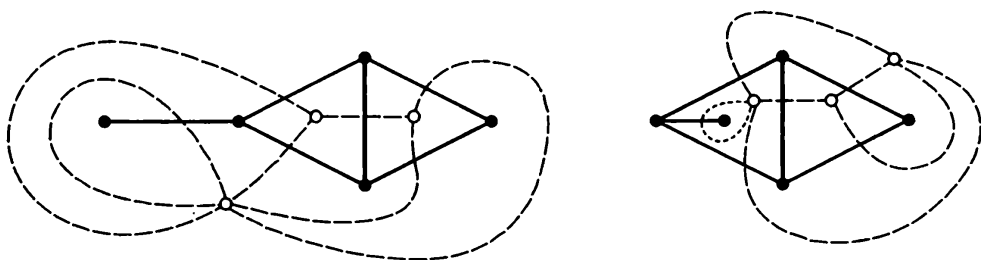
(2)  $G^*$  是连通图.

(3) 若边  $e$  为  $G$  中的环边, 则  $G^*$  中与  $e$  对应的边  $e^*$  为割边; 反之, 若  $e$  为  $G$  的割边, 则  $G^*$  中与  $e$  对应的边  $e^*$  为环边.

(4) 在多数情况下,  $G^*$  为含重边的图.

(5) 两个同构的平面图, 其对偶图不一定是同构的.

例如, 下列两图 (实线边的图) 是同构的, 但它们的对偶图 (虚线所示) 不同构, 因为两个对偶图中环边所关联的顶点的度不相等.



平面图  $G$  与它的对偶图  $G^*$  的顶点数、边数和面数有如下关系.

**定理 7.4.1** 设  $G^*$  是连通平面图  $G$  的对偶图,  $\nu^*, \varepsilon^*, \varphi^*$  和  $\nu, \varepsilon, \varphi$  分别为  $G^*$  和  $G$  的顶点数、边数和面数, 则

(1)  $\nu^* = \varphi$ ;

(2)  $\varepsilon^* = \varepsilon$ ;

(3)  $\varphi^* = \nu$ ;

(4) 设  $G^*$  的顶点  $v_i^*$  位于  $G$  的面  $f_i$  中, 则  $d_{G^*}(v_i^*) = \deg(f_i)$ .

**证明** 由  $G^*$  的构造可知, (1) 和 (2) 是显然的.

(3) 由于  $G$  与  $G^*$  都连通, 因而满足欧拉公式:

$$\nu - \varepsilon + \varphi = 2, \quad \nu^* - \varepsilon^* + \varphi^* = 2.$$

由此及上述 (1)、(2), 便得

$$\varphi^* = 2 + \varepsilon^* - \nu^* = 2 + \varepsilon - \varphi = \nu.$$

(4) 设  $G$  的面  $f_i$  的边界为  $C_i$ . 设  $C_i$  中有  $k_1$  条割边 ( $k_1 \geq 0$ ),  $k_2$  条非割边, 则  $C_i$  的长度为  $k_2 + 2k_1$ , 即  $\deg(f_i) = k_2 + 2k_1$ . 与  $G$  中这  $k_1$  条割边对应, 在  $G^*$  中  $v_i^*$  处有  $k_1$  个环边, 而  $C_i$  的  $k_2$  条非割边对应  $G^*$  中从  $v_i^*$  处引出  $k_2$  条边, 所以

$$d_{G^*}(v_i^*) = k_2 + 2k_1 = \deg(f_i). \quad \text{证毕.}$$

**定理 7.4.2** 设  $G^*$  是具有  $w$  个连通分支的平面图  $G$  的对偶图, 则

- (1)  $\nu^* = \varphi$ ;
- (2)  $\varepsilon^* = \varepsilon$ ;
- (3)  $\varphi^* = \nu - w + 1$ ;
- (4) 设  $v_i^*$  位于  $G$  的面  $f_i$  中, 则  $d_{G^*}(v_i^*) = \deg(f_i)$ .

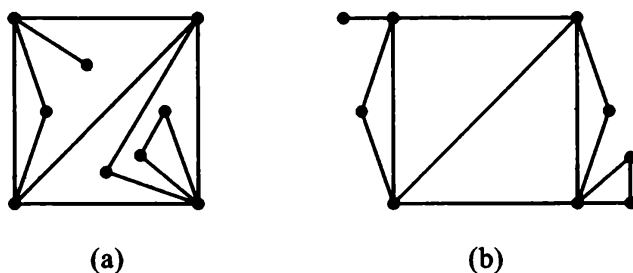
其中  $\nu^*, \varepsilon^*, \varphi^*, \nu, \varepsilon, \varphi$  含义同前.

证明留作练习.

## §7.5 外可平面图

**定义 7.5.1** 若一个图能够被嵌入平面使得所有顶点均在外部面的边界上, 则称该图为外可平面图 (outerplanar graph). 这种使所有顶点均在外部面的边界上的平面嵌入称为外平面嵌入. 外可平面图的外平面嵌入称为外平面图 (outerplane graph).

下图给出了一个外可平面图 (a) 及其外平面嵌入 (b).



容易证明关于外可平面图的下列结论 (习题 7.34).

- (1) 若  $G$  是外可平面图, 则  $G$  的任何子图也是外可平面图;
- (2) 若  $G$  不是外可平面图, 则含有  $G$  作为子图的任何图也不是外可平面图;
- (3) 若  $G$  不是外可平面图, 则  $G$  的任何剖分图也不是外可平面图.

**定理 7.5.1** 设  $G$  是无割点的外可平面简单图, 且  $\nu(G) > 2$ , 则

- (1)  $G$  是 Hamilton 图;
- (2)  $G$  的外平面嵌入, 必使得其顶点全在一个圈上而不在该圈上的边全是该圈的不相交的弦 (这里将连接圈上两点且位于圈的内部的线段称为弦).

**证明** (1) 在定理条件下,  $G$  的外平面嵌入中外部面的边界是一个圈, 它便构成  $G$  的一个 Hamilton 圈.

(2)  $G$  的外平面嵌入形成的外平面图  $G'$  中, 外部面的边界构成一个 Hamilton 圈  $C$ . 显然, 不在圈  $C$  上的边都在圈  $C$  的内部, 且它们互不相交. 证毕.



$K_{2,3}$  不是外可平面图, 因为它不是 Hamilton 图;  $K_4$  也不是外可平面图, 因为它不存在定理 7.5.1 中所述的外平面嵌入.

**定义 7.5.2** 设  $G$  是一个外可平面简单图, 若在  $G$  中任意两个不相邻顶点之间添加一条边所得之图均不是外可平面图, 则称  $G$  为**极大外可平面图** (maximal outerplanar graph). 极大外可平面图的外平面嵌入称为**极大外平面图** (maximal outerplane graph).

下列结论是显然的 (习题 7.35).

- (1) 极大外可平面图必为连通图;
- (2) 极大外可平面图无割点和割边.

**定理 7.5.2** 设  $G$  是  $\nu$  ( $\nu \geq 3$ ) 个点的外平面图. 则  $G$  是极大外平面图的充分必要条件是它的外部面的边界是一个圈且内部面均为三角形.

**证明 充分性** 设  $G$  是  $\nu$  ( $\nu \geq 3$ ) 个点的外平面图, 并且它的外部面的边界  $C$  是一个圈且内部面均为三角形. 任取  $G$  中两个不相邻的点  $u$  和  $v$ , 则  $u$  和  $v$  不在同一个三角形上. 当添加边  $uv$  时, 若将边  $uv$  画在圈  $C$  内, 则  $uv$  将与其他边相交; 若将边  $uv$  画在圈  $C$  外, 则  $G$  的顶点将不全在同一个面的边界上. 总之  $G + uv$  将不再是外平面图, 所以  $G$  是极大外平面图.

**必要性** 设  $G$  是极大外平面图, 则  $G$  连通, 且没有割点和割边, 否则与  $G$  是极大外平面图矛盾. 因此  $G$  的外部面的边界是一个圈.

任取  $G$  的一个内部面  $f$ , 设  $f$  的边界为  $L$ , 则  $L$  是一个圈. 若  $L$  的长  $|L| \geq 4$ , 则  $L$  中必存在不相邻的点对  $u$  和  $v$ , 于是在  $f$  内连接  $u$  和  $v$  将不破坏  $G$  的平面性, 这与  $G$  是极大外平面图矛盾, 故  $|L| < 4$ . 又因  $G$  是简单图, 故  $|L| = 3$ . 证毕.

**定理 7.5.3** 设  $G$  是有  $\nu$  ( $\nu \geq 3$ ) 个点的极大外平面图, 则  $G$  有  $\nu - 2$  个内部面.

**证明** 对点数  $\nu$  用数学归纳法.

当  $\nu = 3$  时,  $G$  必为三角形, 仅有一个内部面, 定理的结论成立.

设  $\nu = k$  时, 结论成立. 下证对有  $\nu = k + 1$  个点的极大外平面图  $G$ , 结论也成立.

先来证明  $G$  必有度为 2 的顶点. 事实上, 设  $G$  的外部面边界形成的圈为  $C$ . 任取  $G$  的一个顶点  $v_1$ , 设它在圈  $C$  上的左右邻点分别是  $v_0$  和  $v_2$ . 若  $v_1$  不是 2 度点, 则在  $C$  上从  $v_1$  出发沿逆时针方向找出  $v_1$  的异于  $v_0$  和  $v_2$  的第一个邻点, 设为  $v_{i_1}$ .  $C$  上  $v_1$  到  $v_{i_1}$  的一段设为  $v_1 v_2 v_3 \cdots v_{i_1}$ , 则  $v_1 v_2 v_3 \cdots v_{i_1} v_1$  构成一个圈  $C_1$ ,  $C_1$  上的点除了  $v_1$  和  $v_{i_1}$  外, 其余点不可能与  $C_1$  之外的点连边. 类似地, 若  $v_2$  不是 2 度点, 则在  $C_1$  上从  $v_2$  出发沿逆时针方向可找出其异于  $v_1$  和  $v_3$  的第一个邻点, 设为  $v_{i_2}$ .  $v_2 v_3 \cdots v_{i_2} v_2$  构成一个圈  $C_2$ ,  $C_2$  上的点除了  $v_2$  和  $v_{i_2}$  外, 其余点不可能与  $C_2$  之外的点连边. 如此类推, 可得到越来越小的圈, 在这个过程中要么找到 2 度顶点, 要么最终获得一个三角形  $v_i v_j v_r, v_j$  不可能与该三角形之外的点连边, 因而是  $G$  的 2 度顶点.

设  $u$  是  $G$  的一个 2 度顶点, 则  $G - u$  中所有顶点仍全在外部面的边界上, 因此  $G - u$  仍是外平面图. 而且, 由于  $G$  是极大外平面图, 故在  $G$  中  $u$  的两个邻点必定相邻, 因此  $G - u$  也是极大外平面图. 但  $G - u$  仅有  $k$  个点, 由归纳假设,  $G - u$  有  $k - 2$  个内部面, 从而  $G$  有  $(k - 2) + 1 = (k + 1) - 2 = \nu - 2$  个内部面. 证毕.

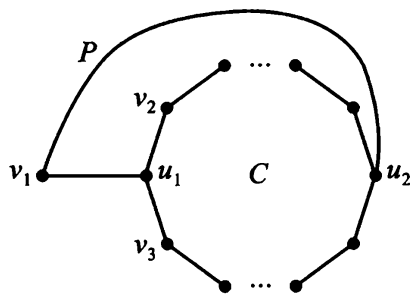
**定理 7.5.4** 图  $G$  是外可平面图当且仅当  $G$  中既不含  $K_4$  的剖分图, 也不含  $K_{2,3}$  的剖分图.

**证明** 由于  $K_4$  和  $K_{2,3}$  都不是外可平面图, 从而它们的剖分图也不是外可平面的. 因此定理的必要性显然成立. 下面用反证法证明充分性.

设存在着这样的图: 它既不含  $K_4$  的剖分图, 也不含  $K_{2,3}$  的剖分图, 但它却不是外平面图. 取这种图中边数最少的一个, 记为  $G$ . 则显然  $G$  是连通简单图.

进一步可证明  $G$  是 2- 连通图. 若不然, 设  $v$  是  $G$  的一个割点,  $G'$  是  $G - v$  的一个连通分支,  $G''$  是  $G - v$  的其余连通分支的并. 令  $G_1 = G[V(G') \cup \{v\}]$ ,  $G_2 = G[V(G'') \cup \{v\}]$ . 则  $G_1$  和  $G_2$  都不含  $K_4$  的剖分图, 也都不含  $K_{2,3}$  的剖分图, 并且它们之中至少有一个不是外平面图. 但这与  $G$  的边数最少性质矛盾. 因此  $G$  中无割点, 从而  $G$  中任二点都共圈.

取  $G$  中最长圈  $C$ , 可以证明  $C$  是  $G$  的 Hamilton 圈. 若不然, 则  $G$  有顶点不在  $C$  上. 取  $v_1 \notin V(C)$ , 使  $v_1$  与  $C$  上一点  $u_1$  相邻. 设  $u_1$  在  $C$  上的左、右邻点分别为  $v_2$  和  $v_3$ . 因  $G$  是 2- 连通的, 故  $G - u_1$  中  $v_1$  与  $C - u_1$  上的点必有路相通. 设  $P$  是  $G - u_1$  中  $v_1$  到  $C - u_1$  上点  $u_2$  的一条路, 不妨设  $u_2$  是  $P$  上仅有的一个属于  $C - u_1$  的点 (若  $u_2$  不是这样的点, 可从  $v_1$  出发沿  $P$  找出第一个在  $C - u_1$  上的点替换  $u_2$ ), 即  $u_2$  是路  $P$  的端点且在  $C - u_1$  上 (如下图所示).



因  $C$  是  $G$  中的最长圈, 故  $u_2 \neq v_2, v_3$  (否则可得到  $G$  中更长的圈). 注意到  $u_1$  与  $v_1, v_2$  和  $v_3$  都相邻,  $u_2$  与  $v_1$  有路  $P$  相连且与  $v_2, v_3$  分别有圈  $C$  上 (不经过  $u_1$  的) 两段相连, 因此构成了完全二部图  $K_{2,3}$  的剖分图 ( $K_{2,3} = (X, Y)$ , 其中  $X = \{u_1, u_2\}$ ,  $Y = \{v_1, v_2, v_3\}$ ), 它完全含于  $G$  中. 这与  $G$  的取法矛盾.

由于  $C$  是  $G$  的 Hamilton 圈而  $G$  不是外可平面图, 因此  $G$  必有边不再  $C$  上, 且若将这些不在  $C$  上的边全画在  $C$  的内部, 则必至少会有两条边相交, 这样便在  $G$  中得到  $K_4$  的剖分图, 与  $G$  的取法矛盾. 证毕.

## §7.6 不可平面图的几个研究方向简介

### 一、最大可平面子图

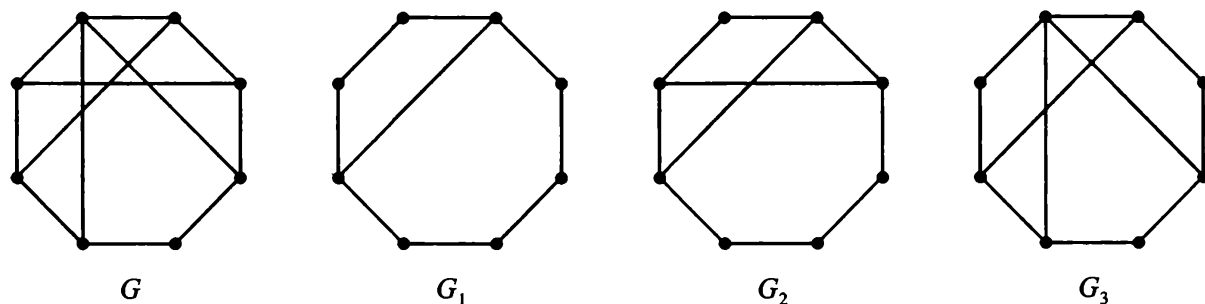
在图的可平面性研究中, 对于一个图  $G$ , 人们首先想要知道的是, 它是不是可平面的. 如果  $G$  不是可平面图, 则不可能将  $G$  的所有边嵌入平面而使得它们互不交叉. 一个自然的问题是: 当我们将  $G$  尽可能地往平面上嵌入并保持嵌入部分的平面性时, 最多能嵌入多少条边? 这便是图的最大可平面子图问题.

**定义 7.6.1** 设  $G$  是一个图,  $G'$  是  $G$  的子图. 如果  $G'$  是可平面图且对  $G$  的每个可平面子图  $G''$  都有  $|E(G'')| \leq |E(G')|$ , 则称  $G'$  是  $G$  的**最大可平面子图**(maximum planar subgraph).

**定义 7.6.2** 设  $G$  是一个图,  $G'$  是  $G$  的子图. 如果  $G'$  是可平面图且给  $G'$  任意添加  $E(G) - E(G')$  中的一条边得到的子图都不再是  $G$  的可平面子图, 则称  $G'$  是  $G$  的**极大可平面子图** (maximal planar subgraph).

显然, 图  $G$  的最大可平面子图一定是  $G$  的极大可平面子图, 但反之不然.

例如, 在下图中,  $G_1$  是  $G$  的一个可平面子图但不是极大可平面子图;  $G_2$  是  $G$  的一个极大可平面子图但不是最大可平面子图;  $G_3$  是  $G$  的一个最大可平面子图, 同时也是  $G$  的一个极大可平面子图.



对于一个给定的图, 一个重要的问题是如何求出它的最大可平面子图. 遗憾的是, 求图的最大可平面子图是一个 NP-hard 问题<sup>[46]~[48]</sup>, 相关的近似算法可参看 [49], 赋权情况下的最大可平面子图问题的近似算法见 [50]~[58].

求图的极大可平面子图有多项式时间算法, 文献 [17] 描述了一个  $O(n^2)$  贪婪算法, 更多的算法见文献 [59]~[66].

### 二、图的交叉数

在许多应用中, 需要将一个图 (不论它是不是可平面的) 画在平面上, 使得边的交叉尽可能少. 因此, 人们希望知道一个图被画在平面上时, 边最少需要交叉几次, 这便是图的交叉数的概念.

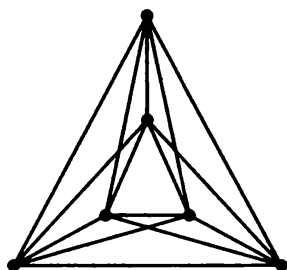
**定义 7.6.3** 将图  $G$  画在平面上时, 边交叉的最少次数, 称为  $G$  的交叉数 (crossing number), 记为  $cr(G)$ .

显然, 所有可平面图的交叉数都为 0. 此外, 容易验证,  $K_5$  和  $K_{3,3}$  的交叉数都为 1.

一些小图的交叉数可以通过直接的方法获得.

**例 7.6.1**  $cr(K_6) = 3$ .

**证明** 由可平面图的性质  $\varepsilon \leq 3\nu - 6$ , 6 个顶点的可平面图最多有 12 条边, 而  $K_6$  有 15 条边, 故  $cr(K_6) \geq 3$ . 另一方面,  $K_6$  的如下图示只有 3 个交叉点, 故又有  $cr(K_6) \leq 3$ . 因此  $cr(K_6) = 3$ . 证毕.



**例 7.6.2** 下图 1 称为完全三部图  $K_{2,2,3}$ , 证明  $cr(K_{2,2,3}) = 2$

**证明**  $K_{2,2,3}$  的另一个图示如图 2, 它只有两个交叉点, 因此  $cr(K_{2,2,3}) \leq 2$ . 另一方面,  $K_{2,2,3}$  含有子图  $K_{3,4}$ . 注意  $K_{3,4}$  是不含三角形的, 因此其可平面子图最多有  $\varepsilon \leq 2\nu - 4 = 2 \cdot 7 - 4 = 10$  条边 (习题 7.12), 但  $K_{3,4}$  本身有 12 条边, 故  $cr(K_{3,4}) \geq 2$ . 因  $K_{2,2,3}$  的任一个图示都包含  $K_{3,4}$  的图示, 因此  $cr(K_{2,2,3}) \geq cr(K_{3,4}) \geq 2$ . 从而  $cr(K_{2,2,3}) = 2$ . 证毕.

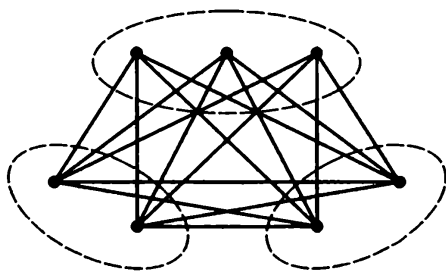


图 1.  $K_{2,2,3}$

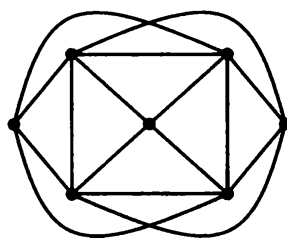


图 2.  $K_{2,2,3}$  的另一种图示

关于一般图的交叉数, 已知的较好下界有

**定理 7.6.1** [67][68] 对于图  $G$ , 若  $\varepsilon(G) \geq 4\nu(G)$ , 则  $cr(G) \geq \frac{\varepsilon^3}{64\nu^2}$ .

**定理 7.6.2** [69] 对于图  $G$ , 若  $\varepsilon(G) \geq 7.5\nu(G)$ , 则  $cr(G) \geq \frac{\varepsilon^3}{33.75\nu^2}$ .

一般地, 有

**定理 7.6.3** <sup>[68][70]</sup> 对于图  $G$ , 若  $\varepsilon(G) \geq c \cdot \nu(G)$ , 则  $\text{cr}(G) \geq \frac{c-3}{c^3} \frac{\varepsilon^3}{\nu^2}$ .

**定理 7.6.4** <sup>[71]</sup> 设图  $G$  的围长为  $g \geq 3$ , 则  $\text{cr}(G) \geq \varepsilon - \frac{g}{g-2}(\nu-2)$ .

对于完全图的交叉数, 目前已知的较好的上、下界如下.

**定理 7.6.5** <sup>[72][73]</sup>  $\text{cr}(K_\nu) \leq \frac{1}{4} \left\lfloor \frac{\nu}{2} \right\rfloor \left\lfloor \frac{\nu-1}{2} \right\rfloor \left\lfloor \frac{\nu-2}{2} \right\rfloor \left\lfloor \frac{\nu-3}{2} \right\rfloor$ .

R.K. Guy [74] [75] 猜想上式的等号成立, 并对  $\nu \leq 10$  的情况给出了证明. 其余情况下, 该猜想仍有待证明.

**定理 7.6.6** <sup>[76]</sup> 当  $\nu \geq 5$  时,  $\text{cr}(K_\nu) \geq \frac{1}{120} \nu(\nu-1)(\nu-2)(\nu-3)$ .

**定理 7.6.7** <sup>[77]</sup> 当  $\nu > 7$  时,  $\text{cr}(K_\nu) \geq \frac{\nu(\nu-1)(\nu-2)(\nu-3)}{20(\nu-6)(\nu-7)} \left\lfloor \frac{\nu-6}{2} \right\rfloor \left\lfloor \frac{\nu-7}{2} \right\rfloor$ .

**定理 7.6.8** <sup>[75]</sup> 对  $\nu \geq 5$ , 有  $\frac{1}{80} \nu^4 + O(\nu^3) \leq \text{cr}(K_\nu) \leq \frac{1}{64} \nu^4 + O(\nu^3)$ .

关于完全二部图的交叉数有如下结果.

**定理 7.6.9** <sup>[78]</sup>  $\text{cr}(K_{m,n}) \leq \left\lfloor \frac{m}{2} \right\rfloor \left\lfloor \frac{m-1}{2} \right\rfloor \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor$ .

Zarankiewicz 曾猜想上式的等号成立并给出了一个“证明”<sup>[78]</sup>, 后来有人指出了这个证明的错误, 有关这方面的研究历史可参看文献 [79]. 目前, 文献 [80] 证明了当  $\min\{m, n\} \leq 6$  时等号成立, [81] 将等式成立扩展到了  $K_{7,n}$ , ( $n \leq 10$ ).

**定理 7.6.10** <sup>[82]</sup>  $\text{cr}(K_{m,n}) \geq \left\lfloor \frac{m(m-1)}{5} \right\rfloor \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor$ .

**定理 7.6.11** <sup>[75]</sup> 如果  $5 \leq m \leq n$ , 则  $\text{cr}(K_{m,n}) \geq \left\lfloor \frac{m(m-1)n(n-2)}{20} \right\rfloor$ .

关于完全图和完全二部图的交叉数的进一步结果可参看文献 [83]~[85].

关于  $n$  维超立方体的交叉数以及图的叉乘积的交叉数的研究情况见文献 [86]~[108]. 特别值得指出的是, 确定  $\text{cr}(C_m \times C_n)$  的研究工作取得了很好的进展, 已经证明  $\text{cr}(C_m \times C_n) = (m-2)n$  在大多数情况下成立<sup>[106]</sup>.

求一般图的交叉数是一个 NPC 问题<sup>[109][110]</sup>, 有关图的平面嵌入的算法都可作为求交叉数的启发式算法. 文献 [111]~[115] 给出了一些新的近似算法, [116] 给出一个非多项式精确算法.

交叉数的一个推广是直边交叉数. 将图  $G$  画在平面上, 且要求每条边都必须画成直线段, 这种条件下边交叉的最少次数, 称为  $G$  的直边交叉数 (rectilinear crossing number), 记为  $\text{cr}'(G)$ .

显然  $\text{cr}'(G) \geq \text{cr}(G)$ . 文献 [117]~[118] 研究了图的交叉数和直边交叉数的关系. 有

关直边交叉数的较新结果和算法方面可参看 [119]~[121]. 关于直边交叉数的进一步变形见 [122]~[127].

关于交叉数研究的更多进展可参看文献 [17] 及 [128]~[161], 其中 [128]~[137] 是关于交叉数的综述文献, 交叉数的应用见 [68]、[136]~[138]. 文献 [150]~[161] 涉及交叉数概念的某些扩展.

### 三、图的厚度

对于一个可平面图, 我们可以将它边不相交地画在一个平面上. 对于一个不可平面图, 我们虽然无法做到这一点, 但总可以把它的边画在若干层平面上 (顶点可复制), 使得画在同一层平面上的边彼此不相交. 一个需要解决的问题是: 要达到这一目的至少需要使用几个平面? 这便涉及图的厚度的概念.

**定义 7.6.4** 设图  $G$  无孤立顶点,  $G_1, G_2, \dots, G_k$  是  $G$  的子图, 如果

$$E(G_1) \cup E(G_2) \cup \dots \cup E(G_k) = E(G) \text{ 且 } E(G_i) \cap E(G_j) = \emptyset, \quad (i \neq j; i, j = 1, 2, \dots, k)$$

则称  $G_1, G_2, \dots, G_k$  是  $G$  的一个子图分解, 进一步地, 若该子图分解中每个子图都是可平面图, 则称为一个可平面子图分解. 图  $G$  的可平面子图分解中所含子图的最少个数称为图  $G$  的厚度 (thickness), 记为  $t(G)$ , 即

$$t(G) = \min\{k | G_1, G_2, \dots, G_k \text{ 是 } G \text{ 的可平面子图分解}\}$$

厚度的一个明显的下界是  $t(G) \geq \left\lceil \frac{\varepsilon}{3\nu - 6} \right\rceil$ . 如果  $G$  是一个不含三角形的图, 则  $t(G) \geq \left\lceil \frac{\varepsilon}{2\nu - 4} \right\rceil$ . 厚度与交叉数显然有关系式  $t(G) \leq \text{cr}(G) + 1$ .

图的厚度在大规模集成电路布线和无线网络等方面有实际的应用<sup>[162][163]</sup>, 关于图的厚度的综述见文献 [17]、[77]、[164]~[165]. 确定图的厚度是一个  $NP$ -hard 问题<sup>[166]</sup>. [111] 及 [165]~[167] 给出了求图的厚度的几种近似算法.

图的厚度是一个较为困难的研究方向, 目前人们只确定出少数几个图类的厚度, 以及给出了厚度的若干上下界. 例如, 完全图的厚度确定如下 (见 [168]、[169]).

**定理 7.6.12**<sup>[168]</sup> 当  $n \geq 1, n \neq 9, n \neq 10$  时,  $t(K_n) = \left\lfloor \frac{n+7}{6} \right\rfloor$ ;  $t(K_9) = t(K_{10}) = 3$ .

完全二部图  $K_{m,n}$  的厚度在大多数情况下也已确定, 详细可参看文献 [170] 和 [171].

**定理 7.6.13**<sup>[170]</sup>  $t(K_{m,n}) = \left\lfloor \frac{m \cdot n}{2(m+n-2)} \right\rfloor$ , 除非  $m, n$  都是奇数且存在整数  $k$  使得  $n = \left\lfloor \frac{2k(m-2)}{m-2k} \right\rfloor$ . 特别地,  $t(K_{n,n}) = \left\lfloor \frac{n+5}{4} \right\rfloor$ .

**定理 7.6.14**<sup>[172]</sup>  $n$  维超立方体的厚度是  $t(Q_n) = \left\lfloor \frac{n+1}{4} \right\rfloor$ .

关于一般图, 有如下结果.

**定理 7.6.15** <sup>[173]</sup> 对任何图  $G$ , 厚度  $t(G) \leq \left\lfloor \sqrt{\frac{\varepsilon}{3}} + \frac{3}{2} \right\rfloor$ .

文献 [174] 和 [175] 研究了图的最大度和最小度与厚度的关系, 得到如下结果.

**定理 7.6.16** <sup>[174]</sup> 对任何图  $G$ ,  $\left\lceil \frac{\delta+1}{6} \right\rceil \leq t(G) \leq \left\lfloor \frac{\Delta}{2} \right\rfloor$ .

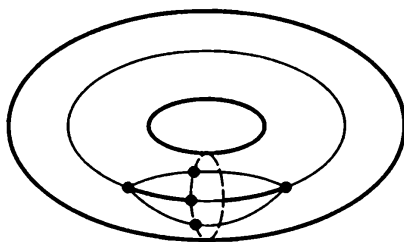
**定理 7.6.17** <sup>[175]</sup> 若  $\Delta(G) \leq 6$ , 则  $t(G) \leq 2$ .

图的书厚度 (book thickness) 是厚度的一种推广. 文献 [176]~[188] 研究了图的书厚度, 其中文献 [186] 和 [187] 证明了任何可平面图都可嵌入一个厚度为 4 页的书, 并给出了一个线性时间嵌入算法.

有关图的厚度的其他文献可参看文献 [189]~[205].

#### 四、图的曲面嵌入、图的亏格

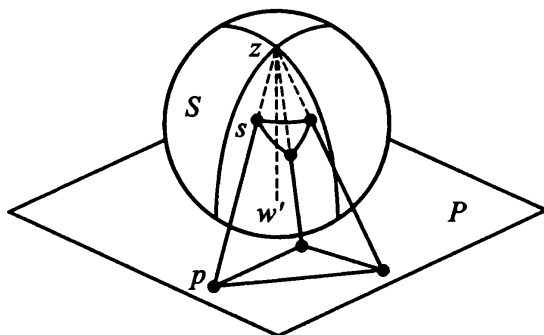
不可平面图有可能可嵌入其他曲面. 例如,  $K_5$  可嵌入环面 (如下图所示).



但是, 可以证明, 一个图可嵌入平面当且仅当它可嵌入球面.

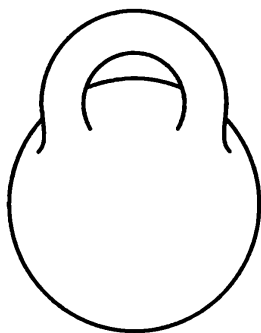
事实上, 将球面  $S$  放在一个平面  $P$  上, 设球面与该平面的切点为  $w$ ,  $w$  通过直径对应的球面上另一点称为极点, 记为  $z$ . 定义映射  $\varphi: S \rightarrow P$  如下: 设  $s$  是球面上任一点, 从极点  $z$  指向  $s$  点的射线交平面  $P$  于点  $p$ , 则  $\varphi(s) = p$ . 特别地,  $\varphi(z) = \infty$ . 显然  $\varphi$  是一个可逆映射.

设  $G'$  是图  $G$  在球面  $S$  上的一个嵌入, 总可以适当放置球面使得极点  $z$  不在  $G'$  的边和顶点上. 则  $G'$  通过  $\varphi$  映射到平面  $P$  上形成的像便是  $G$  的一个平面嵌入. 由  $\varphi$  的可逆性, 反之亦然.



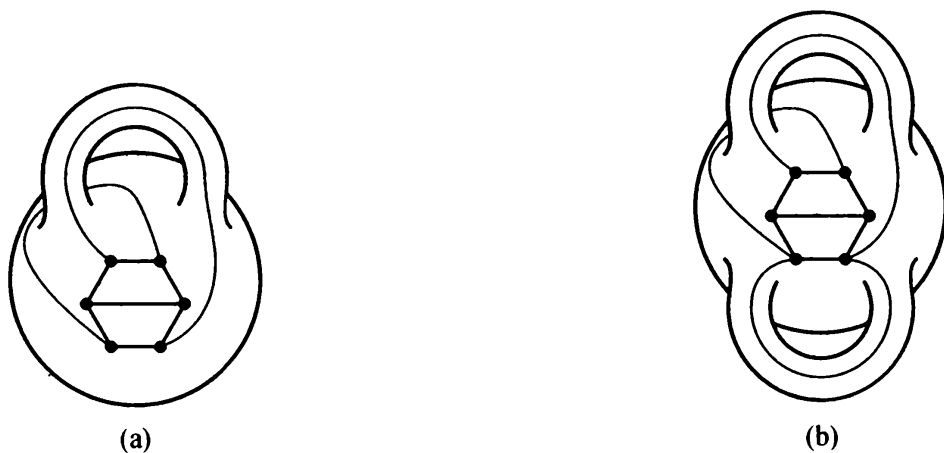
如果一个曲面  $S$  可看成由球面安上若干个柄得到的曲面 (或者另一个角度, 可看成由球体打了若干个洞得到的表面), 则称曲面  $S$  为紧致可定向 2-流形,  $S$  上柄 (或洞) 的个数称为该曲面的亏格. 这里, 一个曲面 “可看成” 另一个曲面的含义是: 一个曲面通过连续的拿捏伸缩后可得到另一个曲面.

平面和球面的亏格是零. 环面可看成由球面安装 1 个柄后得到的 (如图所示), 也可看成由球体打了一个洞后得到的表面, 因此其亏格是 1.



一个图  $G$  通常能 (边不相交地) 嵌入许多不同的曲面中, 这些曲面的亏格不尽相等. 图  $G$  可嵌入的曲面中亏格的最小值称为图  $G$  的亏格.

显然, 可平面图亏格是零. 下列两图分别表示  $K_{3,3}$  在亏格为 1 和亏格为 2 的曲面上的嵌入, 由此可见  $K_{3,3}$  的亏格为 1. 而由此前环面的例子可知,  $K_5$  的亏格也是 1.



设  $G$  是某个图在曲面  $S$  上的一个嵌入, 则由  $G$  的边在曲面上围成的每个连通片称为  $G$  的一个面. 这里 “连通片” 是指曲面的一部分, 其上任二点都能用曲面上不跨越  $G$  的边的曲线连接起来. 例如, 图 a 中  $K_{3,3}$  在亏格为 1 的曲面 (环面) 上的嵌入有三个面, 而图 b 中  $K_{3,3}$  在亏格为 2 的曲面 (双环面) 上的嵌入有两个面.

设  $G$  是图在曲面  $S$  上的一个嵌入,  $f$  是  $G$  的一个面. 如果  $f$  内的任一条连续的封闭曲线都能连续地变形或收缩为一个点, 则称  $f$  为 2-胞腔. 如果  $G$  的每个面都是 2-胞腔, 则称  $G$  是一个 2-胞腔嵌入. 直观地看, 2-胞腔就是没有洞的面. 例如, 图 a



中  $K_{3,3}$  在亏格为 1 的曲面 (环面) 上的嵌入是 2-胞腔嵌入, 而图 b 中  $K_{3,3}$  在亏格为 2 的曲面 (双环面) 上的嵌入不是 2-胞腔嵌入.

**定理 7.6.18** <sup>[206][207]</sup> 设  $G$  是亏格为  $g(G)$  的连通图, 则  $G$  在亏格为  $g(G)$  的曲面上的嵌入必定是 2-胞腔嵌入.

**定理 7.6.19 (Euler 公式在一般曲面嵌入上的推广)** 设  $G$  是某个连通图在亏格为  $r$  的曲面  $S$  上的一个 2-胞腔嵌入,  $G$  有  $\varphi$  个面, 则  $\nu - \varepsilon + \varphi = 2 - 2r$ .

有关该定理的证明可参看文献 [208].

**推论 7.6.1** 可 2-胞腔嵌入到亏格为  $r$  的曲面上的连通简单图最多有  $3(\nu - 2 + 2r)$  条边.

**证明** 设  $G$  是某图在亏格为  $r$  的曲面  $S$  上的一个 2-胞腔嵌入,  $G$  有  $\varphi$  个面. 与平面嵌入中一样, 对于  $G$  的一个面  $f$ , 我们将  $f$  的边界上含有的边数称为面  $f$  的度数, 记为  $\deg(f)$ . 由于  $G$  是连通简单图, 故  $G$  的每个面的度数至少为 3. 当我们将各个面的度数相加时, 每条边被计算了两次, 因此  $3\varphi \leq \sum_{i=1}^{\varphi} \deg(f_i) = 2\varepsilon$ . 由此可知,  $\varphi \leq \frac{2}{3}\varepsilon$ .

由定理 7.6.19,  $\nu - 2 + 2r = \varepsilon - \varphi \geq \varepsilon - \frac{2}{3}\varepsilon = \frac{1}{3}\varepsilon$ , 从而  $\varepsilon \leq 3(\nu - 2 + 2r)$ . 证毕.

由定理 7.6.18 和推论 7.6.1 可知, 若连通简单图  $G$  的亏格为  $g$ , 则  $G$  至多有  $3(\nu - 2 + 2g)$  条边.

容易证明关于图亏格的如下下界.

**定理 7.6.20** 设  $G$  是至少有 3 个顶点的连通简单图, 则其亏格  $g(G) \geq \frac{1}{6}(\varepsilon - 3\nu + 6)$ ; 进一步地, 若  $G$  不含三角形, 则  $g(G) \geq \frac{1}{4}(\varepsilon - 2\nu + 4)$ .

**证明** 这里只给出第一个不等式的证明, 第二个不等式类似可证.

对  $\nu = 3$  的图, 结论是显然的. 下面假定  $\nu \geq 4$ . 设  $G$  已嵌入在亏格为  $g(G)$  的曲面上. 按照定理 7.6.18, 该嵌入必定是 2-胞腔嵌入. 由推论 7.6.1,  $\varepsilon \leq 3(\nu - 2 + 2r)$ . 由此可得,  $g(G) \geq \frac{1}{6}(\varepsilon - 3\nu + 6)$ .

若  $G$  不含三角形, 则  $G$  嵌入亏格为  $g(G)$  的曲面后, 每个面的度数至少为 4, 因此  $4\varphi \leq \sum_{i=1}^{\varphi} \deg(f_i) = 2\varepsilon$ . 由定理 7.6.19,  $2 - 2g(G) = \nu - \varepsilon + \varphi \leq \nu - \varepsilon + \frac{\varepsilon}{2}$ , 从而  $g(G) \geq \frac{1}{4}(\varepsilon - 2\nu + 4)$ . 证毕.

文献 [209] 综述了使得上述不等式严格成立的图的有关结果.

某些特殊图类的亏格已被确定.

**定理 7.6.21** <sup>[210]</sup> 完全图  $K_\nu (\nu \geq 3)$  的亏格为  $g(K_\nu) = \left\lceil \frac{(\nu-3)(\nu-4)}{12} \right\rceil$ .

这个重要的定理称为**完全图定理**, 它在解决著名的 Heawood 地图染色问题中起着关键的作用. 有关这个定理的历史和证明可参看文献 [77]、[211]~[212].

**定理 7.6.22** <sup>[213]</sup> 完全二部图  $K_{m,n} (m, n \geq 2)$  的亏格为  $g(K_{m,n}) = \left\lceil \frac{(m-2)(n-2)}{4} \right\rceil$ .

关于完全  $m$  部图的亏格也已获得丰富的结果. 我们用  $K_{m(n)}$  表示每部分有  $n$  个顶点的完全  $m$  部图.

**定理 7.6.23** <sup>[214]</sup> 完全三部图  $K_{3(n)}$  的亏格为  $g(K_{3(n)}) = \binom{n-1}{2}$ .

该定理的最早证明见 [214], 简化的证明可参看文献 [77] 及 [215]、[216].

**定理 7.6.24** <sup>[217][218][219]</sup> 完全四部图  $K_{4(n)}$  的亏格为  $g(K_{4(n)}) = (n-1)^2, (n \neq 3)$ ; 而  $g(K_{4(3)}) = 5$ .

**定理 7.6.25** <sup>[220][221][222]</sup> 当  $m \not\equiv 2 \pmod{3}$  时,  $g(K_{m(2)}) = \frac{1}{3}(m-3)(m-1)$ .

$n$  维立方体  $Q_n$  的亏格已完全确定.

**定理 7.6.26** <sup>[223][224][225]</sup>  $n$  维立方体  $Q_n$  的亏格为  $g(Q_n) = (n-4)2^{n-3} + 1$ .

确定一般图的亏格是一个 NP 完全问题<sup>[226]</sup>. 有关图的亏格算法方面的文献见 [227]~[228]、[207].

文献 [229] 证明了亏格与厚度之间的关系:  $t(G) \leq g(G) + 1$ .

文献 [230] 改进了上述结果, 得到:  $t(G) \leq 6 + \sqrt{2g(G) - 2}$ .

连通图  $G$  的**最大亏格**  $g_M(G)$  定义为图  $G$  可 2-胞腔嵌入的所有曲面的亏格最大值.

文献 [231] 指出, 对任何连通图  $G$  以及任何满足  $g(G) \leq k \leq g_M(G)$  的正整数  $k$ ,  $G$  在亏格为  $k$  的曲面  $S_k$  上都可 2-胞腔嵌入.

最大亏格的一个常用的上界是  $g_M(G) \leq \left\lceil \frac{1}{2}\beta(G) \right\rceil$ , 此处  $\beta(G) = \varepsilon - \nu + 1$  称为图  $G$  的 Betti 数. 当且仅当  $G$  存在恰有 1 个面 ( $\beta(G)$  为偶数时) 或两个面 ( $\beta(G)$  为奇数时) 的 2-胞腔嵌入时, 上述不等式中的等号成立. 关于最大亏格的进一步研究可参看文献 [232]~[245].

## §7.7 平面图的面染色和四色猜想

### 一、平面图的面染色

**定义 7.7.1** 平面图  $G$  的面正常  $k$  染色 (proper face  $k$ -colouring) 是指  $k$  种颜色  $1, 2, \dots, k$  在  $G$  的面集合  $F(G)$  上的一种分配, 使得有公共边的面所染颜色不同. 用映射的观点看,  $G$  的面正常  $k$  染色是一个映射

$$c: F(G) \rightarrow \{1, 2, \dots, k\},$$

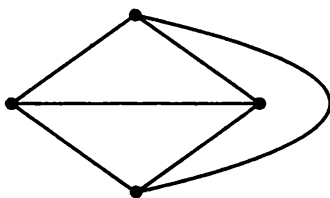
使得对每个  $i (i = 1, 2, \dots, k)$ ,  $c^{-1}(i)$  内的面两两无公共边.

**注** 若令  $F_i = c^{-1}(i) = \{f \in F(G) | c(f) = i\}$ , ( $i = 1, 2, \dots, k$ ), 则  $G$  的一个面正常  $k$  染色可看成是面集合的一种划分  $c = (F_1, F_2, \dots, F_k)$ , 其中每个  $F_i$  中的面彼此不相邻 (无公共边).

**定义 7.7.2** 若平面图  $G$  存在一种面正常  $k$  染色, 则称  $G$  是面  $k$  色可染 (face  $k$ -colourable).

**定义 7.7.3** 正整数  $\chi^*(G) = \min\{k | G \text{ 面 } k \text{ 色可染的}\}$  称为  $G$  的面色数 (face chromatic number).

例如,  $\chi^*(K_4) = 4$ .



**定理 7.7.1** 设  $G^*$  是平面图  $G$  的对偶图, 则  $\chi^*(G) = \chi(G^*)$ .

**证明** 由定义立即可知.

**四色猜想:** 任何地图只需用 4 种颜色来染色, 就可使得任何具有公共边界的两个区域染不同的颜色.

**四色猜想的图论表述:** 对任何平面图  $G$ ,  $\chi^*(G) \leq 4$ .

### 二、四色猜想的几种等价表述

**定理 7.7.2** 下列四个命题等价:

- (1) 每个平面图都是点 4 可染的;
- (2) 每个平面图是面 4 可染的;

(3) 每个 2 边连通 3 正则平面简单图是边 3 色可染的;

(4) 对任何平面图  $G$ , 色多项式  $P(G, 4) > 0$ .

**证明** (1) $\Rightarrow$ (2) 设  $G$  是一个平面图,  $G^*$  是其对偶图. 则  $G^*$  也是平面图. 由 (1),  $G^*$  是点 4 色可染的. 这种点 4 染色对应  $G$  的一个 4 色正常面染色.

(2) $\Rightarrow$ (3) 设  $G$  是 2 边连通 3 正则平面简单图. 由 (2),  $G$  有面正常 4 染色  $\varphi$ . 用  $c_0 = (0, 0), c_1 = (0, 1), c_2 = (1, 0), c_3 = (1, 1)$  来表示四种颜色. 构造  $G$  的一种边染色如下:

对于边  $e \in E(G)$ , 若  $e$  是某两个面  $f$  和  $g$  的公共边, 且  $f$  和  $g$  在  $\varphi$  下的颜色分别为  $c_i, c_j$ , 则给  $e$  染颜色  $C(e) = c_i + c_j \pmod{2}$ .

其中  $c_i$  与  $c_j$  的加法是向量相加, 对各分量分别模 2. 由于  $G$  是 2 边连通的, 故  $G$  中每条边  $e$  都必定是某两个面的公共边, 即  $G$  的每条边都被染色, 并且这种染色只用到颜色  $c_1, c_2, c_3$  [因  $c_0, c_1, c_2, c_3$  中不同的向量两两作加法 (模 2) 只能得到  $c_1, c_2, c_3$ ]. 此外, 由于  $G$  是 3 正则的, 故每个顶点  $u$  关联 3 条边, 在  $G$  的面 4 正常染色下, 这 3 条边所夹三个面的染色互不相同, 分别设为  $c_i, c_j, c_k$ , 则按边的染色规则, 这 3 条边应分别染上色  $c_i + c_j, c_j + c_k, c_k + c_i$ , 这是三种不同的色. 可见相邻的三边必染不同的色. 因此  $C$  是  $G$  的一种正常的边 3 染色.

(3) $\Rightarrow$ (1) 假设 (3) 成立. 下面来证明命题 (1) 成立.

设  $G$  是一个平面图, 我们希望证明  $G$  是点 4 色可染的. 不妨设  $G$  是简单图, 则存在一个三角剖分图  $H$ , 使得  $G$  是  $H$  的生成子图 (习题 7.24), 从而  $H$  的对偶图  $H^*$  是 2 边连通 3 正则的简单图 (习题 7.25). 由 (3),  $H^*$  是边 3 色可染的. 设  $c = (E_1, E_2, E_3)$  是  $H^*$  的一个边正常 3 染色. 对于  $i \neq j$ , 设  $H_{ij}^* = H^*[E_i \cup E_j]$ , (只有  $H_{12}, H_{13}, H_{23}$  三种导出子图). 因  $H^*$  中每个顶点都是 3 度的, 故染色  $c$  的每种颜色都在  $H^*$  的每个顶点处出现, 因而  $H_{ij}^*$  的每个顶点都是 2 度的, 这样的图是边不相交的圈的并.

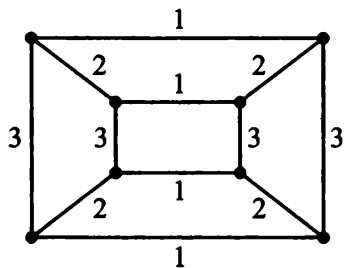


图 1.  $H^*$  及其一个边正常 3 染色

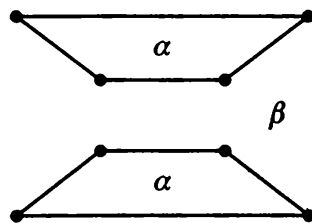
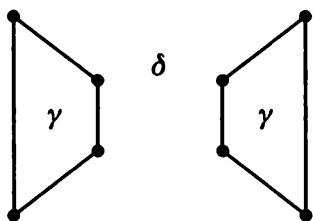
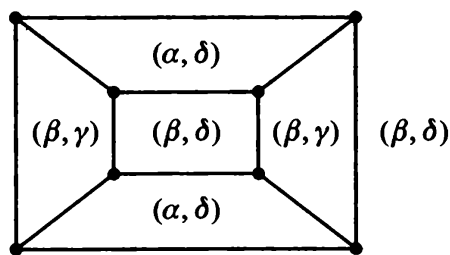


图 2.  $H_{12}^*$  及其一个面正常 2 染色

容易证明此时  $H_{ij}^*$  是面 2 色可染的. 令

$$\varphi_1 : F(H_{12}^*) \rightarrow \{\alpha, \beta\} \quad \varphi_2 : F(H_{23}^*) \rightarrow \{\gamma, \delta\}$$

分别是  $H_{12}^*$  和  $H_{23}^*$  的面正常 2 染色. 于是  $H^*$  的每个面  $f$  是  $H_{12}^*$  的某个面与  $H_{23}^*$  的某个面的交集的一部分. 所以在  $\varphi_1$  和  $\varphi_2$  下, 面  $f$  获得两种颜色  $x$  和  $y$ , 记为  $(x, y)$ ,

图 3.  $H_{23}^*$  及其一个面正常 2 染色图 4.  $H^*$  及其一个面正常 4 染色

其中  $x = \alpha$  或  $\beta$ ,  $y = \gamma$  或  $\delta$ . 定义四种新的颜色  $(\alpha, \gamma)$ ,  $(\alpha, \delta)$ ,  $(\beta, \gamma)$ ,  $(\beta, \delta)$ . 易见  $H^*$  的每个面被这四种颜色之一所染. 由于  $H^* = H_{12}^* \cup H_{13}^*$ , 而且  $H^*$  的相邻两个面得到的颜色对不会相同, 所以得到了  $H^*$  的一个面正常 4 染色. (例如在图 1 和图 2 所示的  $H_{12}^*$  和  $H_{23}^*$  的面正常染色下, 获得的  $H^*$  的面正常 4 染色如图 4 所示, 它实际上只用到 3 种不同的色).

因  $G$  是  $H$  的生成子图, 故  $\chi(G) \leq \chi(H) = \chi^*(H^*) \leq 4$ , 因此,  $G$  是点 4 色可染的. 从而 (3)  $\Rightarrow$  (1) 获证.

由以上结论可知命题 (1)、(2)、(3) 彼此等价. 最后, 由色数和色多项式的定义容易获知, 对任何平面图  $G$ ,  $\chi(G) \leq 4$  当且仅当其色多项式  $P(G, 4) > 0$ , 于是命题 (1) 与 (4) 等价. 证毕.

这个定理给出了四色猜想的四种等价表述:

- (1) 面染色表述: 对任何平面图  $G$ ,  $\chi^*(G) \leq 4$ ;
- (2) 点染色表述: 对任何平面图  $G$ ,  $\chi(G) \leq 4$ ;
- (3) 边染色表述: 每个简单 2 边连通 3 正则平面图是边 3 色可染的;
- (4) 色多项式表述: 对任何平面图  $G$ , 色多项式  $P(G, 4) > 0$ .

### 三、五色定理

**定理 7.7.3 (五色定理, Heawood, 1890)** 对于任何平面图  $G$ ,  $\chi(G) \leq 5$ .

**证明** 因为重边和环边不影响点色数, 故不妨设图  $G$  是平面简单图. 对顶点数  $\nu$  作数学归纳法.

$\nu \leq 5$  时, 定理结论显然成立.

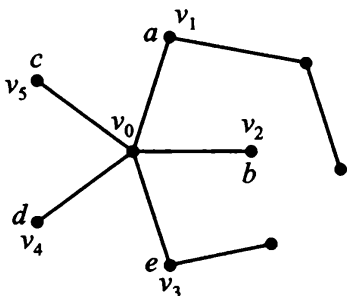
假设对  $\nu \leq n-1$  的所有平面图, 定理结论都成立. 考虑  $\nu = n$  的情形.

由定理 7.2.8, 存在  $v_0 \in V(G)$ , 使得  $d(v_0) \leq 5$ .

(1) 若  $d(v_0) \leq 4$ , 考虑  $G - v_0$ . 由归纳假设,  $\chi(G - v_0) \leq 5$ . 再给  $v_0$  染上与其邻点 (不多于四个) 相异的第五种色即可.

(2) 若  $d(v_0) = 5$ , 设  $v_1, v_2, v_3, v_4, v_5$  是  $v_0$  的邻点. 将它们按顺时针顺序画在平面上, 记  $G_0 = G - v_0$ . 由归纳假设,  $G_0$  可用五种色正常染色. 设顶点  $v_1, v_2, v_3, v_4, v_5$  在  $G_0$  中分别染以色  $a, b, c, d, e$ . 可以假定这五种色互不相同 (如果  $v_1, v_2, v_3, v_4, v_5$

中有染相同色的顶点, 则可用第五种色给  $v_0$  染色). 用  $G_{ac}$  表示  $G_0$  中染  $a$  色和染  $c$  色的顶点导出的子图.



(i) 若  $v_1$  与  $v_3$  分别在  $G_{ac}$  的两个连通分支中, 则在含有  $v_1$  的连通分支中将颜色  $a, c$  互换, 这样便可给顶点  $v_0$  染色  $a$ , 从而得到  $G$  的一个点正常 5 染色.

(ii) 若  $v_1$  与  $v_3$  在  $G_{ac}$  的同一个连通分支中, 则该连通分支中存在一条  $(v_1, v_3)$  路  $P$ , 在  $P$  上顶点的色  $a, c$  交替出现. 从图  $G$  中来看,  $v_0v_1 + P + v_3v_0$  是一个圈. 由于  $G$  已嵌入平面, 故  $v_2$  与  $v_4$  必分别处在该圈的内、外, 因此在  $G_0$  中由  $b, d$  两色的点导出的子图  $G_{bd}$  中,  $v_2$  与  $v_4$  分属于两个连通分支. (否则, 在  $G_{bd}$  中有  $(v_2, v_4)$  路  $P'$ , 它必与路  $P$  相交于一个公共顶点  $u$ , 由于  $u$  在  $P'$  上, 故应染有颜色  $b$  或  $d$ ,  $u$  也在  $P$  上, 又应染有颜色  $a$  或  $c$ , 这是不可能的). 不妨设  $v_2$  处在圈内, 这时可将  $G_{bd}$  中  $v_2$  所在的连通分支中  $b$  色与  $d$  色对换, 再给  $v_0$  染上颜色  $b$ , 这样便可得到  $G$  的一个点正常 5 染色. 证毕.

#### 四、四色猜想简介

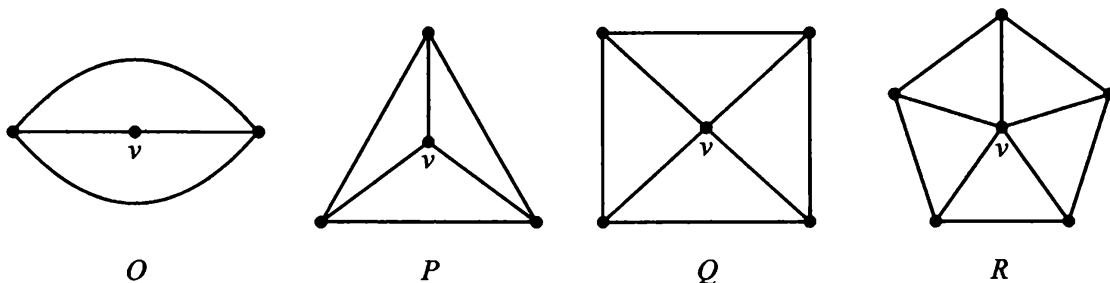
1852 年, 英国学生 Francis Guthrie 在给英国地图染色时发现, 只需要四种颜色就可以将相邻的区域区分开. 于是向他的哥哥 Frederick Guthrie 提出如下猜想: 任何地图只需用 4 种颜色来染色, 就可使得任何具有公共边界的两个区域染不同的颜色.

Frederick 转而请教他的老师、当时伦敦大学数学教授 A. de Morgan, A. de Morgan 后来又写信将此猜想告诉给爱尔兰数学家 W. R. Hamilton. 起初, 这个问题并未引起人们的注意, 但经过一段时间尝试之后, 大家发现这个猜想并不容易被证明或被否定. 1878 年, 当时伦敦数学会负责人 A. Cayley 通过文章《论地图着色》正式公布了这个猜想, 于是四色猜想开始引起数学界的广泛关注.

1879 年, Kempe 提出了对四色猜想的一个“证明”, 但在 1890 年, P. J. Heawood 指出了 Kempe 证明中的错误, 同时, 他利用 Kempe 的方法证明了五色定理. 巧合的是, 1880 年, P. G. Tait 曾断言: 每个 3 正则 3 连通平面图都是 Hamilton 图. 在这个假设下他给出了四色猜想的一个证明. 1891 年, Petersen 指出了 Tait 证明中的错误, 并利用 Tait 的证明思想证明了四色猜想等价于: “每个简单 2 边连通 3 正则平面图是边 3 色可染的”. 后来, W. T. Tutte 于 1946 年构造出一个 3 正则 3 连通非 Hamilton 平面图, 从而否定了 Tait 的断言.

### • Kempe 给出的“证明”

每个内部面都为 3 度面的平面图称为构形 (configuration). 例如, 下列四个平面图都是构形.



设  $F$  是由有限个构形组成的集合, 若任何三角剖分平面图至少含有  $F$  中一个构形, 则称  $F$  是不可免完备集 (unavoidable complete set).

由定理 7.2.8, 每个平面简单图的最小度  $\leq 5$ , 从而任何三角剖分平面图的最小度也不超过 5, 故上列四个构形组成的集合  $F = \{O, P, Q, R\}$  是一个不可免完备集.

此外, 如果有平面简单图  $G$  使得  $\chi(G) = 5$ , 但对任何阶数小于  $\nu(G)$  的平面图  $H$  均有  $\chi(H) \leq 4$ , 则称  $G$  是极小图 (minimal graph). 根据 5 色定理可知, 若四色猜想不真, 则必存在极小图.

1879 年, Kempe 给出了四色猜想的第一个“证明”. 他企图通过证明极小图不存在来证明四色猜想.“证明”使用反证法, 思路如下.

若存在极小图  $G$ , 则易证  $G$  是一个三角剖分图, 从而它必含有  $F = \{O, P, Q, R\}$  中的构形.

若  $G$  含有构形  $O$  或  $P$ , 则由反证法假设,  $\chi(G - v) \leq 4$  ( $v$  为构形的中心点). 设  $\pi = (V_a, V_b, V_c, V_d)$  是图  $G - v$  的一个点正常 4 染色. 则在染色  $\pi$  下,  $v$  的邻点集  $N_G(v)$  至多只需用到 3 种色, 因而 4 色中可有一种色给点  $v$  染色, 可见  $\pi$  也是图  $G$  的正常点 4 染色, 这与  $\chi(G) = 5$  矛盾.

若  $G$  含有构形  $Q$ , 则由反证法假设,  $\chi(G - v) \leq 4$  ( $v$  为构形  $Q$  的中心点). 设  $\pi = (V_a, V_b, V_c, V_d)$  是图  $G - v$  的一个正常点 4 染色, 并设  $N_G(v) = \{v_1, v_2, v_3, v_4\}$ . 不妨假定  $v_1 \in V_a, v_2 \in V_b, v_3 \in V_c, v_4 \in V_d$ , 且  $v_1, v_2, v_3, v_4$  作为  $v$  的邻点按顺时针方向排列. 与五色定理的证明中类似可知,  $v_1, v_4$  在导出子图  $G_{ad} = G[V_a \cup V_d]$  中以及  $v_2, v_3$  在导出子图  $G_{bc} = G[V_b \cup V_c]$  中不可能同时都是连通的 (否则  $G_{ad}$  中  $v_1v_4$  路与  $G_{bc}$  中  $v_2v_3$  路会相交). 不妨设  $v_1$  和  $v_4$  在  $G_{ad}$  中不连通. 这时交换  $G_{ad}$  中含  $v_1$  的连通分支中顶点的颜色, 则  $v_1$  被染上色  $d$ , 而  $v_2, v_3, v_4$  的色不变. 因此可用色  $a$  来给点  $v$  染色, 这样得到图  $G$  的一个正常点 4 染色, 这又与  $\chi(G) = 5$  矛盾.

Kempe 采用同样的方法“证明”了  $G$  也不可能含有构形  $R$ , 从而与  $F$  是不可免完备集矛盾. 这样便“证明”了极小图不存在, 四色猜想成立.

1890年, P. J. Heawood 举出一个反例说明上述证明中  $G$  不含构形  $R$  的结论并非总能成立, 从而否定了 Kempe 的证明.

• 寻找可约构形的不可免完备集

设  $S$  是一个构形, 若它不含在任何极小图中, 则称  $S$  为可约的 (reducible).

Kempe 实际上证明了  $O$ 、 $P$ 、 $Q$  是可约的, 但并没有能够正确地证明  $R$  也是可约的. 尽管如此, Kempe 的证明提供了一条证明四色猜想的途径: 寻找一个由可约构形组成的不可免完备集.

1904年, P. Wernicke 找到一个新的不可免完备集 (图 1).

1913年, G. D. Birkhoff 找出另一个不可免完备集 (图 2).

但他们都不能证明其中最后两个构形是可约的.

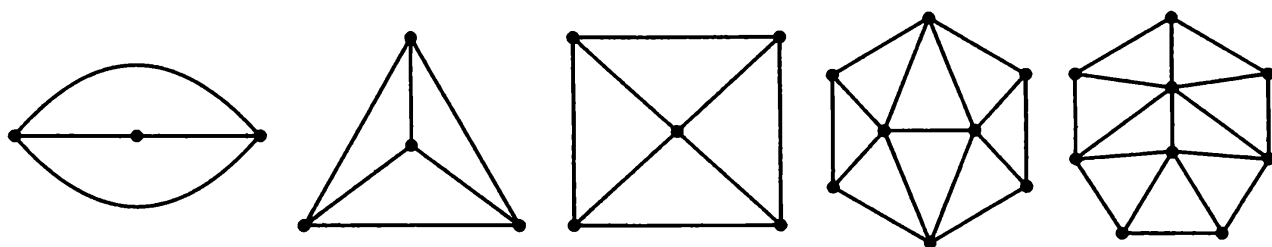


图 1

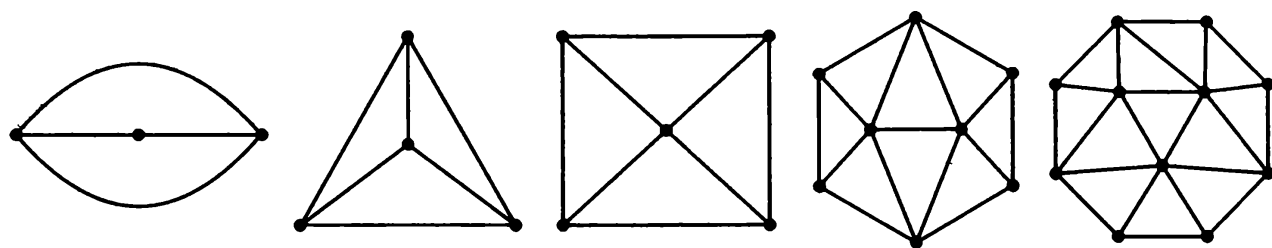


图 2

1913年, G. D. Birkhoff 又发现了一个新的可约构形 (图 3).

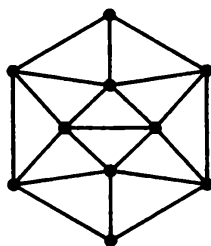


图 3

此后, 人们陆续发现了数千个可约构形.

1960年, H. Heesch 试图用计算机在已发现的可约构形中找出不可免完备集, 并提出一个判断给定的构形集合是否为不可免完备集的算法.



1961年, Y. Shimamoto 宣布找到了一个由可约构形组成的不可免完备集. 但 Whitney 和 Tutte 在 1972 年发现 Shimamoto 在利用计算机判断时误算了一个构形的可约性. 同时, 他们提出了构形可约性分类理论.

1976年, Apple 和 Haken 与 Koch 合作改进了 Heesch 的算法, 并研究了可约构形的范围, 最后利用电子计算机花了 1260 个机时, 进行了上百亿次逻辑判断, 找到了一个由 1936 个可约构形组成的不可免完备集, 从而宣布证明了四色猜想 [246]~[247].

如何使用常规证法而不借助计算机判断来证明四色猜想, 仍是许多人希望努力解决的问题.

### • Heawood 地图染色定理

四色猜想的一个推广是著名的 Heawood 地图染色猜想<sup>[248]</sup>. 它考虑嵌入曲面的图的面染色问题. 设  $S_r$  是一个亏格为  $r$  的曲面, 对 2-胞腔嵌入在其上的一个图  $G$  进行正常面染色所需要的最少的颜色数称为  $G$  的  $S_r$  面色数, 记为  $\chi_{S_r}(G)$ . 所有嵌入在  $S_r$  上的图的  $S_r$  面色数的最大值称为  $S_r$  的色数, 记为  $\chi(S_r)$ . Heawood 地图染色猜想是:

$$\chi(S_r) = \left\lfloor \frac{7 + \sqrt{1 + 48r}}{2} \right\rfloor \quad (r = 1, 2, 3, \dots).$$

这个猜想经过研究者们 80 余年曲折的努力后, 最终在 1968 年被 G. Ringel 和 J. W. T. Youngs 所证明<sup>[210][211]</sup>. 他们证明中的关键步骤是上一节介绍的完全图定理.

如果在上式中取  $r = 0$ , 则可获得四色猜想的结论, 因此四色猜想可看作 Heawood 地图染色猜想表达式的特例. 出人意料的是, 对 Heawood 地图染色猜想的证明却不能适用于四色猜想这个特例.

下面作为一个例子介绍 Heawood 地图染色猜想上界的证明. 下界的证明需要用到上一节介绍的完全图定理.

**例 7.7.1** 设  $S_r$  是一个亏格为  $r$  的曲面, 证明

$$\chi(S_r) \leq \left\lfloor \frac{7 + \sqrt{1 + 48r}}{2} \right\rfloor \quad (r = 1, 2, 3, \dots).$$

**证明** 只需证明对 2-胞腔嵌入在曲面  $S_r$  上的任一个图  $G$ , 都有

$$\chi_{S_r}(G) \leq \left\lfloor \frac{7 + \sqrt{1 + 48r}}{2} \right\rfloor$$

即可. 记  $\left\lfloor \frac{7 + \sqrt{1 + 48r}}{2} \right\rfloor = k$ . 只要能证明  $G$  有度不超过  $k-1$  的顶点, 就可以对  $\nu(G)$  用数学归纳法证明上述不等式. 对于顶点数不超过  $k$  的图, 所述结论自然成立, 因此可假定  $\nu(G) > k$ .

注意到  $k = \left\lfloor \frac{7 + \sqrt{1 + 48r}}{2} \right\rfloor$  是一元二次方程  $k^2 - 7k + (12 - 12r) = 0$  的根, 因此由方程式得  $k - 1 = 6 - \frac{12 - 12r}{k}$ . 由上节推论 7.6.1, 可 2-胞腔嵌入到亏格为  $r$  的曲面上的连通简单图最多有  $3(\nu - 2 + 2r)$  条边. 从而当  $r \geq 1$  时,  $G$  的平均度

$$\frac{2\varepsilon}{\nu} \leq \frac{6(\nu - 2 + 2r)}{\nu} \leq 6 - \frac{12 - 12r}{k} = k - 1.$$

这已经表明  $G$  有度不超过  $k - 1$  的顶点, 证毕.

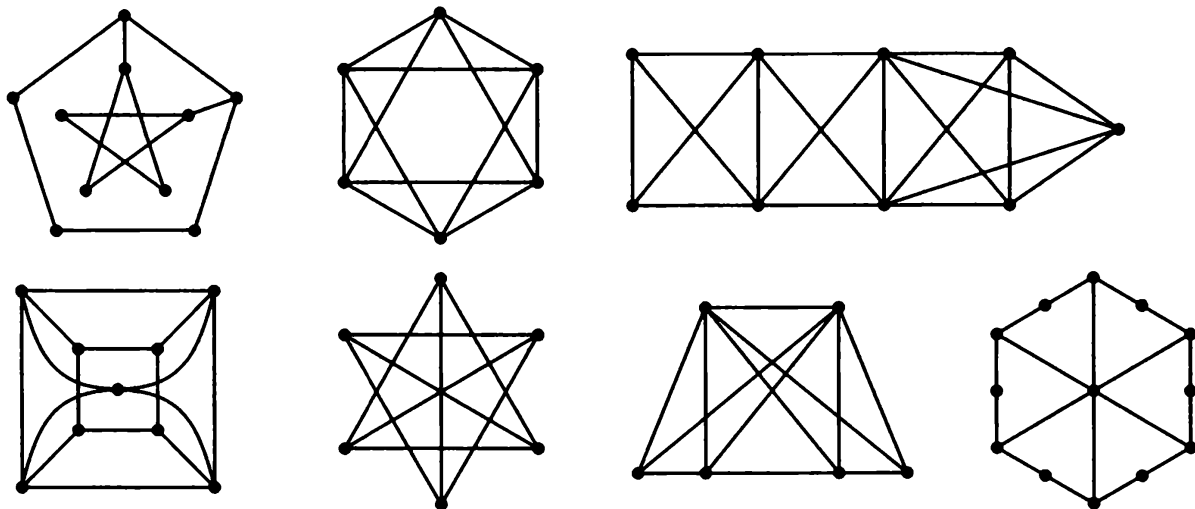
注意在此例结论中若取  $r = 0$  便得到四色定理  $\chi(S_0) \leq 4$ . 可惜在上述证明中的最后一个不等号当  $r = 0$  时不能成立. 因此结论对  $r = 0$  不适用.

有关 Heawood 地图染色定理的更多介绍可参看文献 [212]

有关四色问题的更多内容可参看文献 [249]~[251].

## 习题七

7.1 下列各图哪些是可平面图? 若是, 请画出其平面嵌入; 若不是, 给出证明.



7.2 给出  $K_{2,n}$  的一个平面嵌入, 说明  $K_{2,n}$  是可平面图.

7.3 证明: (1)  $K_5 - e$  是可平面图; (2)  $K_{3,3} - e$  是可平面图.

7.4 画出一个具有 7 条边和 5 个面的连通的平面图.

7.5 证明: 对任何正整数  $n$ , 不存在具有 7 个面且每个顶点的度都为 4 的  $n$  顶点连通平面图.

7.6 证明: 不存在这样的平面图: 它有 5 个面, 且任二面间均至少有一条公共边.

7.7 证明: 一个可平面图的所有平面嵌入都具有相等的面数.

7.8 证明: 一个图是可平面图当且仅当它的每个连通块是可平面图.

7.9 如果图  $G$  满足  $\varepsilon \leq 3\nu - 6$ , 则  $G$  一定是可平面图吗? 为什么?

- 7.10 对任意给定的图  $G$ , 将其每条边都剖分一次, 证明这样得到的图是一个二部图.
- 7.11 设  $G$  是有  $w$  个连通分支的平面图, 证明  $\nu - \varepsilon + \varphi = w + 1$ .
- 7.12 设  $G$  是不含三角形的平面简单图, 且  $\nu(G) \geq 3$ , 证明  $\varepsilon(G) \leq 2\nu(G) - 4$ .
- 7.13 证明: 顶点数少于 12 的可平面简单图必有度不超过 4 的顶点.
- 7.14 设  $G$  是可平面简单图, 证明:
- (1) 若  $\delta(G) = 2$ , 则  $\nu \geq 3$ , 且  $G$  中至少有 3 个顶点的度不超过 5.
  - (2) 若  $\delta(G) = 3$ , 则  $\nu \geq 4$ , 且  $G$  中至少有 4 个顶点的度不超过 5.
  - (3) 若  $\delta(G) = 4$ , 则  $\nu \geq 6$ , 且  $G$  中至少有 6 个顶点的度不超过 5.
  - (4) 若  $\delta(G) = 5$ , 则  $\nu \geq 12$ , 且  $G$  中至少有 12 个顶点的度等于 5.
- 7.15 (1) 设  $G$  是一个极大可平面图, 且顶点数  $\nu(G) \geq 4$ . 证明:  $G$  的最小度  $\delta(G) \geq 3$ .  
 (2) 若  $G$  是  $\nu \geq 3$  的极大平面图, 则  $G$  中至少有 4 个顶点的度不超过 5.  
 (3)  $\nu \geq 4$  的连通可平面简单图  $G$  至少有 4 个顶点的度不超过 5.
- 7.16 设连通的简单图  $G$  是可平面图, 且满足  $\varepsilon = 3\nu - 6$ . 证明  $G$  是极大可平面图.
- 7.17 (1) 设  $G$  是一个  $\nu(G) \geq 4$  的极大可平面图, 用  $n_i$  表示  $G$  中  $i$  度顶点的个数 ( $3 \leq i \leq \Delta(G)$ ), 证明:  $3n_3 + 2n_4 + n_5 = n_7 + 2n_8 + \cdots + (\Delta - 6)n_\Delta + 12$ , 其中  $\Delta = \Delta(G)$ .  
 (2) 设  $G$  是一个  $\nu(G) \geq 4$  的树, 证明:  $n_1 = n_3 + 2n_4 + 3n_5 + \cdots + (\Delta - 2)n_\Delta + 2$ , 其中  $n_i$  及  $\Delta$  含义同上.
- 7.18 设  $G$  是 3 正则平面图, 用  $\varphi_i$  表示  $i$  度面的个数. 证明:
- (1)  $12 = 5\varphi_1 + 4\varphi_2 + 3\varphi_3 + 2\varphi_4 + \varphi_5 - \varphi_7 - 2\varphi_8 - \cdots - (\nu - 6)\varphi_\nu$ .
  - (2)  $G$  中必存在度数小于 6 的面.
- 7.19 证明: 任何平面图都有偶数个度数为奇数的面.
- 7.20 设连通简单图  $G$  是平面图. 证明:
- (1) 若每个面至少由 4 条边围成, 则  $\varepsilon \leq 2\nu - 4$ ;
  - (2) 若每个面至少由 5 条边围成, 则  $3\varepsilon \leq 5\nu - 10$ .
- 7.21 设连通简单图  $G$  是有 6 个顶点 12 条边的平面图, 证明  $G$  的每个面都由 3 条边围成.
- 7.22 设  $G$  是有  $\nu$  个顶点  $\varphi$  个面的简单连通平面图,  $\nu \geq 3$ . 证明  $\varphi \leq 2\nu - 4$ .
- 7.23 证明: 若  $G$  是连通的平面图, 且  $\delta(G) \geq 3$ , 则  $G$  至少有一个面的度数不超过 5.
- 7.24 每个面的度数都是 3 的平面图称为三角剖分平面图. 证明顶点数至少为 3 的每个平面简单图都是某个三角剖分平面图的生成子图.
- 7.25 设  $G$  是至少含 4 个顶点的三角剖分平面简单图. 证明其对偶图  $G^*$  是 2 边连通 3 正则平面简单图.

- 7.26 证明定理 7.4.2.
- 7.27 画出超立方体  $Q_3$  的对偶图.
- 7.28 求轮图  $W_n$  的对偶图.
- 7.29 一个平面图如果与其对偶图同构, 则称之为自对偶图. 证明:  
(1) 若  $G$  为自对偶图, 则  $\varepsilon = 2\nu - 2$ .  
(2) 对每个  $n \geq 4$ , 构造一个  $n$  顶点自对偶图.
- 7.30 设  $G$  是平面图, 证明下列命题等价:  
(1)  $G$  是二部图;  
(2)  $G$  的每个面的长度都是偶数;  
(3)  $G$  的对偶图是 Euler 图.
- 7.31 证明: 如果非平凡平面图是 Euler 图, 则其对偶图是二部图.
- 7.32 证明: 当  $n \geq 8$  时, 圈  $C_n$  的补图  $\bar{C}_n$  是不可平面图.
- 7.33 证明: (1) 若  $G$  是  $\nu \geq 11$  的可平面简单图, 则其补图  $\bar{G}$  是非平面图. (W.T. Tutte 已证明此命题对  $\nu \geq 9$  成立).  
(2) 若  $G$  是  $\nu \geq 8$  的外可平面简单图, 则其补图  $\bar{G}$  不是外可平面图.
- 7.34 证明: (1) 若  $G$  是外可平面图, 则  $G$  的任何子图也是外可平面图;  
(2) 若  $G$  不是外可平面图, 则含有  $G$  作为子图的任何图也不是外可平面图;  
(3) 若  $G$  不是外可平面图, 则  $G$  的任何剖分图也不是外可平面图.
- 7.35 证明: (1) 极大外可平面图必为连通图; (2) 极大外可平面图无割点和割边.
- 7.36 证明: (1) 若  $G$  为极大外可平面图,  $\nu \geq 3$ , 则  $\varepsilon = 2\nu - 3$ ;  
(2) 若  $G$  为外可平面简单图,  $\nu \geq 3$ , 则  $\varepsilon \leq 2\nu - 3$ ;  
(3) 设  $G$  是外可平面简单图,  $\nu \geq 3$ , 则  $G$  是极大外可平面图当且仅当  $\varepsilon = 2\nu - 3$ .  
(4) 设  $G$  是外可平面简单图, 则  $G$  有度最多为 2 的顶点.
- 7.37 设  $G$  是一个外平面图且其内部面没有三角形, 证明  $2\varepsilon \leq 3\nu - 4$ .
- 7.38 平面上由  $n$  个点, 其中任意两个点之间的距离至少为 1. 证明在这  $n$  个点中, 距离恰好为 1 的点对不超过  $3n - 6$ .
- 7.39 证明: (1) 平面图  $G$  是面 2 可染色的当且仅当  $G$  是欧拉图;  
(2) 极大平面图  $G$  是点 3 可染色的当且仅当  $G$  是欧拉图;
- 7.40 证明: 设  $G$  是无割边的连通可平面图,  $\nu(G) \geq 2$ . 证明:  $G$  的平面嵌入是面 2 色可染的当且仅当  $G$  是 Euler 图.
- 7.41 证明: 每个 Hamilton 平面图都是面 4 可染色的.
- 7.42 不使用四色定理和五色定理, 证明: 每个可平面图都是点 6 色可染的.
- 7.43 证明: (1) 图  $G$  的厚度  $t(G) \geq \left\lceil \frac{\varepsilon}{3\nu - 6} \right\rceil$ .

(2) 如果  $G$  是一个不含三角形的图, 则  $t(G) \geq \left\lfloor \frac{\varepsilon}{2\nu - 4} \right\rfloor$ .

(3)  $t(K_n) \geq \left\lfloor \frac{n+7}{6} \right\rfloor$ .

7.44 求 Petersen 图的厚度.

7.45 求完全图  $K_6$ 、 $K_7$  以及完全二部图  $K_{3,4}$ 、 $K_{4,4}$  的交叉数.

7.46 证明: 对  $1 \leq \nu \leq 6$ , 有  $\text{cr}(K_\nu) = \frac{1}{4} \left\lfloor \frac{\nu}{2} \right\rfloor \left\lfloor \frac{\nu-1}{2} \right\rfloor \left\lfloor \frac{\nu-2}{2} \right\rfloor \left\lfloor \frac{\nu-3}{2} \right\rfloor$ .

7.47 证明: 对非平面简单图  $G$ , 交叉数  $\text{cr}(G) \geq \varepsilon - 3\nu + 6$ .

## 参考文献

- [1] K. Kuratowski, Sur le problème des courbes gauches en topologie, *Fund. Math.*, 15(1930), 271–283.
- [2] K. Wagner, Über eine Eigenschaft der ebenen Komplexe, *Math. Ann.*, 114(1937), 570–590.
- [3] 蒋长浩, 图论图网络流, 北京: 中国林业出版社, 2001 年.
- [4] 王树禾, 图论及其算法, 合肥: 中国科技大学出版社, 1990 年.
- [5] 徐俊明, 图论及其应用, 合肥: 中国科技大学出版社, 1998 年.
- [6] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan Press Ltd., 1976. (中译本: 图论及其应用. 吴望名、李念祖等译. 北京: 科学出版社, 1984 年).
- [7] D. B. West, *Introduction to Graph Theory*, Prentice-Hall, Inc., 2001.

## 平面图专著和综述

- [8] L. W. Beineke and R. J. Wilson, *Selected Topics in Graph Theory*, Academic Press, London, 1978.
- [9] S. Stahl, embeddings of a graph- a survey, *J. Graph Theory*, 2(1978) 275–298.
- [10] J. L. Gross and T. W. Tucker, *Topological Graph Theory*, Wiley, New York, 1987.
- [11] T. Nishizeki and N. Chiba, *Planar Graphs: Theory and Algorithms*, volume 32 of *Annals of Discrete Mathematics*, North-Holland, 1988.
- [12] Walter Schnyder, *Planar Graphs and Poset Dimension*, *Order*, 5(1989)323–343.
- [13] T. Nishizeki, Planar graph problems, in *Computational Graph Theory* (Gottfried Tinhofer etc., editors), volume 7 of *Computing Supplementum*, Springer-Verlag Wien New York, 1990, pp53–68.
- [14] A. K. Kelmans, Graph planarity and related topics, in *Graph Structure Theory* (Neil Robertson and Paul Seymour, editors), *Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference on Graph Minors*, Seattle, Washington, USA, June 22–July 5,

1991, volume 147 of Contemporary Mathematics, pp137–147, American Mathematical Society, 1993.

- [15] Y. P. Liu, *Embeddability in Graphs*, Kluwer, Boston, 1995.
- [16] D. Archdeacon, Topological Graph Theory, A Survey, *Congr. Numerantium*, 115(1996), 5–54, available in <http://www.emba.uvm.edu/~archdeac/papers/survey.ps>
- [17] A. Liebers, Planarizing graphs-a survey and annotated bibliography, *Journal of Graph Algorithms and Applications*, 5: 1(2001) 1–74. <http://www.inf.uni-konstanz.de/~liebers/>
- [18] F. Brandenburg, D. Eppstein, M. T. Goodrich, S. Kobourov, G. Liotta and P. Mutzel, Selected Open Problems in Graph Drawing, in *Graph Drawing, GD2003*, Lecture Notes in Computer Science 2912, Springer-Verlag, 2004, pp515–539.

### 平面图刻画

- [19] Y. P. Liu, Boolean planarity characterization of graphs, *数学学报 (英文版)*, 4(1988), 316–329.
- [20] Graham R. Brightwell and Edward R. Scheinerman, Representations of planar graphs, *SIAM J. Discrete Math.*, 6(1993) 214–229.
- [21] Y. Colin de Verdiere, On a new graph Invariant and a criterion for planarity, in *Graph Structure Theory* (Neil Robertson and Paul Seymour, editors), *Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference on Graph Minors*, Seattle, Washington, USA, June 22-July 5, 1991, volume 147 of Contemporary Mathematics, pp137–147, American Mathematical Society, 1993.
- [22] Dan Archdeacon, C. Paul Bonnington and Charles H. C. Little, An algebraic characterization of planar graphs, *J. of Graph Theory*, 19(1995) 237–250.
- [23] Hubert de Fraysseix and P. Ossona de Mendez, Planarity and edge poset dimension, *Europ. J. Combinatorics*, 17(1996) 731–740.
- [24] A. Schrijver, Minor-monotone Graph Invariants, in *Surveys in Combinatorics 1997* (Rosemary A. Bailey, editor), Papers presented by the invited lectures at the 16<sup>th</sup> British Combinatorial Conference, 1997, volume 241 of London Mathematical Society Lecture Note Series, Cambridge University Press, Cambridge-New York, 1997.
- [25] H. Tamaki and T. Tokuyama, A characterization of planar graphs by pseudo-line arrangements, in *Algorithms and Computation, 8<sup>th</sup> International Symposium, ISAAC'97*, Springer-Verlag, Lecture Notes in Computer Science, vol.1350(1997) 133–142.
- [26] C. Gutwenger, P. Nutzel, R. Weiskircher, Inserting an edge into a planar graph, *Algorithmica*, 41(4)(2005), 289–308.
- [27] S. -H. Hong, P. Eades, Drawing planar graphs symmetrically, II: biconnected planar graphs, *Algorithmica*, 42(2)(2005), 159–197.
- [28] S. -H. Hong, P. Eades, Drawing planar graphs symmetrically, III: oneconnected planar graphs, *Algorithmica*, 44(1)(2006), 67–100.

- [29] N. Bonichon, S. Felsner, M. Mosbah, Convex drawing of 3-connected plane graphs, *Algorithmica*, 47(4)(2007), 399–420.
- [30] A. Grigoriev, H. L. Bodlaender, Algorithms for graphs embeddable with few crossings per edge, *Algorithmica*, 49(1)(2007), 1–11.
- [31] Fabrici, Light graphs in families of outerplanar graphs. *Discrete Mathematics*, 307(2007), 866–872.
- [32] R. E. L. Aldred, C. Thomassen, On the maximum number of cycles in a planar graph, *Journal of Graph Theory*, 57(2008), 255–264.

### 平面图判别算法

- [33] John E. Hopcroft, and R. E. Tarjan, Efficient planarity testing, *J. of the Association for Computing Machinery*, 21(1974) 549–568.
- [34] K. S. Booth and G. S. Lueker, Testing for the consecutive ones property, interval graphs and planarity using PQ-tree algorithms, *J. Computer and System Sciences*, 13(1976) 335–379.
- [35] S. G. Williamson, Embedding graphs in the plane – algorithmic aspects, in *Combinatorial Mathematics, Optimal Designs and Their Applications*(J. Srivastava, editor), North-Holland, Annals of Discrete Mathematics, 6(1980) 349–384.
- [36] K. Mehlhorn, *Data Structures and Algorithms 2: Graph Algorithms and NP- Completeness*, Springer, 1984.
- [37] N. Chiba, T. Nishizeki, S. Abe, and T. Ozawa, A linear algorithm for embedding planar graphs using PQ-trees, *J. Computer and System Sciences*, 30(1985) 54–76.
- [38] Y. P. Liu, A new approach to the linearity of testing planarity of graphs, *应用数学学报 (英文版)*, 4(1988), 257–265.
- [39] K. Mehlhorn, P. Mutzel and S. Näher, *An Implementation of the Hopcroft and Tarjan Planarity Testing and Embedding Algorithm*, Technical Report MPI-I-93-151, Max-Planck-Institut für Informatik, Imstadtswald, 66123 Saarbrücken, Germany, 1993. <http://www.mpi-sb.mpg.de/~mutzel/mpireports/MPI-I-93-151.ps.gz>
- [40] P. Mutzel, *The Maximum Planar Subgraph Problem*, PhD thesis, Universität zu Köln, 1994. <http://www.mpi-sb.mpg.de/~mutzel/koelnreports/>
- [41] K. Mehlhorn, and P. Mutzel, On the embedding phase of the Hopcroft and Tarjan Planarity Testing Algorithm, *Algorithmica*, 16(1996) 233–242.
- [42] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis, Algorithms for drawing graphs: an annotated bibliography, *Computational Geometry*, 4(1999) 235–282.
- [43] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice Hall.1999.
- [44] J. Boyer and W. Myrvold, Stop minding your P’s and Q’s: a simplified  $O(n)$  planar

embedding algorithm, in *Proceedings of the 10<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA'99, 1999, 140–146.

- [45] Wei-Kuan Shih and Wen-Lian Hsu, A new planarity test, *Theoretical Computer Science*, 223(1999) 179–191.

### 最大可平面算法

- [46] M. Yannakakis, Node- and edge-deletion NP-complete problems, in *Proceedings of the 10<sup>th</sup> Annual ACM Symposium on Theory of Computing*, STOC'78, 1978, pp253–264.
- [47] P. C. Liu and R. C. Geldmacher, On the deletion of nonplanar edges of a graph, in *Proceedings of the 10<sup>th</sup> Southeastern Conference on Combinatorics, Graph Theory and Computing*, Boca Raton, Florida, USA, 1979, part 2, volume 24 of *Congressus Numerantium*, 1979, pp727–738.
- [48] T. Watanabe, T. Ae, and A. Nakamura, On the NP-hardness of edge-deletion and contraction problems, *Discrete Applied Math.*, 6(1983) 63–78.
- [49] G. Călinescu, C. G. Fernandes, U. Finkler, and H. Karloff, A better approximation algorithm for finding planar subgraphs, *J. Algorithms*, 27(1998) 269–302.
- [50] M. E. Dyer, L. R. Foulds, and A. M. Frieze, Analysis of heuristics for finding a maximum weight planar subgraph, *European J. of Operational Research*, 20(1985) 102–114.
- [51] L. R. Foulds, P. B. Gibbons, and J. W. Giffin, Facilities layout adjacency determination: an experimental comparison of three graph theoretic heuristics, *Operations Research*, 33(1985) 1091–1106.
- [52] R. Jayakumar, K. Thulasiraman, and M. N. S Swamy,  $O(n^2)$  algorithms for graph planarization, *IEEE Trans. on Computer-aided Design*, 8(1989) 257–267.
- [53] Y. Takefuji and Kuo Chun Lee, A near-optimum parallel planarization algorithm, *Science*, 245(1989) 1221–1223.
- [54] R. J. Cimikowski, Branch-and-bound techniques for the maximum planar subgraph problem, *Intern. J. Computer Math.*, 53(1994) 135–147.
- [55] O. Goldschmidt and A. Takvorian, An efficient graph planarization two-phase heuristic, *Networks*, 24(1994) 69–73.
- [56] M. Jünger and P. Mutzel, Maximum planar subgraphs and nice embeddings: practical layout tools, *Algorithmica*, 16(1996) 33–59.
- [57] Mauricio G. C. Resende and Celso C. Ribeiro, A GRASP for graph planarization, *Networks*, 29(1997) 173–189.
- [58] R. J. Cimikowski, An analysis of heuristics for graph planarization, *J. Information Optimization Sciences*, 18(1997) 49–73.



## 极大可平面图算法

- [59] T. Chiba, I. Nishioka, and I. Shirakawa, An algorithm of maximal planarization of graphs, in *Proceedings IEEE Symposium on Circuits and Systems, ISCAS'1979*, 1979, pp649–652.
- [60] G. Di Battista and R. Tamassia, Incremental planarity testing, in *Proceedings of the 30<sup>th</sup> Annual Symposium on Foundations of Computer Science, FOCS'89*, 1989, pp436–441.
- [61] Leizhen Cai, Xiaofeng Han, and Robert E. Tarjan, An  $O(m \cdot \log n)$ -time algorithm for the maximal planar subgraph problem, *SIAM J. Comput.*, 22(1993) 1142–1162.
- [62] G. Kant, An  $O(n^2)$  maximal planarization algorithms based on PQ-trees, *Technical Report, RUU-CS-92-03, Dept. of Computer Science, Utrecht University*, 1992.
- [63] S. Leipert, The problem of computing a maximal planar subgraph using PQ-trees is still not solved, in *Special Proceedings for students at ECMI'94*, Kaiserslautern, 1994.
- [64] H. N. Djidjev, A linear algorithm for the maximal planar subgraph problem; in *Proceedings of 4<sup>th</sup> International Workshop on Algorithms and Datastructures (S.G. Akl, F. Dehne, J-R. Sack, and N. Santoro, editors)*, WADS'95, Springer-Verlag, Lecture Notes in Computer Science, vol.955, 1995, pp369–380.
- [65] Wen-lian Hsu, A linear time algorithm for finding maximal planar subgraphs, in *Algorithms and Computation, 6<sup>th</sup> International Symposium (J. Staples, P. Eades, N. Katoh, and A. Moffat, editors)*, ISAAC'95, Springer-Verlag, Lecture Notes in Computer Science, vol. 1004, 1995, pp 352–361.
- [66] Luérbio Faria, Celina Miraglia Herrera de Figueiredo, and Candido Ferreira Xavier de Mendonça Neto, Splitting number is NP-complete, *Discrete Applied Math.*, 108(2001) 65–83.

## 交叉数

- [67] M. Ajtai, V. Chvátal, M. M. Newborn, and E. Szemerédi, Crossing-free subgraphs, *Theory and Practice of Combinatorics, Ann. Discr. Math.*, 12(1982) 9–12.
- [68] F. T. Leighton, *Complexity Issues in VLSI*, MIT Press, Cambridge, 1983.
- [69] János Pach, and Géza Tóth, Graphs drawn with few crossings per edge, *Combinatorica*, 17(1997) 427–439.
- [70] N. Ajtai, V. Chvátal, M. Newborn, and E. Szemerédi, Crossing-free subgraphs, *Annals of Discrete Mathematics*, 12(1982), 9–12.
- [71] P. C. Kainen, A lower bound for crossing numbers of graphs with application to  $K_n$ ,  $K_{p,q}$ , and  $Q(d)$ . *Journal of Combinatorial Theory, Series B*, 12(1972) 287–298.

## 完全图、完全二部图的交叉数

- [72] O. Aichholzer, F. Aurenhammer, and H. Krasser, On the crossing number of complete graphs, in *Proc. 18<sup>th</sup> Ann. ACM Symp. Computational Geometry, Barcelona, Spain, 2002*, 19–24.

- [73] J. Blazek and N. Koman, A minimal problem concerning complete plane graphs, in *Theory of Graphs and its Applications* (M. Fiedler, editor), Czechoslovak Academy of Sciences, Prague, 1964, pp113–117.
- [74] R. K. Guy, Latest results on crossing numbers, in *Recent Trends in graph Theory* (M. Capobianco, J. B. Frechen, and M. Krolík, editors), *Proceedings of the First New York City Graph Theory Conference*, June 11–13, 1970, Springer-Verlag, Lecture Notes in Mathematics, vol. 186, 1971, pp143–156.
- [75] R. K. Guy, Crossing numbers of Graphs, in *Graph Theory and Applications* (Y. Alavi, D.R. Lick, and A.T. White, editors), *Proceedings of the Conference at Western Michigan University*, May 10–13, 1972, Springer-Verlag, Lecture Notes in Mathematics, vol.303, 1972, pp111–124.
- [76] F. T. Leighton, New lower bound techniques for VLSI, *Math. Systems Theory*, 17(1984) 47–70.
- [77] A. T. White and L.W. Beineke, Topological graph theory, in *Selected Topics in Graph Theory* (L. W. Beineke and R.J. Wilson, editors), Academic Press, New York, 1978, pp15–49.
- [78] K. Zarankiewicz, On a problem of P. Turan concerning graphs, *Fundamenta Mathematicae*, 41(1954), 137–145.
- [79] R. K. Guy, The decline and fall of Zarankiewicz's theory, in *Proof Techniques in Graph Theory* (F. Harary editor), Academic Press, New York, 1969, 63–69.
- [80] D. J. Kleitman, The crossing number of  $K_{5,n}$ , *J. of Combinatorial Theory*, 9(1970), 315–323 .
- [81] D. R. Woodall, Cyclic-order graphs and Zarankiewicz's crossing-number conjecture, *J. of Graph Theory*, 17(1993)657–671.
- [82] R. K. Guy, Sequences associated with a problem of Turán and other problems, in *Proc. Combin. Conf. Balatonfüred 1969*, Bolyai János Matematikai Tarsulat, 1970, pp553–569.
- [83] R. Bruce Richter and Carsten Thomassen, Relations between crossing numbers of complete and complete bipartite graphs, *The American Mathematical Monthly*, 104(1997) 131–137.
- [84] Shengjun Pan, R. B. Richter, The crossing number of  $K_{11}$  is 100, *Journal of Graph Theory*, 56(2007), 128–134.
- [85] Yuanqiu Huang, Tinglei Zhao, The crossing number of  $K_{1,4,n}$ . *Discrete Mathematics*, 308(2008), 1634–1638.

#### 叉乘积的交叉数

- [86] P. C. Kainen, A lower bound for crossing numbers of graphs with application to  $K_n$ ,  $K_{p,q}$ , and  $Q(d)$ , *J. of Combinatorial Theory, Series B*, 12(1972) 287–298.

- [87] P. C. Kainen, On the stable crossing number of cubes, *Proceedings of the American Mathematical Society*, 36(1972) 55–62.
- [88] T. Madej, Bounds for the crossing number of the  $n$ -cube, *J. of Graph Theory*, 15(1991) 81–97.
- [89] O. Sýkora, and I. Vrt'o, On the crossing number of the hypercube and the cube connected cycles, in *Proceedings 17<sup>th</sup> International Workshop on Graph Theoretic Concepts in Computer Science* (G. Schmidt and R. Berghammer, editors), WG'91, Springer-Verlag, Lecture Notes in Computer Science, vol.570, 1992, pp214–218.
- [90] Luérbio Faria and Celina Miraglia Herrera de Figueiredo, On Eggleton and Guy conjectured upper bound for the crossing number of the  $n$ -cube, *Math. Slovaca*, 50(2000) 271–287.
- [91] Giustina Pica, Tomaz Pisanski, and Aldo G. S. Ventre, Cartesian products of graphs and their crossing numbers, in *Proceedings Int. Conf. Finite Geom. Comb. Struct.*, Bari, Italy, 1984, Combinatorics'84, North-Holland, Annals of Discrete Mathematics, vol.30, 1986, pp339–346.
- [92] L. W. Beineke, R. D. Ringeisen, On the crossing number of product of cycles and graphs of order four, *J. Graph Theory*, 4(1980), 145–155.
- [93] D. Bokal, On the crossing number of Cartesian products with paths, Available electronically at <http://www.ijp.si/ftp/pub/preprints/ps/2005/pp1000.pdf>
- [94] Marián Klešč, On the crossing numbers of Cartesian products of stars and paths or cycles, *Math. Slovaca*, 41:2(1991) 113–120.
- [95] Marián Klešč, The crossing numbers of products of paths and stars with 4-vertex graphs, *J. of Graph Theory*, 18(1994) 605–614.
- [96] Marián Klešč, The crossing numbers of certain Cartesian products, *Discussiones Mathematicae - Graph Theory*, 15(1995) 5–10.
- [97] Marián Klešč, The crossing numbers of  $K_{2,3} \times P_n$  and  $K_{2,3} \times S_n$ , *Tatra Mountains Math. Publ.*, 9(1996) 51–56.
- [98] Marián Klešč, The crossing number of  $K_{2,3} \times C_3$ , *Discrete Math.*, 251(2002), 109–117.
- [99] Marián Klešč, R. Bruce Richter and Ian Stobert, The crossing number of  $C_5 \times C_n$ , *J. of Graph Theory*, 22(1996) 239–243.
- [100] R. D. Ringeisen and L. W. Beineke, The crossing number of  $C_3 \times C_n$ , *J. of Combinatorial Theory, Series B*, 24(1978) 134–136.
- [101] Alice M. Dean and R. Bruce Richter, The crossing number of  $C_4 \times C_4$ , *J. of Graph Theory*, 19: 1(1995) 125–129.
- [102] R. Bruce Richter and Carsten Thomassen, Intersections of curve systems and the crossing number of  $C_5 \times C_5$ , *Discrete Comput. Geom.*, 13(1995) 149–159.

- [103] M. Anderson, R. B. Richter and P. Rodney, The crossing number of  $C_7 \times C_7$ , *Congr. Numerantium*, 125 (1997) 97–117.
- [104] F. Shahrokhi, O. Sýkora, L. A. Székely and I. Vrt'o, Intersection of curves and crossing number of  $C_m \times C_n$  on surfaces, *Discrete Comput. Geom.*, 19(1998) 237–247.
- [105] L. Y. Glebsky, G. Salazar, The crossing number of  $C_m \times C_n$  is as conjectured for  $n \geq m(m+1)$ , *J. Graph Theory*, 47(2004), 53–72.
- [106] L. Y. Glebsky, G. Salazar, The conjecture  $cr(C_m \times C_n) = n(m-2)$  is true for all but finitely many  $n$ , for each  $m$ , *submitted*.
- [107] D. Bokal, On the crossing numbers of Cartesian products with paths, *Journal of Combinatorial Theory*, (Ser.B), 97 (2007), 381–384.
- [108] D. Bokal, On the crossing numbers of Cartesian products with trees, *Journal of Graph Theory*, 56(2007), 287–300.

#### 交叉数近似算法

- [109] M. R. Garey and D. S. Johnson, Crossing number is NP-complete, *SIAM J. Alg. Discrete Methods*, 4(1983) 312–316.
- [110] P. Hliněný, Crossing number is hard for cubic graphs, in *Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science 3153, Springer, Berlin, 2004, 772–782; also see *J. of Combinatorial Theory Ser. B*, 96(2006), 455–471.
- [111] T. Odenthal and M. Scharbrodt, Maximal planarization as a tool for approximating thickness and crossing number, in *Student Proceedings 8<sup>th</sup> conference of the European Consortium for Mathematics in Industry*, ECMI'94, Kaiserslautern, Germany, 1994, pp137–151.
- [112] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis, *Graph Drawing: Algorithms for Visualization of Graphs*, Prentice Hall, 1999.
- [113] R. Cimikowski, Algorithms for the fixed linear crossing number problem, *Discrete Applied Mathematics*, 122(2002) 93–115
- [114] G. Even, S. Guha, and B. Schieber, Improved approximations of crossings in graph drawings and VLSI layout areas, in *Proc. 32<sup>nd</sup> Annual ACM Symposium on the Theory of Computing, STOC'00*, 2000, 296–305; also see *SIAM J. Computing*, 32:1(2002) 231–252.
- [115] M. Grohe, Computing crossing numbers in quadratic time, in *Proc. 33<sup>rd</sup> Annual ACM Symposium on the Theory of Computing, STOC'01*, 2001, 231–236; also see *J. Comput. Syst. Sci.*, 68(2004) 285–302.
- [116] C. Buchheim, D. Ebner, M. Jünger, G. W. Klau, P. Mutzel, and R. Weiskircher, Exact crossing minimization, in *Proc. of 13<sup>th</sup> Intl. Symposium on Graph Drawing*, Lecture Notes in Computer Science 3843, Springer, Berlin, 2006.

## 直边交叉数

- [117] D. Bienstock and N. Dean, New results on rectilinear crossing numbers and plane embeddings, *J. of Graph Theory*, 16(1992) 389–398.
- [118] D. Bienstock and N. Dean, Bounds for rectilinear crossing numbers, *J. of Graph Theory*, 17(1993) 333–348.
- [119] A. Brodsky, S. Durocher, E. Gether, The rectilinear crossing number of  $K_{10}$  is 62, *The Electronic J. of Combinatorics*, 8 (2001), Research Paper 23.
- [120] F. Aurenhammer, On the Rectilinear Crossing Number, available electronically at <http://www.igi.tugraz.at/auren/>
- [121] J. T. Thorpe, F. C. Harris, A parallel stochastic optimization algorithm for finding mappings of the rectilinear minimal crossing number, *Ars Combinatorica*, 43 (1996) 135–148.
- [122] F. Shahrokhi, O. Sýkora, L. A. Székely, and I. Vrt'ó, Book embeddings and crossing numbers, in *Proceedings 20<sup>th</sup> International Workshop on Graph-Theoretic Concepts in Computer Science* (Ernst W. Mayr, G. Schmidt, and G. Tinhofer, editors), WG'94, Springer-Verlag, Berlin, 1995, pp256–268.
- [123] F. Shahrokhi, L. A. Székely, O. Sýkora, and I. Vrt'ó, The book crossing number of graphs, *J. Graph Theory*, 21(1996)413–424.
- [124] Michael Jünger and Petra Mutzel, 2-Layer straightline crossing minimization: performance of exact and heuristic algorithms, *Journal of Graph Algorithms and Applications*, 1(1997)1–25.
- [125] Michael Jünger, Eva K. Lee, Petra Mutzel, and Thomas Odenthal, A polyhedral approach to the multi-layer crossing minimization problem, in *Proceedings of the Symposium on Graph Drawing* (Giuseppe Di Battista, editor), GD'97, Springer-Verlag, Lecture Notes in Computer Science, vol.1353, (1997) 13–24.
- [126] A. Yamaguchi and A. Sugimoto, An approximation algorithms for the two-layered graph drawing problem, in *Proceedings 5<sup>th</sup> Annual International Conference on Computing and Combinatorics* (T. Asano, H. Imai, D.T. Lee, Shin-ichi Nakano, and T. Tokuyama, editors), COCOON'99, Springer-Verlag, Lecture Notes in Computer Science, vol.1627, 1999, pp81–91.
- [127] F. Shahrokhi, O. Sýkora, L. A. Székely, and I. Vrt'ó, A new lower bound for the bipartite crossing number with applications, *Theoretical Computer Science*, 245(2000), 281–294.

## 交叉数更多文献

- [128] Imrich Vrt'ó, Crossing Numbers of Graphs: A Bibliography, Available electronically at <ftp://ifi.savba.sk/pub/imrich/crobib.ps.gz>; and <http://www.ifi.savba.sk/imrich/>
- [129] H. Wilf, On crossing numbers, and some unsolved problems, in *Combinatorics, Geometry, and Probability: A Tribute to Paul Erdos, Papers from the Conference in Honor of Paul*

- Erdos's 80<sup>th</sup> Birthday Held at Trinity College, Cambridge, March 1993, (B. Bollobas and A. Thomason, editors), Cambridge, England, Cambridge University Press, 1997, 557–562.
- [130] János Pach and Géza Tóth, Thirteen problems on crossing numbers, *Geombinatorics*, 9(2000), 194–207.
- [131] J. Pach, Crossing numbers, in *Discrete and Computational Geometry Japanese Conference* (J. Akiyama, M. Kano, and M. Urabe, editors), JCDCG'98, Tokyo, Japan, December 1998, Lecture Notes in Computer Science, vol. 1763, Springer-Verlag, Berlin, 2000, 267–273.
- [132] L. A. Székely, Progress on crossing number problems, in *SOFSEM 2005: Theory and Practice of Computer Science: 31<sup>st</sup> Conference on Current Trends in Theory and Practice of Computer Science*, (M. Bieliková et al. editors), Lecture Notes in Computer Science, vol. 3381, Springer-Verlag, 2005, 53–61.
- [133] L. A. Székely, A successful concept for measuring non-planarity of graphs: the crossing number, *Discrete Math.*, 276(2004), 331–352.
- [134] F. Shahrokhi, L. A. Székely, and I. Vrt'ó, Crossing numbers of graphs, lower bound techniques and algorithms: a survey, in *Graph Drawing, Proceedings of the DIMACS International Workshop* (R. Tamassia and I. G. Tollis, editors), GD'94, Springer-Verlag, Lecture Notes in Computer Science, vol.894, 1995, pp131–142.
- [135] D. Bienstock, Some provably hard crossing number problems, *Discrete Comput. Geom.*, 6(1991) 443–459.
- [136] S. N. Bhatt, F. T. Leighton, A framework for solving VLSI graph layout problems, *J. Comput. System Sci.*, 28(1984) 300–343.
- [137] F. Shahrokhi, O. Sýkora, L. A. Székely, and I. Vrt'ó, Crossing numbers: bounds and applications, in *Intuitive Geometry* (I. Bárány and K. Böröczky, editors), Bolyai Society Mathematical Studies 6, Janos Bolyai Mathematical Society, Budapest, 1997, 179–206.
- [138] János Pach, Farhad Shahrokhi, and Mario Szegedy, Applications of the crossing number, *Algorithmica*, 16(1996) 111–117.
- [139] J. Leanos and G. Salazar, On the additivity of crossing numbers of graphs, 2006, Available electronically at <http://www.ifisica.uaslp.mx/~gsalazar/RESEARCH/additivity.pdf>
- [140] S. Fiorini, On the crossing number of generalized Petersen graphs, in *Proceedings Int. Conf. Finite Geom. Comb. Struct.*, Bari, Italy, Combinatorics'84, North-Holland, *Annals of Discrete Mathematics*, vol. 30, 1986, 225–241.
- [141] D. McQuillan and R. Bruce Richter, On the crossing numbers of certain generalized Petersen graphs, *Discrete Mathematics*, 104(1992) 311–320.

- [142] D. McQuillan and R. Bruce Richter, On 3-regular graphs having crossing number at least 2, *J. Graph Theory*, 18(1994) 831–839.
- [143] R. Bruce Richter, Cubic graphs with crossing number two, *J. of Graph Theory*, 12(1988) 363–374.
- [144] R. Bruce Richter and C. Thomassen, Minimal graphs with crossing number at least  $k$ , *J. of Combinatorial Theory, Series B*, 58(1993) 217–224.
- [145] R. Bruce Richter, and J. Širáň, The crossing number of  $K_{3,n}$  in a surface, *J. of Graph Theory*, 21(1996) 51–54.
- [146] K. Asano, The crossing number of  $K_{1,3,n}$  and  $K_{2,3,n}$ , *J. of Graph Theory* 10(1986) 1–8.
- [147] B. Mohar, On the crossing number of almost planar graphs, *Informatica*, 30(2006) 301–303.
- [148] P. Hliněný and G. Salazar, On the crossing number of almost planar graphs, Available electronically at <http://www.math.gatech.edu/~thomas/BANFF/hlineny.pdf>
- [149] János Pach, J. Spencer, and Géza Tóth, New bounds on crossing numbers, *Discrete Comp. Geom.*, 24(2000) 623–644
- [150] János Pach and Géza Tóth, Which crossing number is it, anyway? In *Proceedings of the 39<sup>th</sup> Annual Symposium on Foundations of Computer Science, FOCS'98, 1998*, 617–626; and *J. Com. Theory, Ser. B*, 80(2000), 225–246.
- [151] J. Pach and G. Tóth, Crossing number of toroidal graphs, in *Proc. 13<sup>th</sup> International Symposium on Graph Drawing (GD'05)*, vol. 3843 of Lecture Notes in Computer Science, Springer, Berlin, 2006, pp334–342.
- [152] S. Masuda, K. Nakajima, T. Kashiwabara, and T. Fujisawa, Crossing minimization in linear embeddings of graphs, *IEEE Transactions on Computers*, 39:1(1990) 124–127
- [153] M. Kochol, Construction of crossing-critical graphs, *Discrete Mathematics*, 66(1987) 311–313.
- [154] G. Salazar, Infinite families of crossing critical graphs with given average degree, *Discrete Math.*, 271(2003), 343–350.
- [155] D. Bokal, G. Fijavz, and B. Mohar, The minor crossing number, *SIAM J. Discrete Math.*, 20:2(2006) 344–356.
- [156] L. W. Beineke, Biplanar graphs: a survey, *Computers and Mathematics with Applications*, 34(1997) 1–8.
- [157] Éva Czabarka, O. Sýkora, L. A. Székely, and I. Vrt'o, Crossing numbers and biplanar crossing numbers I: a survey of problems and results, in *More sets, Graphs and Numbers (E. Gyori, G.O.H. Katona, and L. Lovasz, editors)*, Bolyai Society Mathematical Studies 15, Springer-Verlag, 2006, 57–77.
- [158] Éva Czabarka, O. Sýkora, L. A. Székely, and I. Vrt'o, Outerplanar crossing numbers,

the circular arrangement problem and isoperimetric functions, *Electron. J. Combin.*, 11(2004) R81.

- [159] H. N. Djidjev and I. Vrt'o, Planar crossing numbers of genus  $g$  graphs, in *Proc. 33<sup>rd</sup> International Colloquium on Automata, Languages and Programming (ICALP'06)*, vol.4051 of Lecture Notes in Comput. Sci., 2006, pp419–430.
- [160] D. R. Wood and J. A. Telle, Planar decompositions and the crossing number of graphs with an excluded minor, 2006, Available electronically at <http://www.arxiv.org/math/0604467>
- [161] F. Shahrokhi, O. Sýkora, L. A. Székely, and I. Vrt'o,  $k$ -planar crossing numbers, to appear in *Discrete Appl. Math.*

### 厚度

- [162] A. Aggarwal, M. Klawe and P. Shor, Multilayer grid embeddings for VLSI, *Algorithmica*, 6(1991) 129–151.
- [163] R. Ramanathan and E. L. Lloyd, Scheduling algorithms for multihop radio networks, *IEEE/ACM Transactions on Networking*, 1(1993) 166–177.
- [164] L. W. Beineke, Fruited planes, in *Graph Theory, 250<sup>th</sup> Anniv. Conf., Lafayette/Indiana*, 1986, volume 63 of *Congressus Numerantium*, 1988, pp127–138.
- [165] P. Mutzel, T. Odenthal, and M. Scharbrodt, The thickness of graphs: a survey, *Graphs and Combinatorics*, 14(1998) 58–73. <http://citeseer.comp.nus.edu.sg/mutze198thickness.html>
- [166] A. Mansfield, Determining the thickness of graphs is NP-hard, *Mathematical Proceedings of the Cambridge Philosophical Society*, 93: 9(1983) 9–23.
- [167] R. J. Cimikowski, On heuristics for determining the thickness of a graph, *Information Sciences*, 85(1995) 87–98.
- [168] V. B. Alekseev and V. S. Goncakov, The thickness of an arbitrary complete graph, *Math. Sbornik*, 30:2(1976) 187–202. (In Russian)
- [169] J. M. Vasak, The thickness of the complete graph, *Notices of the American Mathematical Society*, 23:A (1976), page479, (Abstract).
- [170] L. W. Beineke, F. Harary, and J. W. Moon, On the thickness of the complete bipartite graph, *Proc. Camb. Phil. Soc.*, 60(1964) 1–5.
- [171] L. W. Beineke, Complete bipartite graphs: decomposition into planar subgraphs, in *A Seminar on Graph Theory* (F. Harary, editor), Holt, Rinehart and Winston, 1967.
- [172] M. Kleinert, Die Dicke des  $n$ -dimensionalen Würfel-Graphen, *J. of Combinatorial Theory*, 3(1967) 10–15.
- [173] A. M. Dean, J. P. Hutchinson, and E. R. Scheinerman, On the thickness and arboricity of a graph, *J. of Combinatorial Theory, Series B*, 52(1991) 147–151.



- [174] Walter Wessel, Über die Abhängigkeit der Dicke eines Graphen von seinen Knotenpunktvalenzen, in *Geometrie und Kombinatorik, 2. Kolloq.*, Karl-Marx-Stadt, GDR 1983, part2, 1984, pp235–238.
- [175] J. H. Halton, On the thickness of graphs of given degree, *Information Sciences*, 54(1991) 219–238.

### 书厚度

- [176] Paul C. Kainen. The book thickness of a graph, II. In *Proc. 20<sup>th</sup> South-eastern Conference on Combinatorics, Graph Theory, and Computer*, Boca Raton, Florida, USA, 1989, volume 71 of *Congressus Numerantium*, 1990, pp127–132.
- [177] S. M. Malitz, Graph with  $E$  edges have pagenumber  $O(\sqrt{E})$ , *J. Algorithms*, 17(1994) pp71–84.
- [178] S. M. Malitz, genus  $g$  graphs have pagenumber  $O(\sqrt{g})$ , *J. Algorithms*, 17(1994) pp85–109.
- [179] Douglas Muder, Margaret L. Weaver, and Douglas B. West, Pagenumber of complete bipartite graphs, *J. of Graph Theory*, 12(1988) pp469–489.
- [180] L. S. Heath, Embedding outerplanar graphs in small books, *SIAM J. Alg. Disc. Meth.*, 8(1987) pp198–218.
- [181] L. S. Heath and S. Istrail, The pagenumber of genus  $g$  graphs is  $O(g)$ , *J. of the Association for Computing Machinery*, 39(1992) pp479–501.
- [182] Fan R. K. Chung, F. T. Leighton, and A. L. Rosenberg, Embedding graphs in books: A layout problem with applications to VLSI design, *SIAM J. Alg. Disc. Meth.*, 8(1987) pp33–58.
- [183] B. Obrenic, Embedding de Bruijn and shuffle-exchange graphs in five pages, *SIAM J. Discrete Math.*, 6(1993), 642–654.
- [184] R. Swaminathan, D. Giriraj, and D. K. Bhatia, The pagenumber of the class of bandwidth- $k$  graphs is  $k - 1$ , *Information Processing Letters*, 55(1995) 71–74.
- [185] F. Shahrokhi and Weiping Shi, Efficient deterministic algorithms for embedding graphs on books, in *Proceedings 2<sup>nd</sup> Annual International Conference on Computing and Combinatorics* (Jin-Yi Cai and Chak Kuen Wong, editors), COCOON'96, Springer-Verlag, Lecture Notes in Computer Science, vol.1090, 1996, pp162–168.
- [186] M. Yannakakis, Fore pages are necessary and sufficient for planar graphs, in *Proceedings of the 18<sup>th</sup> Annual ACM Symposium on Theory of Computing, STOC'86*, 1986, pp104–108.
- [187] M. Yannakakis, Embedding planar graphs in four pages, *J. Computer and System Sciences*, 38(1989) 36–67.
- [188] T. Bilski, Embedding graphs in books, a survey, *IEE Proceedings-E*, vol.139, no.2, March, 1992.

## 厚度更多文献

- [189] Joan P. Hutchinson, Thomas Shermer, and Andrew Vince, On representations of some thickness-two graphs, *Computational Geomerty*, 13(1999) 161–171.
- [190] Peter Horák and Jozef Širáaň. On a Modified concept of thickness of a graph, *Math. Nachr.*, 108(1982) 305–306.
- [191] Peter Horák and Jozef Širáaň. A construction of thickness-minimal graphs, *Discrete Mathematics*, 64(1987)263–268.
- [192] Y. H. Peng, C. C. Chen, and K. M. Koh, On the Factor-thickness of regular graphs, *Graphs and Combinatorics*, 5(1989) 173–188.
- [193] M. B. Dillencourt, D. Eppstein, and D. S. Hirschberg, Geometric thickness of complete graphs, *Journal of Graph Algorithms and Applications*, 4(2000) 5–17.
- [194] D. R. Wood, Geometric thickness in a grid, *Discrete Mathematics*, 273(2003) 221–234.
- [195] Ian Anderson, On the toroidal thickness of graphs, *J. of Graph Theory*, 6(1982) 177–184.
- [196] Ian Anderson, The toroidal thickness of the symmetric quadripartite graph, *J. of Combinatorial Theory, Series B*, 33(1982) 57–59.
- [197] M. Jünger, P. Mutzel, T. Odenthal, and M. Scharbrodt, The thickness of a minor-excluded class of graphs, *Discrete Mathematics*, 182(1998) 169–176.
- [198] K. Asano, On the thickness of graphs with genus 2, *Ars Combinatoria*, 38(1994) 87–95.
- [199] W. Wessel, Thickness-critical graphs—a generalization of Kuratowski’s topic, in *Graph Theory, Proc. of a Conference held in Lagów* (M. Borowiecki, J.W. Kennedy, and Maciej M. Sysło, editors), Poland, February 10–13, 1981, Springer-Verlag, Lecture Notes in Mathematics, vol. 1018, 1983, pp266–277.
- [200] W. Wessel, Construction of critical graphs by replacing edges, in *Graph Theory in Memory of G. A. Dirac* (L. D. Andersen and G.A. Dirac, editors), *Annals of Discrete Mathematics*, North-Holland, vol. 41, 1989, pp473–486.
- [201] W. Wessel, On some variations of the thickness of a graph connected with colouring, in *Graphs and other combinatorial topics*, *Teubner Texte Math.*, 59(1983), 344–348.
- [202] N. K. Bose, and K. A. Prablu, Thickness of graphs with degree constrained vertices, *IEEE Trans. on Circuits and Systems*, 24 (1977) 184–190.
- [203] A. M. Dean, J. P. Hutchinson, and E. R. Scheinerman, On the thickness and arboricity of a graph, *J. of Combinatorial Theory Series B*, 52(1991) 147–151.
- [204] D. L. Boutin, E. Gethner, T. Sulanke, Thickness-two graphs part one: New nine-critical graphs, permuted layer graphs, and Catlin’s graphs, *Journal of Graph Theory*, 57(2008), 198–214.
- J. Fox, C. D. Tóth, On the decay of crossing numbers, *Journal of Combinatorial Theory, (Ser.B)*, 98 (2008), 33–42.

- [205] D. Bokal, G. Fijavz, D. R. Wood, The minor crossing number of graphs with an excluded minor, *The Electronic Journal of Combinatorics*, 15(2008), #R4.

### 亏格

- [206] I. N. Kagno, The triangulation of surfaces and the Heawood color formula, *J. Math. Phys.*, 15(1936) 179–186.
- [207] J. W. T. Youngs, Minimal imbeddings and the genus of a graph, *J. Math. Meth.*, 12 (1963) 303–315.
- [208] 蒋长浩, 图论与网络流, 北京: 中国林业出版社, 2001 年.
- [209] G. Ringel, Non-existence of graph embeddings, in *Theory and Applications of Graphs*, Lecture Notes in Mathematics 642 (Y. Alavi and D.R. Lick editors), Springer-Verlag, Berlin, 1978, pp465–476.

### 完全图和完全二部图的亏格

- [210] G. Ringel and J. W. T. Youngs, Solution of the Heawood map-coloring problem, *Proc. Nat. Acad. Sci. U.S.A.*, 60(1968) 438–445.
- [211] G. Ringel, *Map Color Theorem*, Springer-Verlag, Berlin, 1974.
- [212] A. T. White, The proof of the Heawood conjecture, in *Selected Topics in Graph Theory* (L.W. Beineke and R.J. Wilson editors), Academic Press, New York, 1978, pp50–82.
- [213] G. Ringel, Das geschlecht des vollständigen paaren graphen, *Abh. Math. Sem. Univ. Hamburg*, 38(1965) 139–150.

### 完全多部图的亏格

- [214] G. Ringel and J. W. T. Youngs, Das geschlecht des vollständigen drei-farbaren graphen, *Comment. Math. Helv.*, 45(1970) 152–158.
- [215] A. T. White, The genus of the complete tripartite graph  $K_{m,n,n}$ , *J. Combinatorial Theory*, 7(1969) 283–285.
- [216] S. Stahl and A. T. White, Genus emdeddings for some complete tripartite graphs, *Discrete Math.*, 14(1976) 279–296.
- [217] B. L. Garman, Cayley graph imbeddings and the associated block designs, Ph.D Thesis. Western Michigan University, 1976.
- [218] M. Jungerman, The genus of the symmetric Quadripartite graph, *J. Combinatorial Theory Ser.B*, 19(1975) 181–187.
- [219] A. T. White, Graphs of groups on surfaces, in *Combinatorial Surveys: Proceedings of the Sixth British Combinatorial Conference* (P.J. Cameron editor), Academic Press, London, 1977, pp165–197.
- [220] B. L. Garman, R. D. Ringeisen and A. T. White, On the genus of strong tensor products of graphs, *Canad. J. Math.*, 28(1976) 523–532.

[221] J. L. Gross and S. R. Alpert, Branched coverings of graph imbeddings, *Bull. Amer. Math. Soc.*, 79(1973) 942–945.

[222] M. Jungerman and G. Ringel, The genus of the  $n$ -octahedron: regular cases, *J. Graph Theory*, 2(1978) 69–75.

#### $n$ 维立方体的亏格

[223] G. Ringel, Über drei kombinatorische Probleme am  $n$ -dimensionalen Würfel und Würfelgitter, *Abh. Math. Sem. Univ. Hamburg*, 20(1955) 10–19.

[224] L. W. Beineke and F. Harary, The genus of the  $n$ -cube, *Canad. J. Math.*, 17(1965) 494–496.

[225] J. L. Gross and S. R. Alpert, Components of branched coverings of current graphs, *J. Combinatorial Theory Ser. B*, 20(1976) 283–303.

#### 亏格的算法

[226] C. Thomassen, The Graph genus problem is NP-complete, *J. Algorithms*, 10(1989) 568–576.

[227] H. N. Djidjev and J. H. Reif, An efficient algorithm for the genus problem with explicit construction of forbidden subgraphs, in *Proceedings of the 23th Annual ACM Symposium on Theory of Computing, STOC'91*, 1991, pp337–347.

[228] B. Mohar, Embedding graphs in an arbitrary surface in linear time, in *Proceedings of the 28<sup>th</sup> Annual ACM Symposium on Theory of Computing, STOC'96*, 1996.

#### 亏格与厚度

[229] K. Asano, On the genus and thickness of graphs, *J. of Combinatorial Theory, Ser. B*, 43(1987) 187–192.

[230] A. M. Dean, and J. P. Hutchinson, Relations among embedding parameters for graphs, in *Int' 1 Conf. on the Theory and Applications of Graphs*, Kalamazoo, MI, 1988.

#### 最大亏格

[231] R. A. Duke, The genus, regional number, and Betti number of a graph, *Canad. J. Math. Soc.*, 18(1966) 817–822.

[232] R. D. Ringeisen, Upper and lower embeddable graphs, in *Graph Theory and Applications, Lecture Notes in Mathematics 303 (Y. Alavi et al. editors)*, Springer-Verlag, Berlin, 1972, pp261–268.

[233] M. Jungerman, A characterization of upper-embeddable graphs, *Trans. Amer. Math. Soc.*, 214(1978) 401–406.

[234] N. H. Xuong, How to determine the maximum genus of a graph, *J. Combinatorial Theory Ser. B*, 26(1979) 217–225.

[235] D. Archdeacon, R. Nedela, M Škoviera, Maximum genus, connectivity, and Nebeský's theorem, Manuscript, 1997.

- [236] J. Chen, D. Archdeacon and J. L. Gross, Maximum genus and connectivity, *Discrete Math.*, 149(1996) 19–29.
- [237] J. Chen, S. P. Kanchi and J. L. Gross, A tight lower bound on the maximum genus of a simplicial graph, *Discrete Math.*, 156(1996) 83–102.
- [238] Y. Huang, Y. Liu, Maximum genus and girth of graphs, *Discrete Math.*, 194(1999), pp. 253–259.
- [239] Y. Huang and Y. Liu, Maximum genus and maximum nonseparating independent set of a 3-regular graph, *Discrete Math.*, 176(1997) 149–158.
- [240] Y. Huang and Y. Liu, The maximum genus of graphs with diameter three, *Discrete Math.*, 194(1999) 139–149.
- [241] Y. Huang, The maximum genus on a 3-vertex-connected graph, *Graphs and Combinatorics*, 16(2000) 159–164.
- [242] Hung-lin Fu and Min-chu Tsai, The maximum genus of diameter three graphs, *Australian J. Combin.*, 14(1996) 187–191.
- [243] M. Škoviera, The maximum genus of graphs of diameter two, *Discrete Math.*, 87(1991) 175–180.
- [244] Y. Huang, Maximum genus and chromatic number of graphs, *Discrete Mathematics*, 271 (2003) 117–127.
- [245] Y. Huang and T. Zhao, Maximum genus, connectivity and minimal degree of graphs, *Discrete Mathematics*, 300 (2005) 110–119.

#### 四色猜想

- [246] K. Appel, The proof of the four-colour theorem, *New Scientist*, 72(1976) 711–712.
- [247] K. Appel and W. Haken, Every planar map is four colorable, *Bull. Amer. Math. Soc.*, 82(1976) 711–712.
- [248] P. J. Heawood, Map-colour theorem, *Quart. J. Pure Appl. Math.*, 24(1890) 332–338.
- [249] D. R. Woodall and R. J. Wilson, The Appel-Haken proof of the four-color theorem, *Selected Topics in Graph Theory* (eds. L. W. Beineke, and R.J. Wilson), Academic Press, London, (1978) 83–101.
- [250] R. C. Read, Some applications of computers in graph theory, *Selected Topics in Graph Theory*, (eds. L. W. Beineke, and R.J. Wilson), Academic Press, New York, (1978) 417–444.
- [251] N. Robertson, D. P. Sanders, P. Seymour and R. Thomas, The four colour theorem, *J. of Combinatorial Theory, Ser.B.*, 70(1997), 2–44.

# 第八章 有向图

---

## §8.1 有向图的基本概念

**定义 8.1.1** 每条边都具有一个方向的图称为**有向图** (digraph, directed graph). 严格的说, 一个有向图  $\vec{G}$  是一个有序二元组  $(V(\vec{G}), A(\vec{G}))$ , 其中集合  $V(\vec{G})$  是非空的顶点集, 其元素称为顶点, 集合  $A(\vec{G})$  是  $V \times V$  的一个子集 (有序对, 元素可重复), 它是带有方向的边的集合, 称为**弧集**,  $A(\vec{G})$  中的元素称为**弧** (arc) 或有**向边** (directed edge). 有向图  $\vec{G}$  的顶点个数  $|V(\vec{G})|$  称为  $\vec{G}$  的**阶**, 记为  $V(\vec{G})$ , 弧的数目  $|A(\vec{G})|$  称为  $\vec{G}$  的**弧数**, 记为  $\varepsilon(\vec{G})$ .  $\nu(\vec{G})$  和  $\varepsilon(\vec{G})$  通常简写为  $\nu$  和  $\varepsilon$ .

**例 8.1.1** 设  $\vec{G} = (V, A)$ , 其中  $V = \{v_1, v_2, v_3, v_4, v_5\}$ , 有序对集合

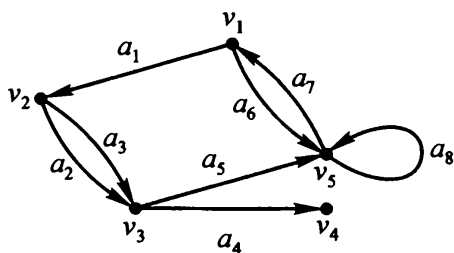
$$A = \{\langle v_1, v_2 \rangle, \langle v_2, v_3 \rangle, \langle v_2, v_3 \rangle, \langle v_3, v_4 \rangle, \langle v_3, v_5 \rangle, \langle v_1, v_5 \rangle, \langle v_5, v_1 \rangle, \langle v_5, v_5 \rangle\}.$$

这便定义出一个有向图.

**定义 8.1.2** 对有向图中的一条弧  $a = \langle u, v \rangle$ , 顶点  $v$  (即  $a$  的箭头指向的顶点) 称为弧  $a$  的**头** (有时也称为**终点**), 另一顶点  $u$  称为弧  $a$  的**尾** (有时也称为**始点**). 头、尾顶点相同的弧称为**环弧**, 两条具有相同头顶点和相同尾顶点的弧称为**并行弧**. 既无环弧又无并行弧的有向图称为**简单有向图** (或**严格有向图**). 简单有向图中两点  $u, v$  间可以有方向相反的两条弧, 这样的两条弧称为**一对反向弧**.

**定义 8.1.3** 有向图的顶点可用平面上的一个点来表示, 弧可用平面上的有向线段来表示 (直的或曲的), 这样画出的图形称为有向图的**图示**.

比如, 例 8.1.1 中有向图的一个图示如下.



其中  $a_8$  是环弧,  $a_2$  和  $a_3$  是并行弧,  $a_6$  和  $a_7$  是一对反向弧.

注 有向图的图示一般不唯一.

**定义 8.1.4** 设  $v$  是有向图  $\vec{G}$  的一个顶点.  $v$  的出度  $d_G^+(v)$  是指从  $v$  出发的弧的数目;  $v$  的入度  $d_G^-(v)$  是指指向  $v$  的弧的数目. 出度和入度可简写为  $d^+(v)$  和  $d^-(v)$ .  $v$  的出度与入度之和称为  $v$  的度, 记为  $d(v)$ , 即  $d(v) = d^+(v) + d^-(v)$ .

关于有向图的度和邻点集, 还有如下概念;

$$\text{最小出度 } \delta^+(\vec{G}) = \min_{v \in V(\vec{G})} \{d^+(v)\}$$

$$\text{最小入度 } \delta^-(\vec{G}) = \min_{v \in V(\vec{G})} \{d^-(v)\}$$

$$\text{最大出度 } \Delta^+(\vec{G}) = \max_{v \in V(\vec{G})} \{d^+(v)\}$$

$$\text{最大入度 } \Delta^-(\vec{G}) = \max_{v \in V(\vec{G})} \{d^-(v)\}$$

$$\text{顶点 } v \text{ 的出邻集 } N_G^+(v) = \{u \mid \langle v, u \rangle \in A(\vec{G})\}$$

$$\text{顶点 } v \text{ 的入邻集 } N_G^-(v) = \{u \mid \langle u, v \rangle \in A(\vec{G})\}$$

注 在一个  $\nu$  阶简单有向图中, 对一个顶点  $v$ , 有可能其度  $d(v) \geq \nu$ , 但总有  $d^+(v) \leq \nu - 1$ ,  $d^-(v) \leq \nu - 1$  以及  $d(v) \leq 2\nu - 2$ .

**定理 8.1.1** 对于任何有向图  $\vec{G}$ , 都有  $\sum_{v \in V(\vec{G})} d(v) = 2\varepsilon$  且  $\sum_{v \in V(\vec{G})} d^+(v) = \sum_{v \in V(\vec{G})} d^-(v) = \varepsilon$ .

证明与无向图情况类似.

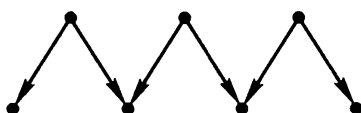
**定义 8.1.5** 与有向图的概念相对应, 以前章节中所说的边不带有方向的图称为无向图. 给定一个无向图  $G$ , 给它的每条边确定一个方向, 便可得到一个有向图, 这个有向图称为  $G$  的一个定向图, 而  $G$  称为这个有向图的基础图或底图 (underlying graph).

注 一个无向图一般有多个定向图.

## §8.2 有向路与有向圈

**定义 8.2.1** 有向图的一条有向途径是指一个有限非空序列  $w = v_0 a_1 v_1 \cdots a_k v_k$ , 它的各项交替地是顶点和弧, 使得  $a_i = \langle v_{i-1}, v_i \rangle, (i = 1, 2, \cdots, k)$ . 弧不重复的有向途径称为有向迹; 顶点不重复的有向途径称为有向路; 首尾相接的有向路称为有向圈.

**注** 有向图的底图中有长的路未必该有向图中就有长的有向路. 例如, 下图所示的有向图其底图中的最长路为 6, 但其最长有向路的长为 1.



令人惊奇的是, 有向图中有向路的长却与其底图的色数有关.

**定理 8.2.1** (Vitaver, 1962; Roy, 1967; Gallai, 1968) <sup>[1][2][3]</sup> 以图  $G$  为底图的有向图  $\vec{G}$  中必有长为  $\chi(G) - 1$  的有向路.

**证明** 设  $A'$  是使有向图  $\vec{G} - A'$  不含有向圈的最小弧子集. 设  $\vec{G} - A'$  中最长有向路的长为  $k$ . 按如下方式将颜色  $1, 2, \cdots, k+1$  分配给  $\vec{G} - A'$  的顶点;

对任何顶点  $v \in V(\vec{G} - A')$ , 如果  $\vec{G} - A'$  中从  $v$  出发的最长有向路长为  $i$ , 则给  $v$  染色  $i+1$ . 用  $V_i$  表示染有第  $i$  种色的顶点的集合. 由于  $\vec{G} - A'$  中最长有向路的长为  $k$ , 因此这种染色方案可使  $G$  的所有顶点获得染色. 下证  $(V_1, V_2, \cdots, V_{k+1})$  是  $G$  的正常  $k+1$  顶点染色. 为此, 先证明关于这种染色的两个结论.

(1)  $\vec{G} - A'$  中任一有向路的起点与终点异色

事实上, 设  $P(u, v)$  是  $\vec{G} - A'$  的一条  $(u, v)$  有向路,  $u \neq v$ . 不妨设  $v \in V_{i+1}$ . 由染色规则可知, 存在从  $v$  出发的长为  $i$  的有向路, 设为  $Q = vv_1v_2 \cdots v_i \subseteq \vec{G} - A'$ . 又因  $\vec{G} - A'$  中没有有向圈, 故  $v_1, v_2, \cdots, v_i$  不在有向路  $P(u, v)$  上. 因此  $P(u, v)$  与  $Q$  接起来是从  $u$  出发的长至少为  $i+1$  的有向路. 于是由染色规则,  $u$  点的染色标号大于  $i+1$ , 故  $u \notin V_{i+1}$ .

(2)  $\vec{G}$  中任一条弧的两端点异色.

事实上, 设弧  $a = \langle u, v \rangle \in A(\vec{G})$ . 若  $a \in A'$ , 则由  $A'$  的最小性,  $(\vec{G} - A') + a$  含有有向圈  $C$ , 且弧  $a$  必在  $C$  上.  $C - a$  是  $\vec{G} - A'$  的一条从  $v$  到  $u$  的有向路, 故由 (1),  $u$  与  $v$  异色; 若  $a \in A$ , 则  $a \in A(\vec{G} - A')$ , 此时  $uv$  是  $\vec{G} - A'$  的一条有向路. 由 (1),  $u$  与  $v$  也异色.

由 (2) 可知, 在上述染色规则下, 无向图  $G$  的任一条边两端点异色. 因此  $(V_1, V_2, \cdots, V_{k+1})$  是  $G$  的正常  $k+1$  顶点染色. 因而  $\chi(G) \leq k+1$ , 即  $k \geq \chi(G) - 1$ . 证毕.



**定理 8.2.2 (Chvátal, Lavász, 1974) [4]** 若  $\vec{G}$  是无环有向图, 若其底图  $G$  无孤立点, 则底图  $G$  中必有一个独立集  $S$ , 使得对  $V(G) - S$  中的每个顶点  $v$ , 存在  $v_0 \in S$ , 在  $\vec{G}$  中从  $v_0$  到  $v$  有长不超过 2 的有向路.

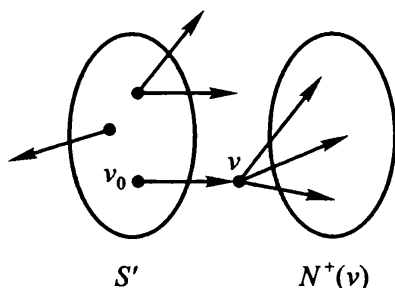
**证明** 无妨设底图  $G$  为连通图 (若不连通, 考虑它的每个连通分支即可). 对顶点数  $\nu$  作归纳.

$\nu = 1$  时定理显然成立.

假设对顶点数少于  $\nu$  的所有有向图  $\vec{G}$ , 结论成立. 考虑顶点数为  $\nu$  的有向图  $\vec{G}$ .

任取  $v \in V(\vec{G})$ , 令  $\vec{G}' = \vec{G} - (\{v\} \cup N^+(v))$ . 由归纳假设, 存在  $\vec{G}'$  的底图  $G'$  的一个独立集  $S'$ , 对  $V(\vec{G}') - S'$  中任何顶点, 可从  $S'$  中的某顶点出发, 经过长度  $\leq 2$  的有向路到达它.

(1) 若存在  $v_0 \in S'$ , 使弧  $\langle v_0, v \rangle \in A(\vec{G})$ , 则  $N^+(v) \cup \{v\}$  中每个顶点可从  $v_0$  出发, 经过长度  $\leq 2$  的有向路到达; 而  $\vec{G}$  中其他点都在  $\vec{G}'$  中, 本来就可从  $S'$  中点出发经过长度  $\leq 2$  的有向路到达. 可见  $S'$  满足定理的要求.



(2) 否则, 对  $\forall v_0 \in S', \langle v_0, v \rangle \notin A(\vec{G})$ , 而且因  $N^+(v) \cap S' = \emptyset$ , 故  $\langle v, v_0 \rangle \notin A(\vec{G})$ , 因此在底图  $G$  中  $v$  与  $S'$  中的点不相邻. 此时  $S = S' \cup \{v\}$  满足定理的要求. 证毕.

上述定理表明, 在无孤立顶点的有向图中, 必存在底图中的点独立集  $S$ , 从  $S$  到  $V(G) - S$  中的任一顶点都可两步到达.

**定理 8.2.3** 有向图  $\vec{G}$  中存在有向圈的充分必要条件是:  $\vec{G}$  中存在有向子图  $\vec{H}$  使得对任意的  $v \in V(\vec{H})$ , 都有  $d_H^+(v) > 0$  (或都有  $d_H^-(v) > 0$ ).

**证明 必要性** 设  $\vec{G}$  中存在有向圈, 则该有向圈即可取作  $\vec{H}$ .

**充分性** 由于对任意的  $v \in V(\vec{H})$ , 都有  $d_H^+(v) > 0$ , 故任取  $\vec{H}$  中一点  $v_1$ , 必在  $\vec{H}$  中存在其出邻点  $v_2$ , 使得  $\langle v_1, v_2 \rangle \in A(\vec{H})$ ; 同理,  $\vec{H}$  中必存在  $v_2$  的出邻点  $v_3$ , 使得  $\langle v_2, v_3 \rangle \in A(\vec{H})$ , 得有向路  $v_1 v_2 v_3$ ; 如此继续, 有向路不断延伸. 由于  $\vec{H}$  中顶点个数有限, 该有向路必定会在某一步返回某个已走过的顶点, 如此便形成一个有向圈.  $d_H^-(v) > 0$  的情况类似可证. 证毕.

**定理 8.2.4** 设  $\vec{G}$  是一个有向图, 且其底图是连通图. 如果对任意的  $v \in V(\vec{G})$ , 都有  $d^+(v) = 1$  (或都有  $d^-(v) = 1$ ), 则  $\vec{G}$  恰有一个有向圈.

**证明** 由定理 8.2.3 知,  $\vec{G}$  中存在有向圈. 下面用反证法证明唯一性 (只证  $d^+(v) = 1$  的情形,  $d^-(v) = 1$  时类似可证).

假如  $\vec{G}$  中有向圈不唯一, 任取其二, 设为  $\vec{C}_1, \vec{C}_2$ .  $\vec{C}_1, \vec{C}_2$  的关系必为如下三种情况之一:

- (1)  $\vec{C}_1$  与  $\vec{C}_2$  有公共弧;
- (2)  $\vec{C}_1$  与  $\vec{C}_2$  无公共弧, 但有公共顶点;
- (3)  $\vec{C}_1$  与  $\vec{C}_2$  无公共点.

在情况 (1) 下,  $\vec{C}_1$  与  $\vec{C}_2$  的所有公共弧必形成一条或几条有向路, 这些有向路的终点的出度至少为 2, 与已知条件矛盾.

在情况 (2) 下, 设  $w$  是  $\vec{C}_1$  与  $\vec{C}_2$  的一个公共顶点, 则  $w$  的出度至少为 2, 也与已知条件矛盾.

在情况 (3) 下, 因  $\vec{G}$  的底图是连通图, 故  $\vec{C}_1$  与  $\vec{C}_2$  间有路相连, 该路上每条边在  $\vec{G}$  中都有方向, 由已知条件, 该路内部的顶点在有向图  $\vec{G}$  中出度都为 1, 因此该路至少有一个端点在  $\vec{G}$  中有弧指向路的内部点, 因这样的端点在圈上还有一个出邻点, 所以其出度至少为 2. 这又与已知条件矛盾.

以上矛盾表明  $\vec{G}$  中有向圈是唯一的. 证毕.

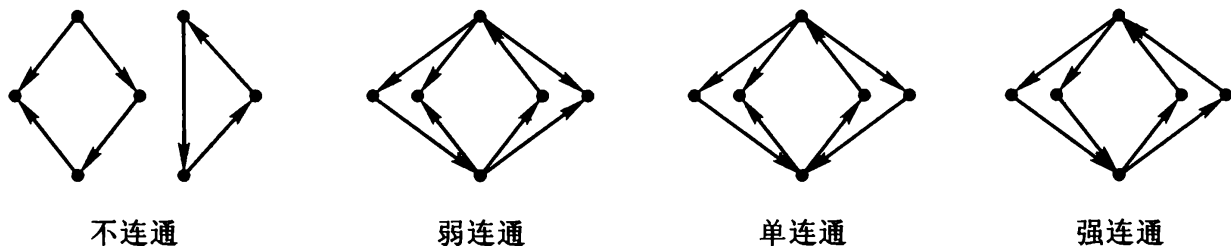
### §8.3 有向图的连通性及无向图的强连通定向

**定义 8.3.1** 设  $\vec{G}$  是一个有向图,

- (1) 若  $\vec{G}$  的底图  $G$  是连通图, 则称  $\vec{G}$  是弱连通的.
- (2) 若对  $\vec{G}$  的任二顶点  $u, v$ , 在  $\vec{G}$  中要么存在有向路  $P(u, v)$ , 要么存在有向路  $P(v, u)$ , 则称  $\vec{G}$  是单连通的.
- (3) 若对  $\vec{G}$  的任二顶点  $u, v$ , 在  $\vec{G}$  中既存在有向路  $P(u, v)$ , 又存在有向路  $P(v, u)$ , 则称  $\vec{G}$  是强连通的 (或称双向连通的).

**注** 易见, 有向图强连通必定单连通, 单连通必定弱连通.

下图中分别列举出了不连通、弱连通、单连通和强连通有向图的例子.



**定理 8.3.1**  $\vec{G}$  是单连通有向图的充分必要条件是  $\vec{G}$  的所有顶点在一条有向途径上.

**证明** 充分性是显然的.

**必要性** 首先用归纳法证明: 对  $\vec{G}$  的任何至少含有 2 个顶点的顶点子集  $S$ , 必有  $u \in S$ , 使得在  $\vec{G}$  中  $u$  到  $S$  中每个其他顶点都有有向路.

当  $S$  只含 2 个顶点时, 由于  $\vec{G}$  单连通, 结论显然成立.

假设对任何含有  $k$  个顶点的子集, 结论成立. 下证对含有  $k+1$  个顶点的子集  $S$ , 结论也成立.

事实上, 任取  $v \in S$ , 由归纳假设,  $S - \{v\}$  中必有某点  $u$ , 它到  $S - \{v\}$  中每个其他顶点都有有向路. 由于  $\vec{G}$  单连通, 在  $\vec{G}$  中  $u, v$  之间必有有向路. 如果  $u$  到  $v$  有有向路, 则  $u$  符合结论的要求; 如果  $v$  到  $u$  有有向路, 则  $v$  符合结论的要求. 归纳法完成.

下面来证明定理必要性的结论.

取  $S_1 = V(\vec{G})$ , 由上述结论, 存在  $u_1 \in S_1$ , 使得  $u_1$  到  $S_1$  中每个其他顶点都有有向路. 令  $S_2 = V(\vec{G}) - \{u_1\}$ , 再由上述结论, 存在  $u_2 \in S_2$ , 使得  $u_2$  到  $S_2$  中每个其他顶点在  $\vec{G}$  中都有有向路. 再令  $S_3 = V(\vec{G}) - \{u_1, u_2\}$ , 又由上述结论, 存在  $u_3 \in S_3$ , 使得  $u_3$  到  $S_3$  中每个其他顶点在  $\vec{G}$  中都有有向路. 以此类推, 可得到含所有顶点的序列, 顶点沿序列依次有有向路, 这些有向路合成一条含所有点的有向途径. 证毕.

**定理 8.3.2**  $\vec{G}$  是强连通有向图的充要条件是  $\vec{G}$  的所有顶点在一条有向闭途径上.

**证明 必要性** 设  $\vec{G}$  是强连通有向图,  $V(\vec{G}) = \{v_1, v_2, \dots, v_\nu\}$ , 则存在有向路  $P_1(v_1, v_2), P_2(v_2, v_3), \dots, P_{\nu-1}(v_{\nu-1}, v_\nu), P_\nu(v_\nu, v_1)$ , 于是  $\bigcup_{i=1}^{\nu} P_i$  即为含所有顶点的有向闭途径.

**充分性** 设  $\vec{G}$  的顶点共处于一个有向闭途径  $C$  上, 则对  $\forall u, v \in V(\vec{G})$ , 在  $C$  上必有  $u$  到  $v$  的一段有向路  $P_1(u, v)$ , 也有  $v$  到  $u$  的一段有向路  $P_1(v, u)$ , 故  $\vec{G}$  是强连通的. 证毕.

**定理 8.3.3 (Robbins, 1939) [5]** 无向图  $G$  可定向成强连通图的充分必要条件为:  $G$  连通且无割边 (即  $G$  是 2-边连通图).

**证明 必要性** 用反证法. 若  $G$  中有割边, 则无论如何对  $G$  的各边定向, 都无法使割边两端的点在所形成的有向图中双向连通. 这与  $G$  可定向成为强连通有向图矛盾.

**充分性** 设  $G$  无割边, 则  $G$  中存在圈. 取其一个圈  $C$ , 记  $G_1 = C$ . 给  $G_1$  定向为顺时针方向. 若  $G_1$  不是  $G$  的生成子图, 则存在  $v_1 \in V(G) - V(G_1)$ , 由第二章 Menger

定理的推论, 存在两条无公共边的路  $P_1, Q_1$ , 它们的一端是  $v_1$ , 另一端在  $G_1$  上. 给  $P_1$  定向为指向  $v_1, Q_1$  定向为指向  $G_1$ , 令  $G_2 = G_1 \cup P_1 \cup Q_1$ , 则  $G_2$  是强连通的.

若  $G_2$  仍不是  $G$  的生成子图, 则存在  $v_2 \in V(G) - V(G_2)$ , 同理, 存在无公共边的路  $P_2, Q_2$ , 其一端在  $v_2$  处, 另一端在  $G_2$  中. 给  $P_2$  定向为指向  $v_2, Q_2$  定向为指向  $G_2$ , 令  $G_3 = G_2 \cup P_2 \cup Q_2$ , 则  $G_3$  是强连通的. 如此反复, 可得一个强连通子图序列  $G_1, G_2, \dots$ , 顶点数严格递增. 因  $|V(G)|$  是有限数, 故必在某一步得到  $G_n$  后终止,  $G_n$  是强连通的且是  $G$  的生成子图. 最后, 把  $G$  中不在  $G_n$  中的边任意定向, 得到以  $G$  为底图的一个强连通有向图. 证毕.

该定理充分性的证明是构造性的, 由此可得到一个将 2-边连通图定向为强连通图的算法. 但其中要反复求点到已定向子图的无公共边的, 计算量较大. 下面给出一个更为简便的定向算法.

**定义 8.3.2** 如果一个有向图在不考虑方向时是一棵树, 则称这个有向图为有向树.

设  $\vec{G}$  是一个有向图, 若  $\vec{G}$  的子有向图  $T$  是一棵有向树且  $\vec{G}$  的所有顶点都在  $T$  上, 则称  $T$  是  $\vec{G}$  的一棵有向生成树.

按照定理 8.3.3, 任何 2 边连通无向图  $G$  都可定向成强连通有向图. 利用有向生成树的概念可得到一个将 2 边连通图定向为强连通图的简便算法. 其基本思想是: 从任何一点出发, 先按照树的生长过程 (深度优先) 生长出一棵有向生成树  $T$ , 然后对每条不在  $T$  上的每条边  $e = uv$ , 按  $u, v$  进入有向树  $T$  的顺序由后向前给  $e$  定向: 若在生长  $T$  的过程中, 顶点  $v$  比  $u$  先进入  $T$ , 则将  $e$  定向为从  $u$  指向  $v$ .

这一算法称为 Hopcroft-Tarjan 定向算法. 算法步骤如下.

**Hopcroft-Tarjan 定向算法**

**输入:** 2 边连通图  $G = (V, E)$ .

**输出:**  $G$  的强连通定向图.

**第 1 步:** 任取  $G$  中一个顶点  $w$ , 令  $l(w) := 0, L := \{w\}, U := V - \{w\}, A := \phi$ . 转下步.

**第 2 步:** 在  $L$  中找出满足如下条件的一点  $u$ :

- (1)  $u$  在顶点集  $U$  中有邻点;
- (2)  $u$  是符合 (1) 的点中  $l(u)$  最大的.

转下步.

**第 3 步:** 在  $U$  中任取  $u$  的一个邻点  $v$ , 将  $u, v$  间的连边定向为从  $u$  指向  $v$ . 令  $l(v) := l(u) + 1, L := L \cup \{v\}, U := V - \{v\}, A := A \cup \{(u, v)\}$ . 转下步.

**第 4 步:** 若  $L \neq V$ , 则转第 2 步; 否则转下步.

**第 5 步:** 对任何尚未获得定向的边  $e = xy$ , 若  $l(x) > l(y)$ , 则将  $e$  定向为从  $x$  指向  $y$ ; 否则, 定向为从  $y$  指向  $x$ . 如此将所有边定向后, 算法结束.

在算法中,  $l(v)$  表示顶点  $v$  的标号,  $L$  表示已获得标号的顶点之集,  $U$  表示尚未获得标号的顶点之集,  $A$  表示已获得定向的弧的集合. 在算法第 1 步 ~ 第 4 步中, 通过对  $G$  的顶点按深度优先方式逐次标号来实现有向生成树的生长过程, 并在第 1 步 ~ 第 4 步反复循环结束时获得一棵有向生成树  $T$ . 由算法可见, 沿  $T$  的任一条有向路, 顶点的标号依次严格递增.

**定理 8.3.4** Hopcroft-Tarjan 定向算法在有限步内结束, 且所得定向图是强连通图.

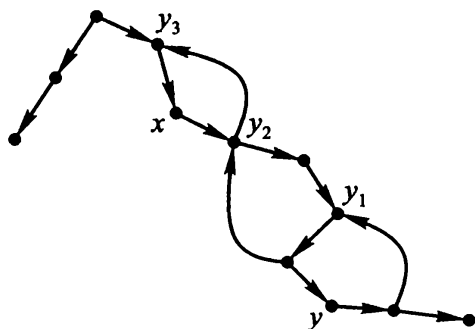
**证明** 首先, 若  $U \neq \phi$ , 则算法第 2 步中要找的顶点  $u$  总是存在的. 这是因为  $G$  是 2 边连通的, 故已获得标号的顶点集合  $L$  与尚未获得标号的顶点集合  $U$  之间必有连边. 由于算法第 2 步每循环一次, 恰有一个点获得标号, 故有向生成树的生长过程 (第 1 步 ~ 第 4 步) 会在  $\nu - 1$  次循环后结束. 在执行算法第 5 步时, 已获得有向生成树  $T$ . 对  $G$  中任一条尚未定向的边  $e = xy$ , 不妨设在算法第 1 步 ~ 第 4 步循环时  $x$  先于  $y$  获得标号, 由于算法执行深度优先搜索过程, 因此在  $x$  获得标号后, 算法必定继续搜索完从  $x$  有路可达 (在  $G$  中) 的所有未标顶点, 然后才会返回  $x$  的祖先搜索其他顶点. 由于在  $G$  中  $x, y$  间有边相连 (当然有路可达), 故  $y$  在  $T$  中必定成为  $x$  的后代, 因此  $x, y$  必位于  $T$  的一条有向路上. 而在  $T$  的有向路上的顶点, 其标号严格递增, 故  $l(x) \neq l(y)$ , 因此第 5 步总可以给每条未在前四步定向的边进行定向. 又因  $G$  的边数是有限的, 故第 5 步会在有限步结束. 因此, 在有限步之后, 所有边都获得定向, 算法结束.

设算法第 1 步 ~ 第 4 步循环结束时, 获得的有向生成树是  $T$ , 算法结束后获得的定向图为  $\vec{G}$ . 为叙述方便, 对有向生成树  $T$  中任二点  $u, v$ , 若  $T$  中从  $u$  到  $v$  有有向路, 则称  $u$  是  $v$  的祖先, 而称  $v$  是  $u$  的后代. 对  $T$  中任一点  $x$ , 我们将  $T$  中  $x$  及其所有祖先和所有后代导出的子有向树称为  $T$  的  $x$  支. 显然, 对  $T$  中任二点,  $u, v$ , 若  $v$  不在  $T$  的  $u$  支中, 则  $u$  也不在  $T$  的  $v$  支中. 此时  $u, v$  两点在  $G$  中不可能有连边. 若不然, 不妨设在算法第 1 步 ~ 第 4 步循环时  $u$  先于  $v$  获得标号. 按深度优先搜索过程, 在  $u$  获得标号后, 算法必定继续搜索完从  $u$  有路可达 (在  $G$  中) 的所有未标顶点, 然后才会返回  $u$  的祖先搜索其他顶点. 由于当  $u$  获得标号时,  $v$  尚未被标号, 且  $u, v$  在  $G$  中有连边, 故  $v$  在  $T$  中必定成为  $u$  的后代, 这与  $v$  不在  $u$  支中相矛盾.

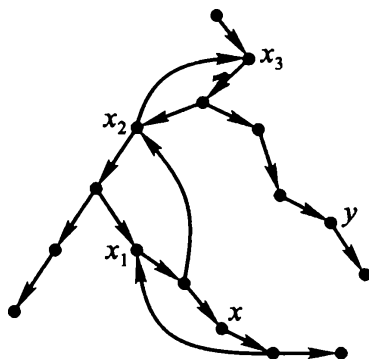
任取  $\vec{G}$  中两点  $x, y$ , 下分两种情况证明  $x, y$  间互相有有向路.

(1) 若  $x, y$  在  $T$  中的一条有向路上, 不妨设  $x$  是  $y$  的祖先. 由于  $G$  是 2 边连通的,  $y$  或  $y$  的后代在  $G$  中必定与  $y$  的某个祖先  $y_1$  有不在  $T$  上的连边 (否则, 因  $y$  及其后代与  $y$  支之外的点在  $G$  中无连边,  $T$  中指向  $y$  的弧在  $G$  中会成为割边). 按算法, 这样的边被定向为由  $y$  或  $y$  的后代指向  $y_1$ . 同理,  $y_1$  或  $y_1$  的后代在  $G$  中必定有连到  $y_1$  的某个祖先  $y_2$  的边, 按算法此边应被定向为指向  $y_2$ . 如此递推, 总可到达某个

$y_k, y_k$  是  $x$  祖先, 且  $y_{k-1}$  或  $y_{k-1}$  的后代有定向到  $y_k$  的弧. 易见在  $\vec{G}$  中有  $y$  到  $x$  的有向路, 如下图所示.



(2) 若  $x, y$  在  $T$  中不在一条有向路上, 则  $x$  不在  $T$  的  $y$  支中,  $y$  也不在  $x$  支中. 由于且  $G$  是 2 边连通的, 故在  $G$  中  $x$  或  $x$  的后代必定与  $x$  的某个祖先  $x_1$  有不在  $T$  上的边 (否则, 因  $x$  及其后代与  $x$  支之外的点在  $G$  中无连边,  $T$  中指向  $x$  的弧在  $G$  中会成为割边), 按照算法, 该边被定向为从  $x$  或  $x$  的后代指向  $x_1$ . 同理,  $x_1$  或  $x_1$  的后代在  $G$  中必定有连到  $x_1$  的某个祖先  $x_2$  的边, 按算法应被定向为指向  $x_2$ . 如此递推, 总可到达某个  $x_k$ , 它是  $x$  和  $y$  的共同祖先, 且  $x_{k-1}$  或  $x_{k-1}$  的后代有定向到  $x_k$  的弧 (如图所示).



易见, 在  $\vec{G}$  中有从  $x$  到  $y$  的有向路. 同理可知,  $y$  到  $x$  也有有向路. 因此,  $\vec{G}$  是强连通的. 证毕.

容易检查 Hopcroft-Tarjan 定向算法的计算复杂度是  $O(v^2)$ .

### §8.4 Euler 有向图和 Hamilton 有向图

**定义 8.4.1** 经过有向图  $\vec{G}$  的所有弧恰一次的有向迹称为  $\vec{G}$  的一条 Euler 有向迹. 经过有向图  $\vec{G}$  的所有弧恰一次的有向闭迹称为  $\vec{G}$  的一条 Euler 有向闭迹. 存在 Euler 有向闭迹的有向图称为 Euler 有向图. 经过有向图  $\vec{G}$  的所有顶点恰一次的有向路称为  $\vec{G}$  的一条 Hamilton 有向路; 经过有向图  $\vec{G}$  的所有顶点恰一次的有向圈称为  $\vec{G}$  的一个 Hamilton 有向圈. 存在 Hamilton 有向圈的有向图称为 Hamilton 有向图.

**定理 8.4.1** 非平凡弱连通有向图  $\vec{G}$  是 Euler 有向图的充分必要条件是对任何  $v \in V(\vec{G})$ , 都有  $d^+(v) = d^-(v)$ .

**定理 8.4.2** 非平凡弱连通有向图  $\vec{G}$  是 Euler 有向图的充分必要条件是  $\vec{G}$  可分解为有向圈的并, 即:  $\vec{G} = \bigcup_{i=1}^k C_i$ , 其中  $C_i$  是  $\vec{G}$  的有向圈, 且  $A(C_i) \cap A(C_j) = \emptyset$ , ( $1 \leq i, j \leq k$ ),  $k$  是一个正整数.

**定理 8.4.3** 非平凡弱连通有向图  $\vec{G}$  有 Euler 有向迹的充分必要条件是  $\vec{G}$  中存在两个顶点  $u$  和  $w$  满足  $d^+(u) = d^-(u) + 1, d^+(w) = d^-(w) - 1$ , 而其他顶点都有  $d^+(v) = d^-(v)$ .

上述定理的证明与无向 Euler 图中相关定理的证明类似, 留做习题 (习题 8.17~习题 8.19).

**定理 8.4.4 (Meyniel, 1973) [6]** 设  $\vec{G}$  是  $\nu$  阶强连通简单有向图, 如果对任二互不相邻顶点  $u, v$  都有

$$d(u) + d(v) \geq 2\nu - 1,$$

则  $\vec{G}$  是 Hamilton 有向图.

证明可参看文献 [6]~[8].

**推论 8.4.1 (Woodall, 1972) [9]** 设  $\vec{G}$  是  $\nu$  阶非平凡简单有向图, 如果对任二满足  $\langle u, v \rangle \notin A(\vec{G})$  的顶点  $u, v$  都有

$$d^+(u) + d^-(v) \geq \nu,$$

则  $\vec{G}$  是 Hamilton 有向图.

**证明** 先证  $\vec{G}$  是强连通的.

任取  $\vec{G}$  中两点  $v_1, v_2$ , 我们来证明  $\vec{G}$  中必有  $(v_1, v_2)$  有向路. 事实上, 如果  $\langle v_1, v_2 \rangle$  是  $\vec{G}$  中的一条弧, 则结论显然; 如果  $\langle v_1, v_2 \rangle$  不是  $\vec{G}$  中一条弧, 则由条件  $d^+(v_1) + d^-(v_2) \geq \nu$ , 至少存在一个异于  $v_1, v_2$  的顶点  $w$ , 使  $\langle v_1, w \rangle, \langle w, v_2 \rangle$  是两条弧, 从而  $v_1 w v_2$  是一条  $(v_1, v_2)$  有向路.

同理可知,  $\vec{G}$  中也存在一条  $(v_2, v_1)$  有向路. 由  $v_1, v_2$  的任意性,  $\vec{G}$  是强连通的.

设  $u$  和  $v$  是任意两个不相邻顶点. 由条件,  $d^+(u) + d^-(v) \geq \nu$ , 且  $d^+(v) + d^-(u) \geq \nu$ , 因此  $d(u) + d(v) \geq 2\nu$ , 由定理 8.4.4,  $\vec{G}$  是 Hamilton 有向图. 证毕.

由定理 8.4.4 和推论 8.4.1, 下面两个推论显然成立.

**推论 8.4.2 (Ghouilà-Houri, 1960) [10]** 设  $\vec{G}$  是  $\nu$  阶强连通简单有向图, 且对每个顶点  $v$  都有  $d(v) \geq \nu$ , 则  $\vec{G}$  是 Hamilton 有向图.

**推论 8.4.3 (Ghouilà-Houri, 1960)** <sup>[10][11]</sup> 设  $\vec{G}$  是  $\nu$  阶简单有向图, 且对每个顶点  $v$ , 都有

$$d^+(v) \geq \frac{\nu}{2}, d^-(v) \geq \frac{\nu}{2},$$

则  $\vec{G}$  是 Hamilton 有向图.

**定理 8.4.5** 设  $\vec{G}$  是  $\nu$  阶非平凡简单有向图, 如果对任二互不相邻顶点  $u, v$  都有

$$d(u) + d(v) \geq 2\nu - 3,$$

则  $\vec{G}$  含有 Hamilton 有向路.

**证明** 给  $\vec{G}$  添加一个新顶点  $w$ , 并将  $w$  与  $\vec{G}$  的每个顶点用两条方向相反的弧相连, 由此构成一个新的有向图  $\vec{G}'$ . 显然  $\vec{G}'$  是强连通图. 任取  $\vec{G}'$  中两个不相邻顶点  $u, v$ , 则  $u, v$  也是  $\vec{G}$  的不相邻顶点, 由条件知,

$$d_{\vec{G}'}(u) + d_{\vec{G}'}(v) = d_{\vec{G}}(u) + d_{\vec{G}}(v) + 4 \geq (2\nu - 3) + 4 = 2\nu + 1 = 2(\nu + 1) - 1.$$

因  $\vec{G}'$  是  $\nu + 1$  阶强连通有向图, 故由定理 8.4.4,  $\vec{G}'$  包含一个 Hamilton 有向圈. 从该圈删去  $w$  便得到  $\vec{G}$  的一条 Hamilton 有向路. 证毕

类似于推论 8.4.1 ~ 推论 8.4.3, 有如下推论 (习题 8.21 和习题 8.22).

**推论 8.4.4** 设  $\vec{G}$  是  $\nu$  阶非平凡简单有向图, 如果对任二满足  $\langle u, v \rangle \notin A(\vec{G})$  的顶点  $u, v$  都有

$$d^+(u) + d^-(v) \geq \nu - 1,$$

则  $\vec{G}$  含有 Hamilton 有向路.

**推论 8.4.5** 设  $\vec{G}$  是  $\nu$  阶简单有向图, 且对每个顶点  $v$  都有  $d(v) \geq \nu - 1$ , 则  $\vec{G}$  含有 Hamilton 有向路.

**推论 8.4.6** 设  $\vec{G}$  是  $\nu$  阶简单有向图, 且对每个顶点  $v$  都有

$$d^+(v) \geq \frac{\nu - 1}{2}, d^-(v) \geq \frac{\nu - 1}{2},$$

则  $\vec{G}$  含有 Hamilton 有向路.

## §8.5 竞赛图

设在若干选手间进行的循环赛中, 任意两选手之间都会比赛一场, 每场比赛都要决出胜负, 不许出现平局. 若以选手作为顶点, 选手  $u$  胜了选手  $v$ , 则由  $u$  向  $v$  连一条弧, 由此可获得完全图的一个定向图, 这样的有向图称为竞赛图. 竞赛图的确切定义如下.



**定义 8.5.1** 完全图的每条边确定一个方向后所得的有向图称为竞赛图.

### 一、竞赛图中的王

**定理 8.5.1** 竞赛图中必有一个顶点  $v$ , 它到其他任何顶点都有长不超过 2 的有向路.

**证明** 因竞赛图的底图的任何独立集只含一个顶点, 故由定理 8.2.2, 结论成立. 证毕.

**定义 8.5.2** 如果竞赛图中一个顶点  $v$  到其他任何顶点都有长不超过 2 的有向路, 则称  $v$  是该竞赛图的一个王.

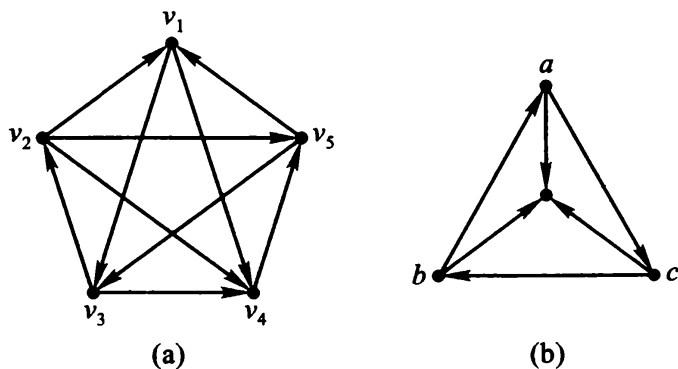
**注** 直观地讲, 若  $v$  是王, 则所有其他选手要么直接败给了  $v$ , 要么败给了  $v$  的手下败将.

定理 8.5.1 表明任何竞赛图中王总是存在的. 此外, 还有如下定理.

**定理 8.5.2** 竞赛图中出度最大的顶点必为王.

**证明** 设  $\vec{K}_\nu$  是  $\nu$  阶竞赛图, 并设  $v$  是  $\vec{K}_\nu$  中出度最大的顶点. 如果  $v$  的出度为  $\nu - 1$ , 则  $v$  显然是王. 如果  $v$  的出度为  $k < \nu - 1$ , 设  $v$  的出邻点为  $v_1, v_2, \dots, v_k$ , 而它的人邻点为  $v_{k+1}, v_{k+2}, \dots, v_{\nu-1}$ , 则对每个  $v_j (k+1 \leq j \leq \nu-1)$ ,  $v_1, v_2, \dots, v_k$  不会全是  $v_j$  的出邻点 (否则,  $v_j$  的出度为  $k+1$ , 这与最大出度为  $k$  矛盾). 因此  $v_1, v_2, \dots, v_k$  至少有一点是  $v_j$  的人邻点. 可见  $v$  到每个  $v_j$  都有长至多为 2 的有向路, 从而  $v$  是王. 证毕.

此定理表明, 在循环赛中得胜最多的选手必定是王. 但反之不真. 事实上, 王未必唯一. 例如, 下图 (a) 中, 顶点  $v_1, v_2$  都是王, 但  $v_1$  的出度为 2, 而  $v_2$  的出度为 3. 在图 (b) 中, 顶点  $a, b, c$  都是王.



**定理 8.5.3** 竞赛图  $\vec{K}_\nu$  中一个顶点  $v$  是唯一的王当且仅当  $v$  的出度为  $\nu - 1$ .

**证明 必要性** 设  $v$  是  $\vec{K}_\nu$  中唯一的王. 假如  $v$  的出度小于  $\nu - 1$ , 则它有人邻

点. 设  $v$  的所有入邻点导出的子竞赛图为  $\vec{G}$ , 则由上一定理,  $\vec{G}$  有自己的王. 设  $u$  是  $\vec{G}$  的一个王. 因  $u$  到  $v$  有弧, 故  $u$  也是  $\vec{K}_\nu$  的王. 这与  $v$  是唯一的王矛盾.

**充分性** 设  $v$  的出度为  $\nu - 1$ , 则  $v$  显然是王. 若  $\vec{K}_\nu$  还有另一王  $v'$ , 则由王的定义, 要么存在弧  $v'v$ , 要么存在长为 2 的有向路  $v'wv$ . 无论如何,  $v$  的入度至少为 1, 从而其出度至多为  $\nu - 2$ , 这与前提条件矛盾. 证毕.

这个定理说明, 如果没有全胜的选手, 则相应的竞赛图至少有两个王.

## 二、竞赛图的 Hamilton 性质

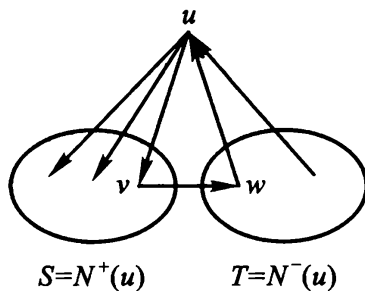
**定理 8.5.4 (Rédei, 1934)**<sup>[12]</sup> 竞赛图  $\vec{K}_\nu$  中必存在 Hamilton 有向路.

**证明** 因色数  $\chi(K_\nu) = \nu$ . 故由定理 8.2.1, 竞赛图  $\vec{K}_\nu$  中有长为  $\chi(K_\nu) - 1 = \nu - 1$  的有向路, 此即为  $\vec{K}_\nu$  的一条 Hamilton 有向路. 证毕.

**定理 8.5.5 (Moon, 1966)**<sup>[13]</sup>  $\nu \geq 3$  的强连通竞赛图的每个顶点都含在长为  $k$  的有向圈上 ( $k = 3, 4, \dots, \nu$ ).

**证明** 设  $\vec{K}_\nu$  是一个强连通竞赛图,  $u$  是  $\vec{K}_\nu$  的任一个顶点. 下面证明  $u$  在  $\vec{K}_\nu$  的长为 3 的有向圈上.

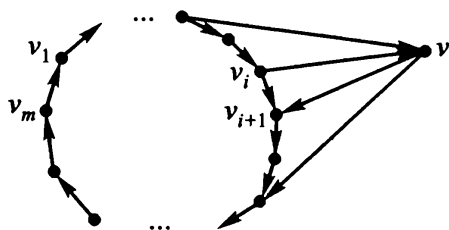
取  $S = N^+(u), T = N^-(u)$ , 则  $S$  与  $T$  非空 (因  $\vec{K}_\nu$  是强连通的, 既有从  $u$  出发的弧, 又有指向  $u$  的弧). 用  $(S, T)$  表示尾在  $S$  而头在  $T$  中的弧的集合. 由于  $\vec{K}_\nu$  是强连通竞赛图, 故  $S \cup T = V - \{u\}$ , 且  $(S, T)$  非空, 即存在弧  $(v, w) \in (S, T)$ , 于是  $u$  在三角形  $uvw$  上, 它是一个 3 阶有向圈.



下面对  $k$  用归纳法.

假设  $\vec{K}_\nu$  中有长为  $3, 4, \dots, m$  的有向圈含有  $u$  ( $m < \nu$ ). 下证必有长为  $m + 1$  的有向圈含有  $u$ .

设  $\vec{C} = v_1 v_2 \dots v_m$  是  $\vec{K}_\nu$  中含有  $u$  的一个长为  $m$  的有向圈.



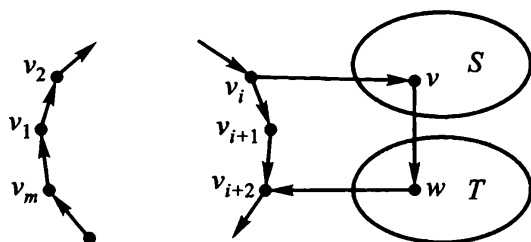
(1) 若在  $V(\vec{K}_\nu) - V(\vec{C})$  中存在顶点  $v$  满足:  $v$  是尾在  $\vec{C}$  上的一条弧的头, 又是头在  $\vec{C}$  上的一条弧的尾, 则在  $\vec{C}$  上有顶点  $v_i, v_{i+1}$ , 使得  $v_i v \in A(\vec{K}_\nu)$  且  $v_{i+1} v \in A(\vec{K}_\nu)$ , (注意  $\vec{K}_\nu$  是竞赛图, 任二顶点间都有弧). 此时  $u$  在长为  $m+1$  的圈  $v_1 \cdots v_i v v_{i+1} v_{i+2} \cdots v_m v_1$  上(如上图所示).

(2) 否则, 用  $S$  表示  $V(\vec{K}_\nu) - V(\vec{C})$  中从圈  $\vec{C}$  有弧指向的顶点之集,  $T$  表示  $V(\vec{K}_\nu) - V(\vec{C})$  中有弧指向  $\vec{C}$  的顶点之集. 由于  $m < \nu$ , 故  $S \cup T \neq \emptyset$ , 又因  $\vec{G}$  是强连通竞赛图, 从而

$$S \neq \emptyset, T \neq \emptyset, (S, T) \neq \emptyset.$$

(若  $S = \emptyset$ , 则  $\vec{C}$  到  $T$  无有向路; 同理  $T \neq \emptyset$ ; 若  $(S, T) = \emptyset$ , 则  $S$  到  $T$  无有向路).

由于当前情况不满足(1)的要求, 故  $S \cap T = \emptyset$ , 因此存在  $v \in S, w \in T$ , 使得  $(v, w) \in (S, T)$  (如下图).



注意  $w$  与  $\vec{C}$  之间的每条弧都指向  $\vec{C}$  (否则将导致情况 (1)). 同理  $v$  与  $\vec{C}$  之间的每条弧都指向  $v$ . 因此总可适当选择  $i$  使得  $u$  在圈  $v_1 v_2 \cdots v_i v w v_{i+2} \cdots v_m v_1$  上, 这是一个长为  $m+1$  的圈. 证毕.

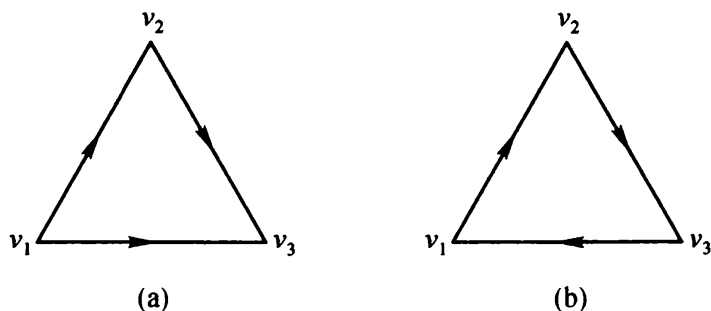
由定理 8.5.5, 如下推论显然成立.

**推论 8.5.1 (Camion, 1959)**<sup>[14]</sup> 强连通竞赛图是 Hamilton 有向图.

**注** 若一个  $\nu$  阶图 (或有向图) 有长为 3 直到长为  $\nu$  的圈 (或有向圈), 则称这个图 (或有向图) 具有泛圈性. 定理 8.5.5 表明强连通竞赛图具有泛圈性. 实际上定理 8.5.5 证明了强连通竞赛图的一个更强的性质: 如果竞赛图  $\vec{K}_\nu$  是强连通的, 则  $\vec{K}_\nu$  中过每个顶点都有长从 3 直到  $\nu$  的有向圈. 如何判断一般图 (或有向图) 是否具有泛圈性, 仍是一个悬而未决的难题.

### 三、竞赛图的传递性

在一个有向图  $\vec{G}$  中, 如果对任意的  $\langle u, v \rangle, \langle v, w \rangle \in A(\vec{G})$ , 总有  $\langle u, w \rangle \in A(\vec{G})$ , 则称  $\vec{G}$  是传递的 (transitive). 如下图中, (a) 是传递的, 但 (b) 不是传递的.



**定理 8.5.6** 一个竞赛图  $\vec{K}_\nu$  是传递的当且仅当它不含有向圈.

**证明 充分性** 设竞赛图  $\vec{K}_\nu$  不含有向圈. 对任意的  $\langle u, v \rangle, \langle v, w \rangle \in A(\vec{K}_\nu)$ , 因  $\vec{K}_\nu$  不含有向圈, 故  $\langle w, u \rangle \notin A(\vec{K}_\nu)$ , 从而  $\langle u, w \rangle \in A(\vec{K}_\nu)$ , 这表明  $\vec{K}_\nu$  是传递的.

**必要性** 用反证法. 设竞赛图  $\vec{K}_\nu$  是传递的但它包含有向圈  $C = v_1 v_2 \cdots v_k v_1$ . 由  $\vec{K}_\nu$  的传递性, 因  $\langle v_1, v_2 \rangle, \langle v_2, v_3 \rangle \in A(\vec{K}_\nu)$ , 故  $\langle v_1, v_3 \rangle \in A(\vec{K}_\nu)$ , 而  $\langle v_1, v_3 \rangle, \langle v_3, v_4 \rangle \in A(\vec{K}_\nu)$  又导致  $\langle v_1, v_4 \rangle \in A(\vec{K}_\nu)$ , 如此类推, 知  $\langle v_1, v_k \rangle \in A(\vec{K}_\nu)$ , 这与  $\langle v_k, v_1 \rangle \in A(\vec{K}_\nu)$  矛盾. 证毕.

将一个竞赛图  $\vec{K}_\nu$  的各个顶点的出度由小到大依次排列得到的序列称为  $\vec{K}_\nu$  的出度序列; 将  $\vec{K}_\nu$  的各个顶点的入度由小到大依次排列得到的序列称为  $\vec{K}_\nu$  的入度序列.

**定理 8.5.7** 一个非负整数的不减序列  $d_1, d_2, \cdots, d_\nu$  是某个传递竞赛图  $\vec{K}_\nu$  的出度序列当且仅当  $d_i = i - 1, (i = 1, 2, \cdots, \nu)$ .

**证明 充分性** 对序列  $0, 1, 2, \cdots, \nu - 1$ , 构造竞赛图  $\vec{K}_\nu$  如下: 顶点为  $v_1, v_2, \cdots, v_\nu$ , 顶点  $v_1, v_2, \cdots, v_i$  都向  $v_{i+1}$  连弧,  $(i = 1, 2, \cdots, \nu - 1)$ . 易见这样得到的竞赛图  $\vec{K}_\nu$  是传递的, 且其出度序列恰为  $0, 1, 2, \cdots, \nu - 1$ .

**必要性** 设  $\vec{K}_\nu$  是传递竞赛图, 非负整数的不减序列  $d_1, d_2, \cdots, d_\nu$  是  $\vec{K}_\nu$  的出度序列. 先来证明  $d_1, d_2, \cdots, d_\nu$  互不相等. 事实上, 对任意两点  $u, v \in V(\vec{K}_\nu)$ , 不妨设  $\langle u, v \rangle \in A(\vec{K}_\nu)$ , 用  $W$  表示  $v$  的所有出邻点构成的集合. 对任意  $w \in W$ , 因  $\langle u, v \rangle \in A(\vec{K}_\nu)$  且  $\langle v, w \rangle \in A(\vec{K}_\nu)$ , 由  $\vec{K}_\nu$  的传递性, 有  $\langle u, w \rangle \in A(\vec{K}_\nu)$ . 因此,  $d^+(u) \geq |W| + 1 = d^+(v) + 1$ . 这表明  $d_1, d_2, \cdots, d_\nu$  互不相等. 但因  $d_1, d_2, \cdots, d_\nu$  都是非负整数且不超过  $\nu - 1$ , 从而  $d_1, d_2, \cdots, d_\nu$  只能是  $0, 1, 2, \cdots, \nu - 1$ . 证毕.

同理可证如下定理 (习题 8.27).

**定理 8.5.8** 一个非负整数的不减序列  $d_1, d_2, \cdots, d_\nu$  是某个传递竞赛图  $\vec{K}_\nu$  的入度序列当且仅当  $d_i = i - 1, (i = 1, 2, \cdots, \nu)$ .

由以上定理易得下列两个推论 (习题 8.28).

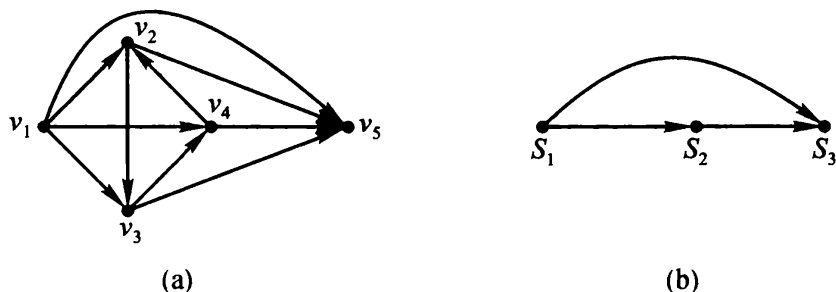
**推论 8.5.2** 在同构意义下,  $\nu$  阶传递竞赛图是存在且唯一的.

**推论 8.5.3** 对每个正整数  $\nu$ , 恰存在一个无有向圈的  $\nu$  阶竞赛图.

**定义 8.5.3** 设  $\vec{G}$  是一个有向图,  $\vec{S}$  是  $\vec{G}$  中一部分顶点导出的有向子图, 若  $\vec{S}$  是强连通的, 且给  $\vec{S}$  任意添加  $\vec{G} - \vec{S}$  中一点后导出的有向子图都不再是强连通图, 则称  $\vec{S}$  是  $\vec{G}$  的一个强连通分支.

设  $\vec{S}_1, \vec{S}_2, \dots, \vec{S}_k$  是竞赛图  $\vec{G}$  的全部强连通分支 (注意  $\bigcup_{i=1}^k V(\vec{S}_i) = V(\vec{G})$ , 且有些分支可能只含有一个顶点), 由强连通的性质,  $\vec{S}_1, \vec{S}_2, \dots, \vec{S}_k$  中任意两个都互相没有公共顶点, 且对任意的  $i, j, (1 \leq i \leq k, 1 \leq j \leq k, i \neq j)$ , 如果存在  $u \in V(\vec{S}_i), v \in V(\vec{S}_j)$ , 使得  $\langle u, v \rangle \in A(\vec{G})$ , 则对任意  $x \in V(\vec{S}_i), y \in V(\vec{S}_j)$ , 都有  $\langle x, y \rangle \in A(\vec{G})$  而  $\langle y, x \rangle \notin A(\vec{G})$ . 因此, 可以将  $\vec{S}_1, \vec{S}_2, \dots, \vec{S}_k$  作为顶点, 当且仅当存在  $u \in V(\vec{S}_i), v \in V(\vec{S}_j)$ , 使得  $\langle u, v \rangle \in A(\vec{G})$  时, 从顶点  $S_i$  向顶点  $S_j$  连弧, 这样便获得一个新的有向图, 记为  $\vec{G}^*$ , 称为  $\vec{G}$  的强连通分支图.

例如, 下图 (a) 中的竞赛图共有 3 个强连通分支;  $\vec{S}_1 = \vec{G}[v_1]$ ,  $\vec{S}_2 = \vec{G}[v_2, v_3, v_4]$ ,  $\vec{S}_3 = \vec{G}[v_5]$ , 其强连通分支图如 (b) 所示.



**定理 8.5.9** 若  $\vec{K}_\nu$  是恰有  $r$  个强连通分支的竞赛图, 则其强连通分支图  $\vec{K}_r^*$  是  $r$  阶传递竞赛图.

**证明** 显然  $\vec{K}_r^*$  中任二顶点间都有一条弧, 因此是  $r$  阶竞赛图. 而且, 对于  $\vec{K}_r^*$  中任意三个顶点  $\vec{S}_i, \vec{S}_j, \vec{S}_k$ , 如果在  $\vec{K}_r^*$  中有弧  $(\vec{S}_i, \vec{S}_j)$  和  $(\vec{S}_j, \vec{S}_k)$ , 则必定有弧  $(\vec{S}_i, \vec{S}_k) \in \vec{K}_r^*$ . 否则, 在  $\vec{K}_\nu$  中, 有从强连通分支  $\vec{S}_i$  到  $\vec{S}_j$  的弧、从  $\vec{S}_j$  到  $\vec{S}_k$  的弧、以及从  $\vec{S}_k$  到  $\vec{S}_i$  的弧, 根据前面的分析, 此时  $V(\vec{S}_i) \cup V(\vec{S}_j) \cup V(\vec{S}_k)$  生成  $\vec{K}_\nu$  的一个更大的强连通子图, 这与  $\vec{S}_i, \vec{S}_j, \vec{S}_k$  是  $\vec{K}_\nu$  的强连通分支矛盾. 因此,  $\vec{K}_r^*$  是传递的竞赛图. 证毕.

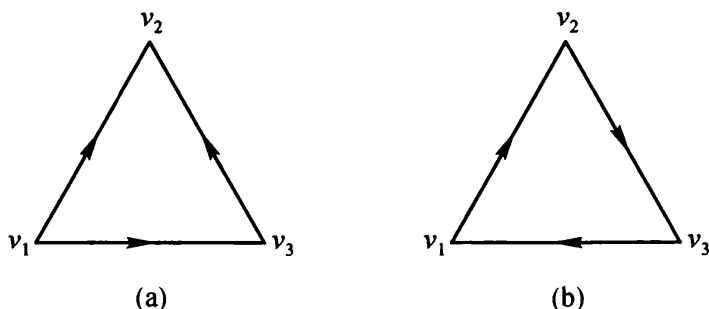
上述定理揭示了竞赛图的结构: 任何竞赛图都是由它的强连通分支 (强连通子竞赛图, 可以只含单个顶点) 以传递竞赛图的方式连接而成的. 对一般有向图, 类似的结论为: 任何有向图的强连通分支图都是传递的 (习题 8.30).

#### 四、竞赛图的应用 —— 循环赛名次排列问题

**循环赛名次的排列问题:** 在若干选手间进行循环赛, 已知各对选手间比赛的胜负关系 (任意两选手之间都比赛一场, 不出现平局), 求比赛结束后所有选手的名次排列.

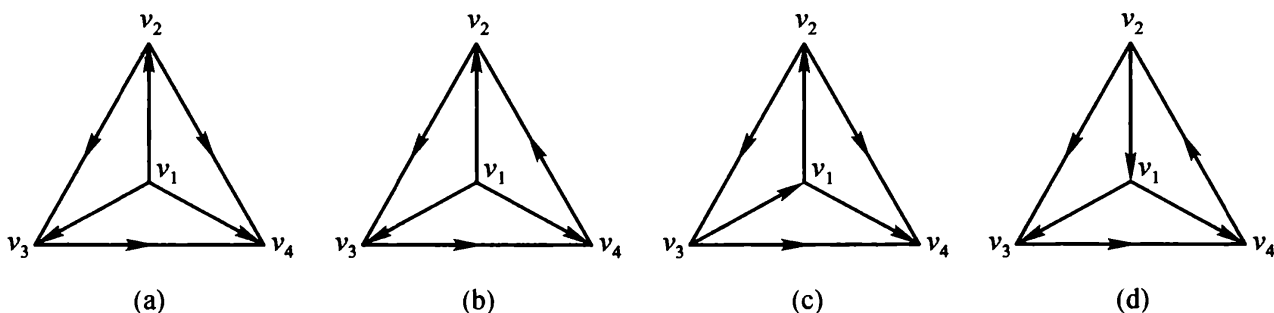
**建立该问题的图论模型:** 以选手作为顶点, 若选手  $u$  胜了选手  $v$ , 则由  $u$  向  $v$  连一条弧. 这样可获得一个竞赛图  $\vec{K}_n$  (假定共有  $n$  名选手). 选手  $u$  在比赛中获胜的次数体现为  $u$  在  $\vec{K}_n$  中的出度. 问题转化为: 将竞赛图  $\vec{K}_n$  的所有顶点依出度由大到小排序.

先来考察三个顶点的竞赛图, 有如下两种情况 (不考虑顶点的标号).



我们将竞赛图各个顶点的出度排成的序列称为**得分向量**. 对于情况 (a), 三个选手  $(v_1, v_2, v_3)$  的得分向量为  $(2, 0, 1)$ , 名次排列显然应是  $\{v_1, v_3, v_2\}$ , 注意此时竞赛图是传递的; 对于情况 (b), 三个选手各胜一场, 得分依次为  $(1, 1, 1)$ , 三个选手名次相同.

再看四个顶点的竞赛图, 在同构意义下共有如下四种形式.



对情况 (a), 四个选手  $(v_1, v_2, v_3, v_4)$  的得分向量为  $(3, 2, 1, 0)$ , 名次排列无疑应为  $\{v_1, v_2, v_3, v_4\}$ , 注意此时竞赛图是传递的; 对情况 (b), 选手  $v_1$  显然应排第一, 其余三个选手名次相同, 四个选手得分依次为  $(3, 1, 1, 1)$ ; 对情况 (c), 选手  $v_4$  应排在最后, 其余三个选手名次相同, 四个选手得分依次为  $(2, 2, 2, 0)$ ; 对情况 (d), 四个选手得分依次为  $(2, 2, 1, 1)$ . 后三种情况, 出现并列排名.

由前面的定理及以上例子可见, 若竞赛图  $\vec{K}_n$  是传递的, 则各顶点的出度互不相同, 可以得出唯一确定的排名. 若  $\vec{K}_n$  不是传递的, 则会出现并列排名. 对于并列排名, 有些是无法继续排出名次先后的. 比如上述三个顶点的情况 (b) 和四个顶点的情况 (b)、(c), 得分向量分别为  $(1, 1, 1)$ 、 $(3, 1, 1, 1)$  和  $(2, 2, 2, 0)$ , 从图中观察可知得分

相等的顶点完全对称, 并列的选手无法进一步排名. 但有些情况下并列选手可进一步排名, 比如上述四个顶点的情况 (d), 得分向量为  $(2, 2, 1, 1)$ ,  $v_1$  与  $v_2$  得分都为 2, 他们各自打败了两个对手. 但是, 被  $v_1$  打败的两个对手  $v_3, v_4$  的出度之和为 2, 被  $v_2$  打败的两个对手  $v_1, v_3$  的出度之和为 3, 这表明  $v_2$  打败的对手总体上比  $v_1$  打败的对手实力更强些, 因此应将  $v_2$  排在  $v_1$  之前.

在竞赛图中, 一个顶点的出度表示该选手所打败的选手数, 因此就是该选手的得分. 定义竞赛图的邻接矩阵  $A = (a_{ij})_{\nu \times \nu}$  如下:

$$a_{ij} = \begin{cases} 1 & \text{存在从顶点 } i \text{ 到 } j \text{ 的有向边} \\ 0 & \text{否则} \end{cases}, \quad (i, j = 1, 2, \dots, \nu).$$

则一个竞赛图的邻接矩阵中第  $i$  行的行和等于该竞赛图中第  $i$  个顶点的出度, 即第  $i$  个选手的得分. 令  $e = (1, 1, \dots, 1)^T$ , 则竞赛图的顶点得分向量为

$$s^{(1)} = Ae. \quad (*)$$

我们称此得分向量为直接得分向量或一阶得分向量.

对上述四个顶点的情况 (a), 不难按 (\*) 式算出其一阶得分向量为  $(3, 2, 1, 0)$ , 由此立即可得出选手的排名. 对四个顶点的情况 (d), 得分向量为  $s^{(1)} = A \cdot e = (2, 2, 1, 1)^T$ , 此时由一阶得分向量  $s^{(1)}$  只能给出部分排名  $\{(v_1, v_2), (v_3, v_4)\}$ , 其中圆括号内的选手需进一步排名.

注意到矩阵  $A^2$  的第  $i$  行第  $j$  列位置元素表示有向图中从顶点  $v_i$  出发两步可以到达的顶点的数目,  $A^2$  的第  $i$  行元素之和反映了被选手  $v_i$  打败的那些选手获胜的次数之和, 即选手  $v_i$  间接 (两步) 打败的选手的数量之和. 将  $A^2$  的行和构成的向量记为  $s^{(2)}$ , 称为竞赛图的二阶得分向量. 显然,  $s^{(2)} = A^2 e$ .

在两个选手的一阶得分相同的前提下, 应进一步比较它们的二阶得分. 例如对上述四个顶点的情况 (d), 其二阶得分向量为  $s^{(2)} = A^2 \cdot e = (3, 2, 1, 2)^T$ , 因此两部分内部排名应分别为  $\{v_1, v_2\}, \{v_4, v_3\}$ . 从而四个顶点的完全排名应为  $\{v_1, v_2, v_4, v_3\}$ .

若由  $s^{(2)}$  仍不能确定顶点的完全排名, 再将矩阵  $A^2$  中所有非零元改写为 1, 得矩阵  $\tilde{A}^2$ .  $\tilde{A}^2$  中第  $i$  行第  $j$  列元素的 0、1 状态表示从  $v_i$  到  $v_j$  是否有长为 2 的有向途径, 即是否  $v_i$  间接地打败了  $v_j$ . 而矩阵  $B = \tilde{A}^2 \cdot A$  的第  $i$  行元素的和表示  $v_i$  间接 (两步) 打败的选手再打败选手的数量之和. 该和可用三阶得分向量  $s^{(3)} = \tilde{A}^2 \cdot A \cdot e$  来计算. 此前局部尚未排出名次部分可按  $s^{(3)}$  中对应的得分进行内部排序. 类似地, 将矩阵  $B = \tilde{A}^2 \cdot A$  中所有非零元改写为 1, 得矩阵  $\tilde{A}^3$ , 并求出四阶得分向量  $s^{(4)} = \tilde{A}^3 \cdot A \cdot e$ , 此前局部尚未排出名次部分可按  $s^{(4)}$  中对应的得分进行内部排序. 如此递推, 直至求出  $\nu$  阶得分向量  $s^{(\nu)}$  时仍未确定名次的部分, 可作为并列排名.

## §8.6 根树及其应用

## 一、根树

**定义 8.6.1** 设  $T$  是一棵有向树. 若  $T$  中有一个顶点的入度为 0, 其余顶点的入度均为 1, 则称  $T$  为根树. 入度为 0 的顶点称为树根, 入度为 1 且出度为 0 的顶点称为树叶, 其余顶点称为内点, 树根和内点称为分支点.

**注** (1) 设  $v$  是根树  $T$  中一个顶点, 从树根到顶点  $v$  的通路长度称为  $v$  的层数,  $T$  中各点层数的最大值称为根树  $T$  的树高. 如下图 2 中的根树有 4 层, 分别为第 0 层 ~ 第 3 层, 树高为 3.

(2) 由于根树中任何两条有向边不会指向同一个顶点, 因此各有向边的方向是一致的, 所以画根树的图示时通常将树根画在最上方, 并省去各边 (向下) 的箭头 (如下图 3 所示).

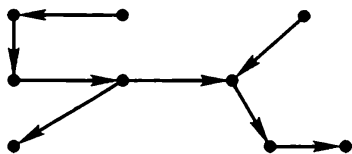


图 1. 有向树

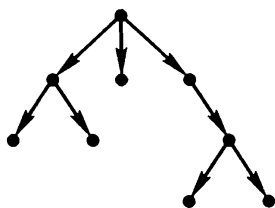


图 2. 一棵高为 3 的根树

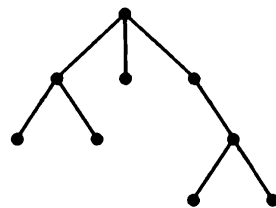


图 3. 根树省去箭头的图示

**定义 8.6.2** 设  $T$  为一棵非平凡的根树,  $u, v$  是  $T$  中两点. 若  $u$  到  $v$  有有向路, 则称  $u$  是  $v$  的祖先,  $v$  是  $u$  的后代; 特别地, 若  $u$  到  $v$  有有向边, 则称  $u$  是  $v$  的父亲,  $v$  是  $u$  的儿子. 若  $u$  和  $v$  有同一个父亲, 则称  $u$  和  $v$  是兄弟.

**定义 8.6.3** (1) 若将根树  $T$  中层数相同的顶点都标定次序 (比如从左到右), 则称  $T$  为有序树.

(2) 若根树  $T$  的每个顶点至多有  $r$  个儿子, 则称  $T$  为  $r$  叉树; 若  $r$  叉树又是有序的, 则称它为  $r$  叉有序树.

(3) 若根树  $T$  的每个非叶子顶点都恰好有  $r$  个儿子, 则称  $T$  为  $r$  叉正则树; 若  $r$  叉正则树又是有序的, 则称它为  $r$  叉正则有序树.

(4) 若  $T$  为  $r$  叉正则树, 且每个树叶的层数都等于树高, 则称  $T$  为  $r$  叉完全正则树; 若  $r$  叉完全正则树又是有序的, 则称它为  $r$  叉完全正则有序树.

**定义 8.6.4** 设  $T$  为一棵根树,  $v$  是  $T$  中一个顶点. 称  $v$  及其后代的导出子图为  $T$  的以  $v$  为根的子树. 2 叉正则有序树的每个非叶子顶点的两个儿子导出的子树分别称为左子树和右子树.



## 二、2 叉树与淘汰赛

**定理 8.6.1** 有  $m$  个分支点的  $t$  叉正则树  $T$  的阶为  $\nu = tm + 1$ .

**证明** 因为  $T$  中除了树根外, 其余点都是儿子. 每个分支点有  $t$  个儿子, 故  $m$  个分支点共有  $t \cdot m$  个儿子. 因此  $\nu = tm + 1$ . 证毕.

**推论 8.6.1** 有  $m$  个分支点的  $t$  叉正则树  $T$  有  $(t-1)m + 1$  个叶子.

**证明** 因  $T$  中除了分支点便是叶子, 故由上述定理,  $T$  共有  $\nu - m = (t \cdot m + 1) - m = (t-1)m + 1$  个叶子. 证毕.

下一个推论是显然的.

**推论 8.6.2** 有  $m$  个分支点的 2 叉正则树  $T$  有  $2m + 1$  个顶点, 其中有  $m + 1$  个叶子.

**定理 8.6.2** 设  $T$  是高为  $h$  的  $\nu$  阶 2 叉树, 则  $h + 1 \leq \nu \leq 2^{h+1} - 1$ .

**证明** 设  $T$  中第  $i$  层的顶点有  $n_i$  个 ( $0 \leq i \leq h$ ), 则  $\sum_{i=0}^h n_i = \nu$ . 另一方面, 因  $1 \leq n_i \leq 2^i$ , 故

$$h + 1 = \sum_{i=0}^h 1 \leq \sum_{i=0}^h n_i \leq \sum_{i=0}^h 2^i = 2^{h+1} - 1.$$

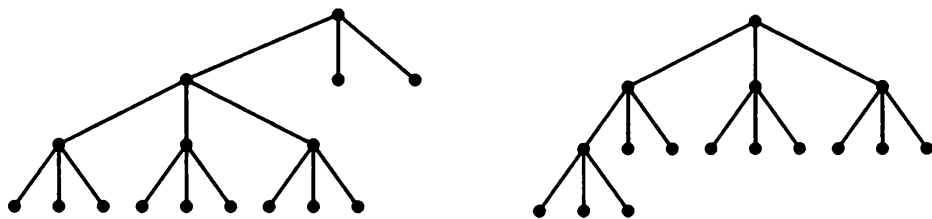
因此  $h + 1 \leq \nu \leq 2^{h+1} - 1$ . 证毕.

**定理 8.6.3** 设  $T$  是高为  $h$  的  $\nu$  阶 2 叉树, 则  $h \geq \left\lceil \log_2 \left( \frac{\nu + 1}{2} \right) \right\rceil$ .

**证明** 由上一定理,  $\nu \leq 2^{h+1} - 1$ , 即  $2^h \geq \frac{\nu + 1}{2}$ . 从而  $h \geq \left\lceil \log_2 \left( \frac{\nu + 1}{2} \right) \right\rceil$ . 证毕.

**例 8.6.1** 假设某台计算机有一条加法指令, 每次执行该指令可计算 3 个数之和. 如果要计算 11 个数之和, 问至少需要执行几次这个加法指令.

**解** 指令执行过程可用一棵 3 叉正则树来表示. 将 11 个数作为树叶, 每三个数执行一次加法指令, 所得之数作为它们的上一层分支点 (父节点), 在下一层无法参与运算的数直接进入上一层参与运算 (或在下一层先做一次运算). 则因有 11 个叶子, 且叉数  $t=3$ , 由推论 8.6.1, 分支点有  $m = \frac{11-1}{3-1} = 5$  个, 故至少需执行这个加法指令 5 次. 例如, 下列两棵 3 叉树表示两种不同的执行方案, 都执行了 5 次加法指令.



2 叉树可用于描述淘汰赛. 在淘汰赛中, 一旦一个选手输了一场比赛, 就被淘汰出局. 有  $n$  个选手参加的淘汰赛可用一棵有  $n$  片叶子的 2 叉树来表示. 树叶表示选手, 选手间两两配对进行比赛. 两个选手比赛一场后, 胜者作为它们的上一层 2 分叉点 (父节点), 未能被配对的一个选手 (如果存在的话) 直接进入上一层. 上一层节点 (选手) 再分别配对比赛后, 决定出更上一层节点, 如此反复, 直至得到树根, 便产生出冠军. 所形成的树的树高表示最长赛程. 一个选手所在的层数表示他要取得冠军需要获胜的比赛次数.

### 三、2 叉树与二元前缀码

**定义 8.6.5** 设  $\alpha_1\alpha_2\cdots\alpha_n$  是长为  $n$  的符号串, 其子串  $\alpha_1, \alpha_1\alpha_2, \cdots, \alpha_1\alpha_2\cdots\alpha_{n-1}$  分别称为该符号串的长度为  $1, 2, \cdots, n-1$  的前缀. 设  $A = \{\beta_1, \beta_2, \cdots, \beta_m\}$  为一个符号串集合. 若对  $\forall \beta_i, \beta_j \in A, (i \neq j), \beta_i$  与  $\beta_j$  都互不为前缀, 则称  $A$  为前缀码. 若  $A$  中每个符号串中都只出现 0, 1 两个符号, 则称  $A$  为二元前缀码.

例如,  $\{1, 00, 011, 0101, 01001, 01000\}$  是二元前缀码, 而  $\{1, 00, 011, 0101, 0100, 01001\}$  不是前缀码.

利用 2 叉树可产生二元前缀码.

**定理 8.6.4** 由一棵给定的 2 叉树可以产生一个二元前缀码.

**证明** 设  $T$  是具有  $t$  片树叶的 2 叉树.  $T$  的任何分支点  $v$  必有 1 个或 2 个儿子. 若  $v$  有两个儿子, 在由  $v$  引出的两条边上, 左边的标上 0, 右边的标上 1. 若  $v$  只有一个儿子, 由它引出的边可以标 0 也可以标 1. 设  $u$  是  $T$  的任一片树叶, 从树根到  $u$  的通路各边的标号按顺序组成的一个符号串, 将其放在  $u$  处. 因  $u$  处的符号串的前缀均在  $u$  所在的通路上取得, 故树叶处获得的符号串互相不是前缀. 因此  $t$  片树叶对应的  $t$  个符号串组成的集合构成一个二元前缀码. 证毕.

**例 8.6.2** 由下图 1 中的 2 叉树所产生的一个二元前缀码为  $\{01, 11, 000, 0010, 0011\}$ . 产生过程如图 2 所示.

如果将其中只有一个儿子的点发出的边标上 0, 则得另一个二元前缀码:

$$\{01, 10, 000, 0010, 0011\}.$$

容易看出, 若一个 2 叉树是正则的, 则由这个正则 2 叉树只能产生唯一的二元前

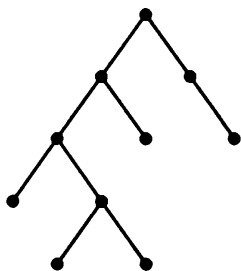


图 1

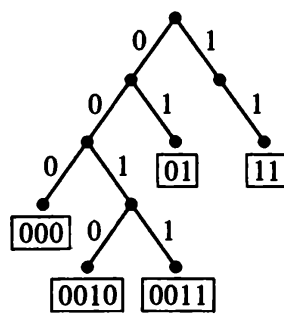


图 2

缀码.

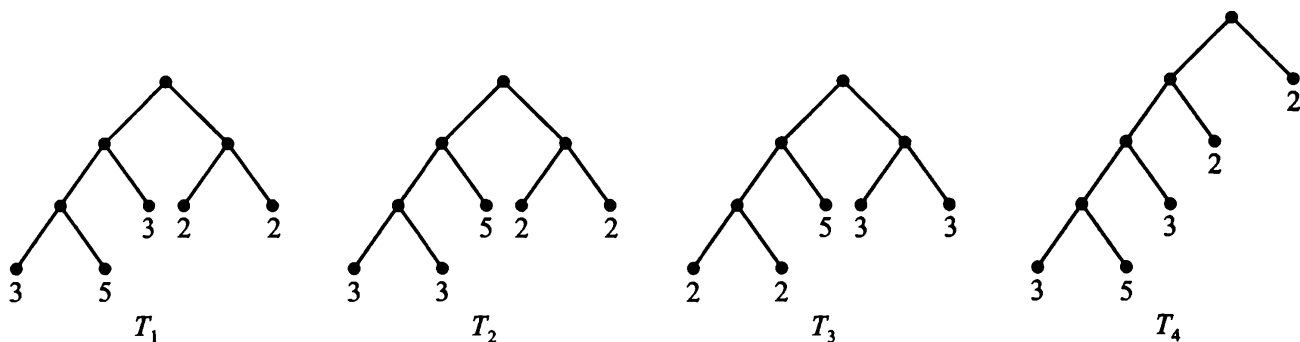
一个二元前缀码中的码字有长有短, 每个码字都可以代表某个符号, 比如上例中第一组前缀码中的 5 个码字可以分别代表字母  $A, B, C, D, E$ . 当这些字母在文字中出现的频率不同时, 用哪个码字代表哪个字母, 需要优化安排, 因为不同的安排方案其效率是不同的 (使用 0,1 符号的数量不同). 一个基本的想法是, 尽可能用较短的码字代表使用频率较高的字母. 这种使得 0、1 符号的数量最为节省的二元前缀码安排称为最佳前缀码. 最佳前缀码可通过最优 2 叉树得到.

**定义 8.6.6** 若 2 叉树  $T$  有  $t$  片树叶, 分别标有权 (数字)  $w_1, w_2, \dots, w_t$ , 则称  $T$  是带权  $w_1, w_2, \dots, w_t$  的二叉树. 若权  $w_1, w_2, \dots, w_t$  对应的树叶分别为  $v_1, v_2, \dots, v_t$ , 它们的层高分别为  $l(v_1), l(v_2), \dots, l(v_t)$ , 则称

$$W(T) = \sum_{i=1}^t w_i l(v_i)$$

为树  $T$  的权. 在所有含  $t$  片叶子. 带权  $w_1, w_2, \dots, w_t$  的 2 叉树中, 权最小的称为最优 2 叉树.

容易看出, 最优 2 叉树必是 2 叉正则树. 此外, 注意叶子的权  $w_1, w_2, \dots, w_t$  是一个无序数组. 如果两组权分别按由小到大排序后, 权值对应相等, 便认为这两组权相同, 因此通常可将权值按由小到大顺序书写. 对给定的 2 叉树及同样的一组权, 各片叶子上的权可以有不同的带法; 具有不同结构的两棵 2 叉树, 其叶子的带权也可能相同. 例如, 下图所示的 4 棵树都是带权 2,2,3,3,5 的 2 叉树. 它们的权分别为  $W(T_1) = 38, W(T_2) = 36, W(T_3) = 34, W(T_4) = 47$ .



**定理 8.6.5** 在带权  $w_1 \leq w_2 \leq \cdots \leq w_t$  的最优二叉树  $T$  中, 总可以使带权  $w_1$  和  $w_2$  的树叶为兄弟, 且它们的层数为树高.

**证明** 设  $v_1, v_2$  是最优二叉树  $T$  中带权  $w_1, w_2$  的叶子顶点,  $v$  是  $T$  中层数最大的分支点之一,  $v$  的两个儿子(必为树叶)设为  $x$  和  $y$ ,  $x$  带权  $w_x, y$  带权  $w_y$ . 不妨设  $w_x \leq w_y$ . 由于  $x$  和  $y$  的层数等于树高, 故层数  $l(x) \geq l(v_1), l(y) \geq l(v_2)$ .

下面来证明, 在不影响  $T$  的最优性的前提下, 总可以使  $w_x = w_1, w_y = w_2$ , 从而完成定理的证明.

事实上, 当  $l(x) = l(v_1)$  时, 若  $w_x \neq w_1$ , 则可将  $w_x$  与  $w_1$  对调, 使  $x$  带权  $w_1$ , 这样做不会影响  $T$  的权.

当  $l(x) > l(v_1)$  时, 必定有  $w_x = w_1$ . 若不然,  $w_x > w_1$ , 将  $w_x$  与  $w_1$  对调, 使  $x$  带权  $w_1$  而  $v_1$  带权  $w_x$ , 得带权  $w_1, w_2, \cdots, w_t$  的二叉树  $T'$  (与  $T$  的结构相同, 但叶子点的权有变化), 并且

$$W(T) - W(T') = [w_x l(x) + w_1 l(v_1)] - [w_1 l(x) + w_x l(v_1)] = [l(x) - l(v_1)](w_x - w_1)$$

因  $l(x) > l(v_1)$  且  $w_x > w_1$ , 故  $W(T) > W(T')$ . 这与  $T$  是最优二叉树矛盾.

这样便证明了在最优二叉树  $T$  中, 总可以使  $x$  的权  $w_x = w_1$ . 同理可证, 总可以使  $y$  的权  $w_y = w_2$ . 证毕.

**定理 8.6.6** 设权  $w_1 \leq w_2 \leq \cdots \leq w_t, T'$  是带权  $w_1 + w_2, w_3, \cdots, w_t$  的最优二叉树. 给  $T'$  中带权  $w_1 + w_2$  的树叶添加两个儿子并分别赋以权  $w_1, w_2$ , 则所得之树  $T$  是带权  $w_1, w_2, \cdots, w_t$  的最优二叉树.

**证明** 由  $T$  的构造知,  $W(T) = W(T') + w_1 + w_2$ . 设  $T^*$  是带权  $w_1, w_2, \cdots, w_t$  的最优二叉树. 由定理 8.6.5, 总可以认为在  $T^*$  中带权  $w_1$  和  $w_2$  的顶点  $v_1$  和  $v_2$  是兄弟, 且它们所在的层数为树高. 设  $v_1$  和  $v_2$  的父亲为  $u$ , 在  $T^*$  中删除顶点  $v_1$  和  $v_2$ , 并给它们的父亲  $u$  赋权  $w_1 + w_2$ , 记所得的树为  $\bar{T}$ , 则有

$$W(T^*) = W(\bar{T}) + w_1 + w_2.$$

因  $\bar{T}$  是带权  $w_1 + w_2, w_3, \cdots, w_t$  的二叉树, 而  $T'$  是带权  $w_1 + w_2, w_3, \cdots, w_t$  的最优二叉树, 故  $W(T') \leq W(\bar{T})$ , 从而

$$W(T) = W(T') + w_1 + w_2 \leq W(\bar{T}) + w_1 + w_2 = W(T^*).$$

但因  $T$  是带权  $w_1, w_2, w_3, \cdots, w_t$  的二叉树, 而  $T^*$  是带权  $w_1, w_2, \cdots, w_t$  的最优二叉树, 因此只可能  $W(T) = W(T^*)$ , 这表明  $T$  也是带权  $w_1, w_2, \cdots, w_t$  的最优二叉树. 证毕.

根据定理 8.6.6, 为了求带  $t$  个权  $w_1, w_2, \dots, w_t$  的最优二叉树, 我们可以先画出两个兄弟顶点  $v_1, v_2$ , 分别赋以权  $w_1$  和  $w_2$ , 然后求出带  $t-1$  个权  $w_1+w_2, w_3, \dots, w_t$  的最优二叉树  $T'$ , 将  $v_1, v_2$  作为儿子与  $T'$  中带权  $w_1+w_2$  的叶子顶点连边即可. 从而将求带  $t$  个权  $w_1, w_2, \dots, w_t$  的最优二叉树的问题转化为求带  $t-1$  个权  $w_1+w_2, w_3, \dots, w_t$  的最优二叉树. 同理, 先找出  $w_1+w_2, w_3, \dots, w_t$  中两个最小的权, 又可以将求带  $t-1$  个权的最优二叉树的问题转化为求带  $t-2$  个权的最优二叉树问题. 依此类推, 最后归结为求带有两个权的最优二叉树问题. 由此可得到求最优二叉树的 Huffman 算法.

### Huffman 算法—求最优二叉树

输入: 实数 (权)  $w_1, w_2, \dots, w_t$ .

输出: 带权  $w_1, w_2, \dots, w_t$  的最优 2 叉树.

第 0 步: 引入顶点  $v_1, v_2, \dots, v_t$ , 并给各顶点依次赋权  $w(v_k) = w_k, (k = 1, 2, \dots, t)$ .

令

$$V = \{v_1, v_2, \dots, v_t\}, T = \{v_1, v_2, \dots, v_t\}, i := 1.$$

第 1 步: 从  $V$  中选取权最小的两个顶点  $v', v''$ , 引入一个新顶点  $u_i$ , 作为  $v', v''$  的父顶点 (即  $u_i$  与  $v', v''$  分别连边), 令  $w(u_i) = w(v') + w(v'')$ ,  $V := (V - \{v', v''\}) \cup \{u_i\}$ ,  $T := T \cup \{u_i\}$ , 转下步.

第 2 步: 若  $i = t - 1$ , 则输出所得的最优 2 叉树  $T$ , 结束; 否则,  $i := i + 1$ , 转第 1 步.

算法中,  $T$  记录形成 2 叉树过程中得到的树顶点的集合, 其中初始步得到的  $T = \{v_1, v_2, \dots, v_t\}$  中均为 2 叉树的叶子点, 后续循环第 1 步过程中逐步得到的  $u_i$  为 2 叉树的分支顶点. 2 叉树的连边可以通过给  $u_i$  做标号记录.  $V$  表示每轮循环生成分支点  $u_i$  时可供选择的顶点集合.

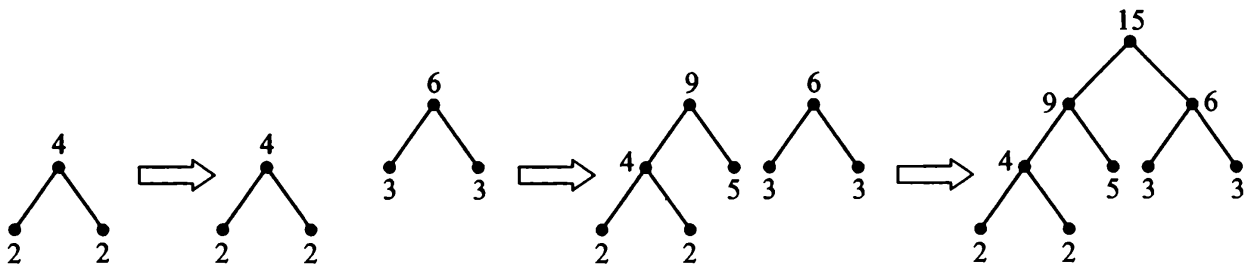
算法所得之图显然无圈, 且有  $t$  片叶子. 算法每一次循环产生 1 个分支点. 由于  $t$  片叶子的 2 叉树有  $t-1$  个分支点 (习题 8.32), 因此算法在循环  $t-1$  次后结束. 此时共产生出  $t-1$  个分支点. 因而所得之图有  $\nu = t + (t-1) = 2t-1$  个顶点, 且因每个分支点对应有 2 条边, 故有  $\varepsilon = 2(t-1) = 2t-2 = \nu-1$  条边. 可见所得之图是一棵树, 且为 2 叉树. 进一步地, 由定理 8.6.6 易知, 算法得到的 2 叉树是最优 2 叉树.

算法的主要计算量在于循环执行第 1 步. 每循环一次第 1 步,  $V$  集合中减少一个顶点, 在第  $i$  次循环执行第 1 步时, 要从  $|V| = t - (i-1)$  个顶点中选出 2 个权最小的顶点, 这需要  $(|V|-1) + (|V|-2) = 2t - 2i - 1$  次比较, 此外需进行 1 次加法、2 次集合运算、2 次赋值. 而执行第 2 步需 1 次判断、1 次赋值. 故算法第  $i$  次循环的计算量为  $2t - 2i + 6$ .  $t-1$  轮循环总的计算量为

$$\sum_{i=1}^{t-1} (2t - 2i + 6) = O(t^2).$$

例 8.6.3 求带权 2, 2, 3, 3, 5 的最优 2 叉树.

解 运用 Huffman 算法, 构造如下:



最终得到一棵最优 2 叉树, 权为  $2 \times 3 + 2 \times 3 + 5 \times 2 + 3 \times 2 + 3 \times 2 = 34$ . (注意权的一种简便计算方式:  $4 + 9 + 6 + 15 = 34$ ).

利用最优二叉树可求得最佳前缀码.

例 8.6.4 在某种通信中, 使用八进制数字串. 0~7 各个数字出现的频率如下:

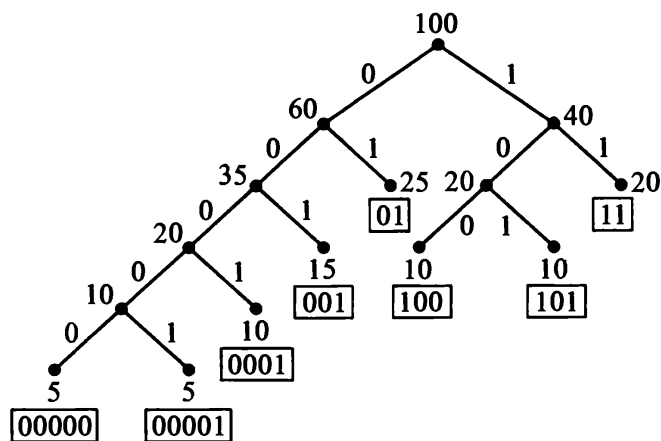
0 : 25%; 1 : 20%; 2 : 15%; 3 : 10%; 4 : 10%; 5 : 10%; 6 : 5%; 7 : 5%.

(1) 求传输它们的最佳前缀码.

(2) 问传输  $10^n$  个按上述比例出现的八进制数字需要多少个二进制数字位 (bit 数, 即 0,1 的个数)?

(3) 如果用等长码字 (长为 3) 传输  $10^n$  个按上述比例出现的八进制数字, 需要多少个二进制数字位?

解 用 100 乘以各个数字出现的频率作为权, 将权由小到大排列;  $w_0 = 5, w_1 = 5, w_2 = 10, w_3 = 10, w_4 = 10, w_5 = 15, w_6 = 20, w_7 = 25$ , (记住它们对应哪个数字). 用 Huffman 算法求出最优 2 叉树如下:



由此获得前缀码 {01, 11, 001, 100, 101, 0001, 00000, 00001}, 且各码字对应传输数字为: 01 传 0; 11 传 1; 001 传 2; 100 传 3; 101 传 4; 0001 传 5; 00000 传 6; 00001 传 7.

由 2 叉树的最优性可知所得前缀码 {01, 11, 001, 100, 101, 0001, 00000, 00001} 是最佳的.

用所得前缀码按题中给定频率发送 100 个八进制数字所用的二进制数字位的个数为:

$$100(5 \times 5\% + 5 \times 5\% + 4 \times 10\% + 3 \times 15\% + 3 \times 10\% + 3 \times 10\% + 2 \times 25\% + 2 \times 20\%) = 285.$$

故传输  $10^n$  个按上述频率出现的八进制数字需要  $285 \times 10^{n-2} = 2.85 \times 10^n$  个二进制数字.

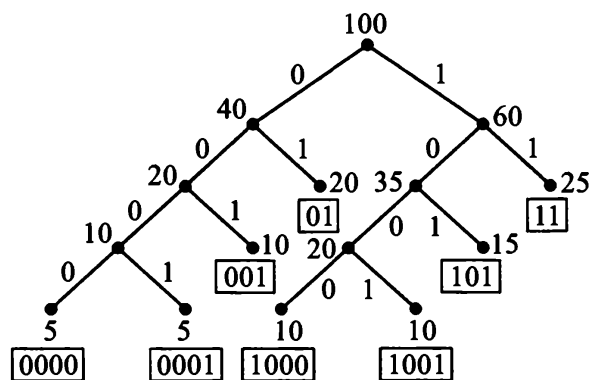
若用等长前缀码 {000, 001, 010, 011, 100, 101, 110, 111} 传输  $10^n$  个按上述频率出现的八进制数字需要  $3 \times 10^n$  个二进制数字位.

注 (1) 利用上述最优 2 叉树发送 100 个八进制数字所用的二进制数字位数, 恰等于该最优 2 叉树的权. 这个最优 2 叉树的权也可用分支点上的权累加计算:

$$W(T) = 10 + 20 + 35 + 20 + 60 + 40 + 100 = 285.$$

(2) 最佳前缀码并不唯一. 由于每一步选择两个最小权的选法不同, 且两个权对应的顶点所放左右位置不同, 得到的最优树可能不同, 由此可能得到不同的最佳前缀码.

例如, 如下的 2 叉树也是一棵最优树, 权为  $10 + 20 + 20 + 35 + 40 + 60 + 100 = 285$ , 因此, 由这个最优 2 叉树给出的前缀码也是满足上例要求的最佳前缀码.



前缀码 {01, 11, 001, 101, 0000, 0001, 1000, 1001}, 各码字对应传输的数字为:

11 传 0; 01 传 1; 101 传 2; 001 传 3; 1000 传 4; 1001 传 5; 0000 传 6; 0001 传 7.

#### 四、2 叉树的遍历

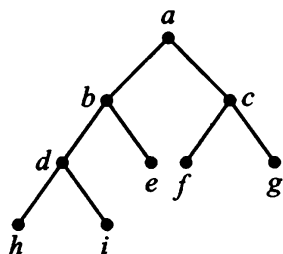
访问一棵根树的每个顶点一次且仅一次称为遍历这棵树. 遍历一棵有序正则二叉树的方法主要有以下三种;

1. 中序遍历法: 访问次序为“左子树、树根、右子树”;
2. 前序遍历法: 访问次序为“树根、左子树、右子树”;

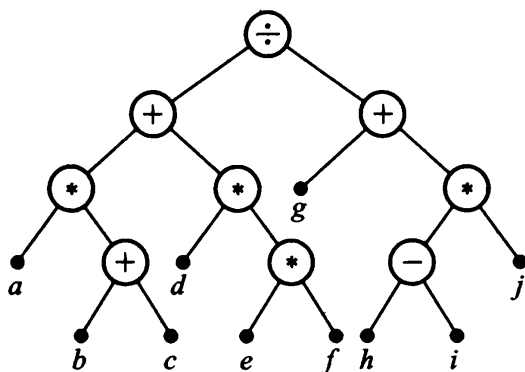
3. 后序遍历法: 访问次序为“左子树、右子树、树根”。

例如, 下图所示的 2 叉有序正则树, 按中序、前序、后序遍历的结果分别如下:

$$((hdi)be)a(fcg); a(b(dhi)e)(cfg); ((hid)eb)(fgc)a$$



利用 2 叉有序正则树和确定的遍历方法可表示四则运算的算式. 一般将最高层次的运算符放在树根上, 然后依次将运算符放在根子树的树根上., 参加运算的数字放在树叶上, 被除数、被减数总是放在左子树树叶上. 例如算式  $(a * (b + c) + d * e * f) \div (g + (h - i) * j)$  在中序遍历法下可用下列 2 叉有序正则树表示:



关于有向图的更多内容可参看文献 [15]~[32].

## 习题八

- 8.1 证明: 对任何有向图  $\vec{G}$ ,  $\sum_{v \in V(\vec{G})} d^-(v) = \sum_{v \in V(\vec{G})} d^+(v) = \varepsilon(\vec{G})$ .
- 8.2 在一次围棋擂台赛中, 双方各出  $n$  名选手. 比赛的规则是双方先各自排定一个次序, 设甲方排定的次序为  $x_1, x_2, \dots, x_n$ , 乙方排定的次序为  $y_1, y_2, \dots, y_n$ .  $x_1$  与  $y_1$  先比赛, 胜的一位与对方的下一位选手比赛. 按这种方法进行比赛, 直到有一方的最后一位选手出场比赛并且输给对方时, 比赛结束. 问最多进行几场比赛便可定出胜负 (假定比赛不出现平局)?
- 8.3 设  $G$  是具有  $\nu$  个顶点  $\varepsilon$  条边的无向简单图, 问  $G$  共有多少个定向图?
- 8.4 证明对 Peterson 图以下结论不成立: 每个图都有一个定向图, 使得对每个  $S \subseteq V$ ,  $(S, \bar{S})$  和  $(\bar{S}, S)$  的弧数相差最多为 1.



- 8.5 证明: 任一图  $G$  都有一个定向图  $\vec{G}$ , 使得对每个顶点  $v$  都有  $|d^+(v) - d^-(v)| \leq 1$ .
- 8.6 (1) 任何连通的简单图  $G$  都有一种定向方式, 使得所得的有向图  $\vec{G}$  中最长路的长等于  $\chi(G) - 1$ ;  
 (2) 任何连通的简单图  $G$  都有一种定向方式, 使得所得的有向图  $\vec{G}$  中每条有向路的长度不超过  $\Delta(G)$ .
- 8.7 设  $\vec{G}$  是一个简单有向图, 且  $\max\{\Delta^-(\vec{G}), \Delta^+(\vec{G})\} = k$ , 证明:  $\vec{G}$  中必包含长大于或等于  $k$  的有向路.
- 8.8 (1) 证明: 若有向图  $\vec{G}$  中无有向圈, 则  $\delta^- = \delta^+ = 0$ ;  
 (2) 证明或举出反例: 若有向图  $\vec{G}$  满足  $\delta^- = \delta^+ = 0$ , 则  $\vec{G}$  中无有向圈.
- 8.9 设  $n$  名棋手进行循环赛 ( $n \geq 3$ ), 没有出现平局. 证明: 若不出现甲胜乙、乙胜丙、丙又胜甲的情况, 则必有一人在所有比赛中全胜, 也必有一人在所有比赛中全败.
- 8.10 将 Peterson 图定向, 使其分别成为: (1) 强连通图; (2) 单连通图但不强连通.
- 8.11 设  $\vec{G}$  是有  $\nu$  个顶点  $\varepsilon$  条边的有向简单图, 证明:  
 (1) 若  $\vec{G}$  是弱连通的, 则  $\nu - 1 \leq \varepsilon \leq \nu(\nu - 1)$ ;  
 (2) 若  $\vec{G}$  是强连通的, 则  $\nu \leq \varepsilon \leq \nu(\nu - 1)$ .
- 8.12 设  $\vec{G}$  是一个  $\nu \geq 2$  的有向图. 证明下列命题等价:  
 (1)  $\vec{G}$  是强连通图;  
 (2)  $\vec{G}$  是连通的, 且  $\vec{G}$  的每条弧都在一个有向圈上;  
 (3) 对  $\forall S \subset V(G)$ , 且  $S \neq \phi$ , 都有  $(S, \bar{S}) \neq \phi, (\bar{S}, S) \neq \phi$ .
- 8.13 利用上题结论证明: 对于简单有向图  $\vec{G}$ , 如果对满足  $\langle u, v \rangle \notin A(\vec{G})$  的任何两点  $u, v$ , 都有

$$d^+(u) + d^-(v) \geq \nu - 1,$$

则  $\vec{G}$  是强连通的.

- 8.14 有  $n$  种昆虫, 每两种必有一种能吃掉另一种, 证明可以把这些昆虫排序, 使得每一种昆虫能吃掉它前面的那种昆虫.
- 8.15 证明: 对每个竞赛图, 要么它本身是强连通的, 要么可以只改变一条弧的方向而使其变成强连通的.
- 8.16 证明每个竞赛图都有  $\sum_{v \in V} [d^+(v)]^2 = \sum_{v \in V} [d^-(v)]^2$ .
- 8.17 证明定理 8.4.1: 非平凡弱连通有向图  $\vec{G}$  是 Euler 有向图的充分必要条件是对任何  $v \in V(\vec{G})$ , 都有  $d^+(v) = d^-(v)$ .
- 8.18 证明定理 8.4.2: 非平凡弱连通有向图  $\vec{G}$  是 Euler 有向图的充分必要条件是  $\vec{G}$  可

分解为有向圈的并, 即:  $\vec{G} = \bigcup_{i=1}^k C_i$ , 其中  $C_i$  是  $\vec{G}$  的有向圈, 且  $A(C_i) \cap A(C_j) = \emptyset$ ,  $(1 \leq i, j \leq k)$ ,  $k$  是一个正整数.

8.19 证明定理 8.4.3: 非平凡弱连通有向图  $\vec{G}$  有 Euler 有向迹的充分必要条件是  $\vec{G}$  中存在两个顶点  $u$  和  $w$  满足  $d^+(u) = d^-(u) + 1, d^+(w) = d^-(w) - 1$ , 而其他顶点都有  $d^+(v) = d^-(v)$ .

8.20 证明: 如果有向图  $\vec{G}$  中存在两个顶点  $u$  和  $w$  满足  $d^+(u) = d^-(u) + k, d^+(w) = d^-(w) - k$ , ( $k$  是一个正整数), 而其他顶点都有  $d^+(v) = d^-(v)$ , 则  $\vec{G}$  中从  $u$  到  $w$  有  $k$  条无公共弧的有向路.

8.21 设  $\vec{G}$  是  $\nu$  阶非平凡简单有向图, 证明:

(1) 若对任二满足  $\langle u, v \rangle \notin A(\vec{G})$  的顶点  $u, v$  都有  $d^+(u) + d^-(v) \geq \nu - 1$ , 则  $\vec{G}$  含有 Hamilton 有向路.

(2) 若对  $\vec{G}$  的每个顶点  $v$  都有  $d^+(v) \geq \frac{\nu-1}{2}, d^-(v) \geq \frac{\nu-1}{2}$ , 则  $\vec{G}$  含有 Hamilton 有向路.

8.22 设  $\vec{G}$  是  $\nu$  阶简单有向图, 且对每个顶点  $v$  都有  $d(v) \geq \nu - 1$ , 则  $\vec{G}$  含有 Hamilton 有向路.

8.23 证明每个竞赛图中至多有一个满足  $d^+ = 0$  (或  $d^- = 0$ ) 的顶点, 它是一条有向 Hamilton 路的终点 (或起点).

8.24 证明或举出反例: 对每一个非平凡的竞赛图  $\vec{G}$ , 都有:

(1) 若  $\delta^- = 0$ , 则  $\delta^+ = 0$ ;

(2) 若  $\delta^- = \delta^+ = 0$ , 则  $\vec{G}$  是传递的.

8.25 设  $\vec{G}$  是一个  $\nu$  阶非平凡的正则竞赛图 (各点的出度都相等、各点的入度也都相等), 证明:

(1)  $\nu(\vec{G})$  是奇数, 且  $\vec{G}$  的每个顶点  $v$  都有  $d^+(v) = d^-(v) = \frac{\nu-1}{2}$ ;

(2)  $\vec{G}$  中任二顶点都在一个长为 3 的有向圈上;

(3)  $\vec{G}$  是强连通的.

8.26 设  $\vec{G}$  是一个竞赛图,  $l$  是一个正整数 ( $l \geq 3$ ). 证明: 若  $\vec{G}$  中有长为  $l$  的有向圈, 则必有长为  $3, 4, \dots, l$  的有向圈.

8.27 证明定理 8.5.8: 一个非负整数的不减序列  $d_1, d_2, \dots, d_\nu$  是某个传递竞赛图  $\vec{K}_\nu$  的入度序列当且仅当  $d_i = i - 1, (i = 1, 2, \dots, \nu)$ .

8.28 证明推论 8.5.2 及推论 8.5.3, 即:

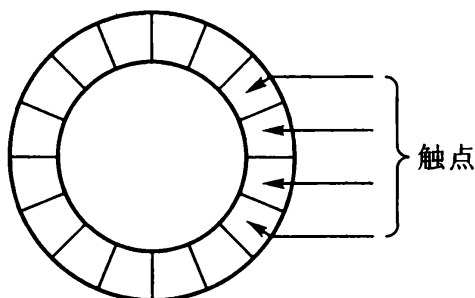
(1) 在同构意义下,  $\nu$  阶传递竞赛图是存在且唯一的;

(2) 对每个正整数  $\nu$ , 恰存在一个无有向圈的  $\nu$  阶竞赛图.

8.29 证明: 任何有向图的强连通分支图都是传递的, 由此证明任何有向图的强连通分

支图都不含有向圈.

- 8.30 (1) 设  $P = v_1 v_2 \cdots v_k$  是竞赛图  $\vec{K}_n$  中的极大有向路. 证明: 若  $P$  不是 Hamilton 有向路, 则对不在  $P$  上的任一顶点  $v$ , 都存在  $i$  使得  $(v_i, v)$  和  $(v, v_{i+1})$  都是  $\vec{K}_n$  中的弧;
- (2) 利用上述结论给出一个求竞赛图中 Hamilton 有向路的多项式时间算法.
- 8.31 证明: (1) 恰含  $t$  片叶子的 2 叉树有  $t-1$  个分支点;
- (2) 任何 2 叉树必有奇数个顶点.
- 8.32 设有  $n$  种工件  $J_1, J_2, \cdots, J_n$  需要在某台机器上加工. 在加工完一种工件, 开始下一种工件之前, 机器必须进行调整. 设机器从加工工件  $J_i$  到加工工件  $J_j$  所需的调整时间为  $t_{ij}$ . 现在需要确定这些工件的加工顺序, 使得所需的调整时间之和最短. 试适当地建立这一问题的图论模型, 以便获得较优解, 并描述求解方法.
- 8.33 计算机旋转鼓轮如下图所示. 一个圆环被等分为  $2^n$  个单元, 每个单元标有 0 或 1 (分别表示该单元为绝缘体或导体). 鼓轮有  $n$  个触点, 当鼓轮每旋转一个单元时,  $n$  个触点便读出一个  $n$  元二进制数.



现在希望设计各个单元的 0、1 标识, 使得鼓轮旋转一周后, 恰好能得到所有  $2^n$  个不同的  $n$  元二进制数. 试建立这一问题的图论模型, 并描述求解方法.

## 参考文献

- [1] L. M. Vitaver, Determination of minimal coloring of vertices of a graph by means of Boolean powers of the incidence matrix (Russian). *Dokl. Akad. Nauk. SSSR*, 147(1962), 758-759.
- [2] B. Roy, Nombre chromatique et plus longs chemins d'un graphe. *Rev. Francaise Automat. Informat. Recherche Operationelle sér. Rouge*. 1(1967), 127-132.
- [3] T. Gallai, On directed paths and circuits. In *Theory of Graphs—Proc. Tihany 1966* (ed. P. Erdos and G. Katona), Academic Press, 1968, 115-118.
- [4] V. Chvatal, L. Lovasz, Every directed graph has a semi-kernel. In *Hypergraph Seminar* (eds. C. Berge and D.K. Ray-Chaudhuri), Springer-Verlag, New York, 1974, p175.

- [5] H. E. Robbins, A theorem on graphs, with an application to a problem of traffic control. *Amer. Math. Monthly*, 46 (1939), 281-283.
- [6] H. Meyniel, Une condition suffisante d'existence d'un circuit Hamiltonien dans un graph orienté. *Journal of Combinatorial Theory (B)*, 14(1973), 137-147.
- [7] J.A. Bondy, C. Thomassen, A short proof of Meyniel's Theorem. *Discrete Mathematics*, 19(1977), 195-197.
- [8] D.B. West, *Introduction to Graph Theory* (second edition), Prentice-Hall, Inc., 2001, p420. (中译本; 图论导引, 李建中、骆吉周译, 北京: 机械工业出版社, 2006, p336).
- [9] D.R. Woodall, Sufficient conditions for circuits in graphs. *Proc. London. Math. Soc.*, 24(1972), 739-755.
- [10] A. Ghouilà-Houri, Une condition suffisante d'existence d'un circuit Hamiltonien. *C. R. Acad. Sci. Paris*, 156(1960), 495-497.
- [11] C.St. J.A. Nash-Williams, Hamiltonian circuits in graphs and digraphs. *The Many Facets of Graph Theory, Lecture Notes in Mathematics 110* (ed. G. Chartrand and S. F. Kapoor), Springer-Verlag, New York, 1969, 237-243.
- [12] L. Rédei, Ein kombinatorischer Satz. *Acta Litt. Sci. Szeged*, 7(1934), 39-43.
- [13] J.W. Moon, On subtournaments of a tournament. *Canad. Math. Bull.*, 9(1966), 297-301.
- [14] P. Camion, Chemins et circuits hamiltoniens des graphes complets. *C. R. Acad. Sci. Paris*, 249(1959), 2151-2152.
- [15] S. Khuller, B. Raghavachari and N. Yong, Approximating the minimum equivalent digraph. *Proceedings of the 5<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms*, 41(4)(1995), 859-872.
- [16] S. Khuller, B. Raghavachari and N. Yong, On Strongly Connected Digraphs with Bounded Cycle Length, *Discrete Applied Mathematics*, 69(3) (1996), 281-289.
- [17] V. Vetta, Approximating the minimum strongly connected subgraph via a matching lower bound, *Proc. 12<sup>th</sup> Annual SODA*, pp417-426, 2001.
- [18] Jørgen Bang-Jensen and Gregory Gutin, *Digraphs: Theory, Algorithms and Applications*, Springer, 2001.
- [19] L. Khachiyan, E. Boros, K. M. Elbassioni, V. Gurvich, On enumerating minimal dicuts and strongly connected subgraphs, *Algorithmica*, 50(1)(2008), 159-172.
- [20] R. Yuster, Decomposing oriented graphs into transitive tournaments. *Discrete Mathematics*, 306(2006), 166-170.
- [21] J. M. Brunat, A. Montes, On polynomial digraphs. *Discrete Mathematics*, 306(2006), 401-412.
- [22] R. P. Stanley, Acyclic orientations of graphs. *Discrete Mathematics*, 306(2006), 905-909.
- [23] V. Borodin, On acyclic colorings of planar graphs. *Discrete Mathematics*, 306(2006), 953-972.

- 
- [24] Romeo Rizzi, Acyclically pushable bipartite permutation digraphs: An algorithm. *Discrete Mathematics*, 306(2006), 1177-1188.
- [25] R. Rizzi, M. Rospocher, Covering partially directed graphs with directed paths. *Discrete Mathematics*, 306(2006), 1390-1404.
- [26] R. Rao, The number of reachable pairs in a digraph. *Discrete Mathematics*, 306(2006), 1595-1600
- [27] M. Zwonek, A note on total colourings of digraphs. *Discrete Mathematics*, 306(2006), 2282-2291.
- [28] G. Gutin, A. Rafiey, S. Severini, A. Yeo, Hamilton cycles in digraphs of unitary matrices. *Discrete Mathematics*, 306(2006), 3315-3320.
- [29] Hao Li, E. Flandrin, Jinlong Shu, A sufficient condition for cyclability in directed graphs. *Discrete Mathematics*, 307(2007), 1291-1297.
- [30] Kim A. S. Factor, Larry J. Langley, Characterization of digraphs with equal domination graphs and underlying graphs. *Discrete Mathematics*, 308(2008), 34-43.
- [31] V. Gurvich, On cyclically orientable graphs. *Discrete Mathematics*, 308(2008), 129-135.
- [32] Hubenko, On a cyclic connectivity property of directed graphs. *Discrete Mathematics*, 308(2008), 1018-1024.

# 第九章 网络流理论与算法

## §9.1 网络与网络流的基本概念

### 一、网络

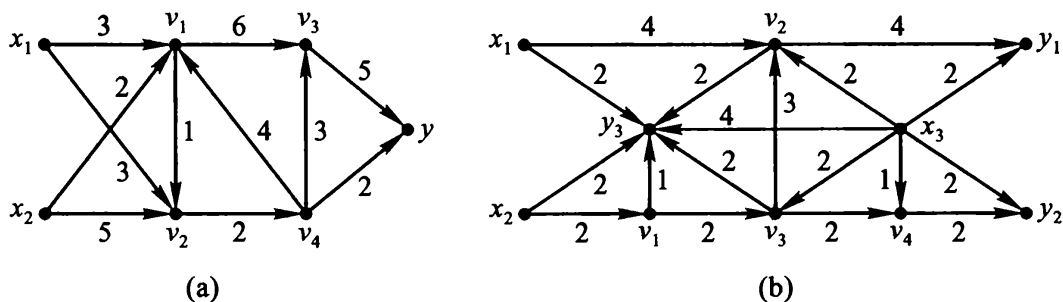
现实应用中经常需要考虑网络及网络上的流, 比如公路货运或客运网络、输电网络、油气管线网络、通信网络等等. 这些网络的共同特点是: 它们都是有发点、收点、中转点的有向图, 每条弧上都有传输能力的限制.

**定义 9.1.1** 一个网络  $N = (V, A)$  是指一个连通无环弧且满足下列条件的有向图:

- (1) 有一个顶点子集  $X$ , 其每个顶点的入度都为 0;
- (2) 有一个与  $X$  不相交的顶点子集  $Y$ , 其每个顶点的出度都为 0;
- (3) 每条弧都有一个非负的权值, 称为弧的容量.

**注** 上述网络  $N$  可记作  $N = (V, X, Y, A, C)$ ,  $X$  称为网络的发点集或源点集,  $Y$  称为网络的收点集或汇点集, 网络中其他顶点称为中转点,  $C$  称为网络的容量函数. 它是定义在弧集  $A$  上的非负函数.

例如, 在下图 (a) 所示的网络中,  $\{x_1, x_2\}$  是源点集,  $\{y\}$  是汇点集, 其他顶点是中转点, 弧上的数字是弧容量; 在图 (b) 所示的网络中,  $\{x_1, x_2, x_3\}$  是源点集,  $\{y_1, y_2, y_3\}$  是汇点集, 其他顶点是中转点, 弧上的数字是弧容量.



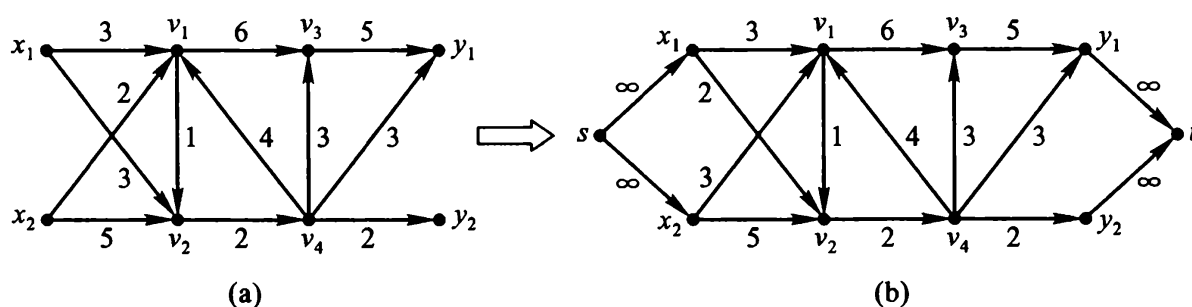
如果一个网络中的源点集和汇点集都只含一个顶点, 则称该网络为单源单汇网络.

任一网络  $N = (V, X, Y, A, C)$  都可导出一个单源单汇网络. 方法如下:

- (1) 给  $N$  添加两个新顶点  $s$  和  $t$ ;
- (2) 对  $\forall x \in X$ , 从  $s$  向  $x$  连一条弧, 其容量为  $\infty$  (或  $\sum_{v \in N^+(x)} c(x, v)$ );
- (3) 对  $\forall y \in Y$ , 从  $y$  向  $t$  连一条弧, 其容量为  $\infty$  (或  $\sum_{u \in N^-(y)} c(u, y)$ ).

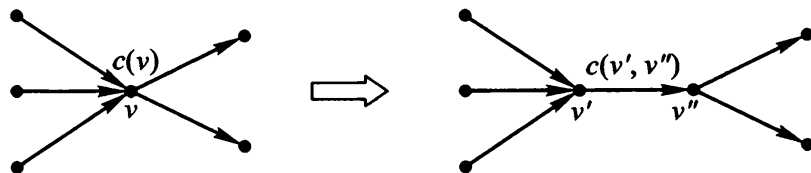
其中  $N^+(x)$  表示顶点  $x$  的出邻点集合  $\{u | (x, u) \in A\}$ ,  $N^-(y)$  表示顶点  $y$  的入邻点集合  $\{u | (u, y) \in A\}$ . 新添的顶点  $s$  和  $t$  分别称为人工源和人工汇.

例如, 对下图 (a) 所示的多源多汇网络, 可按上述方法导出图 (b) 所示的单源单汇网络.



由于多源多汇网络中的流问题均可转化为单源单汇网络中的流问题, 因而此后的讨论中主要考虑单源单汇网络.

一些实际应用中需要考虑弧和顶点都有容量限制的网络. 比如在运输网络中, 中转点的中转能力可能会有限制. 这样的网络可转化为顶点无容量限制的网络, 方法为: 将每个中转点  $v$  分裂为两个点  $v', v''$ , 将原来顶点  $v$  的入弧都连向顶点  $v'$ , 将原来顶点  $v$  的出弧都连给顶点  $v''$ , 同时从  $v'$  向  $v''$  连一条弧, 弧容量为原来顶点  $v$  的容量 (如下图所示).



## 二、网络的流与割

**定义 9.1.2** 网络  $N = (V, X, Y, A, C)$  中的一个 (可行) 流是指定义在  $A$  上的一个整值函数  $f$ , 使得:

- (1) 对  $\forall a \in A, 0 \leq f(a) \leq c(a)$ , (容量约束);
- (2) 对  $\forall v \in V - (X \cup Y), f^-(v) = f^+(v)$ , (流量守恒).

其中  $f^-(v)$  表示点  $v$  处入弧上的流量之和,  $f^+(v)$  表示点  $v$  处出弧上的流量之和.

**注** (1) 可行流总是存在的, 比如  $\forall a \in A, f(a) = 0$ , 便是任一网络中的一个可行流, 这个流称为**零值流**.

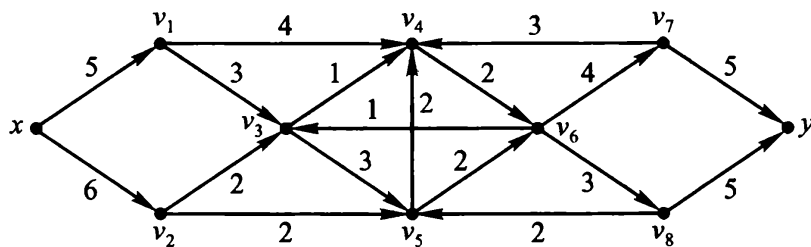
(2) 对网络  $N$  中任一可行流  $f$  和任一顶点子集  $S, f^+(S)$  表示从  $S$  中流出的流量, 即从  $S$  中顶点指向  $S$  外顶点的弧上的流量之和,  $f^-(S)$  表示流入  $S$  的流量, 即从  $S$  外的顶点指向  $S$  的弧上流量之和.

**定义 9.1.3** 设  $f$  是网络  $N = (V, X, Y, A, C)$  中的一个可行流, 则必有  $f^+(X) = f^-(Y)$ .  $f^+(X)$  (或  $f^-(Y)$ ) 称为流  $f$  的**流量**, 记为  $\text{Val } f$ .  $N$  中流量最大的可行流称为  $N$  的**最大流**.

**注** 流量为零的流称为**零流**. 零流未必是零值流.

**定义 9.1.4** 设  $N = (V, x, y, A, C)$  是一个单源单汇网络.  $S \subseteq V, \bar{S} = V - S$ . 用  $(S, \bar{S})$  表示尾在  $S$  中而头在  $\bar{S}$  中的所有弧的集合 (即从  $S$  中的点指向  $S$  之外点的所有弧之集). 若  $x \in S$ , 而  $y \in \bar{S}$ , 则称弧集  $(S, \bar{S})$  为网络  $N$  的一个**割**. 一个割  $(S, \bar{S})$  的**容量**是指  $(S, \bar{S})$  中各条弧的容量之和, 记为  $\text{Cap}(S, \bar{S})$ .

例如, 对下图所示的网络, 令  $S = \{x, v_1, v_2, v_3, v_5\}$ , 则割  $(S, \bar{S}) = \{v_1v_4, v_3v_4, v_5v_4, v_5v_6\}$ , 割容量  $\text{Cap}(S, \bar{S}) = 9$ .



**注** 一个网络  $N$  可能有多个割, 各个割的容量不一定相等, 其中容量最小的割称为  $N$  的**最小割**. 最小割的确切定义如下.

**定义 9.1.5** 网络  $N$  的一个割  $K$  称为**最小割**, 如果网络  $N$  中不存在割  $K'$  使得  $\text{Cap } K' < \text{Cap } K$ .

**引理 9.1.1** 对网络  $N$  中任一流  $f$  和任一割  $(S, \bar{S})$ , 均有  $\text{Val } f = f^+(S) - f^-(S)$ .

**证明** 设  $f$  是网络  $N = (V, x, y, A, C)$  中的流,  $(S, \bar{S})$  是  $N$  的一个割, 由流的定义,

$$f^+(x) = \text{Val } f, f^-(x) = 0, f^+(v) - f^-(v) = 0, (\forall v \in S - \{x\}).$$



故

$$\begin{aligned} \text{Val} f &= f^+(x) - f^-(x) + \sum_{v \in S - \{x\}} [f^+(v) - f^-(v)] = \sum_{v \in S} [f^+(v) - f^-(v)] \\ &= \sum_{v \in S} \left[ \sum_{u \in V} f(v, u) - \sum_{u \in V} f(u, v) \right] = \sum_{v \in S} \sum_{u \in V} f(v, u) - \sum_{v \in S} \sum_{u \in V} f(u, v) \\ &= \sum_{v \in S} \sum_{u \in S} f(v, u) + \sum_{v \in S} \sum_{u \notin S} f(v, u) - \sum_{v \in S} \sum_{u \in S} f(u, v) - \sum_{v \in S} \sum_{u \notin S} f(u, v) \end{aligned}$$

上述推导中, 若点  $u$  与  $v$  之间无弧, 则视为  $f(u, v) = 0$ . 最后一行中第一式和第三式都表示  $S$  内部弧上流量之和, 故相等; 第二式表示从  $S$  的顶点指向  $S$  之外点的所有弧上流量之和, 因此它等于  $f^+(S)$ ; 第四式表示从  $S$  之外的顶点指向  $S$  中点的所有弧上的流量之和, 因此等于  $f^-(S)$ . 从而  $\text{Val} f = f^+(S) - f^-(S)$ . 证毕.

**定义 9.1.6** 设  $N = (V, x, y, A, C)$  是一个网络,  $f$  是  $N$  的一个可行流. 对  $N$  中任一条弧  $a$ ,

若  $f(a) = 0$  则称  $a$  是  $f$  零的;

若  $f(a) > 0$  则称  $a$  是  $f$  正的;

若  $f(a) = c(a)$ , 则称  $a$  是  $f$  饱和的;

若  $f(a) < c(a)$  则称  $a$  是  $f$  非饱和的.

**定理 9.1.1** 对网络  $N$  中任一可行流  $f$  和任一割  $K = (S, \bar{S})$ , 均有  $\text{Val} f \leq \text{Cap} K$ . 其中等式成立当且仅当  $(S, \bar{S})$  中每条弧都是  $f$  饱和的而  $(\bar{S}, S)$  中每条弧都是  $f$  零的.

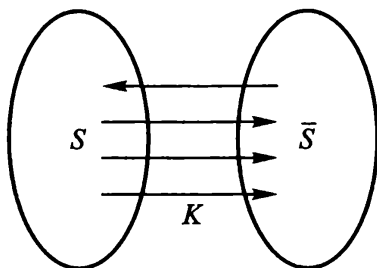
**证明** 因  $f$  是可行流, 故

$$f^+(S) = \sum_{a \in K} f(a) \leq \sum_{a \in K} c(a) = \text{Cap} K, \text{ 并且 } f^-(S) = \sum_{a \in (\bar{S}, S)} f(a) \geq 0.$$

由引理 9.1.1,  $\text{Val} f = f^+(S) - f^-(S) \leq \text{Cap} K$ , 可见第一个结论成立.

当  $(S, \bar{S})$  中每条弧都是  $f$  饱和弧, 而  $(\bar{S}, S)$  中每条弧都是  $f$  零流弧时,  $f^+(S) = \text{Cap} K$  且  $f^-(S) = 0$ , 故

$$\text{Val} f = f^+(S) - f^-(S) = f^+(S) = \text{Cap} K;$$



反之, 若  $\text{Val}f = \text{Cap}K$ , 则由于  $f^+(S) \leq \text{Cap}K$ , 故

$$\text{Cap}K = \text{Val}f = f^+(S) - f^-(S) \leq \text{Cap}K - f^-(S).$$

但  $f^-(S) \geq 0$ , 因此必定  $f^-(S) = 0$ , 且  $f^+(S) = \text{Cap}K$ . 证毕.

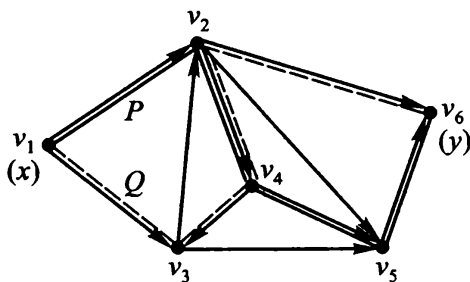
**推论 9.1.1** 设  $f$  是网络  $N$  的一个可行流,  $K$  是  $N$  的一个割, 若  $\text{Val}f = \text{Cap}K$ , 则  $f$  是最大流而  $K$  是最小割.

**证明** 设网络  $N$  中的最大流为  $f^*$ , 最小割为  $K^*$ , 则由定理 9.1.1,  $\text{Val}f^* \leq \text{Cap}K^*$ , 从而  $\text{Val}f \leq \text{Val}f^* \leq \text{Cap}K^* \leq \text{Cap}K$ . 由已知条件,  $\text{Val}f = \text{Cap}K$ , 故由上式知,  $\text{Val}f = \text{Val}f^*$ ,  $\text{Cap}K^* = \text{Cap}K$ . 证毕.

**定义 9.1.7** 设  $u, v$  是网络  $N = (V, x, y, A, C)$  中任意两点,  $P$  是  $N$  的底图中一条连接  $u$  与  $v$  的 (无向) 路, 若规定路  $P$  的走向为从  $u$  到  $v$ , 则称这样规定了走向的路  $P$  为网络  $N$  中一条从  $u$  到  $v$  的路, 简称为  $u-v$  路. 特别地, 一条从源  $x$  到汇  $y$  的路称为一条  $x-y$  路.

**定义 9.1.8** 设  $P = uv_1 \cdots v_kv$  是网络  $N = (V, x, y, A, C)$  中一条  $u-v$  路, 若弧  $\langle v_i, v_{i+1} \rangle \in A$ , 则称此弧为  $u-v$  路  $P$  的一条正向弧 (或称前向弧, 顺向弧), 若弧  $\langle v_{i+1}, v_i \rangle \in A$ , 则称此弧为  $u-v$  路  $P$  的一条反向弧 (或称后向弧, 逆向弧). 将  $u-v$  路  $P$  所经过的弧 (无论正向弧或反向弧) 称为路  $P$  上的弧.

例如, 在下图所示的  $x-y$  路  $P = xv_2v_4v_5y$  上, 所有弧都是正向弧; 而在  $x-y$  路  $Q = xv_3v_4v_2y$  上, 弧  $\langle v_1, v_3 \rangle$  和  $\langle v_2, v_6 \rangle$  是正向弧, 而  $\langle v_2, v_4 \rangle$  和  $\langle v_4, v_3 \rangle$  是反向弧. 可见, 一条弧是正向弧还是反向弧与路的选择有关.



**定义 9.1.9** 设  $f$  是网络  $N = (V, x, y, A, C)$  中的一个可行流,  $u$  是  $N$  中任意一点,  $P$  是  $N$  中一条  $x-u$  路. 如果对路  $P$  上的任一条弧  $a$ , 都有

- (1) 若弧  $a$  是  $P$  的正向弧, 则  $c(a) - f(a) > 0$ ;
- (2) 若弧  $a$  是  $P$  的反向弧, 则  $f(a) > 0$ .

则称  $P$  是  $N$  中一条  $f$  可增  $x-u$  路. 特别地,  $N$  中一条  $f$  可增  $x-y$  路简称为  $N$  的一条  $f$  可增路.

对  $N$  中任一条  $f$  可增路  $P$  和  $P$  上任一条弧  $a$ , 令

$$\Delta f(a) = \begin{cases} c(a) - f(a), & \text{若 } a \text{ 是 } P \text{ 的正向弧,} \\ f(a), & \text{若 } a \text{ 是 } P \text{ 的反向弧,} \end{cases}$$

则沿路  $P$  可增加的流量为  $\Delta f(P) = \min_{a \in P} \Delta f(a)$ , 该值称为  $f$  可增路  $P$  上流的增量或可增量.

例如, 在下图所示的网络中, 每条弧上括号内的数字表示弧的容量, 括号外的数字是当前流  $f$  在弧上的流值. 图 1 中虚线所示的  $x$ - $y$  路  $P$  是一条  $f$  可增路. 因  $\Delta f(v_1, v_3) = 6 - 1 = 5$ ,  $\Delta f(v_3, v_4) = 2$ ,  $\Delta f(v_4, v_2) = 5$ ,  $\Delta f(v_2, v_6) = 4 - 0 = 4$ , 故可增路  $P$  上流  $f$  的可增量  $\Delta f(P) = \min\{5, 2, 5, 4\} = 2$ . 沿  $P$  增流后的流网络如图 2 所示. 网络的流量从 5 增加至 7.

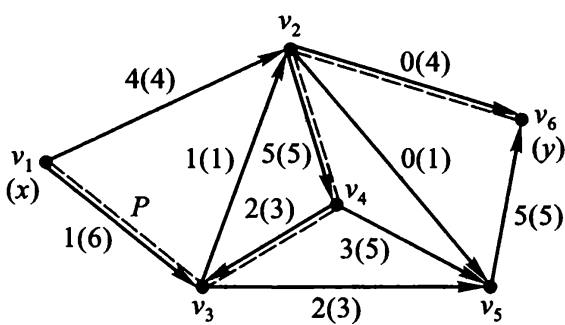


图 1

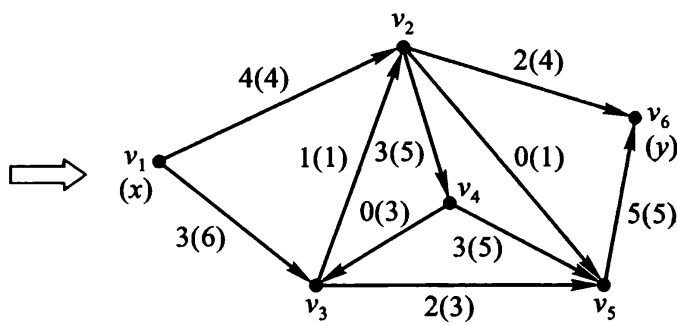


图 2

**定理 9.1.2 (最大流最小割定理, L. R. Ford, D. R. Fulkerson, 1956)** <sup>[1]</sup> 任一个网络  $N = (V, x, y, A, C)$  中, 最大流的流量等于最小割的容量.

**证明** 设  $f^*$  是网络  $N$  中的最大流,  $K^*$  是最小割, 则由定理 9.1.1,  $\text{Val} f^* \leq \text{Cap} K^*$ . 另一方面, 令

$$S = \{u \in V \mid N \text{ 中有 } f^* \text{ 可增 } x\text{-}u \text{ 路}\} \cup \{x\},$$

则汇  $y \notin S$  (否则可沿  $x$  到  $y$  的可增路增流, 得到更大的流), 而源  $x \in S$ , 从而  $K = (S, \bar{S})$  是  $N$  中的一个割.

对  $(S, \bar{S})$  中的任一条弧  $a = \langle u, v \rangle$ , 必定  $f^*(a) = c(a)$  (否则, 若  $f^*(a) < c(a)$ , 因  $u \in S$ ,  $N$  中有  $f^*$  可增  $x$ - $u$  路  $P$ , 从而  $P + a$  是  $N$  的一条  $f^*$  可增  $x$ - $v$  路, 因此  $v \in S$ , 矛盾); 同理, 对于  $(\bar{S}, S)$  中任一条弧  $a = \langle v, u \rangle$ , 必定  $f^*(a) = 0$ . 这说明  $(S, \bar{S})$  中每条弧都是  $f^*$  饱和的而  $(\bar{S}, S)$  中每条弧都是  $f^*$  零的. 由定理 9.1.1,  $\text{Val} f^* = \text{Cap}(S, \bar{S}) \geq \text{Cap} K^*$ .

由以上两方面, 便知  $\text{Val} f^* = \text{Cap} K^*$ . 证毕.

由最大流最小割定理易知下列推论成立.

**推论 9.1.2 (整值最大流定理)** 任一网络中, 若所有弧的容量都是整数, 则最大流的流量也必为整数.

## §9.2 最大流问题及其标号算法

**最大流问题:** 给定网络  $N = (V, x, y, A, C)$ , 如何求  $N$  中的最大流?

由可增路的概念, 对网络  $N$  中一个可行流  $f$ , 如果能找到  $N$  中一条  $f$  可增  $x$ - $y$  路  $P$ , 则可沿着  $P$  修改流的值, 得到一个流量更大的可行流  $\hat{f}$ . 修改办法如下:

$$\hat{f}(a) = \begin{cases} f(a) + \Delta f(P), & \text{若 } a \text{ 是 } P \text{ 的正向弧,} \\ f(a) - \Delta f(P), & \text{若 } a \text{ 是 } P \text{ 的反向弧,} \\ f(a), & a \text{ 不在 } P \text{ 上.} \end{cases} \quad (*)$$

修改后流的流量为:  $\text{Val } \hat{f} = \text{Val } f + \Delta f(P)$ .

这给出了求网络  $N$  的最大流的一个途径: 反复找  $N$  中的可增路, 沿可增路将流量扩大, 直到找不出可增路为止. 直观上想, 此时应该已达到最大流. 下面的定理证实了这一点.

**定理 9.2.1** 网络  $N(V, x, y, A, C)$  中的可行流  $f$  是  $N$  的最大流当且仅当  $N$  中不存在  $f$  可增路.

**证明 必要性** 若  $N$  有  $f$  可增路  $P$ , 则  $f$  不是最大流, 因为沿  $P$  按 (\*) 式修改流可得流量更大的流  $\hat{f}$ .

**充分性** 设  $N$  中不存在  $f$  可增路, 我们来证明  $f$  是最大流. 令

$$S = \{v \in V \mid \text{从源 } x \text{ 到 } v \text{ 有 } f \text{ 可增路}\} \cup \{x\},$$

则与最大流最小割定理同样可证  $K = (S, \bar{S})$  是  $N$  中的一个割, 且  $\text{Val } f = \text{Cap } K$ . 再由推论 9.1.1 便知  $f$  是最大流 (而且当前的  $K$  是最小割). 证毕.

由此我们得到求网络  $N$  中最大流的一种思想: 从一个已知流 (例如零值流) 开始, 递推地构作流量严格增加的流序列. 在每一个新的流  $f$  得出之后, 若能找到一条  $f$  可增路  $P$ , 则沿  $P$  修改流的值, 得到一个更大的流  $\hat{f}$ , 作为这个序列中的下一个流; 若不存在  $\hat{f}$  可增路, 则由定理 9.2.1,  $f$  就是最大流, 算法终止.

**问题** 对网络  $N$  中的一个可行流  $f$ , 如何找  $f$  可增路或判断  $f$  可增路不存在?

解决这个问题的一种方法是 Ford-Fulkerson 标号法, 标号过程描述如下.

设网络  $N = (V, x, y, A, C)$  中当前可行流为  $f$ . 从源点  $x$  开始. 首先给  $x$  标上  $\infty$ , 即  $l(x) = \infty$  ( $x$  称为已标未查顶点, 其他顶点称为未标未查顶点).

任选一已标未查顶点  $u$ , 检查其所有尚未标号的邻点:

(1) 对  $u$  的尚未标号的出邻点  $v$  (即  $\langle u, v \rangle \in A$ ), 若  $c(u, v) > f(u, v)$ , 则给  $v$  标号:

$$l(v) = \min \{l(u), c(u, v) - f(u, v)\} \quad (v \text{ 称为已标未查顶点});$$

否则, 不给  $v$  标号.

(2) 对  $u$  的尚未标号的入邻点  $v$  (即  $\langle v, u \rangle \in A$ ), 若  $f(v, u) > 0$ , 则给  $v$  标号:

$$l(v) = \min \{l(u), f(v, u)\}, \quad (v \text{ 称为已标未查顶点});$$

否则, 不给  $v$  标号.

当检查完  $u$  的所有邻点之后,  $u$  称为已标已查顶点.

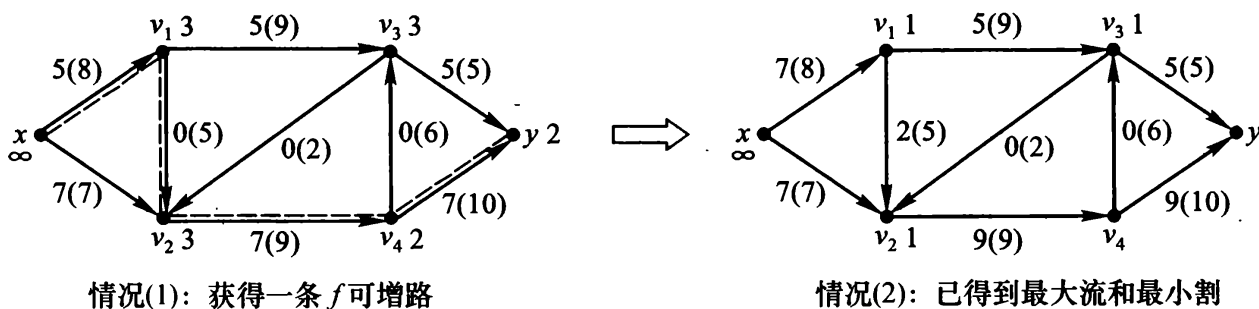
反复进行上述标、查过程, 最终必出现两种结果之一:

(i) 汇点  $y$  获得标号, 此时已得到  $f$  可增路. 可沿该路按 (\*) 式增流.

(ii)  $y$  点未获得标号, 但已没有已标未查顶点 (所有已标顶点都已被查, 没有更多的新标号顶点).

此时当前的流  $f$  即为最大流 (设当前已标已查的顶点之集为  $S$ , 则  $(S, \bar{S})$  为最小割).

例如, 下图情况 (1) 显示了一个网络  $N(V, x, y, A, C)$ , 以及在该网络当前的可行流  $f$  下利用标号方法得到的可增路. 其中每条弧上括号内数字是弧容量, 括号外的数字是弧上的流值, 顶点旁的数字是顶点获得的标号, 一条可增  $x$ - $y$  路是  $xv_1v_2v_4y$ . 情况 (2) 显示沿上述可增路增流后的流网络, 以及新一轮的顶点标号. 在新一轮标号下, 对  $x, v_1, v_2, v_3$  进行“标”、“查”过程后, 已没有“已标未查点”, 但此时汇点  $y$  尚未获得标号, 因而此时不再有可增路, 当前的网络流已经是最大流了. 同时还获得最小割  $(S, \bar{S}) = \{v_3y, v_2v_4\}$ , 其中  $S = \{x, v_1, v_2, v_3\}$ .



### 求网络最大流的 Ford-Fulkerson 标号算法<sup>[2]</sup>

输入: 网络  $N(V, x, y, A, C)$ .

输出: 网络  $N$  中一个最大流.

第 0 步 (初始化); 对  $\forall a \in A$ , 令  $f(a) = 0$ :

第 1 步: 给源点  $x$  标号  $(x, \infty)$ .  $L := \{x\}$ ,  $S = \emptyset$ . 其中  $L$  表示已标未查集,  $S$  表示已标已查集.

第 2 步: 任取  $u \in L$ , 检查  $u$  的每个尚未标号的邻点  $v$ :

(1) 若  $v \in N^+(u)$ ,  $v$  尚未标号且  $c(u, v) > f(u, v)$ , 则给  $v$  标号  $(u, +, l(v))$ , 其中  $l(v) = \min\{l(u), c(u, v) - f(u, v)\}$ . 令  $L := L \cup \{v\}$ .

(2) 若  $v \in N^-(u)$ ,  $v$  尚未标号且  $f(v, u) > 0$ , 则给  $v$  标号  $(u, -, l(v))$ , 其中  $l(v) = \min\{l(u), f(v, u)\}$ . 令  $L := L \cup \{v\}$ .

第 3 步: 令  $L := L - \{u\}$ ,  $S := S \cup \{u\}$ . 若汇点  $y$  已被标号, 则转第 5 步; 否则转下步.

第 4 步: 若  $L = \emptyset$ , 则算法结束, 当前流是最大流; 否则转第 2 步.

第 5 步: 令  $z = y$ .

第 6 步: 若  $z$  的标号为  $(w, +, l(z))$ , 则令  $f(w, z) = f(w, z) + l(y)$ ; 若  $z$  的标号为  $(w, -, l(z))$ , 则令  $f(z, w) = f(z, w) - l(y)$ . 转下步.

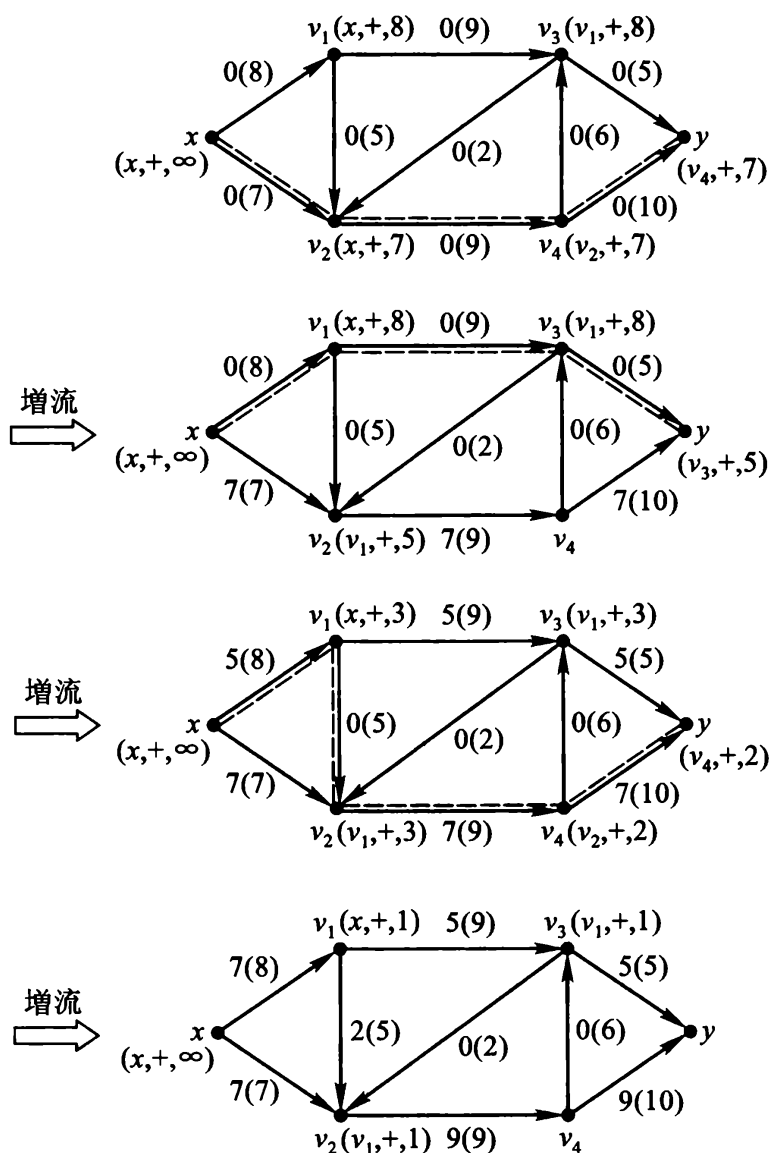
注 不论  $z$  是哪个点, 上两式中的减项总是  $l(y)$ , 因  $l(y)$  是沿可增路的最大的可增量.

第 7 步: 若  $w \neq x$ , 则令  $z = w$ , 转第 6 步; 否则, 去掉所有的顶点标号, 转第 1 步.

算法中第 1 步 ~ 第 4 步是找可增路的反复标、查过程, 第 5 步 ~ 第 7 步是反向追踪找到的可增路并逐条弧修改流的循环过程.

下图演示了对一个网络从零值流开始, 利用 Ford-Fulkerson 标号算法进行 3 次增流, 得到最大流的过程. 在每条弧上, 括号里的数字表示容量, 括号外的数字表示流值. 每个顶点的三元标号中, 第三个符号是一个数字, 表示该顶点的标号值; 第一个符号表示当前该点的标号值是通过哪个点 (称为标号的前一点) 获得的; 第二个符号的正或负表示标号的前一点到当前点是通过正向弧还是反向弧连接的. 第一个符号用于最后反向追踪可增路, 第二个符号标识在增流时是在弧的原有基础上加还是减流值, 标号值表示从源点  $x$  到该点通过当前找到的可增路可以增加的流值.

在第一个图中, 标号结束后汇点  $y$  获得了标号  $(v_4, +, 7)$ , 根据标号的第一项逐步反向追踪获得可增路  $xv_2v_4y$ , 由第三项的值可知沿该可增路可以增加 7 个单位的流量, 增流后网络成为第二个图. 在第二个图中, 标号结束后汇点  $y$  获得了标号  $(v_3, +, 5)$ , 根据标号的第一项逐步反向追踪获得可增路  $xv_1v_3y$ , 由第三项的值可知沿该可增路可以增加 5 个单位的流量, 增流后网络成为第三个图. 注意第二个图中顶点  $v_4$  未获得标号, 这是因为在给  $v_4$  标号之前汇点  $y$  已获得标号, 可增路已找到, 没有必要继续给  $v_4$  标号了. 在第四个图中, 对  $x, v_1, v_2, v_3$  进行“标”、“查”过程后, 已没有“已标未查点”, 而汇点  $y$  尚未获得标号, 因而此时不再有可增路, 当前的网络流已经是最大流了. 算法结束时还获得最小割  $(S, \bar{S}) = \{v_3y, v_2v_4\}$ , 其中  $S = \{x, v_1, v_2, v_3\}$ .



**定理 9.2.2** 标号算法结束时所得的流是最大流.

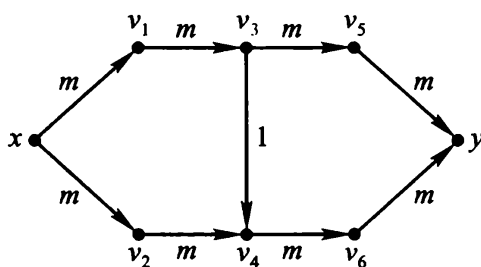
**证明** 算法结束时,  $L = \emptyset$ , 所有已标顶点都已查,  $S$  是所有已标已查顶点的集合.  $\bar{S} \neq \emptyset$  (因  $y \in \bar{S}$ ). 按照标号规则, 对  $\forall a \in (S, \bar{S}), f(a) = c(a)$ , 且对  $\forall a \in (\bar{S}, S), f(a) = 0$ , (否则可得到新的标号顶点, 算法不会终止). 从而根据引理 9.1.1,

$$\text{Val } f = \sum_{a \in (S, \bar{S})} f(a) - \sum_{a \in (\bar{S}, S)} f(a) = \sum_{a \in (S, \bar{S})} f(a) = \sum_{a \in (S, \bar{S})} c(a) = \text{Cap}(S, \bar{S}).$$

由推论 9.1.1,  $f$  是最大流, 而  $(S, \bar{S})$  是最小割. 证毕.

关于标号算法的计算复杂度, 我们先来看一个例子. 例如, 对下列网络运用 Ford-Fulkerson 标号法求最大流时, 有可能反复交替地沿路  $xv_1v_3v_4v_6y$  和路  $xv_2v_4v_3v_5y$  增流, 每次只能增 1 个单位流量, 需要  $2m$  次找路和增流过程才能得到最大流. 可见, 标号算法的计算量并不完全依赖于问题的规模——点数  $\nu$  和弧数  $\varepsilon$ , 还依赖于弧容量. 我们将计算量虽然是问题规模的多项式函数但又依赖于其他参量的算法称为伪多项

式算法. Ford-Fulkerson 标号算法是伪多项式算法.



实际上, 设网络中最大的弧容量为  $c_{\max}$ . 因网络源点的出弧不超过  $\nu - 1$ , 故网络的最大流的流量至多为  $\nu - 1 \cdot c_{\max}$ . 运用 Ford-Fulkerson 标号算法每次沿可增路增流时, 至少增加一个单位的流量, 因此, 找可增路增流的次数不超过  $\nu - 1 \cdot c_{\max}$  次. 而用标号算法寻找可增路并沿可增路增流的计算量为  $O(\varepsilon)$ . 从而 Ford-Fulkerson 标号算法的计算复杂度为  $O(\varepsilon \nu c_{\max})$ , 它是一个伪多项式时间算法而不是多项式时间算法. 当最大弧容量  $c_{\max}$  为指数次幂时, 算法的计算量是指数阶的. 更为严重的是, L. R. Ford 和 D. R. Fulkerson 构造出一个例子, 说明当弧容量为无理数时, 标号算法不能在有限步终止, 并且计算过程中得到的流序列不收敛于最大流<sup>[3][4][5]</sup>. 当然, 这种情况在现实的大规模计算中是不会发生的, 因为电子计算机在计算过程中处理的数据都是有理数.

幸运的是, J. Edmonds 和 R. M. Karp 发现, 对标号算法略做修改就可以将其改造为计算复杂度为  $O(\nu \varepsilon^2)$  的多项式时间算法<sup>[6][7]</sup>. 他们提出的修改策略是: 在 Ford-Fulkerson 标号算法的第 2 步检查已标顶点时, 采用“先标先查”规则, 其余步骤不变. 也就是说, 在选择已标未查点进行检查时, 按获得标号的先后次序从  $L$  集合中选择顶点, 检查其所有尚未标号的邻点. 这种先标先查规则, 相当于按广度优先算法 (广探法) 搜索已标未查点, 实质上是求  $x$  到  $y$  的最短可增路进行增流. E. A. Dinic 提出了另一种算法<sup>[8]</sup>, 通过引入增量网络的概念并对增量网络进行分层, 来实现沿最短可增路增流的思想. 这种方法将在下一节介绍.

### §9.3 求最大流的 Dinic 算法

本节介绍一种求网络最大流的最短可增路算法. 这种算法基于 Dinic 提出的对增量网络进行分层的思想.

#### 一、增量网络

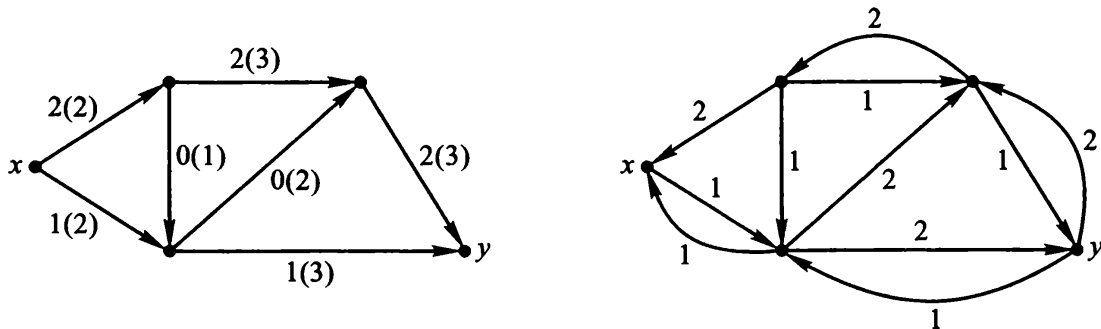
**定义 9.3.1** 对于网络  $N = (V, x, y, A, C)$  和  $N$  上的一个可行流  $f$ , 构造一个新的网络  $N(f) = (V, x, y, A(f), C')$ , 其中  $A(f)$  及容量函数  $C'$  定义如下:

- (1) 若  $\langle u, v \rangle \in A$  且  $f(u, v) < c(u, v)$ , 则  $\langle u, v \rangle \in A(f)$ , 且  $c'(u, v) = c(u, v) - f(u, v)$ ;
- (2) 若  $\langle u, v \rangle \in A$  且  $f(u, v) > 0$ , 则  $\langle v, u \rangle \in A(f)$ , 且  $c'(v, u) = f(u, v)$ .

这样构造的网络  $N(f)$  称为网络  $N$  关于流  $f$  的增量网络.



例如, 下列左图显示了一个网络, 以及该网络的一个可行流  $f$ . 右图显示该网络关于流  $f$  构成的增量网络  $N(f)$ .



网络  $N$  及其一个可行流  $f$ , 弧上括号外数字为流量, 括号内数字为容量.

增量网络  $N(f)$

注 (1) 对应于  $N$  的每条饱和弧  $\langle u, v \rangle$ ,  $N(f)$  中只有一条弧  $\langle v, u \rangle$ , 且

$$c'(v, u) = c(u, v) = f(u, v);$$

对应于  $N$  的每条零流弧  $\langle u, v \rangle$ ,  $N(f)$  中只有一条弧  $\langle u, v \rangle$ , 且

$$c'(u, v) = c(u, v) = c(u, v) - f(u, v);$$

对应于  $N$  的每条非零流非饱和弧  $\langle u, v \rangle$ ,  $N(f)$  中有两条弧  $\langle u, v \rangle$  和  $\langle v, u \rangle$ , 且

$$c'(u, v) = c(u, v) - f(u, v), \quad c'(v, u) = f(u, v).$$

(2)  $N(f)$  中每条弧的容量恰为  $N$  中相应弧的流可增量.

(3) 严格来讲, 增量网络不符合前述网络的定义 (无出度为 0 的点 (源) 和入度为 0 的点 (汇)), 但因它是在原网络  $N$  基础上定义的, 有明显的源和汇的痕迹, 因此我们仍将其看作网络, 其中  $x$  和  $y$  分别为源和汇, 该网络上的一个可行流  $f$  定义为满足

$$\begin{cases} f(x) = f^+(x) - f^-(x) \geq 0, \\ f(y) = f^+(y) - f^-(y) \leq 0, \\ f(u) = f^+(u) - f^-(u) = 0 \end{cases}$$

的流.

(4) 将  $N(f)$  中从  $x$  到  $y$  的有向路称为增量网络  $N(f)$  的  $x$ - $y$  有向路. 明显地,  $N(f)$  中的  $x$ - $y$  有向路自然地对应出  $N$  中的  $x$ - $y$  路, 且构成  $N$  的  $f$  可增路. 因此, 寻找  $N$  中  $f$  可增路的问题可转化为在增量网络  $N(f)$  中找  $x$ - $y$  有向路.

按照增量网络的概念及以上分析, 下面一个定理显然成立.

**定理 9.3.1** 设  $f$  是网络  $N$  中的一个可行流. 则

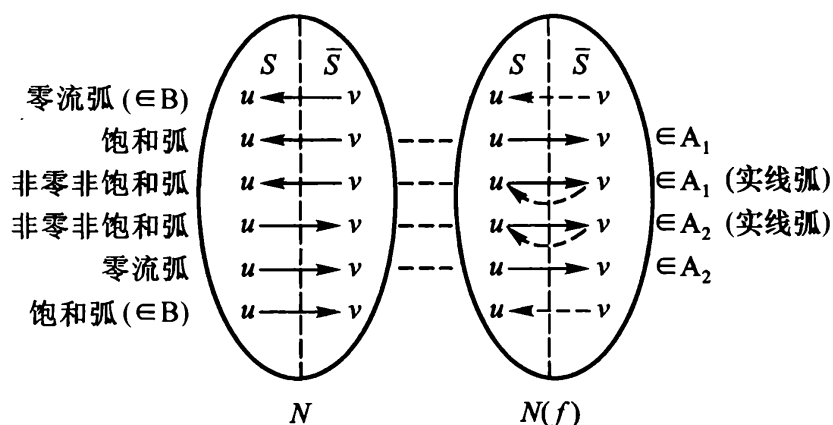
(1)  $N$  中  $f$  可增路与  $N(f)$  中的  $x$ - $y$  有向路一一对应.

(2)  $N$  中  $f$  可增路  $P$  的可增流值  $\Delta f(P)$  等于  $N(f)$  中对应的  $x$ - $y$  有向路  $\vec{P}$  上最小的弧容量.

证明留作习题 (习题 9.6).

**定理 9.3.2** 设  $f^*$  是网络  $N$  中的最大流,  $f$  是  $N$  中的一个可行流, 则增量网络  $N(f)$  中的最大流的流量为  $\text{Val}f^* - \text{Val}f$ .

**证明** 考虑  $N(f)$  中的任一个割  $(S, \bar{S})$ , 对  $\forall \langle u, v \rangle \in (S, \bar{S})$ , 在  $N(f)$  中弧  $\langle u, v \rangle$  的容量  $c'(u, v)$  只有两种可能: 要么  $c'(u, v) = f(v, u)$ , 要么  $c'(u, v) = c(u, v) - f(u, v)$ , [见前述注 (1)], 其中  $c(u, v)$  和  $f(u, v)$  分别表示在  $N$  中弧  $\langle u, v \rangle$  上的容量和流值. 用  $A_1$  表示  $N(f)$  中  $(S, \bar{S})$  间前一种弧的集合,  $A_2$  表示后一种弧的集合, 则  $A_1 \cap A_2 = \emptyset$  且  $A_1 \cup A_2 = (S, \bar{S})$ , 注意  $N(f)$  中的割  $(S, \bar{S})$  自然对应出  $N$  中顶点集的一个划分, 形成  $N$  中的割  $(S, \bar{S})$ , 它由  $N(f)$  中  $A_2$  对应的弧及  $N$  中从  $S$  到  $\bar{S}$  的所有饱和弧组成. 我们将  $N$  中从  $S$  到  $\bar{S}$  的所有饱和弧的集合记为  $B$ , 另将  $N$  中从  $\bar{S}$  到  $S$  的所有零流弧的集合记为  $L$ . 则对  $\forall \langle u, v \rangle \in B, f(u, v) = c(u, v)$ , 对  $\forall \langle v, u \rangle \in L, f(v, u) = 0$ .  $N(f)$  中  $S$  与  $\bar{S}$  间的弧与  $N$  中  $S$  与  $\bar{S}$  间的弧的对应如下图所示.



设  $\text{Cap}'(S, \bar{S})$  和  $\text{Cap}(S, \bar{S})$  分别为割  $(S, \bar{S})$  在  $N(f)$  和  $N$  中的割容量, 则

$$\begin{aligned}
 \text{Cap}'(S, \bar{S}) &= \sum_{\langle u, v \rangle \in (S, \bar{S})} c'(u, v) = \sum_{\langle u, v \rangle \in A_1} c'(u, v) + \sum_{\langle u, v \rangle \in A_2} c'(u, v) \\
 &= \sum_{\langle u, v \rangle \in A_1} f(v, u) + \sum_{\langle u, v \rangle \in A_2} (c(u, v) - f(u, v)) \\
 &= \sum_{\langle u, v \rangle \in A_2} c(u, v) + \sum_{\langle u, v \rangle \in A_1} f(v, u) - \sum_{\langle u, v \rangle \in A_2} f(u, v) \\
 &= \sum_{\langle u, v \rangle \in A_2} C(u, v) + \sum_{\langle u, v \rangle \in B} C(u, v) + \sum_{\langle u, v \rangle \in A_1 \cup L} f(u, v) - \\
 &\quad \left( \sum_{\langle u, v \rangle \in A_2} f(u, v) + \sum_{\langle u, v \rangle \in B} f(u, v) \right) \\
 &= \text{Cap}(S, \bar{S}) + f^-(S) - f^+(S) = \text{Cap}(S, \bar{S}) - \text{Val}f.
 \end{aligned}$$

当  $f$  给定后  $\min \text{Cap}'(S, \bar{S}) = \min \text{Cap}(S, \bar{S}) - \text{Val}f$ . 由最大流最小割定理,  $\text{Val}f^* =$

$\text{Val } f^* - \text{Val } f$ . 其中  $\tilde{f}^*$  表示  $N(f)$  中的最大流. 证毕.

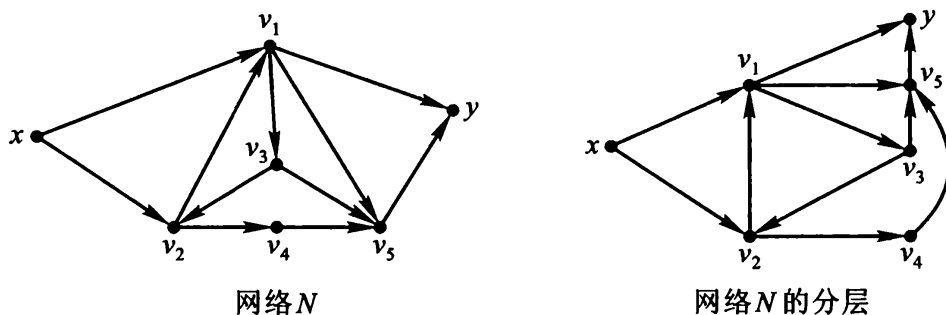
## 二、网络顶点的分层

设网络  $N = (V, x, y, A, C)$ , 令:  $V_i = \{v \in V \mid N \text{ 中 } x \text{ 到 } v \text{ 的最短有向路的长为 } i\}$ . 设  $x$  到  $y$  的最短有向路的长为  $n$ , 则 (1)  $x \in V_0, y \in V_n$ ; (2)  $V_i \cap V_j = \emptyset, (j \neq i)$ .  $V_i$  中的顶点称为网络  $N$  的第  $i$  层顶点.

注 此处一条有向路的长指该路上有向边的数目, 两点间的最短有向路指该两点间含有向边数最少的有向路.

例如, 下图显示了一个网络及其顶点的分层结果:

$$V_0 = \{x\}, V_1 = \{v_1, v_2\}, V_2 = \{v_3, v_4, v_5, y\}.$$

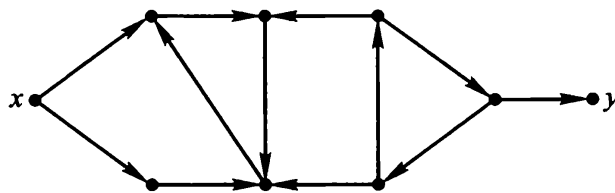


注 (1) 网络顶点分层后, 弧有三种可能的情况:

- 从第  $i$  层顶点指向第  $i+1$  层顶点;
- 从第  $i$  层顶点指向第  $i$  层顶点;
- 从第  $i$  层顶点指向第  $j$  层顶点 ( $j < i$ );

(2) 不存在从第  $i$  层顶点指向第  $i+k$  层顶点的弧 ( $k \geq 2$ ).

(3) 并非所有网络都能分层. 例如, 下列网络不能分层.



使用广度优先搜索法可获得网络顶点分层的一个简单算法.

### 网络分层算法

输入: 网络  $N = (V, x, y, A, C)$ .

输出: 网络分层后的各层顶点.

第 0 步: 令  $V_0 = \{x\}, S_0 = V_0, i := 0$ .

第 1 步: 若  $S_i = V$ , 分层完毕, 输出  $V_0, V_1, \dots, V_i$ ; 否则令

$$V_{i+1} = N^+(S_i) = \{S_i \text{ 的所有出邻点}\},$$

转下步.

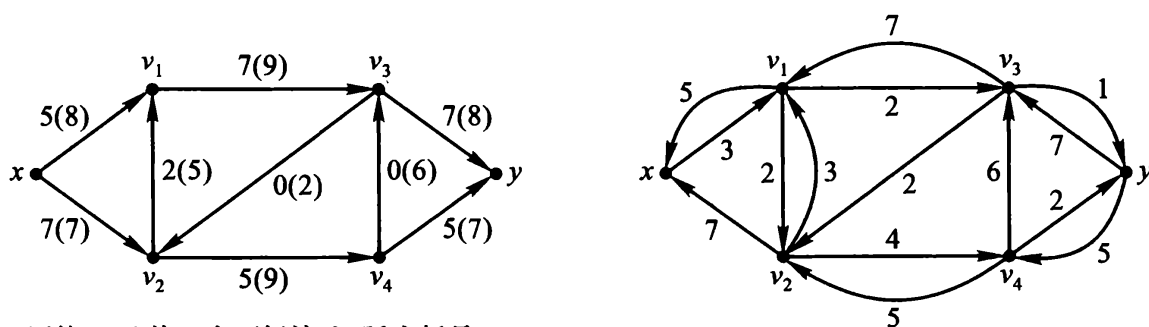
第 2 步: 若  $V_{i+1} = \emptyset$ , 则停止, 网络不能继续分层; 否则, 令  $S_{i+1} = S_i \cup V_{i+1}$ , 并令  $i := i + 1$ , 转第 1 步.

分层算法第 1 步求  $V_{i+1}$  时需要检查  $S_i$  的出弧, 最坏情况下每循环一次第 1 步只检查一条出弧, 且在算法执行过程中每条弧都作为出弧, 算法需循环  $\varepsilon$  次. 因此算法的时间复杂度为  $O(\varepsilon)$ .

### 三、辅助网络

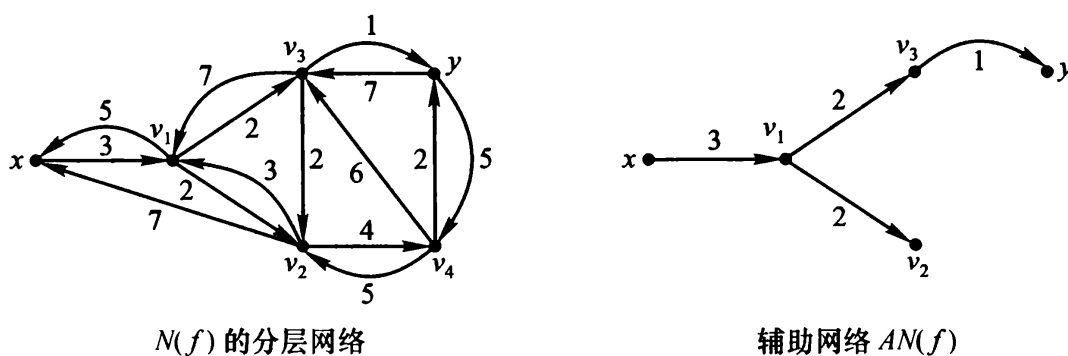
**定义 9.3.2** 设  $f$  是网络  $N(V, x, y, A, C)$  的一个可行流,  $N(f)$  是  $N$  的关于流  $f$  的增量网络. 对  $N(f)$  的顶点按最短有向路分层后, 删去比  $y$  层数高的顶点及与  $y$  同层的顶点 (保留  $y$ ), 再删去从高层指向低层的弧及同层顶点间的弧, 所剩的各条弧上的容量与  $N(f)$  中相同. 这样所得的网络是  $N(f)$  的子网络, 称为  $N$  的关于流  $f$  的辅助网络, 记为  $AN(f)$ .

例如下图中显示了一个网络  $N$ 、流  $f$  以及相应的增量网络、分层网络和辅助网络.



网络  $N$  及其一个可行流  $f$ , 弧上括号外数字为流值, 括号内数字为容量.

增量网络  $N(f)$ , 弧上数字是容量



$N(f)$  的分层网络

辅助网络  $AN(f)$

## 四、求最大流的 Dinic 算法

### 1. 算法的基本思想

从任一可行流  $f$  开始, 执行如下过程:

(i) 构造增量网络  $N(f)$ ;

(ii) 对  $N(f)$  分层并构造辅助网络  $AN(f)$ ;

(iii) 求  $AN(f)$  中一条  $x$ - $y$  有向路  $P$  (它是  $N$  中一条  $f$  可增路);

(iv) 在  $N$  中沿  $P$  增流得到更大的流, 并去掉因增流在  $AN(f)$  中所导致的饱和弧.

若此时  $AN(f)$  中仍存在  $x$ - $y$  有向路, 则再沿新的  $x$ - $y$  有向路 (可增路) 在  $N$  中增流, 直到  $N(f)$  的剩余网络中没有  $x$ - $y$  有向路为止.

(v) 反复执行上述四步直至新流  $f$  的增量网络  $N(f)$  不能分层到达  $y$  为止, 此时  $N$  中不再有  $f$  可增路, 因而得到最大流.

注 (1) 在 (ii) 中对  $N(f)$  分层时只要到达  $y$  即可停止, 此时已经有  $x$ - $y$  有向路, 尚未分层的顶点要么与  $y$  同层, 要么比  $y$  层数高, 均可直接删除.

(2) 辅助网络  $AN(f)$  中, 每条弧上只有容量, 没有其他权. 因此 (iii) 中求  $AN(f)$  中的  $x$ - $y$  有向路, 是指非赋权图的有向路.

(3) 上述 (iv) 中沿路  $P$  增流, 得到一个流值更大的新流  $\hat{f}$ , 方法如下: 对  $N$  中任一条弧  $a$ ,

$$\hat{f}(a) = \begin{cases} f(a) + \Delta f(P), & \text{若 } a \text{ 是 } P \text{ 的正向弧,} \\ f(a) - \Delta f(P), & \text{若 } a \text{ 是 } P \text{ 的反向弧,} \\ f(a), & \text{若 } a \text{ 不在 } P \text{ 上.} \end{cases} \quad (*)$$

修改后的流量为  $\text{Val} \hat{f} = \text{Val} f + \Delta f(P)$ . 其中  $\Delta f(P)$  是路  $P$  上的最小容量值.

更详细地说, Dinic 算法从网络  $N$  的任一可行流  $f_0$  开始, 构造增量网络  $N(f_0)$ , 对  $N(f_0)$  分层并构造辅助网络  $AN(f_0)$ . 在  $AN(f_0)$  中找一条  $x$ - $y$  有向路  $P$ , 在  $N$  中按 (\*) 式沿  $P$  增流, 得到可行流  $f_1^1$ , 并去掉增流所导致的  $AN(f_0)$  中的饱和弧. 若此时  $AN(f_0)$  中仍有  $x$ - $y$  有向路, 再沿新的  $x$ - $y$  有向路对  $N$  增流, 得到可行流  $f_1^2$ , 并去掉增流在  $AN(f_0)$  中造成的饱和弧. 反复执行这个过程, 直到剩余网络中没有  $x$ - $y$  有向路为止. 此时所得的流记为  $f_1^0$ .

然后对  $f_1^0$  继续上述过程. 在新的辅助网络  $AN(f_1^0)$  中反复找  $x$ - $y$  有向路, 并沿这些有向路增流, 并删去增流后  $AN(f_1^0)$  中的饱和弧, 得递增的流序列  $f_1^1, f_1^2, \dots, f_1^q$ . 设在得到  $f_1^q$  后  $AN(f_1^0)$  的剩余网络中不再有  $x$ - $y$  路, 则令  $f_1^q$  为  $f_2^0$ , 继续上述过程.

如此反复. 经有限次循环后, 得一可行流  $f_k^0$ , 此时构造的辅助网络  $AN(f_k^0)$  中不再有任何  $x$ - $y$  有向路, 即  $N$  中不存在  $f_k^0$  可增路, 从而  $f_k^0$  便是最大流.

注 构造增量网络的目的是将求网络  $N$  中可增路的问题化为求  $N(f)$  中的  $x$ - $y$  有向路; 而对  $N(f)$  分层并作辅助网络的目的是简化  $N(f)$ , 进一步将求  $N(f)$  中  $x$ - $y$

有向路的问题转化为求  $N(f)$  中的最短  $x$ - $y$  有向路.

## 2. Dinic 算法的步骤

输入: 网络  $N(V, x, y, A, C)$ .

输出: 网络  $N$  中一个最大流.

第 0 步: 在网络  $N(V, x, y, A, C)$  中任取一个可行流  $f_0$ , 令  $p = 0, q = 0, f_p^q = f_0$ .

第 1 步: 构造增量网络  $N(f_p^q)$ .

第 2 步: 对增量网络  $N(f_p^q)$  分层并构造辅助网络  $AN(f_p^q)$ . 如果在分层时汇点  $y$  得不到标号 (即分层不能到达  $y$ ), 则结束, 当前的  $f_p^q$  就是  $N$  的最大流; 否则, 转第 3 步.

第 3 步: (找  $AN(f_p^q)$  中的  $x$ - $y$  有向路, 深度优先法) 在辅助网络  $AN(f_p^q)$  中执行如下步骤:

3.0. 将源点  $x$  标号为  $(-1, \infty)$ , 令  $i := 0, v_i = x$ ;

3.1. 若  $v_i$  在  $AN(f_p^q)$  中没有出弧, 转 3.4; 否则在  $AN(f_p^q)$  任取  $v_i$  的一条出弧  $(v_i, v_j)$ , 转 3.2;

3.2. 设  $v_i$  的标号为  $(k, \delta_i)$ , 令  $\delta_j = \min\{\delta_i, \bar{c}_{ij}\}$  (其中  $\bar{c}_{ij}$  是  $AN(f_p^q)$  中弧  $(i, j)$  当前的容量), 将  $v_j$  标号为  $(i, \delta_j)$ ;

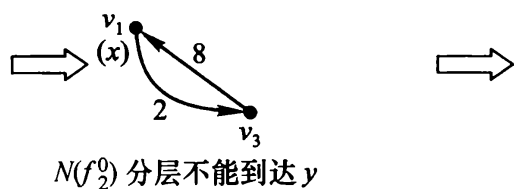
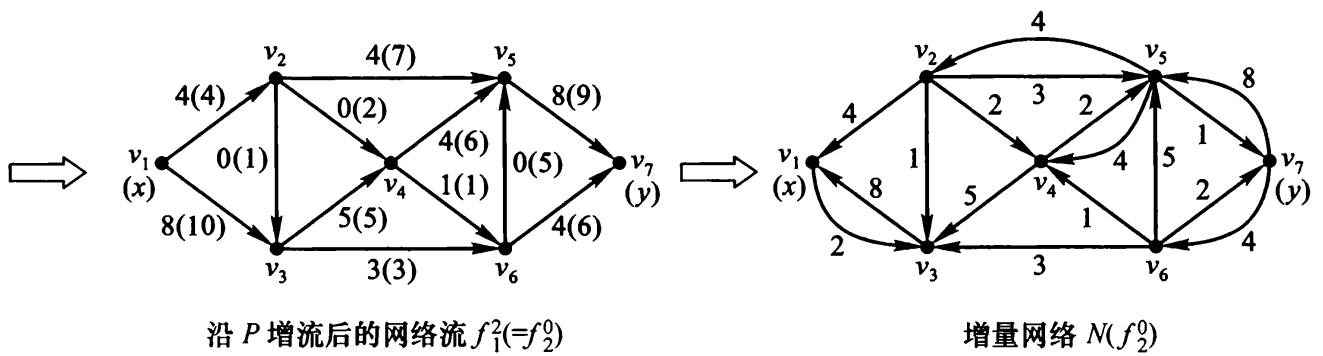
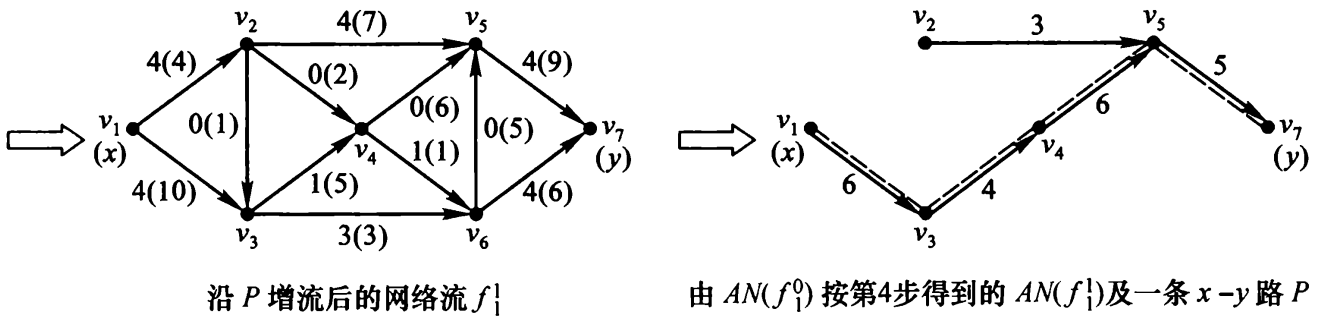
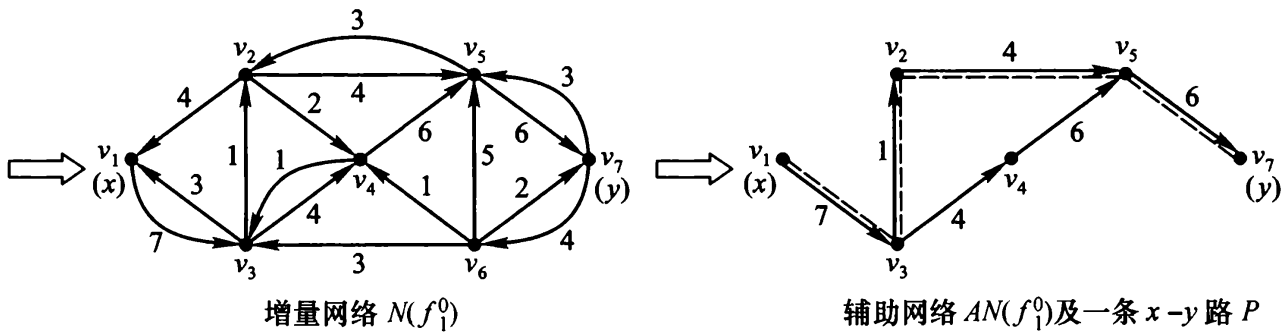
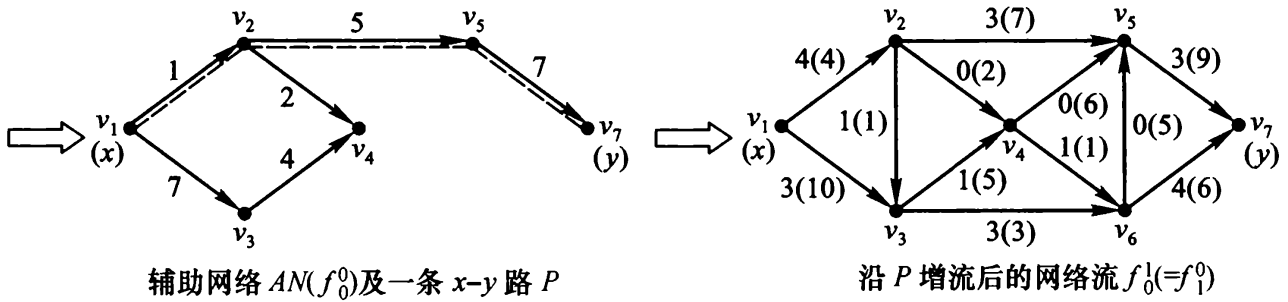
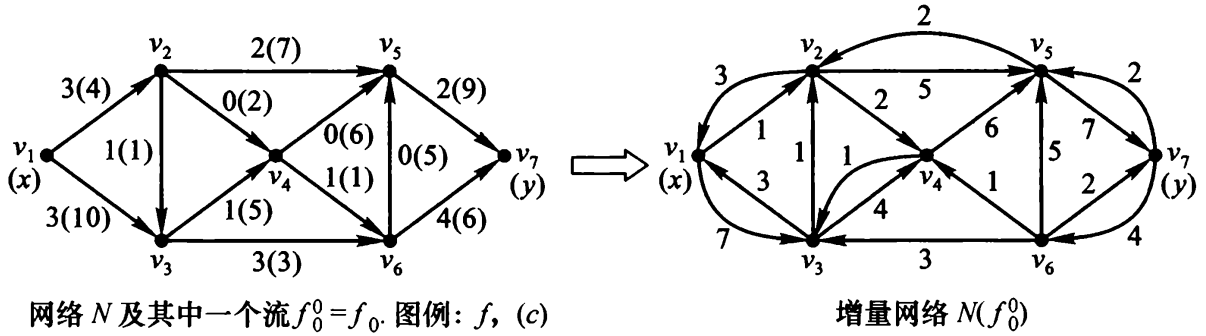
3.3. 若  $v_j = y$ , 转第 4 步; 否则令  $i = j$ , 转 3.1;

3.4. 设  $v_i$  的标号为  $(k, \delta_i)$ . 若  $k \neq -1$ , 在  $AN(f_p^q)$  中删去  $v_i$  及其所有入弧, 所得网络仍记为  $AN(f_p^q)$ , 令  $i = k$ , 转 3.1; 否则, 令  $f_{p+1}^0 := f_p^q, p := p + 1, q := 0$ , 转第 1 步.

第 4 步: (增流) 从顶点  $y$  出发按第一个标号反向追踪, 求出  $AN(f_p^q)$  的  $x$ - $y$  有向路  $P$ , 沿  $P$  对流  $f_p^q$  按 (\*) 式增流得到  $f_p^{q+1}$  (在 (\*) 中取  $\Delta f(P) = \delta_y$ ), 同时在  $AN(f_p^q)$  中将  $P$  上每条弧的容量  $\bar{c}_{ij}$  改为  $\bar{c}_{ij} - \delta_y$ , 删去容量为 0 的弧. 将所得网络记为  $AN(f_p^{q+1})$ , 并令  $q := q + 1$ . 去掉网络  $AN(f_p^q)$  中所有点的标号, 转第 3 步.

在算法中, 顶点  $v_i$  的标号  $(k, \delta_i)$  的第二项  $\delta_i$  表示  $AN(f_p^q)$  中当前走出的  $(x, v_i)$  有向路上各弧的最小容量. 因此在第 3.2 步中当这条有向路继续延伸时, 到达的新顶点  $v_j$  的标号第二项应为  $\delta_j = \min\{\delta_i, \bar{c}_{ij}\}$ . 标号  $(k, \delta_i)$  中的第一项表示  $v_i$  的当前标号是从哪个点获得的, 它是  $(x, v_i)$  有向路上  $v_i$  的后退一个点的下标, 用于反向追踪找到的最短有向路. 在第 3.1 步转第 3.4 步中, 在当前最短有向路到达的顶点  $v_i$  没有出弧时, 若  $k \neq -1$ , 则表明该点不是源  $x$  点, 因此应在  $AN(f_p^q)$  中删去  $v_i$  及其所有入弧, 以免后续找路循环中再次走到该点.

例如, 下图演示了用 Dinic 算法求网络  $N$  中最大流的循环过程.



算法结束, 当前流  $f_2^0$  是所求的最大流

我们来分析 Dinic 算法的计算复杂度. 明显地, 算法在反复执行“构造增量网络 (第 1 步)→对增量网络分层获得辅助网络 (第 2 步)→在辅助网络中找  $x-y$  有向路并在  $N$  中沿找到的可增路增流 (第 3 步、第 4 步)”大循环. 可以证明, 随着这种大循环次数的增加, 其中获得的辅助网络的层数严格单调递增<sup>[5][9]</sup>. 因此, 这种大循环至多执行  $\nu-1$  次. 在每轮大循环中, 构造增量网络并对其分层获得辅助网络 (第 1 步、第 2 步) 的计算复杂度为  $O(\varepsilon)$ ; 而在辅助网络中找  $x-y$  有向路有两种可能的结果: 要么到达一个没有出弧的顶点而删去至少一条弧, 要么找到一条  $x-y$  有向路并沿该路增流后, 删掉这条路上至少一条弧. 因此这种找路循环至多能进行  $\varepsilon$  次, 而在 (分层) 辅助网络中找一条  $x-y$  有向路的计算量为  $O(\nu)$ . 因此算法总的计算复杂度为  $O((\nu-1)(\varepsilon+\varepsilon\cdot\nu)) = O(\nu^2\varepsilon)$ .

## §9.4 求最大流的推拉流算法

上一节介绍的算法, 将网络  $N$  中的最大流问题转化为反复求辅助网络  $AN(f)$  中的  $x-y$  有向路, 在  $AN(f)$  中每找到一条  $x-y$  有向路, 便对应  $N$  中一条  $f$  可增路, 沿这条可增路增流后, 继续在辅助网络  $AN(f)$  中找  $x-y$  有向路, 直至辅助网络  $AN(f)$  中没有  $x-y$  有向路时, 再构造新的辅助网络. 在这一过程中, 往往会有前后几条可增路重复经过同一条弧或同一段路的情况, 导致在这些弧或路段上反复增流. 这在计算复杂度的角度来看, 是不经济的. 本节介绍的推拉流算法, 主要是针对这种情况的改进, 它在辅助网络  $AN(f)$  中可一次找出“一束”路来增流, 从而降低了计算复杂度.

我们先通过例子来描述推拉流算法的基本思想. 设  $AN(f)$  是流  $f$  的辅助网络. 对  $AN(f)$  中每个点  $v_i$ , 定义其通过能力 (throughput) 为该点出弧容量之和与入弧容量之和的较小者, 记为  $t(v_i)$ . 特别地, 定义源点  $x$  的通过能力  $t(x)$  为其出弧上的容量之和, 汇点  $y$  的通过能力  $t(y)$  为其入弧上的容量之和. 例如, 设图 1 是某个网络  $N$  中的流  $f$  的辅助网络, 则其中  $t(v_1) = 3, t(v_2) = 4, t(v_9) = 3$ , 而  $t(x) = 9, t(y) = 7$ .

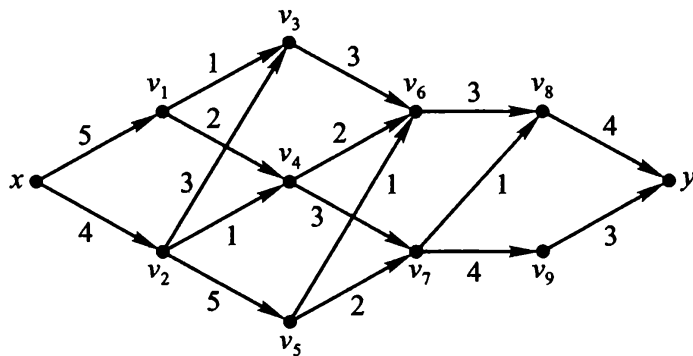


图 1. 一个辅助网络  $AN(f)$ , 弧上的数字是容量

取所有顶点中通过能力最小的一个, 比如  $v_1$ , 其通过能力为 3. 我们希望从  $v_1$  出发将 3 个单位的流推至汇点. 为此, 应当向  $v_3$  推 1 个单位, 并向  $v_4$  推 2 个单位. 推向  $v_3$  的 1 个单位可以一路经  $v_6, v_8$  到达汇点  $y$ . 这样我们已得到  $AN(f)$  中一条从  $v_1$  到



$y$  的有向路, 可对应出  $N$  中一条可增路实施增流. 但我们暂不这样做. 因为另 2 个单位的流也许会汇集到  $v_6$ 、 $v_8$ , 到时我们一并增流, 以免做重复的处理和计算. 在  $v_4$  点, 将到达的 2 个单位全部推向  $v_6$ , 与先前到达的 1 个单位合并, 形成一个 3 单位流, 经过  $v_8$  推进到汇点. 这样我们成功地将  $v_3$  所能承受的 3 个单位的流推到了汇点. 注意在  $v_4$  点处, 我们没有将流分成两个 1 单位流分别推向  $v_6$  和  $v_7$ . 这是因为我们总是希望动用尽可能少的顶点以节省计算量. 因此, 在向前推流时, 算法总是尽可能饱和一条选定的弧. 此外, 在推流的过程中, 我们并不担心推进的流到达某点后由于其出容量太小而被迫退回, 因为我们选择的出发点  $v_1$  是通过能力最小的点, 3 个单位的流能够通过  $v_1$ , 就必定能通过所有其他顶点.

事情到此并未结束, 因为要使  $v_1$  有 3 个单位的流推向汇点, 就必须通过  $v_1$  的人邻点逐步向源点“索取” 3 个单位的流, 以便形成完整的  $x$ - $y$  增流路束, 这个过程称为“拉流”. 拉流过程是前述推流过程的反向操作, 做法与推流完全一样. 在现在的例子中, 因为源点  $x$  恰好是  $v_1$  点的人邻点, 因此直接从  $x$  点拉 3 个单位的流即可. 形成的  $x$ - $y$  增流路束如下图 2 所示.

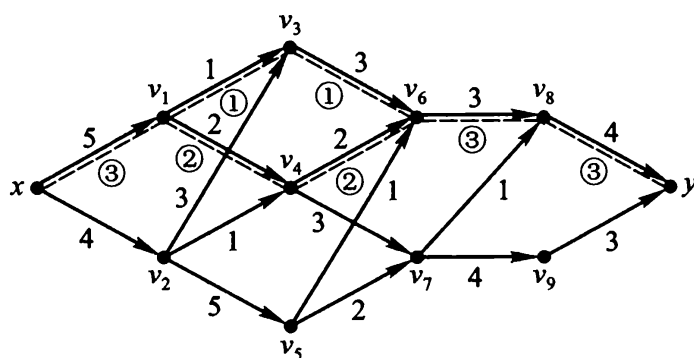


图 2. 虚线所示为找到的增流路束, 圆圈中的数字为弧增流量.

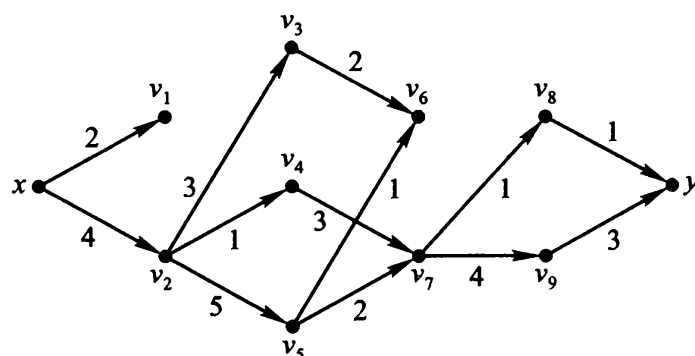


图 3. 删去饱和弧、修改弧容量后剩余的辅助网络.

沿“增流路束”对  $N$  增流后, 去掉在当前辅助网络  $AN(f)$  中所导致的饱和弧, 并修改相关弧上的容量, 剩余的辅助网络如图 3 所示. 继续递推地删去没有出弧或没有入弧的点及其关联的弧 (源点和汇点除外), 得到图 4.

在残余的辅助网络 (图 4) 中选定通过能力最小的点  $v_4$  继续施行推拉流过程, 得

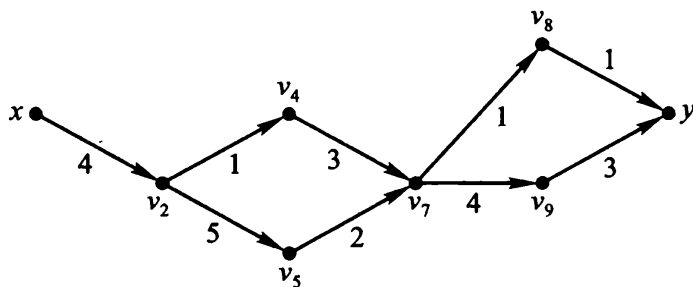


图 4. 递推地删去图 3 中没有出弧或没有入弧的点及其关联的弧后, 所得之图.

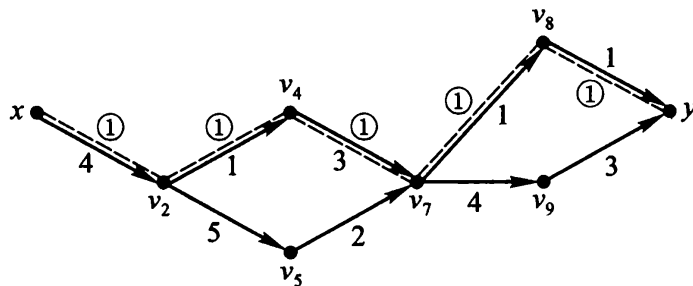


图 5. 对图 4 从  $v_4$  出发施行推拉流过程, 得到的可增路束.

到可增路束如图 5 所示. 沿这个可增路束对  $N$  增流 1 个单位后, 删去在当前辅助网络中所导致的饱和弧, 并修改相关弧上的容量, 然后递推地删去没有出弧或没有入弧的点及其关联的弧 (源点和汇点除外), 得到图 6. 在图 6 中从通过能力最小的  $v_5$  出发, 进行推拉流操作, 得到可增  $x-y$  路 (现在就是图 6 本身). 沿这条路对  $N$  增流 2 个单位后, 删除在当前辅助网络中所导致的饱和弧, 发现残余图中已没有  $x-y$  路 (图 7), 该轮循环结束. 此时, 在  $N$  中现有流基础上重新构造辅助网络, 开始下一轮循环.

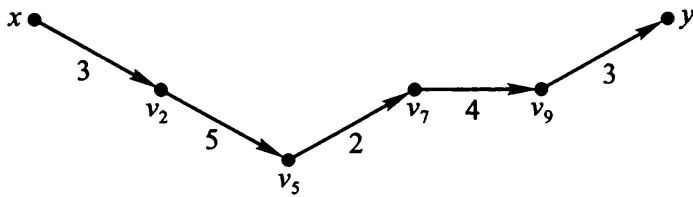


图 6. 按图 5 增流后, 递推地删去没有出弧或没有入弧的点及其关联的弧后, 所得之图.

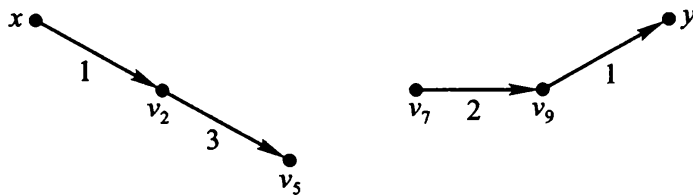


图 7. 本轮循环最终所得之图, 无  $x-y$  路

推拉流算法的步骤描述如下, 更为详细的描述见文献 [4].

### 推拉流算法

输入: 网络  $N = (V, x, y, A, C)$

输出: 网络  $N$  的一个最大流

第 1 步: (初始化) 任取  $N$  中一个可行流作为初始流  $f$  (比如零值流).

第 2 步: 构造增量网络  $N(f)$ , 并对  $N(f)$  分层. 若汇点  $y$  得不到分层, 则停止, 当前流  $f$  即为最大流. 否则, 在分层基础上获得辅助网络  $AN(f)$  (同 Dinic 算法), 转下步.

第 3 步: 求  $AN(f)$  中各顶点的通过能力.

$$t(v) = \min \left\{ \sum_{v_i \in N^-(v)} c(v_i, v), \sum_{v_j \in N^+(v)} c(v, v_j) \right\} \quad (v \neq x, v \neq y),$$

$$t(x) = \sum_{v_i \in N^+(x)} c(x, v_i), \quad t(y) = \sum_{v_i \in N^-(y)} c(v_i, y).$$

取通过能力最小的一点  $v_0$ , 转下步.

第 4 步: 在  $AN(f)$  中从  $v_0$  出发将  $t(v_0)$  个单位的流向  $y$  推进. 推进过程中采用先进先出的规则. 先处理完本层顶点, 再处理下层顶点. 在任意顶点处, 总是先选定一条出弧安排推流, 当它饱和后再选择另一条出弧安排剩余流, 如此类推. 当  $t(v_0)$  个单位的流全部推进到  $y$  时, 记录经过的弧以及各弧上推进的流量, 转下步.

第 5 步: 从  $v_0$  开始向  $x$  点逆行将  $t(v_0)$  个单位的流从  $x$  拉到  $v_0$ , 拉流规则和做法同第 4 步, 只是将出弧改为入弧. 拉流过程完成时, 得到  $AN(f)$  中从  $x$  到  $y$  的一个流  $f'$  (没有推拉流的弧上流量视为 0), 记录各弧上的  $f'$  流量, 并转下步.

第 6 步: 对原网络流  $f$  增流, 增流计算办法如下:

$$\text{对 } N \text{ 中任何弧 } a, f(a) := \begin{cases} f(a) + f'(a), & \text{若 } a \text{ 是正向弧,} \\ f(a) - f'(a), & \text{若 } a \text{ 是反向弧.} \end{cases}$$

转下步.

第 7 步: 修改  $AN(f)$  中各弧的容量: 对  $AN(f)$  中任何弧  $a, c(a) := c(a) - f'(a)$ . 删去在这一过程中所得的容量为 0 的弧. 然后再逐次删去出度为 0 的顶点及其关联的弧, 和入度为 0 的顶点及其关联的弧 ( $x, y$  除外), 所得网络仍记为  $AN(f)$ . 若  $AN(f)$  中有  $x-y$  路, 则转第 3 步; 否则, 用当前流替换  $N$  中原有的流, 转第 2 步.

我们来考虑推拉流算法的正确性. 首先, 由于选择从通过能力最小的点出发进行推、拉流操作, 且推、拉流是等量的, 因此拉流和推流分别能顺利到达  $x$  和  $y$ , 形成  $t(v_0)$  个单位的  $x-y$  可增路束. 其次, 在推拉流过程中, 始终遵守容量限制和流量守恒约束, 因此推拉流获得的流  $f'$  是辅助网络 (增量网络) 中的可行流, 从而按第 6 步增流后得到的  $f$  是  $N$  中的可行流. 另外, 算法在新的增量网络  $N(f)$  不能分层时终止, 此时增量网络  $N(f)$  中不再有从  $x$  到  $y$  的有向路, 因此当前的增量网络中最大流量为 0, 结合定理 9.3.2 得,  $\text{Val}f^* - \text{Val}f = 0$ , 其中  $f^*$  表示网络  $N$  中的最大流. 这表明推拉流算法结束时获得的流  $f$  就是  $N$  中的最大流.

关于推拉流算法的时间复杂度,容易证明每次在第 2 步重新构造辅助网络后,辅助网络中的  $x$ - $y$  有向路的长度是严格增加的,因此第 2 步至第 7 步的大循环至多重复  $\nu$  次. 在每一轮大循环中,一条弧不论在哪个阶段达到饱和,就会被去掉,因此处理饱和弧的计算次数不超过  $O(\varepsilon)$ . 如果某条弧经过一次处理没被饱和,则在这轮大循环中可能还会被多次处理. 因为在任何阶段  $AN(f)$  至多有  $\nu$  层顶点,按照算法,一个可增路束每经过一层弧时,在其使用的弧中至多会有一条不被饱和. 因此在产生每个可增路束时,最多留下  $\nu - 1$  条被其使用过但未饱和的弧,需在后续找新的可增路束时重新处理. 而每轮大循环中至多从  $\nu$  个点出发进行推拉流过程,产生可增路束,也就是说,每轮大循环中至多产生  $\nu$  条可增路束,因此,重复处理弧的次数不超过  $\nu(\nu - 1)$ . 于是完成每次大循环需要的计算量不超过  $O(\varepsilon + \nu^2)$ . 由此可知,推拉流算法的计算复杂度为  $O(\nu(\varepsilon + \nu^2)) = O(\nu^3)$ .

本节介绍的推拉流算法,反复在辅助网络  $AN(f)$  中找“一束”有向路,沿这束路将流的增量向前推进. 这里蕴含了一种称为预流推进 (preflow-push) 的思想,这种思想后来得到了进一步的发展. 有兴趣的读者可参看文献 [5]、[10]~[16].

## §9.5 最大流问题的一些扩展

### 一、可行循环流

**定义 9.5.1** 设  $\vec{G}(V, A)$  是一个有向图,在  $\vec{G}$  的每条弧  $a \in A$  上,都带有两个非负实数  $l(a)$  和  $c(a)$ ,满足  $l(a) \leq c(a)$ ,分别称为弧  $a$  的下容量和上容量. 如果存在弧集  $A$  上的一个实函数  $f: \langle v_i, v_j \rangle \rightarrow f(v_i, v_j), (\langle v_i, v_j \rangle \in A)$ , 满足:

$$(1) \text{ 对 } \forall v_i \in V, \sum_{v_j \in N^+(v_i)} f(v_i, v_j) - \sum_{v_j \in N^-(v_i)} f(v_j, v_i) = 0, \forall v_i \in V;$$

$$(2) \text{ 对 } \forall \langle v_i, v_j \rangle \in A, l(v_i, v_j) \leq f(v_i, v_j) \leq c(v_i, v_j).$$

则称  $f$  是  $\vec{G}$  的一个可行循环流 (feasible circulation).

由以上定义可见,我们要考虑的是一般有向图上的循环流问题,这个问题中并不要求有源和汇点. 有源和汇的网络是其中的特例. 另一点与以前不同的是,此处每条弧上既有上容量,又有下容量,可行流既要满足上容量限制,又要满足下容量限制. 容易看出在这样的要求下,可行循环流未必存在. 什么条件下存在可行循环流? 如何求可行循环流? 这是本节要考虑的问题.

我们从  $\vec{G}(V, A)$  出发构造一个网络  $N = (V', x, y, A', C')$  如下: 给  $\vec{G}$  添加一个附加源  $x$  和一个附加汇  $y$ .  $x$  向  $\vec{G}$  的每个顶点引一条弧,弧容量为该弧指向顶点在  $\vec{G}$  中的所有入弧的下容量之和;  $\vec{G}$  的每个顶点向  $y$  引一条弧,弧容量为该弧引出顶点在  $\vec{G}$  中的所有出弧的下容量之和;  $\vec{G}$  的原有弧全部保留,但弧容量 (上界) 改为原来上下容

量之差 (下界为 0). 即

$$\begin{aligned} V' &= V \cup \{x, y\}, \\ A' &= A \cup \{\langle x, v_i \rangle | v_i \in V\} \cup \{\langle v_i, y \rangle | v_i \in V\}, \\ \forall v_i \in V, c'(x, v_i) &= \sum_{v_j \in N_G^-(v_i)} l(v_j, v_i), \quad c'(v_i, y) = \sum_{v_j \in N_G^+(v_i)} l(v_i, v_j), \\ \forall \langle v_i, v_j \rangle \in A, c'(v_i, v_j) &= c(v_i, v_j) - l(v_i, v_j). \end{aligned}$$

这个网络称为带有上下容量的有向图  $\vec{G}(V, A)$  的伴随网络. 带有上下容量的有向图  $\vec{G}(V, A)$  中可行循环流的存在性与其伴随网络  $N = (V', x, y, A', C')$  中的最大流有关. 这体现为下述定理.

**定理 9.5.1** 带有上下容量的有向图  $\vec{G}(V, A)$  中存在可行循环流当且仅当其伴随网络  $N = (V', x, y, A', C')$  中最大流的流量等于  $\vec{G}(V, A)$  中所有弧的下容量之和.

**证明 必要性** 设伴随网络  $N = (V', x, y, A', C')$  中最大流  $f^*$  的流量等于  $\vec{G}(V, A)$  中所有弧的下容量之和. 则由网络  $N = (V', x, y, A', C')$  的构造可知,  $f^*$  在每条从  $x$  发出的弧和每条指向  $y$  的弧上都是饱和的 (即达到弧容量). 基于流  $f^*$  构造  $\vec{G}(V, A)$  中一个流  $f$  如下:

$$\forall \langle v_i, v_j \rangle \in A, f(v_i, v_j) = f^*(v_i, v_j) + l(v_i, v_j)$$

则  $f$  是  $\vec{G}(V, A)$  中一个可行循环流. 这是因为:

(1) 对  $\forall \langle v_i, v_j \rangle \in A$ ,  $f^*$  在  $N$  中满足容量约束

$$0 \leq f^*(v_i, v_j) \leq c'(v_i, v_j) = c(v_i, v_j) - l(v_i, v_j),$$

故

$$l(v_i, v_j) \leq f^*(v_i, v_j) + l(v_i, v_j) \leq c(v_i, v_j),$$

即  $l(v_i, v_j) \leq f(v_i, v_j) \leq c(v_i, v_j)$ . 这表明  $f$  满足容量约束.

(2) 对  $\forall v_i \in V$ ,  $f^*$  在  $N$  中满足流量守恒:  $\sum_{v_j \in N^-(v_i)} f^*(v_j, v_i) = \sum_{v_k \in N^+(v_i)} f^*(v_i, v_k)$ ,

即

$$\sum_{v_j \in N^-(v_i) \setminus \{x\}} f^*(v_j, v_i) + f^*(x, v_i) = \sum_{v_k \in N^+(v_i) \setminus \{y\}} f^*(v_i, v_k) + f^*(v_i, y).$$

因  $f^*(x, v_i) = c'(x, v_i) = \sum_{v_j \in N_G^-(v_i)} l(v_j, v_i)$ , 而  $f^*(v_i, y) = c'(v_i, y) = \sum_{v_k \in N_G^+(v_i)} l(v_i, v_k)$ ,

将此二式代入上式, 得

$$\sum_{v_j \in N^-(v_i) \setminus \{x\}} f^*(v_j, v_i) + \sum_{v_j \in N_G^-(v_i)} l(v_j, v_i) = \sum_{v_k \in N^+(v_i) \setminus \{y\}} f^*(v_i, v_k) + \sum_{v_k \in N_G^+(v_i)} l(v_i, v_k),$$

注意到  $N^-(v_i) \setminus \{x\} = N_G^-(v_i)$ ,  $N^+(v_i) \setminus \{y\} = N_G^+(v_i)$ , 故

$$\sum_{v_j \in N_G^-(v_i)} (f^*(v_j, v_i) + l(v_j, v_i)) = \sum_{v_j \in N_G^+(v_i)} (f^*(v_i, v_j) + l(v_i, v_j)),$$

从而  $\sum_{v_j \in N_G^-(v_i)} f(v_j, v_i) = \sum_{v_j \in N_G^+(v_i)} f(v_i, v_j)$ . 这表明  $f$  满足流量守恒.

**充分性** 设  $f$  是  $\vec{G}(V, A)$  中一个可行循环流. 构造  $N$  中的一个流  $f^*$  如下:

$$\text{对 } \forall \langle v_i, v_j \rangle \in A, f^*(v_i, v_j) = f(v_i, v_j) - l(v_i, v_j),$$

$$\text{对 } \forall v_i \in V, f^*(x, v_i) = c'(x, v_i), \quad f^*(v_i, y) = c'(v_i, y).$$

容易验证  $f^*$  是  $N$  的一个可行流, 且因它饱和了从源  $x$  发出的每条弧, 因此是  $N$  中的最大流, 其流量为  $\text{Val} f^* = \sum_{v_i \in V} f^*(x, v_i) = \sum_{v_i \in V} C'(x, v_i) = \sum_{v_i \in V} \sum_{v_j \in N_G^-(v_i)} l(v_j, v_i) =$

$$\sum_{\langle u, v \rangle \in A} l(u, v). \quad \text{证毕.}$$

按照这个定理可得到在带有上下容量的有向图  $\vec{G}(V, A)$  中求可行循环流的一个算法.

### 求可行循环流的算法

输入: 带有上下容量约束的有向图  $\vec{G}(V, A)$ .

输出:  $\vec{G}(V, A)$  中一个可行循环流.

第 1 步: 按照定理 9.5.1 中的方法, 由  $\vec{G}(V, A)$  构造伴随网络  $N = (V', x, y, A', C')$ .

第 2 步: 运用最大流算法求  $N$  中的最大流  $f^*$ .

第 3 步: 若  $f^*$  的流量等于  $\vec{G}(V, A)$  中所有弧的下容量之和, 则流  $f$ :

$$\forall \langle v_i, v_j \rangle \in A, f(v_i, v_j) = f^*(v_i, v_j) + l(v_i, v_j)$$

是  $\vec{G}(V, A)$  中的一个可行循环流, 停止; 否则,  $\vec{G}(V, A)$  中不存在可行循环流, 停止.

该算法的主要计算量在第 2 步, 因此其计算复杂度取决于第 2 步中使用的求最大流算法的计算复杂度. 按推拉流算法, 该算法的复杂度为  $O(\nu^3)$ .

## 二、上下容量网络中的最大流

设  $N = (V, x, y, A, C, L)$  是一个单源单汇网络, 但每条弧上带有两个容量  $C$  和  $L$ ;  $\forall \langle v_i, v_j \rangle \in A, l(v_i, v_j) \leq f(v_i, v_j) \leq c(v_i, v_j)$ ,  $C$  和  $L$  分别称为网络的上容量和下容量. 这样的网络称为上下容量网络. 上下容量网络中可行流的定义同普通网络中基本一样, 只是除了要满足流量守恒约束外, 还需要满足上下容量约束. 上下容量网络中不

一定总存在可行流. 如果存在可行流, 则除了需要考虑最大可行流外, 往往也需要考虑最小可行流.

我们先来讨论如何判断上下容量网络中是否存在可行流. 给定一个上下容量网络  $N = (V, x, y, A, C, L)$ , 添加一条从汇到源的弧  $\langle y, x \rangle$ , 其上下容量规定为  $c(y, x) = \infty, l(y, x) = 0$ . 这样得到一个带有上下容量的有向图  $\vec{G}$ . 显然  $N = (V, x, y, A, C, L)$  中存在可行流当且仅当有向图  $\vec{G}$  中存在可行循环流. 有向图  $\vec{G}$  中可行循环流的存在性判别条件已在上一节定理 9.5.1 中给出.

下面考虑如何求上下容量网络中的最大可行流. 设  $f$  是上下容量网络  $N = (V, x, y, A, C, L)$  中一个可行流,  $P$  是  $N$  中一条  $x$ - $y$  路. 若  $P$  的每条前向弧  $\langle v_i, v_j \rangle$  都有  $f(v_i, v_j) < c(v_i, v_j)$ , 而每条后向弧  $\langle v_i, v_j \rangle$  都有  $f(v_i, v_j) > l(v_i, v_j)$ , 则称  $P$  是  $N$  中一条从  $x$  到  $y$  的  $f$ -可增路. 类似可定义从  $y$  到  $x$  的  $f$ -可增路.

设  $(S, \bar{S})$  是  $N$  中一个割. 由于

$$\sum_{\langle v_i, v_j \rangle \in (S, \bar{S})} f(v_i, v_j) - \sum_{\langle v_j, v_i \rangle \in (\bar{S}, S)} f(v_j, v_i) = \text{Val}f,$$

易知  $c(S, \bar{S}) - l(\bar{S}, S) \geq \text{Val}f$ , 且  $l(S, \bar{S}) - c(\bar{S}, S) \leq \text{Val}f$ . 因此, 若某个可行流  $f^*$  满足:  $c(S, \bar{S}) - l(\bar{S}, S) = \text{Val}f^*$ , 则  $f^*$  是最大流; 若  $f^*$  满足:  $l(S, \bar{S}) - c(\bar{S}, S) = \text{Val}f^*$ , 则  $f^*$  是最小流.

与最大流最小割定理类似可证明如下定理.

**定理 9.5.2** 设  $f$  是上下容量网络  $N = (V, x, y, A, C, L)$  中一个可行流.  $f$  是  $N$  的一个最大流的充分必要条件是  $N$  中不存在从  $x$  到  $y$  的  $f$ -可增路;  $f$  是  $N$  的一个最小流的充分必要条件是  $N$  中不存在从  $y$  到  $x$  的  $f$ -可增路.

由此可得求上下容量网络中最大流的一个算法.

### 求上下容量网络中最大流算法

输入: 上下容量网络  $N = (V, x, y, A, C, L)$ .

输出:  $N$  中一个最大可行流.

第 1 步: 给  $N$  添加一条从汇到源的弧  $\langle y, x \rangle$ , 其上下容量规定为  $c(y, x) = \infty, l(y, x) = 0$ . 得到一个带有上下容量的有向图  $\vec{G}$ . 运用上节的算法求  $\vec{G}$  中的可行循环流. 若可行循环流存在, 则得到  $N$  的一个可行流  $f$ , 转下步; 否则,  $N$  中不存在可行流, 停止.

第 2 步: 在  $N$  中求从  $x$  到  $y$  的  $f$ -可增路. 若不存在从  $x$  到  $y$  的  $f$ -可增路, 则当前流是最大可行流, 停止; 否则, 求出一条从  $x$  到  $y$  的  $f$ -可增路  $P$ , 转下步.

第 3 步: 沿  $f$ -可增路  $P$  增流, 得到  $N$  中流值更大的新可行流, 取代原有的流  $f$ , 转第 2 步.

在算法第 2 步中, 找  $f$ -可增路可以使用前几节介绍的方法, 如标号法、Dinic 方法等. 第 3 步沿可增路增流的方法与 §9.2 节中相同.

算法的计算复杂度取决于第 1 步中求  $\vec{G}$  中的可行循环流的复杂度和第 2 步求  $f$ -可增路的复杂度. 由上节知, 求  $\vec{G}$  中的可行循环流的复杂度为  $O(\nu^3)$ . 若按照 Dinic 方法求  $f$ -可增路, 则计算复杂度为  $O(\nu^2\varepsilon)$ . 因此该算法总的计算复杂度不超过  $O(\nu^2\varepsilon)$ . 若对第二、三步加以改造, 在求  $f$ -可增路时使用推拉流算法思想, 其计算复杂度可降低为  $O(\nu^3)$ .

### 三、供需网络的可行流

设  $N = (V, X, Y, A, C)$  是一个网络, 其中有一个顶点子集  $X = \{x_1, x_2, \dots, x_m\}$ , 每个  $x_i$  的总发量不能少于总收量, 这些点称为广义源; 另有一个顶点子集  $Y = \{y_1, y_2, \dots, y_n\}$ , 每个  $y_j$  的总收量不能少于总发量, 这些点称为广义汇. 这种网络称为有供需约束的网络, 简称供需网络.

设对每个广义源  $x_i$  有一个非负整数  $\sigma(x_i)$ , 称为供应量; 对每个广义汇  $y_j$  也有一个非负整数  $\rho(y_j)$ , 称为需求量. 如果网络  $N = (V, X, Y, A, C)$  中一个流函数  $f(a) (a \in A)$ , 满足

- (1) 对  $\forall a \in A, 0 \leq f(a) \leq c(a)$  (容量约束);
- (2) 对  $\forall v \in V - (X \cup Y), f^-(v) = f^+(v)$  (流量守恒);
- (3)  $f^+(x_i) - f^-(x_i) \leq \sigma(x_i), i = 1, 2, \dots, m$ ;
- (4)  $f^-(y_j) - f^+(y_j) \geq \rho(y_j), j = 1, 2, \dots, n$ .

则称  $f(a)$  为网络  $N = (V, X, Y, A, C)$  中一个符合供需要求的可行流, 简称为供需可行流. 其中约束条件 (3) 表示对每个广义源点, 求不能大于供, 而条件 (4) 表示对于每个广义汇点, 给不能小于需. 供需可行流在有些文献中也称为相容流 (consistent flow).

在一个供需网络  $N = (V, X, Y, A, C)$  中, 供需可行流未必存在. 我们关心的一个问题是, 什么条件下供需可行流存在? 如果存在, 如何求出一个供需可行流? 下面的定理回答这些问题.

**定理 9.5.3** 一个供需网络  $N = (V, X, Y, A, C)$  中存在供需可行流的充分必要条件是对于所有  $S \subseteq V$  都有  $c(S, \bar{S}) \geq \rho(\bar{S} \cap Y) - \sigma(\bar{S} \cap X)$ .

**证明** 基于网络  $N$  构造一个新网络  $N'$  如下:

- (1) 在  $N$  中添加两个新顶点  $x$  和  $y$ ;
- (2) 对每个  $x_i \in X$ , 添加从  $x$  指向  $x_i$  的弧  $\langle x, x_i \rangle$ , 容量为  $\sigma(x_i)$ ;
- (3) 对每个  $y_j \in Y$ , 添加从  $y_j$  指向  $y$  的弧  $\langle y_j, y \rangle$ , 容量为  $\rho(y_j)$ ;
- (4)  $N$  中其他弧保留, 容量不变.

则  $N'$  是以  $x$  为源点  $y$  为汇点的一个单源单汇网络. 容易看出,  $N$  中有供需可行流当



且仅当  $N'$  中有饱和割  $(Y, \{y\})$  的每条弧的可行流. 而由  $N'$  的结构, 饱和割  $(Y, \{y\})$  的每条弧的可行流必定是  $N'$  的最大流, 其流量为

$$\rho(Y) = c(Y, \{y\}) = \sum_j \rho(y_j).$$

反之,  $N'$  中流量等于割容量  $c(Y, \{y\}) = \sum_j \rho(y_j)$  的可行流必是  $N'$  的最大流且饱和  $(Y, \{y\})$  的每条弧. 由此可知,  $N$  中有供需可行流当且仅当  $N'$  中的最大流流量等于  $c(Y, \{y\}) = \sum_j \rho(y_j)$ , 进一步地, 由最大流最小割定理,  $N'$  中最大流的流量等于  $c(Y, \{y\})$  当且仅当,  $(Y, \{y\})$  是  $N'$  的最小割. 因此,  $N$  中有供需可行流当且仅当  $N'$  的每个割  $(S \cup \{x\}, \bar{S} \cup \{y\}) (\forall s \in V)$  都有  $c(S \cup \{x\}, \bar{S} \cup \{y\}) \geq \rho(Y)$ . 我们用  $c'$  来表示  $N'$  中弧 (以及割) 的容量. 由  $N'$  的结构, 割  $(S \cup \{x\}, \bar{S} \cup \{y\})$  中的弧包括三部分:  $(S, \bar{S}), (S \cap Y, \{y\}), (\{x\}, \bar{S} \cap X)$ , 故

$$\begin{aligned} c(S \cup \{x\}, \bar{S} \cup \{y\}) &= c'(S, \bar{S}) + c'(S \cap Y, \{y\}) + c'(\{x\}, \bar{S} \cap X) \\ &= c(S, \bar{S}) + \rho(S \cap Y) + \sigma(\bar{S} \cap X), \end{aligned}$$

从而  $c(S \cup \{x\}, \bar{S} \cup \{y\}) \geq \rho(Y)$  等价于  $c(S, \bar{S}) \geq \rho(Y) - \rho(S \cap Y) - \sigma(\bar{S} \cap X)$ , 即

$$c(S, \bar{S}) \geq \rho(\bar{S} \cap Y) - \sigma(\bar{S} \cap X). \quad \text{证毕.}$$

由上述定理的证明, 可得求供需可行流的一个算法: 先基于网络  $N$  构造网络  $N'$ , 然后用最大流算法求  $N'$  中的最大流. 如果所得最大流的流量等于  $\rho(Y) = \sum_j \rho(y_j)$  (即饱和  $(Y, \{y\})$  的每条弧), 则当前的流便产生  $N$  中的一个供需可行流, 否则,  $N$  中不存在供需可行流. 这个算法的计算复杂度显然取决于所用的最大流算法的复杂度.

## §9.6 最小费用流问题

**定义 9.6.1** 对网络  $N = (V, x, y, A, C)$ , 给其每条弧  $\langle u, v \rangle$  赋以一个非负实数 (权)  $w(u, v)$ , 称为弧  $\langle u, v \rangle$  的费用. 这种每条弧都带有费用的网络, 称为费用网络, 记为  $N = (V, x, y, A, C, w)$  或  $N = (V, A, C, w)$ , 其中  $w$  为费用函数. 费用网络  $N = (V, x, y, A, C, w)$  中流  $f$  的费用定义为

$$w(f) = \sum_{\langle u, v \rangle \in A} [w(u, v) \cdot f(u, v)].$$

**最小费用流问题:** 对于网络  $N = (V, x, y, A, C, w)$  和一个给定的数  $v_0 \geq 0$ , 求  $N$  中流量为  $v_0$  且费用最小的可行流.

下面介绍两种求解最小费用流问题的算法. 为方便起见, 今后将流量为  $v_0$  的流简称为  $v_0$  流.

### 一、求最小费用流的负费用圈算法<sup>[17]</sup>

**定义 9.6.2 (费用网络的增量网络)** 设  $f$  是网络  $N = (V, x, y, A, C, w)$  上的一个可行流,  $N$  关于  $f$  的增量网络为  $N(f) = (V, x, y, A(f), C', w')$ , 其中  $A(f), C'$ , 及  $w'$  如下:

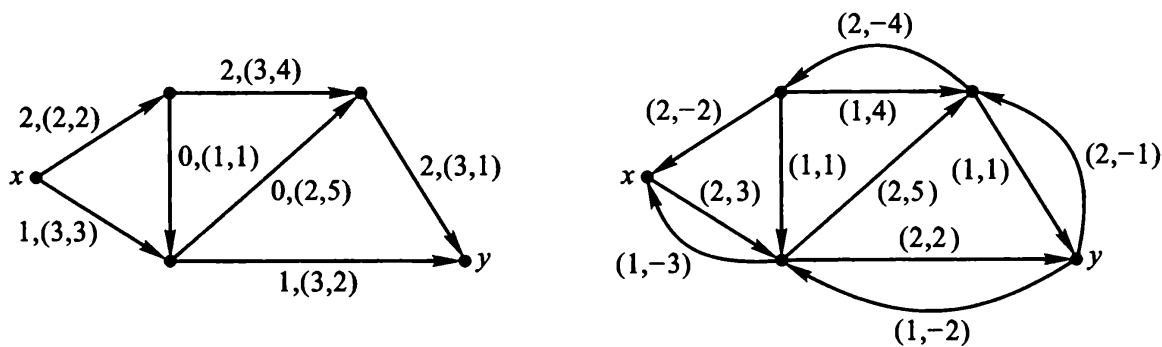
(1) 若  $\langle u, v \rangle \in A$  且  $f(u, v) < C(u, v)$ , 则  $\langle u, v \rangle \in A(f)$ , 且

$$C'(u, v) = C(u, v) - f(u, v), w'(u, v) = w(u, v);$$

(2) 若  $\langle u, v \rangle \in A$  且  $f(u, v) > 0$ , 则  $\langle v, u \rangle \in A(f)$ , 且

$$C'(v, u) = f(u, v), w'(v, u) = -w(u, v).$$

**例 9.6.1** 下面第一个图显示了一个网络  $N$  及其上的一个可行流  $f$ , 第二个图显示了  $N$  关于  $f$  的增量网络  $N(f)$ .



网络  $N$  及其一个可行流  $f$ , 每条弧上的三元数组分别表示  $f, (c, w)$ .

增量网络  $N(f)$ , 每条弧上的二元数组分别表示  $(c', w')$ .

**定义 9.6.3** 设  $f$  是  $N$  的一个可行流,  $f'$  是增量网络  $N(f)$  的一个可行流, 则可得  $N$  中一个新的可行流  $f^*$ : 对  $\forall \langle u, v \rangle \in A$ ,

$$f^*(u, v) = f(u, v) + f'(u, v) - f'(v, u),$$

称  $f^*$  是  $f'$  在  $f$  上的叠加, 记为  $f \dot{+} f'$ .

**引理 9.6.1** 设  $f$  是  $N$  的可行流,  $f'$  是增量网络  $N(f) = (V, x, y, A(f), C', w')$  的一个可行流, 则

$$\text{Val}(f \dot{+} f') = \text{Val} f + \text{Val} f',$$

$$w(f \dot{+} f') = w(f) + w'(f').$$

**证明** 令  $f^* = f \dot{+} f'$ . 对  $N$  的任何割  $(S, \bar{S})$ ,

$$\begin{aligned} \text{Val } f^* &= f^{*+}(S) - f^{*-}(S) = \sum_{\langle u,v \rangle \in (S, \bar{S})} f^*(u,v) - \sum_{\langle u,v \rangle \in (\bar{S}, S)} f^*(u,v) \\ &= \sum_{\langle u,v \rangle \in (S, \bar{S})} (f(u,v) + f'(u,v) - f'(v,u)) - \\ &\quad \sum_{\langle u,v \rangle \in (\bar{S}, S)} (f(u,v) + f'(u,v) - f'(v,u)) \end{aligned}$$

$$\begin{aligned} &= \left( \sum_{\langle u,v \rangle \in (S, \bar{S})} f(u,v) - \sum_{\langle u,v \rangle \in (\bar{S}, S)} f(u,v) \right) \\ &+ \left( \sum_{\langle u,v \rangle \in (S, \bar{S})} f'(u,v) + \sum_{\langle u,v \rangle \in (\bar{S}, S)} f'(v,u) \right) \\ &- \left( \sum_{\langle u,v \rangle \in (S, \bar{S})} f'(v,u) + \sum_{\langle u,v \rangle \in (\bar{S}, S)} f'(u,v) \right) \\ &= \text{Val } f + \text{Val } f'^+(S) - \text{Val } f'^-(S) = \text{Val } f + \text{Val } f'. \end{aligned}$$

零流弧  
非零非饱和弧  
饱和弧  
零流弧  
非零非饱和弧  
饱和弧

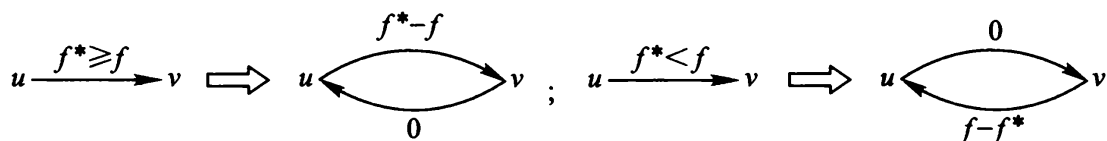
$$\begin{aligned} w(f^*) &= \sum_{\langle u,v \rangle \in A} f^*(u,v) \cdot w(u,v) \\ &= \sum_{\langle u,v \rangle \in A} (f(u,v) + f'(u,v) - f'(v,u)) \cdot w(u,v) \\ &= \sum_{\langle u,v \rangle \in A} f(u,v) \cdot w(u,v) + \sum_{\langle u,v \rangle \in A} f'(u,v) \cdot w(u,v) + \\ &\quad \sum_{\langle u,v \rangle \in A} f'(v,u) \cdot (-w(u,v)) \\ &= w(f) + w'(f'). \quad \text{证毕.} \end{aligned}$$

**引理 9.6.2** 设  $f$  和  $f^*$  是  $N$  中两个可行流, 则增量网络  $N(f) = (V, x, y, A(f), C', w')$  中必存在可行流  $f'$ , 满足  $f^* = f \dot{+} f'$ .

**证明** 利用  $f$  和  $f^*$  来定义  $N(f)$  中一个流  $f'$  如下: 对  $\forall \langle u,v \rangle \in A$ ,  
若  $f^*(u,v) \geq f(u,v)$ , 则定义  $f'(u,v) = f^*(u,v) - f(u,v)$ , 且  $f'(v,u) = 0$ ;  
若  $f^*(u,v) < f(u,v)$ , 则定义  $f'(u,v) = 0$  且  $f'(v,u) = f(u,v) - f^*(u,v)$ .

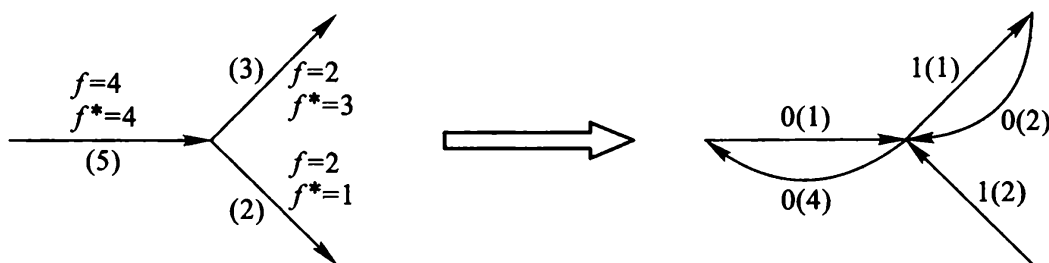
特别地, 对  $N$  中一条弧  $\langle u,v \rangle$ , 如果  $f(u,v) = 0$ , 则由  $N(f)$  的构造可知, 在  $N(f)$  中只有弧  $\langle u,v \rangle$  而没有弧  $\langle v,u \rangle$ , 此时, 只定义弧  $\langle u,v \rangle$  上的流值:  $f'(u,v) = f^*(u,v)$ . 同

理,如果  $f(u, v) = c(u, v)$ , 则  $N(f)$  中只有弧  $\langle v, u \rangle$  而无弧  $\langle u, v \rangle$ , 此时只定义  $\langle v, u \rangle$  上的流值  $f'(u, v) = f(u, v) - f^*(u, v)$ .



易验证这样定义的  $f'$  是  $N(f)$  中的一个可行流, 且  $f^* = f + f'$ . 证毕.

例 9.6.2 下图是由  $N$  中两个可行流  $f$  和  $f^*$  来构造  $N(f)$  中一个流  $f'$  的示意图.



$N$  中两个可行流  $f$  和  $f^*$ , 括号中是弧容量

$N(f)$  及其一个可行流  $f'$ , 括号中是弧容量

定义 9.6.4 设  $f$  是网络  $N$  的一个可行流, 但不是零值流, 如果  $N$  中存在一个有向圈  $C$ , 使得对  $\forall \langle u, v \rangle \in A - A(C)$ , 都有  $f(u, v) = 0$ , 则称  $f$  是  $N$  的关于圈  $C$  的圈流, 记为  $f_C$ .

定义 9.6.5 设  $f_1$  与  $f_2$  都是网络  $N$  中的可行流, 定义  $f_1$  与  $f_2$  的和  $f$  为:

$$\text{对 } \forall \langle u, v \rangle \in A, \quad f(u, v) = f_1(u, v) + f_2(u, v).$$

引理 9.6.3 设  $f$  是网络  $N$  上一个流量为 0 的可行流, 但  $f$  不是零值流, 则  $f$  可表示为  $N$  中若干个圈流之和.

证明 首先注意, 网络中流量为 0 的可行流称为零流, 而零值流是指每条弧上流值都为 0 的流. 令  $A_1 = \{\langle u, v \rangle \in A \mid f(u, v) > 0\}$ .

因  $f$  不是零值流, 故  $A_1 \neq \emptyset$ , 设  $N_1$  是  $N$  的由弧集  $A_1$  导出的子有向图, 则  $N$  的源、汇点  $\notin N_1$ , 且

$$\{f(u, v) \mid \langle u, v \rangle \in A_1\}$$

是  $N_1$  的一个可行流, 即在每条弧上满足容量约束且在  $N_1$  上每个顶点处都满足流量守恒条件. 故  $N_1$  中每点至少有一条出弧. 由定理 8.2.3 知,  $N_1$  有一个有向圈  $C_1$ . 令

$$\delta = \min\{f(u, v) \mid \langle u, v \rangle \in A(C_1)\},$$

则可定义  $N$  中一个圈流  $f_{C_1}$  和一个新流  $f_1$  如下:

$$f_{C_1}(u, v) = \begin{cases} \delta & \langle u, v \rangle \in A(C_1), \\ 0, & \langle u, v \rangle \notin A(C_1), \end{cases}$$

$$f_1(u, v) = \begin{cases} f(u, v) - \delta, & \langle u, v \rangle \in A(C_1), \\ f(u, v), & \langle u, v \rangle \notin A(C_1). \end{cases}$$

显然  $f = f_{C_1} + f_1$ , 且  $f_1$  至少比  $f$  多一条流值为 0 的弧.

再考虑  $A_2 = \{\langle u, v \rangle \in A \mid f_1(u, v) > 0\}$  在  $N$  中导出的子有向图  $N_2$ , 重复上述过程, 知  $N_2$  中含有有向圈  $C_2$ , 且可定义出  $N$  中一个圈流  $f_{C_2}$  和一个新流  $f_2$ , 使得  $f_1 = f_{C_2} + f_2$ , 且  $f_2$  又比  $f_1$  至少多一条流值为 0 的弧. 而  $f = f_{C_1} + f_{C_2} + f_1$ .

如此继续, 经有限步后, 必须得到  $f_{k-1} = f_{C_k} + f_k$ , 且使得  $f_k$  是一个零值流 (其中  $f_{C_k}$  是一个圈流). 因此  $f = f_{C_1} + f_{C_2} + \cdots + f_{C_k}$ . 证毕.

**定义 9.6.6** 设  $f$  是网络  $N$  的一个可行流,  $C$  是增量网络  $N(f) = (V, x, y, A', C', w')$  中的一个有向圈, 如果

$$w'(C) = \sum_{\langle u, v \rangle \in A'(C)} w'(u, v) < 0,$$

则称  $C$  是  $N(f)$  的负费用有向圈.

**定理 9.6.1** 设  $f$  是  $N$  中流量为  $v_0$  的可行流, 则  $f$  是  $N$  中的最小费用  $v_0$  流当且仅当增量网络  $N(f) = (V, x, y, A', C', w')$  中没有负费用有向圈.

**证明 必要性** 用反证法. 假设  $f$  是  $N$  中最小费用  $v_0$  流, 但  $N(f)$  中却有负费用有向圈  $C$ , 即

$$\sum_{\langle u, v \rangle \in A'(C)} w'(u, v) < 0.$$

令  $\delta = \min\{c'(u, v) \mid \langle u, v \rangle \in A'(C)\}$ , 则  $\delta > 0$ . 在  $N(f)$  上定义一个圈流  $f_C$  如下:

$$f_C(u, v) = \begin{cases} \delta, & \langle u, v \rangle \in A'(C), \\ 0, & \langle u, v \rangle \notin A'(C), \end{cases}$$

于是

$$w'(f_C) = \sum_{\langle u, v \rangle \in A'(C)} (\delta \cdot w'(u, v)) = \delta \cdot \left( \sum_{\langle u, v \rangle \in A'(C)} w'(u, v) \right) < 0.$$

令  $f^* = f + f_C$ , 由引理 9.6.1 知

$$\text{Val}f^* = \text{Val}f + \text{Val}f_C = \text{Val}f = v_0.$$

$$w(f^*) = w(f) + w'(f_C) < w(f).$$

可见  $f$  不是  $N$  的费用最小的  $v_0$  流, 矛盾.

**充分性** 设  $N(f)$  不含负费用圈, 对  $N$  中任一流量为  $v_0$  的可行流  $f^*$ , 由引理 9.6.2,  $N(f)$  中存在可行流  $f'$  使得  $f^* = f + f'$ ,

(1) 如果  $f'$  是  $N(f)$  中的零值流, 则  $f^* = f, w(f^*) = w(f)$ .

(2) 如果  $f'$  不是  $N(f)$  中的零值流, 则由引理 9.6.1 知

$$\text{Val}f' = \text{Val}f^* - \text{Val}f = v_0 - v_0 = 0,$$

从而由引理 9.6.3,  $f' = f_{C_1} + f_{C_2} + \cdots + f_{C_k}$ , 其中  $f_{C_i}$  为  $N(f)$  中的圈流 ( $i = 1, 2, \cdots, k$ ), 故

$$w'(f') = \sum_{i=1}^k w'(f_{C_i}).$$

因  $N(f)$  不含负费用圈, 故  $w'(f_{C_i}) \geq 0$  ( $i = 1, 2, \cdots, k$ ), 故由引理 9.6.1,

$$w(f^*) = w(f) + w'(f') \geq w(f),$$

由 (1)、(2) 知,  $f$  为  $N$  中的最小费用  $v_0$  流. 证毕.

由定理 9.6.1 可得一个求最小费用  $v_0$  流的负费用圈算法. 其基本思想为: 先利用最大流算法找可增路反复增流, 求出  $N$  中一个流量为  $v_0$  的可行流. (为防止增流超过  $v_0$ , 增流量应取  $\delta = \min\{\Delta f(P), v_0 - \text{Val}f\}$ . 如果最大流算法在求得  $N$  的最大流时流量仍未达到  $v_0$ , 则表明  $N$  中不存在  $v_0$  可行流). 设求得  $v_0$  可行流  $f$ , 然后在增量网络  $N(f)$  中找负费用圈, 并按定理 9.6.1 的方法构造圈流  $f_C$ , 由  $f^* = f + f_C$  获得费用更小的  $v_0$  流. 在  $N(f^*)$  中重复上述过程. 如此反复, 使得所得流的流量保持为  $v_0$ , 而费用逐次降低, 直至新的增量网络中无负费用圈为止. 此时由定理 9.6.1 知, 当前的流便是最小费用  $v_0$  流.

#### 求最小费用 $v_0$ 流的负费用圈算法

输入: 费用网络  $N(V, x, y, A, C, w)$ , 流量值  $v_0$ .

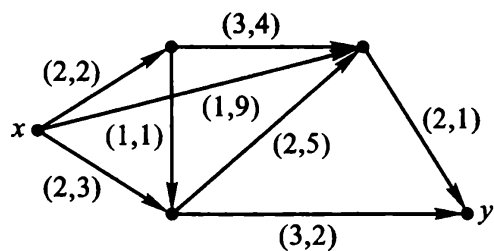
输出:  $N$  中流量为  $v_0$  的最小费用流.

第 1 步: 在  $N$  中运行最大流算法, 在算法中沿可增路  $P$  增流时取  $\delta = \min\{\Delta f(P), v_0 - \text{Val}f\}$ . 要么求出流量为  $v_0$  的可行流  $f$ , 转第 2 步; 要么判定  $N$  中不存在流量为  $v_0$  的可行流. 停止.

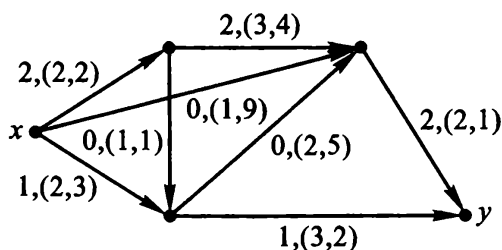
第 2 步: 构造增量网络  $N(f)$ , 在  $N(f)$  中找负费用有向圈. 如果不存在负费用有向圈, 则结束, 当前  $f$  为最小费用  $v_0$  流; 否则, 找到一个负费用有向圈  $C$ , 转第 3 步.

第 3 步: 求  $C$  上的最小容量  $\delta$ , 构造  $N(f)$  的圈流  $f_C$ , 令  $f := f + f_C$ , 转第 2 步.

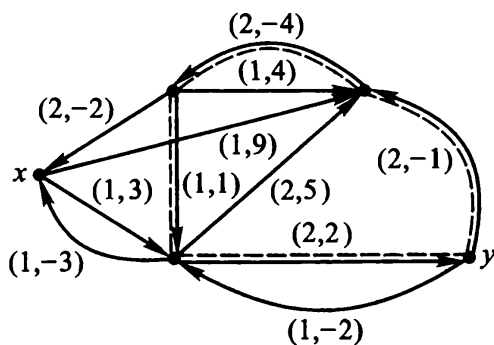
**例 9.6.3** 求下列网络中从  $x$  到  $y$  的流量为 3 的最小费用流.

网络  $N$ , 每条弧上的二元数组表示  $(c, w)$ .

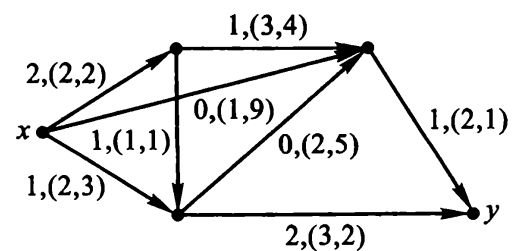
解 (1) 用最大流算法求流量为 3 的可行流  $f_1$ :

网络  $N$  的一个可行流  $f$ , 每条弧上的三元数组分别表示  $f, (c, w)$ 

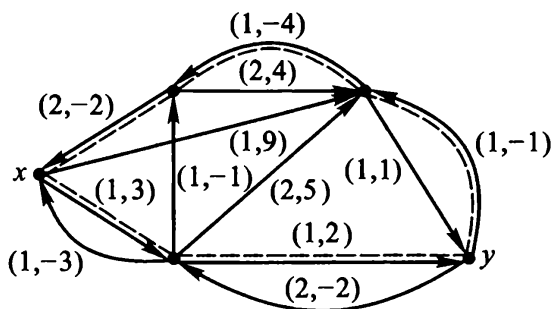
(2) 构造增量网络  $N(f_1)$ , 并找  $N(f_1)$  中的负费用有向圈.

增量网络  $N(f_1)$  及其一个负费用圈(虚线所示), 每条弧上的二元数组表示  $(c', w')$ .

(3) 沿负费用圈修改流, 得  $N$  中流量为 3 的新流  $f_2$ .

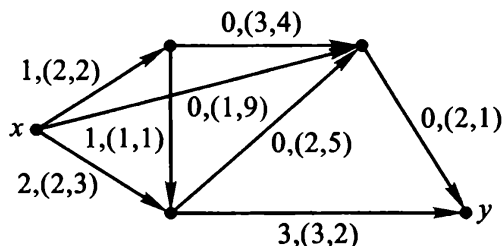


(4) 构造增量网络  $N(f_2)$ , 并找  $N(f_2)$  中的负费用有向圈.

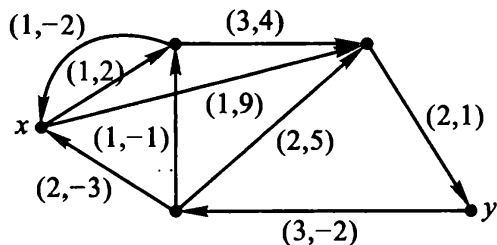


增量网络  $N(f_2)$  及其一个负费用圈(虚线所示), 每条弧上的二元数组表示  $(c', w')$ .

(5) 沿负费用圈修改流, 得流量为 3 的新流  $f_3$ .



(6) 构造增量网络  $N(f_3)$ , 并找  $N(f_3)$  中的负费用有向圈.



增量网络  $N(f_3)$ , 每条弧上的二元数组表示  $(c', w')$ . 其中无负费用圈

因  $N(f_3)$  中已无负费用有向圈, 故  $f_3$  是流量为 3 的最小费用流.

我们来分析负费用圈算法的计算复杂度. 第 1 步可使用 Dinic 最大流算法求  $N$  中流量为  $v_0$  的可行流, 计算复杂度为  $O(v^2\varepsilon)$ . 第 2 步的主要计算量在于找  $N(f)$  的负费用有向圈, 这可使用 Floyd-Warshall 最短路算法<sup>[5][18][19]</sup>, 其计算复杂度为  $O(v^3)$ . 设  $c_{\max}$  和  $w_{\max}$  分别表示网络  $N$  中所有弧上容量的最大值和单位费用的最大值, 则  $N$  中任何可行流的费用都不会超过  $\varepsilon \cdot c_{\max} \cdot w_{\max}$ . 由于每次执行第 2 步和第 3 步, 都是找一个负费用有向圈并沿该圈增流而使可行流的费用降低至少一个单位, 因此第 2 步和第 3 步的循环至多进行  $\varepsilon \cdot c_{\max} \cdot w_{\max}$  次. 从而算法总的计算复杂度为  $O(v^2\varepsilon + v^3 \cdot \varepsilon \cdot c_{\max} \cdot w_{\max}) = O(v^3 \cdot \varepsilon \cdot c_{\max} \cdot w_{\max})$ . 可见负费用圈算法是一个伪多项式算法, 而不是多项式时间算法. 但文献 [20] 已证明, 只要在迭代过程中适当地选择负费用有向圈, 便可将该算法改造成为多项式时间算法. 此外, 如果网络  $N$  中的弧容



量全为整数, 且  $v_0$  是非负整数, 则负费用有向圈算法可在有限步结束. 但如果弧容量为无理数, 则该算法未必能在有限步结束<sup>[21]</sup>.

## 二、求最小费用流的最小费用路算法

本节沿用前几节的所有记号. 在上节负费用圈算法中, 我们总是保持可行流的流量等于给定的值  $v_0$ , 而通过寻找增量网络中的负费用有向圈来逐步降低可行流的费用, 最终得到费用最小的  $v_0$  流. 现在我们从另一个角度来考虑: 从一个流量较小的最小费用可行流 (比如零值流) 开始, 设法逐步增大流量使其达到给定的值  $v_0$ , 在扩流过程中始终保持所得可行流的费用在等流量可行流中的最小性. 如何才能做到流量增加后所得可行流仍然是新流量下的最小费用流呢? 一个自然的想法是增流时增加的单位费用应该尽可能小. 为达到此目的, 人们很容易想到沿着增量网络  $N(f)$  中的最小费用  $x$ - $y$  有向路来增流. 这种想法的正确性由下述定理来保证.

**定理 9.6.2** 设  $f$  是网络  $N = (V, x, y, A, C, w)$  上的一个最小费用可行流, 其流量为  $v(f)$ ,  $P$  是增量网络  $N(f)$  中一条最小费用  $x$ - $y$  有向路,  $\delta$  是  $N(f)$  中  $P$  上最小的弧容量. 若  $\delta > 0$ , 且  $\theta$  是满足  $0 < \theta \leq \delta$  的一个常数, 则沿  $P$  对  $f$  增流  $\theta$  后所得的流  $f'$  是  $N$  的流量为  $v(f) + \theta$  的最小费用流. 其中

$$f'(a) = \begin{cases} f(a) + \theta, & \text{若 } a \text{ 是 } P \text{ 的正向弧,} \\ f(a) - \theta, & \text{若 } a \text{ 是 } P \text{ 的反向弧,} \\ f(a), & \text{若 } a \text{ 不在 } P \text{ 上} \end{cases} \quad (\forall a \in A).$$

**证明** 由定理条件及 §9.2 易知,  $f'$  是  $N$  的流量为  $V(f) + \theta$  的可行流. 下面用反证法证明  $f'$  是  $N$  的费用最小的  $V(f) + \theta$  流.

假设  $f'$  不是  $N$  的最小费用流, 则由定理 9.6.1, 增量网络  $N(f')$  中有负费用有向圈, 取其一记为  $C$ . 注意增量网络  $N(f')$  可看作在  $N(f)$  的基础上沿最小费用  $x$ - $y$  有向路  $P$  增添一些 (相对于  $P$  的) 反向弧、并且删除一些 (相对于  $P$  的) 正向弧得到的. 因  $f$  是网络  $N$  的一个最小费用流, 由定理 9.6.1,  $N(f)$  中没有负费用圈. 因此  $C$  上的弧不可能都在  $N(f)$  中出现. 设  $C$  上有  $k$  条弧  $(v_{i_1}, v_{j_1}), (v_{i_2}, v_{j_2}), \dots, (v_{i_k}, v_{j_k})$  不在  $N(f)$  中, 则它们必是沿  $x$ - $y$  有向路  $P$  增添的那些反向弧中的一些, 因此在  $x$ - $y$  有向路  $P$  (它在  $N(f)$  中) 上有与它们对应的方向相反的弧  $(v_{j_1}, v_{i_1}), (v_{j_2}, v_{i_2}), \dots, (v_{j_k}, v_{i_k})$ . 这  $2k$  条弧组成的集合记为  $A_1$ . 构造一个新的网络  $N_1$  如下:  $V(N_1) = V(C) \cup V(P)$ , 而弧集  $A(N_1)$  由  $C$  上的弧和  $P$  上的弧合并组成 (不是做集合的并, 因此可能出现并行弧), 则  $A_1$  中的  $2k$  条弧全在  $N_1$  中. 明显地, 在  $N_1$  中,  $x$  点的出弧比入弧多一条,  $y$  点的入弧比出弧多一条, 其他点处出弧数等于入弧数. 从  $N_1$  中去掉  $A_1$  中的  $2k$  条弧以及由此出现的孤立点, 得到的有向图记为  $\vec{G}$ , 则  $\vec{G}$  的顶点仍然满足上述性质且  $\vec{G}$  的弧全在  $N(f)$  中. 由定理 8.4.1 ~ 定理 8.4.3 易知,  $\vec{G}$  可分解为  $N(f)$  中一条  $x$ - $y$  有向路

$P'$  与若干有向圈的并. 设  $\vec{G} = P' \cup C_1 \cup C_2 \cup \cdots \cup C_r$ . 由于这些圈  $C_i (i = 1, 2, \cdots, r)$  是  $N(f)$  中的圈, 故每个圈上各弧费用之和均非负. 为方便起见, 我们用  $w(\vec{G})$ 、 $w(A)$  等符号表示所涉及的网络或弧集中各弧的费用之和, 则上述结论可表述为  $w(C_i) \geq 0 (i = 1, 2, \cdots, r)$ . 此外, 按照增量网络的构造, 一对方向相反的弧上的费用互为相反数, 故  $w(A_1) = 0$ . 从而

$$w(N_1) = w(\vec{G}) + w(A_1) = w(\vec{G}) = w(P') + \sum_{i=1}^r w(C_i) \geq w(P').$$

另一方面, 因圈  $C$  是负费用圈, 故  $w(C) < 0$ . 从而

$$w(N_1) = w(P) + w(C) < w(P).$$

由此可得,  $w(P') < w(P)$ , 这与  $P$  是  $N(f)$  中的最小费用  $x-y$  有向路矛盾. 这个矛盾说明反证法假设不真. 证毕.

许多人分别独立地得到上述定理, 包括 W. S. Jewell<sup>[22]</sup>, P. G. Busacker, P. J. Gowen<sup>[23]</sup>, M. Iri<sup>[24]</sup>. 我们这里的表述来自 Ford 和 Fulkerson<sup>[3]</sup>. 由这一定理可得求最小费用流的最小费用路算法. 基本思想是:

先从流量较小的最小费用流 (比如零值流) 开始, 构造增量网络  $N(f)$ , 并求  $N(f)$  中的最小费用  $x-y$  路  $P$ , 沿  $P$  对  $f$  增流. (增流时, 为防止新流的流量超过  $v_0$ , 沿  $P$  的增流量应取为  $y = \min\{\Delta f(P), v_0 - \text{Val}f\}$ ). 反复进行上述操作, 逐步增大流量, 并保持所得可行流是同流量的可行流中费用最小的, 直至获得流量为  $v_0$  的最小费用流. 如果在得到  $v_0$  流之前, 某个流的增量网络中已无  $x-y$  路, 则表明  $N$  中不存在 (最小费用)  $v_0$  流.

### 求最小费用流的最小费用路算法

输入: 费用网络  $N(V, x, y, A, C, w)$ , 流量值  $v_0$ .

输出:  $N$  中流量为  $v_0$  的最小费用流.

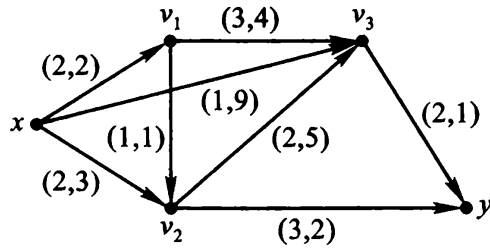
第 0 步: 取零值流作为初始可行流  $f$ .

第 1 步: 若  $\text{Val} f = v_0$ , 则停止, 当前的  $f$  即为所求的最小费用流, 输出  $f$ ; 否则, 转下步.

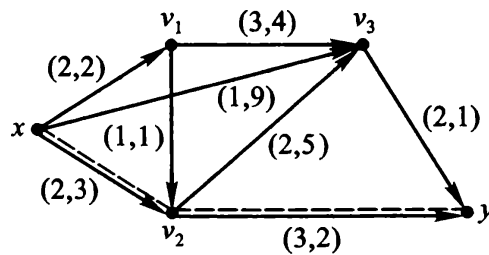
第 2 步: 构造增量网络  $N(f)$ . 若  $N(f)$  中不存在  $x-y$  有向路, 则  $N$  中无流值为  $v_0$  的可行流, 停止; 否则, 找出  $N(f)$  的一条最小费用  $x-y$  有向路  $P$ , 转下步.

第 3 步: 用  $c(P)$  表示  $P$  上最小的弧容量, 令  $\theta = \min\{c(p), v_0 - \text{Val}f\}$ , 在  $N$  中沿  $P$  对  $f$  增流  $\theta$ , 得新流  $f$ , 转第 1 步.

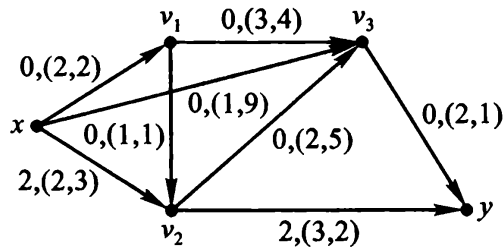
作为例子我们运用最小费用路算法来求上一小节例 9.6.3 网络中流量为 3 的最小费用流 (网络如下).

网络  $N$ , 每条弧上的二元数组表示  $(c, w)$ .

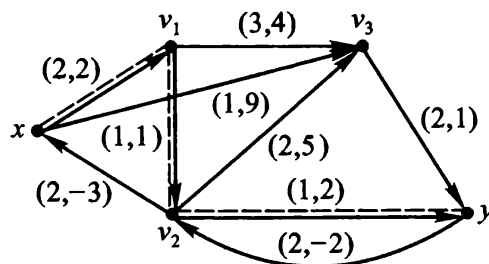
从零值流  $f_0$  开始, 按算法第 1 步, 构造增量网络  $N(f_0)$ , 仍为  $N$ . 由算法第 2 步, 求出  $N(f_0)$  的一条最小费用有向路  $xv_2y$ .

增量网络  $N(f_0)$  及其一条最小费用路.  
每条弧上的二元数组表示  $(c, w)$ .

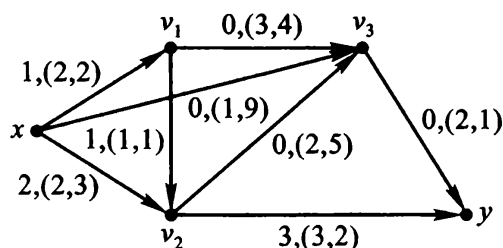
按算法第 3 步, 沿最小费用路增流 2, 得到新流  $f$ .

沿最小费用路增流后得到的新流  $f$ . 每条弧上的三元数组分别表示  $f, (c, w)$ .

构造  $f$  的增量网络  $N(f)$ , 并求出其中一条最小费用路  $xv_1v_2y$ .

增量网络  $N(f)$  及其一条最小费用路.

沿最小费用路增流 1, 得到流量为 3 的最小费用流  $f^*$ . 算法结束.



沿最小费用路增流后得到的流量为3  
的最小费用流.

在最小费用路算法的过程中, 每次增流至少增 1 个单位, 因而经过至多  $v_0$  次找最小费用有向路并增流, 要么可判定网络  $N$  中没有流量为  $v_0$  的可行流, 要么获得最小费用  $v_0$  流. 而每次求最小费用有向路可运用最短路算法, 所花费的计算量不超过  $O(\nu^3)$ . 此外, 第 3 步中沿可增路增流的过程计算量不超过  $O(\nu)$ . 因此最小费用路算法的计算复杂度为  $O((\nu^3 + \nu) \cdot v_0) = O(\nu^3 v_0)$ .

本节讨论的最小费用流问题中, 弧上的费用都是指单位流量的费用, 即认为费用随流量的增加是线性的. 如果某些弧上的费用是流量的非线性函数, 则相应的最小费用流问题更为复杂. 当费用是流量的凸函数时, 最小费用流问题有类似于本节的负费用圈算法和最小费用路算法. 对这方面有兴趣的读者可参阅文献 [5]、[11] 或 [173]、[176]~[179]. 此外, 在许多情况下需要寻求费用网络的最小费用循环流, 这一问题已得到较好的研究, 相应的算法有状态法 (out-of-kilter method) 等, 读者可参阅文献 [5]、[11] 或 [25]~[28].

网络流问题在工程技术、计算机科学与技术、电子科学与技术、通信网络技术、工业制造、交通运输、管理科学、区域规划、物流安排、财经分析等领域有广泛的应用<sup>[11]</sup>. 从 20 世纪 50 年代以来, 经过众多学者大量的研究积累, 网络流方法已形成了较为成熟的理论体系. 一大批专著和综述已经问世<sup>[3][11][21][29]~[47]</sup>.

经典的网络流理论主要由最大流、最小费用流、多品种流等内容组成. 最大流问题早期的最著名算法是由 Ford 和 Fulkerson 提出的标号算法<sup>[1][2]</sup>, 这种算法由 Edmonds、Karp 及 Dinic 等人加以修改后成为多项式时间算法<sup>[7][8]</sup>. 稍后, Karzanov 及 Malhotra 等人提出了更快的算法——预流推进算法 (preflow-push algorithm)<sup>[5][10]~[12]</sup>. 文献 [48]~[51] 研究了求最大流的变尺度方法 (scaling algorithm). 一些随机化方法也已得到研究<sup>[52]~[54]</sup>. 对上述这些算法的后续改进见文献 [13]~[16] 及 [55]~[57]. 对于在平面图网络中求最大流, 较快的算法见 [58]~[60]. 有关最大流问题的新进展可参看 [61]~[72].

关于最小费用流的几类经典算法是负费用圈法 (Klein 算法)<sup>[17][20][21][73]~[76]</sup>、状态法 (out-of-kilter 方法)<sup>[5][11][25]~[28]</sup>、最短路法<sup>[22]~[24][77]</sup>、松弛法<sup>[78][79]</sup>、变尺度方法 (scaling algorithm)<sup>[7][80]~[88]</sup> 等. 求解最小费用流的多项式时间算法首先由 Tardos 得到<sup>[89]</sup>, 此后, 更多的多项式时间算法被提出<sup>[21][74][84]~[87][89]~[94]</sup>, 特别是在网络单纯型

法算法方面得到了丰富的研究成果<sup>[78][86][95]~[112]</sup>.

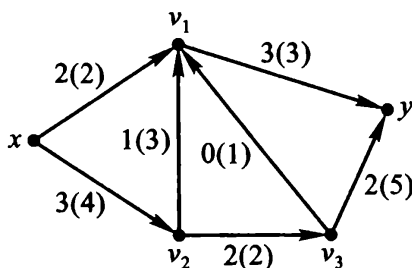
多品种流问题是在一个网络中安排多个不同品种的流, 它们互相不能融合但要共享网络的弧, 需要在一定的要求下, 合理地安排这些流, 以使流量最大或费用最小. 许多实际问题可以归结为多品种流问题<sup>[11][113]~[120]</sup>, 而且, 一些 *NP-hard* 问题可以利用多品种流算法得到近似解<sup>[121][122]</sup>. 多品种流问题在 20 世纪 70 年代以来得到了大量的研究<sup>[98][123]~[134]</sup>. 有关综述见 [11][29][33][34][118][119][134]~[136]. 多品种流问题的大多数精确算法都是基于线性规划思想, 用原始单纯型或原始对偶单纯型方法进行分析<sup>[4][11][98][135]~[137]</sup>. 另一方面, 多品种流问题近似算法的研究取得了较好的进展<sup>[122][124][138]~[144]</sup>, 一些近似比为  $(1 + \epsilon)$  的近似算法被提出<sup>[94][123]</sup>. 最近, 内点算法 (interior-point) 也已受到关注<sup>[144]~[150]</sup>. 关于多品种流的更多文献见 [151]~[159].

近年来, 网络技术的迅猛发展促使网络流研究朝着新的方向拓展. 动态网络流<sup>[160][161]</sup>、多目标网络流<sup>[162]~[172]</sup>、非线性网络流<sup>[173]~[181]</sup>、广义网络流 (generalized network flows)<sup>[182]~[195]</sup>、子模流 (submodular flow)<sup>[196]~[200]</sup> 等方向的研究已经取得了一定进展. 关于网络流的更多文献见 [201]~[215].

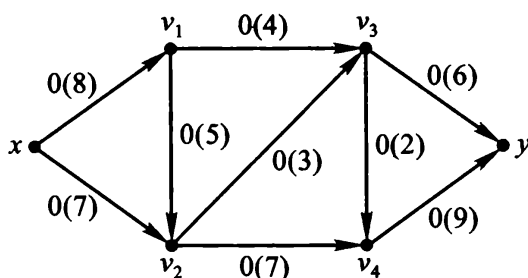
## 习题九

9.1 在下面的网络中, 弧上第一个数字表示流量, 第二个数字表示容量.

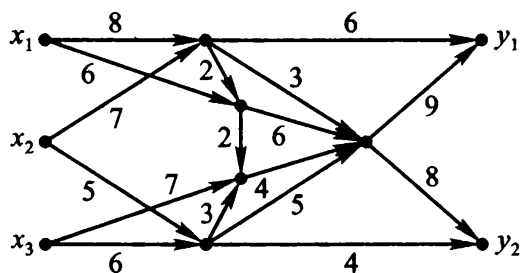
- (1) 确定所有的割;
- (2) 求最小割的容量;
- (3) 证明给定流是最大流.



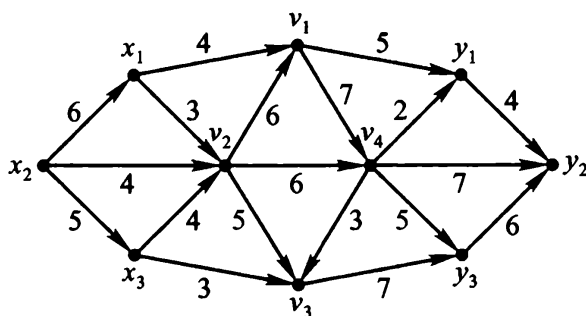
9.2 对如下网络, 从零值流开始, 分别用 Ford-Fulkson 标号算法和 Dinic 算法求出其最大流, 写出或画出算法过程. (注: 括号内数字表示弧的容量, 括号外数字表示当前流值.)



- 9.3 若  $(S, \bar{S})$  和  $(T, \bar{T})$  都是  $N$  中的最小割, 证明  $(S \cup T, \overline{S \cup T})$  及  $(S \cap T, \overline{S \cap T})$  也都是  $N$  的最小割.
- 9.4 设  $(S, \bar{S})$  是网络  $N(V, x, y, A, C)$  的一个最小割.
- (1) 证明或反驳: 若  $f_1$  和  $f_2$  是  $N$  的两个最大流, 则在  $(S, \bar{S})$  和  $(\bar{S}, S)$  的每条弧上,  $f_1$  和  $f_2$  的流值都相等;
  - (2) 证明或反驳: 若在  $(S, \bar{S})$  和  $(\bar{S}, S)$  的每条弧上,  $N$  的两个可行流  $f_1$  和  $f_2$  的流值都相等, 则  $f_1$  和  $f_2$  都是  $N$  的最大流.
- 9.5 设在网络  $N(V, x, y, A, C)$  中, 每条弧的容量都为 1, 证明:
- (1)  $N$  中最大流的流量等于  $N$  中无公共弧的  $x$ - $y$  路的最大数目;
  - (2)  $N$  中最小割的容量等于断开所有  $x$ - $y$  路所必须删去的弧的最小数目.
- 9.6 证明定理 9.3.1: 设  $f$  是网络  $N$  中的一个可行流,  $N(f)$  是  $N$  关于流  $f$  的增量网络, 则
- (1)  $N$  中  $f$  可增路与  $N(f)$  中的  $x$ - $y$  有向路一一对应;
  - (2)  $N$  中  $f$  可增路  $P$  的可增流值  $\Delta f(P)$  等于  $N(f)$  中对应的  $x$ - $y$  有向路  $\vec{P}$  上最小的弧容量.
- 9.7 假设给定了一个从甲地到乙地的单行公路网 (由若干公路段错综复杂地连接而成), 其中每条路段上单位时间内 (例如一天) 能够通过的最多车辆数已经知道. 为了保证公路网的畅通, 交通管理部门对每条路段在单位时间内通过的车辆数目设定了上限. 现在交通管理部门想知道, 在给定的上限限制下, 这个公路网到底在单位时间内最多能通过多少车辆. 请你帮助交通管理部门建立这个问题的网络流模型, 并描述一种求解算法.
- 9.8 如何求多源多汇网络中的最大流?
- 9.9 设计一个在上下容量网络中求最小流的算法.
- 9.10 试将求可行循环流问题转化为在供需网络中求供需可行流问题.
- 9.11 设在网络  $N(V, x, y, A, C)$  中, 每个顶点都有通过量限制, 即对  $\forall v_i \in V$ , 都对应有一个非负实数  $t(v_i)$ ,  $N$  中可行流  $f$  除了要满足弧容量约束和流量守恒外, 还要满足  $\sum_{v \in N^-(v_i)} f(v, v_i) \leq t(v_i)$ . 如何在这种约束下求流值最大的可行流?
- 9.12 不同地点的三家工厂  $x_1, x_2, x_3$  生产某种商品, 通过如下网络运送到两个销售地  $y_1, y_2$ , 弧上的数字是相应线路上的运送能力. 求工厂到销售地一次能运送该商品的 最大数量, 并给出相应的运输方案.

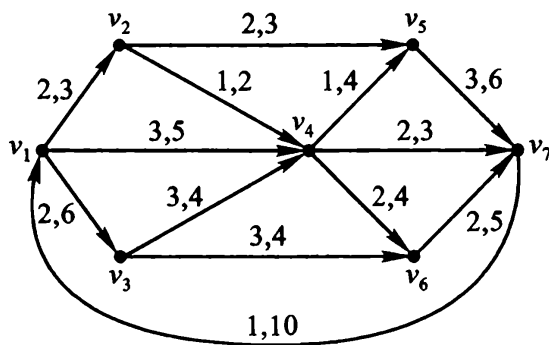


9.13 如图, 一种商品有三个产地  $x_1, x_2, x_3$ , 产量分别是 3, 10, 6 个单位;  $y_1, y_2, y_3$  是需求地, 分别需要这种商品至少 6, 9, 4 个单位. 问  $y_1, y_2, y_3$  的需求是否都可得到满足? 弧上的数字是相应线路上的运送能力.



9.14 在下图所示的上下容量网络中, 弧上的第一个数字表示下容量, 第二个数字表示上容量.

(1) 求该网络中一个可行循环流; (2) 求从  $v_1$  到  $v_7$  的最大流和最小流.

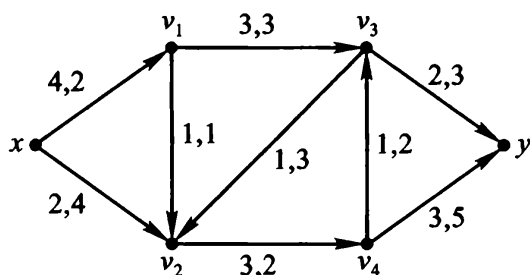


9.15 设有甲、乙、丙、丁四个人, 甲擅长拉小提琴、大提琴和弹吉它, 乙擅长拉大提琴和弹钢琴, 丙擅长拉小提琴和大提琴, 丁会弹吉它和钢琴. 现在这四人要同台演出, 每人演奏自己擅长的一种乐器, 四个人演奏的乐器不能相同. 试用网络流方法为他们设计一种乐器配置方案. 满足要求的乐器配置方案共有几种?

9.16 某单位的女士们将要举办一次别开生面的三八妇女节聚会. 每个人会做若干种食品 (各人会做的数量和品种可能不尽相同), 到时候她们会把自己最拿手的不超过  $k$  样食品各做一盘带到聚会上, 和其他同胞一起分享. 但考虑到同种食品带得太多了会因为吃不完而浪费, 大家给每种食品的总盘数都分别规定了一个上限值. 但这样让她们很伤脑筋; 究竟应该怎样安排每个人带的食品, 才能让聚

会上的食品总盘数尽量多呢? 请建立该问题的网络流模型, 并描述求解思路.

- 9.17 在单源单汇无向网络中如何求最大流和最小费用流?
- 9.18 设有一个给定的网络  $N(V, x, y, A, C, w)$ , 其中  $C$  表示弧容量,  $w$  表示弧上的单位费用. 请设计一个算法, 用于求该网络中一个费用最小的最大流.
- 9.19 设有一个给定的网络  $N(V, x, y, A, C, w)$ , 其中  $C$  表示弧容量,  $w$  表示弧上的单位费用, 另外给定了一个常数  $w_0 \geq 0$ . 请设计一个算法, 用于求该网络中一个费用不超过  $w_0$  的最大流.
- 9.20 在下图所示的网络中, 弧上的第一个数字表示容量, 第二个数字表示单位费用.
- (1) 求从  $x$  到  $y$  的一个流量为 3 的最小费用流;
  - (2) 求该网络的最小费用最大流.



- 9.21 运输调度部门经常遇到这样的问题: 欲将一批货物从甲地经过一个单行公路网运到乙地, 公路网中不同的路段上单位重量的货物运费不等, 且每条路段运送货物的总量有限定. 令运输部门头痛的是, 如何安排这批货物的运输才能使得总运费最省? 请你为运输部门提供一个解决此类问题的模型和算法.
- 9.22 某机械制造厂与一家销售企业签订了一份供货合同, 规定须于当年四个季度分别向销售企业提供 10、25、25、30 台同一规格的机械. 已知该厂四个季度生产这种机械的生产能力及生产每台机械的成本如下表.

季度	生产能力	单位生产成本 (万元/台)
一	35	10.8
二	35	11.1
三	20	11.0
四	15	11.5

若生产的机械在当季度不交货, 则每台机械积压一个季度需储存、维护等费用 0.15 万元. 试求在确保完成合同的情况下, 该厂有关此合同费用最小的全年生产方案 (包括储存, 维护等费用). 试用网络流方法解决这一问题.



## 参考文献

- [1] L. R. Ford, D. R. Fulkerson, Maximal flow through a network, *Canadian Journal of Mathematics*, 8 (1956), 399-404.
- [2] L. R. Ford, D. R. Fulkerson, *Flows in Networks*, Princeton University Press, 1962.
- [3] L. R. Ford, D. R. Fulkerson, A simple algorithm for finding maximal network flows and an application to the Hitchcock problem. *Canadian Journal of Mathematics*, 9 (1957), 210-218.
- [4] C. H. Papadimitry and K. Steiglitz, *Combinatorial Optimization – Algorithms & Complexity*, Prentice-Hall, Englewood Cliffs, New Jersey, 1982.
- [5] 谢政. 网络算法与复杂性理论. 长沙: 国防科技大学出版社, 2003 年.
- [6] J. Edmonds, R. M. Karp, Theoretical improvement in algorithmic efficiency for network flow problems, *J. ACM(J. Assoc. Comput. Mach.)*, 19(2) (1972), 248-264.
- [7] 张先迪. 李正良. 图论及其应用, 北京: 高等教育出版社, 2005.
- [8] E.A. Dinic, Algorithm for solution of a problem of maximum flow in networks with power estimation, *Soviet Mathematics Doklady*, 11(1970), 1277-1280.
- [9] 王树禾. 图论. 北京: 科学出版社, 2004.
- [10] A. V. Karzanov, Determining the maximal flow in a network by the method of preflows, *Soviet Mathematics Doklady*, 15(1974) 434-437.
- [11] R. K. Ahuja, T. L. Magnanti, and J.B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Pearson Education, Inc., 1993. (影印本: 网络流: 理论、算法与应用. 北京: 机械工业出版社, 2005)
- [12] V.M. Malhotra, M. P. Kumar, S. N. Maheshwari, A  $O(|V|^3)$  algorithm for finding maximum flows in networks, *Inf. Proc. Letters*, 7(6) (1978), 277-278.
- [13] Y. Shiloach, U. Vishkin, An  $O(n^2 \log n)$  parallel max-flow algorithm, *Journal of Algorithms*, 3(1982), 128-146.
- [14] J. Cheriyan, S. N. Maheshwari, Analysis of preflow push algorithms for maximum network flow, *SIAM Journal on Computing*, 18(1989), 1057-1086.
- [15] A. V. Goldberg, R. E. Tarjan, A new approach to the maximum flow problem, *Journal of ACM*, 35(1988), 921-940.
- [16] B. V. Cherkassky, A. V. Goldberg, On implementing the push-relabel method for the maximum flow problem, *Algorithmica*, 19(4)(1997), 390-410.
- [17] M. Klein, A primal method for minimal cost flows with application to the assignment and transportation problems, *Management Science*, 14(3) (1967), 205-220.
- [18] R.W. Floyd, Algorithm 97- Shortest Path, *Comm., ACM*, 5(6) (1962) 345.
- [19] S. Warshall, A theorem on Boolean matrices, *J. ACM*, 9(1) (1962), 11-12.

- [20] A. V. Goldberg, R. E. Tarjan, Finding minimum cost circulations by canceling negative cycles, *Journal of ACM*, 36(1989), 873-886.
- [21] Katta G. Murty, *Network Programming*, Prentice-Hall, Englewood Cliffs, New York, 1992.
- [22] W. S. Jewell, Optimal flow throuht networks, Tech. Report 8, OR Center, MIT, Cambridge Mass, 1958.
- [23] R. G. Busacker, P. J. Gowan, A procedure for determining a family of minimum-cost network flow patterns, ORO Tech Report 15, John Hopkins University, 1961.
- [24] M. Iri, A new method of solving transportation-network problems, *Journal of the Operations Research Society of Japan*. 3(1960), 27-87.
- [25] D. R. Fulkerson, An out-of-kilter method for minimal cost flow problems, *J. SIAM*, 9(1) (1961), 18-27.
- [26] G.J. Minty, Monotone networks, *Proceedings of the Royal Society of London*, 257A, (1960), 194-212.
- [27] H. A. Aashtiani, T. L. Magnanti, Implementing primal-dual network flow algorithms, *Technical Report OR 055-76*, Operations Research Center, MIT, Cambridge, MA, 1976.
- [28] N. Zadeh, A simple alternative to the out-of Kilter algorithm, *Technical Report No.35*, Dept. of Operations Research, Stanford University, 1979.

#### 专著、一般性综述

- [29] T. C. Hu, Multicommodity network flows *Operations Research*, 11(1963), 344-360.
- [30] T. C. Hu, *Integer Programming and Network Flows*, Addison-Wesley, 1969.
- [31] S. Even, *Graph Algorithms*, Computer Science Press, 1979.
- [32] T. C. Hu, *Combinatorial Algorithms*, Addison-Wesley, 1982.
- [33] D. T. Phillips, A. Garcia-Diaz, *Fundamentals of Network Analysis*, Prentice-Hall, Englewood Cliffs, 1981.
- [34] J. R. Evans, Multicommodity network flows, in *Fundamentals of Network Analysis* (eds. D.T. Phillips and Alberto Garcia-Diaz), Prentice-Hall, 1981, pp386-424.
- [35] R. T. Rockafellar, *Network Flows and Monotropic Optimization*, Wiley Inter-Science, New York, 1984.
- [36] É. Tardos, A Strongly Polynomial Minimum Cost Circulation Algorithm, *Combinatorica*, 5(1985), 247-255.
- [37] 田丰、马仲蕃, 图与网络流理论, 北京: 科学出版社, 1987.
- [38] F. Glover, D. Klingman, N. Phillips, *Network Models in Optimization and their Applications in Practice*, Wiley, New York, 1992.
- [39] D. Bertsekas, *Network Optimization: Continuous and Discrete Models*, Athena Scientific, Belmont, 1998.

- [40] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, A. Schrijver, *Combinatorial Optimization*, Wiley, New York, 1998.
- [41] D. Jungnickel, *Graphs, Networks and Algorithms*, Springer, Berlin, 1999.
- [42] B. Korte, J. Vygen, *Combinatorial Optimization*, Springer, Berlin, 2000.
- [43] A. Asano, Y. Asano, *Recent developments in maximum flow algorithms*, *Journal of the Operations Research Society of Japan*, 43(2000), 2-31.
- [44] R.K. Ahuja, T. L. Magnanti, J.B. Orlin, The minimum cost flow problem, *Encyclopedia of Optimization*, Vol. 3, edited by C.A. Floudas and P. Pardalos, Kluwer Academic Publishers, 2001, 292-302.
- [45] R.K. Ahuja, T. L. Magnanti, J.B. Orlin, Maximum flow problem, *Handbook of Applied Optimization*, edited by M. Resende and P. Pardalos, Oxford University Press, 2002, 363-374.
- [46] A. Schrijver, On the history of the transportation and maximum flow problems, *Mathematical Programming*, 19(2002), 437-445.
- [47] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer-Verlag, Berlin, 2003.

### 最大流改进

- [48] H.N. Gabow, Scaling algorithms for network problems, *Journal of Computer and System Sciences*, 31(1985), 148-168.
- [49] R.K. Ahuja, J.B. Orlin, A fast and simple algorithm for the maximum flow problem, *Operations Research*, 37(1989), 748-759.
- [50] R.K. Ahuja, J.B. Orlin, Distance-directed augmenting path algorithms for maximum flow and parametric maximum flow problems, *Naval Research Logistics Quarterly*, 38(1991), 413-430.
- [51] R. K. Ahuja, J. B. Orlin, R.E. Tarjan, Improve time bounds for the maximum flow problem, *SIAM Journal on Computing*, 18(1989), 939-954.
- [52] J. Cheriyan, T. Hagerup, A randomized maximum-flow algorithm, *Proceedings of the 30<sup>th</sup> IEEE Conference on the Foundations of Computer Science*, 1989, 118-123.
- [53] J. Cheriyan, T. Hagerup, K. Mehlhorn, Can a maximum flow be computed in  $O(nm)$  time? *Proceedings of the 17<sup>th</sup> International Colloquium on Automata, Languages and Programming*, 1990, 235-248.
- [54] N. Alon, Generating pseudo-random permutations and maximum flow algorithms, *Information Processing Letters*, 35(1990), 201-204.
- [55] K. Iwano, S. Misono, S. Tezuka, S. Fujishige, A new scaling algorithm for the maximum mean cut problem, *Algorithmica*, 11(1994), 243-255.
- [56] R.K. Ahuja, J.B. Orlin, A capacity scaling algorithm for the constrained maximum flow problem, *Networks*, 25(1995), 89-98.

- [57] R. K. Ahuja, J. B. Orlin, C. Stein, R. E. Tarjan, Improved algorithms for bipartite network flow problems, *SIAM Journal on Computing*, 23(5)(1994), 906-933.

#### 平面图上的最大流

- [58] A. Itai, Y. Shiloach, Maximum flow in planar networks, *J. SIAM Comp.*, 8(2) (1979), 135-150.
- [59] D. B. Johnson, S. Venkatesan, Using divided and conquer to find flows in directed planar networks in  $O(n^{3/2} \log n)$  time, *Proceedings of the 20<sup>th</sup> Annual Allerton Conference on Communication, Control, and Computing*, University of Illinois, (1982), 898-905.
- [60] R. Hassin, D. B. Johnson, An  $O(n \log^2 n)$  algorithm for maximum flow in undirected planar networks, *SIAM Journal on Computing*, 14(1985), 612-624.

#### 最大流更多文献

- [61] D. Goldfarb, M. D. Grigoriadis, A computational comparison of the dinic and network simplex methods for maximum flow, *Annals of Operations Research*, 13(1)(1988), 81-123.
- [62] A. V. Goldberg, Recent developments in maximum flow algorithms, Technical Report #98-045, NEC Research Institute, Inc., April, 1998.
- [63] M. Shigeno, A survey of combinatorial maximum flow algorithms on a network with gains, *Journal of the Operational Research Society of Japan*, 47(4)(2004), 244-264.
- [64] V. King, S. Rao, R. Tarjan, A faster deterministic maximum flow algorithm, *Journal of Algorithms*, 17(1994), 447-474.
- [65] L. Fleischer, M. Skutella, Quickest flows over time, *SIAM Journal on Computing*, 36(6) (2007), 1600-1630.
- [66] C. Fremuth-Paeger, D. Jungnickel, Balanced network flows VII: primal-dual algorithms, *Networks*, 39(1)(2002), 35-42.
- [67] R. K. Ahuja, J. B. Orlin, Equivalence of primal simplex and dual simplex algorithms for the maximum flow problem, *Operations Research Letters*, 20(1997), 101-108.
- [68] É. Tardos, K. D. Wayne, Simple maximum flow algorithms in lossy networks, *Proceedings of 6<sup>th</sup> International Integer Programming and Combinatorial Optimization Conference*, Lecture Notes in Computer Science, vol.1412, (1998), 310-324.
- [69] M. Martens, M. Skutella, Flows on few paths: algorithms and lower bounds, *Networks*, 48(2)(2006), 68-76.
- [70] Donglei Du, R. Chandrasekaran, The maximum residual flow problem: NP-hardness with two-arc destruction, *Networks*, 50(3)(2007), 181-182.
- [71] C. C. Aggarwal, J. B. Orlin, On multiroute maximum flows networks, *Networks*, 39(1) (2002), 43-52.
- [72] R. K. Ahuja, M. Kodialam, A. K. Mishra, J. B. Orlin, Computational investigation of maximum flow algorithms, *European Journal on Operational Research*, 97(1997), 509-542.

## 最小费用流

- [73] F. Barahona, E. Tardos, Note on Weintraub's minimum cost circulation algorithm, *SIAM Journal on Computing*, 18(1989), 579-583.
- [74] C. Wallacher, U. T. Zimmermann, A combinatorial interior point method for network flow problems, *Proceedings of the 14<sup>th</sup> International Symposium on Mathematical Programming*, Amsterdam.1991.
- [75] M. Shigeno, S. Iwata, S.T. McCormick, Relaxed most negative cycle and most positive cut canceling algorithms for minimum cost flow, *Mathematics of Operations Research*, 25(1)(2000), 76-104.
- [76] P.T. Soddalingam, R.K. Ahuja, J.B. Orlin, New polynomial-time cycle-canceling algorithms for minimum-cost flows, *Networks*, 36(1)(2000), 53-63.
- [77] N. Tomizawa, On some techniques useful for solution of transportation network problems, *Networks*, 1(1972), 173-194.
- [78] M. D. Grigoriadis, An efficient implementation of the network simplex method , *Mathematical Programming Study*, 26(1986), 83-110 .
- [79] D.P. Bertsekas, P. Tseng, Relaxation methods for minimum cost ordinary and generalized network flow problems, *Operation Research*, 36(1988), 93-114.
- [80] H.Röck, Scaling techniques for minimal cost network flows, In *Discrete Structures and Algorithms*, Edited by V. Page. Carl Hanser Munich, (1980),181-191.
- [81] R. G. Bland, D. L. Jensen, On the computational behavior of polynomial-time network flow algorithm, *Mathematical Programming*, 54(1985), 1-39.
- [82] A. V. Goldberg, An Efficient Implementation of a Scaling Minimum-Cost Flow Algorithm, *Journal of Algorithms*, 22(1997), 1-29.
- [83] R.K. Ahuja, A.V. Goldberg, J.B. Orlin, R.E. Tarjan, Finding minimum-cost flows by double scaling, *Mathematical Programming*, 53(1992), 243-266.
- [84] A.V. Goldberg, R.E. Tarjan, Solving minimum cost flow problem by successive approximation, *Mathematics of Operations Research*, 15(1990), 430-466.
- [85] J.B. Orlin, A faster Strongly polynomial minimum cost flow algorithm, *Proceedings of the 20<sup>th</sup> ACM Symposium on the Theory of Computing*, 1988, 377-387. Also see *Operations Research*, 41(1993), 338-350.
- [86] S. Plotkin, E. Tardos, Improved dual network simplex, *Proceedings of the First ACM-SIAM Symposium on Discrete Algorithms*, 1990, 367-376.
- [87] H. N. Gabow, and R. E. Tardos, Fast scaling algorithm for network problems, *SIAM Journal on Computing*, 18(1989), 1013-1036.
- [88] D. Goldfarb, Z. Jin, A new scaling algorithm for the minimum cost network flow problem, *Operations Research Letters*, 25(1999), 205-211.

- [89] É. Tardos, A strongly polynomial minimum cost circulation algorithm, *Combinatorica*, 5(1985), 247-255.
- [90] S. Fujishige, A capacity-rounding algorithm for the minimum cost circulation problem: A dual framework of Tardos' algorithm, *Mathematical Programming*, 35(1986), 298-308.
- [91] Z. Galil, E. Tardos, An  $O(n^2(m + n \log n) \log n)$  min-cost flow algorithm, *Journal of ACM*, 35(1987), 374-386.
- [92] T. R. Ervolina, and S. T. McCormick, Canceling most helpful cuts for minimum cost network flow, *Networks*, 23(1993), 41-52.
- [93] T. R. Ervolina, and S. T. McCormick, Two strongly polynomial cut canceling algorithms for minimum cost network flow, *Discrete Applied Mathematics*, 46(1993), 133-165.
- [94] P. M. Vaidya, Speeding up linear programming using fast matrix multiplication, *Proceedings of the 30<sup>th</sup> Annual Symposium on the Foundations of Computer Science*, 1989, 332-337.
- [95] E.L. Johnson, Network and basic solution, *Operation Research*, 14(1966), 619-624.
- [96] V. Srinivasan, G.L. Thompson, Benefit-cost analysis of coding techniques for primal transportation algorithm, *Journal of ACM*, 20(1973), 194-213.
- [97] F. Glover, D. Karney, D. Klingman, A. Napier, A computational study on start procedures, basic change criteria, and solution algorithms for transportation problem, *Management Science*, 20(1974), 793-813.
- [98] J. L. Kennington, R. V. Helgason, *Algorithms for Network Programming*, Wiley-Interscience, New York, 1980.
- [99] M. S. Bazaraa, J. J. Jarvis, H.D. Sherali, *Linear Programming and Network Flows*, Wiley, New York, 1990.
- [100] F. Glover, D. Karney, D. Klingman, Implementation and computational comparisons of primal, dual and primal-dual compute codes for minimum cost network flow problem, *Networks*, 4(1974), 191-212.
- [101] J. Mulvey, Pivot strategies for primal-simplex network codes, *Journal of ACM*, 25(1978), 266-270.
- [102] G. Bradley, G. Grown, G. Graves, Design and implementation of large scale primal transshipment algorithms, *Management Science*, 24(1977), 1-34.
- [103] M. D. Chang, C. J. Chen, An improved primal simplex variant for pure processing networks, *ACM Transactions on Mathematical Software*, 15(1989), 64-78.
- [104] W. H. Cunningham, A network simplex method, *Mathematical Programming*, 11(1976), 105-116.
- [105] W. H. Cunningham, Theoretical properties of the network simplex method, *Mathematics of Operations Research*, 4(1979)196-208.

- [106] D. Goldfarb, J. Hao, S. Kai, Anti-stalling pivot rules for the network simplex algorithm, *Networks*, 20(1990), 79-91.
- [107] R. E. Tarjan, Efficiency of the primal network simplex algorithm for the minimum-cost circulation problem, *Mathematics of Operations Research*, 16(1991), 272-291.
- [108] R. E. Bixby, The simplex method: it keeps getting better, *Proceedings of the 14<sup>th</sup> International Symposium on Mathematical Programming*, 1991, Amsterdam.
- [109] J.B. Orlin, A polynomial time primal network simplex algorithm for minimum cost flow, *Mathematical Programming*, 78(1997), 109-129.
- [110] T. C. Sharkey, H. E. Romeijn, A simplex algorithm for minimum-cost network-flow problems in infinite networks, *Networks*, 52(2008).
- [111] R. Hassin, The minimum cost flow problem: a unifying approach to dual algorithms and a new tree-search algorithm, *Mathematical Programming*, 25(1983), 228-239.
- [112] R. K. Ahuja, D. S. Hochbaum, J.B. Orlin, A cut-based algorithm for the nonlinear dual of the minimum cost network flow problem, *Algorithmica*, 39(3)(2004), 189-208.

#### 多品种流

- [113] R.W. Hall, D. Lotspeich, Optimized lane assignment on an automated highway, *Transportation Research-C*, 4C(4), (1996), 211-229.
- [114] A. Haghani, S. C. Oh, Formulation and solution of a multi-commodity, multi-modal network flow modal for disaster relief operations, *Transportation Research-A*, 30A(3), (1996), 231-250.
- [115] F. T. Leighton, S. B. Rao, A. Srinivasan, Multicommodity flow and circuit switching, *Proceedings of Hawaii International Conference on System Sciences*, 1998.
- [116] B. L. Golden, A minimum cost multicommodity network flow problem concerning imports and exports, *Networks*, 5(1975), 331-356.
- [117] T. Crainic, J. A. Ferland, J. M. Rousseau, A tactical planning model for rail freight transportation, *Transportation Science*, 18(1984), 165-184.
- [118] A.V. Goldberg, E. Tardos, R.E. Tarjan, Network flow algorithm, in *Algorithms and Combinatorics vol.9: Path, flows, and VLSI-layout* (eds: B.Korte, L.Lovasz, H.J. Promel, and A. Schrijver), Springer-Verlag, New York, 1990.
- [119] A. Srinivasan, A survey of the role of multicommodity flow and randomization in network design and routing, in *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 43(1999), 271-302.
- [120] A.I. Ali, D. Barnett, K. Farhangian, J.L. Kennington, B. Patty, B. Shetty, B. McCarl, P. Wong, Multicommodity network problems: applications and computations, *IIE Transactions*, 16(1984), 127-134.
- [121] P. Klein, A. Agrawal, R. Ravi, S. Rao, Approximation through multi-commodity flow, *Proceedings of 31<sup>st</sup> IEEE Symposium on Foundations of Computer Science*, 1990, 726-727.

- [122] T. Leighton, S. Rao, An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms, *Proceedings of 29<sup>th</sup> Annual Symposium on Foundations of Computer Science*, 1988, 422-431. Also see *J. ACM* 46(1999) 787-832.
- [123] T. Leighton, F. Makedon, S. Plotkin, C. Stein, Eva Tardos, S. Tragoudas, Fast approximation algorithms for multicommodity flow problems, *J. Computer and System Sciences*, 50 (1995), 228-243.
- [124] A.V. Goldberg, J.D. Oldham, S. Plotkin, C. Stein, An implementation of a combinatorial approximation algorithm for minimum-cost multicommodity flow, in *Integer Programming and Combinatorial: Proceedings of 6<sup>th</sup> International IPCO Conference*, Houston, Texas, 1998.
- [125] A. Ali, R. Helgason, J. Kennington, H. Lall, Computational comparison among three multicommodity network flow algorithms, *Operations Research*, 28(1980), 995-1000.
- [126] M. D. Grigoriadis, W.W. White, A partitioning algorithm for the multicommodity network flow, *Mathematical Programming*, 3(1972),157-177.
- [127] F. Shahrokhi, D.W. Matula, The maximum concurrent flow problem, *J. Assoc. Comp. Mach.*, 37(1990), 318-334.
- [128] A.V. Goldberg, A natural randomization strategy for multicommodity flow and related algorithms, *Information Processing Letters*, 42(1992), 249-256.
- [129] J.R. Evans, J.J. Jarvis, R.A. Duke, Graphic matroids and the multicommodity transportation problem, *Mathematical Programming*, 13(1977), 323-328.
- [130] J.R. Evans, J.J. Jarvis, Network topology and integral multicommodity flow problems, *Networks*, 8(1978), 107-119.
- [131] J.R. Evans, On equivalent formulations of certain multicommodity networks as single commodity flow problems, *Mathematical Programming*, 15(1978), 92-99.
- [132] A.M. Geoffrion, G.W. Graves, Multicommodity distribution system design by Benders decomposition, *Management Science*, 20(1974), 822-844.
- [133] L.R. Ford, A suggested computation for maximal multicommodity network flows, *Management Science*, 20(1974), 822-844.
- [134] J. L. Kennington, A survey of linear cost multicommodity flow, *Operations Research*, 26(1978), 209-236.
- [135] A. A. Assad, Multicommodity network flows: a survey, *Networks*, 8(1978), 37-91.
- [136] C. Barnhart, *A network-based primal-dual solution methodology for the multicommodity network flow problem*, Ph.D. Dissertation, Department of Civil Engineering, MIT, Cambridge, MA.1988.
- [137] J. Castro, N. Nabona, An implementation of linear and nonlinear multicommodity network flows, *European Journal of Operational Research*, 92(1), (1996), 37-53.



- [138] P. Klein, S.A. Plotkin, C. Stein, E. Tardos, Faster approximation algorithms for the unit capacity concurrent flow problem with applications to routing and finding spare cuts, *Proceedings of 22<sup>nd</sup> Annual ACM Symposium on Theory of Computing*, 1990, 310-321.
- [139] B. Awerbuch, T. Leighton, A simple local-control approximation algorithms for the multicommodity flow, *Proceedings of 34<sup>rd</sup> IEEE Symposium on Foundations of Computer Science*, 1993, 459-466.
- [140] M. D. Grigoriadis, L.G. Khachiyan, Approximate minimum-cost multicommodity flows in  $O(\epsilon^{-2}knm)$  time, *Mathematical Programming*, 75(3), (1996), 477-482.
- [141] D. Karger, S. Plotkin, Adding multiple cost constraints to combinatorial optimization problems, with applications to multicommodity flows, *Proceedings of 27<sup>th</sup> Annual ACM Symposium on Theory of Computing*, vol.27, 1995, 18-25.
- [142] T. Leong, P. Shor, C. Stein, Implementation of a combinatorial multicommodity flow algorithm, in *Network Flows and Matching* (D.S. Johnson and C. C. McGeoch, editors), *Series in Discrete Mathematics and Theoretical Computer Science*, vol.12, 1993, 387-405.
- [143] T. Radzik, Fast deterministic approximation for the multicommodity flow problem, *Mathematical Programming*, 78(1), (1997), 43-58.

## 内点法

- [144] A. Kamath, O. Palmon, Improved interior point algorithms for exact and approximate solution of multicommodity flow problems, *Proceedings of the 6<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms*, vol.6, (1995), 502-511.
- [145] S. Kapoor, P.M. Vaidya, Speeding up Karmarkar's algorithm for multicommodity flows, *Mathematical Programming*, 73(1996), 111-127.
- [146] E. Yamakawa, Y. Matsubara, M. Fukushima, A parallel primal-dual interior point method for multicommodity flow problems with quadratic cost, *Journal of the Operations Research Society of Japan*, 39(4)(1996), 566-591.
- [147] J.J. Júdice, L.F. Portugal, M.G.C. Resende, G. Veiga, A Truncated interior point method for the solution of minimum cost flow problems on an undirected multicommodity network, *Proceedings of First Portuguese National Telecommunications Conference*, 1997, 381-384.
- [148] J. Castro, A specialized interior-point algorithm for multicommodity network flows, *SIAM Journal on Optimization*, 10(3)(2000), 852-877.
- [149] R. Chardaire, A. Lissier, Simplex and interior point specialized algorithms for solving nonoriented multicommodity flow problems, *Operations Research*, 50(2002), 260-276.
- [150] M. G. C. Resende, G. Veiga, An annotated bibliography of network interior point methods, *Networks*, 42(2)(2003), 114-121.

### 多品种流更多文献

- [151] A. Gersht, A. Shulman, A new algorithm for the solution of the minimum cost multi-commodity flow problem, *Proceedings of the IEEE Conference on Decision and Control*, 26(1987), 748-758.
- [152] R. Schneur, *Scaling algorithms for multicommodity flow problems and network flow problems with side constraints*, Ph. D. Dissertation, Department of Civil Engineering, MIT, Cambridge, MA.1991.
- [153] Y.K. Lin, Study on the multicommodity reliability of a capacitated-flow network, *Computer and Mathematics with Applications*, 42(2001), 255-264.
- [154] C. Duhamel, P. Mahey, Multicommodity flow problems with a bounded number of paths: a flow deviation approach, *Networks*, 49(1)(2007), 80-89.
- [155] P. Belotti, F. Malucelli, Multicommodity network design with discrete node costs, *Networks*, 49(1)(2007), 90-99.
- [156] F. Alvelos, J.M. Valério de Carvalho, An extended model and a column generation algorithm for the planar multicommodity flow problem, *Networks*, 50(1)(2007), 3-16.
- [157] M. G. Scutella, The maximum congested cut problem and its robust counterpart: exact and approximation algorithms for the single and the multicommodity case, *Networks*, 51(2)(2008), 102-112.
- [158] A. Bompadre, J. B. Orlin, A simple method for improving the primal simplex method for the multicommodity flow problem, *Networks*, 51(1)(2008), 63-77.
- [159] K. Holmberg, M. Joborn, K. Melin, Lagrangian based heuristics for the multicommodity network flow problem with fixed costs on paths, *European Journal on Operational Research*, 188(2008), 101-108.

### 动态网络流

- [160] Jay E. Aronson, A survey of dynamic network flows, *Annals of Operations Research*, 20(1), (1989), 1-66.
- [161] B. Awerbuch, T. Leighton, Improved approximation algorithms for the multi-commodity flow problem and local competitive routing in dynamic networks, *Proceedings of 26<sup>th</sup> Annual ACM Symposium on Theory of Computing*, Montreal, Canada, 1994, 487-495.

### 多目标网络流

- [162] R. Fabregat, Y. Donoso, B. Baran, F. Solano, Jose L. Marzo, Multi-objective optimization scheme for multicast flows: a survey, a model and a MOEA solution, *Proceedings of LANC'05*, October 10-20, 2005, 73-86, Cali, Colombia.
- [163] Y. Donoso, R. Fabregat, Jose L. Marzo, Multi-objective optimization algorithm for multicast routing with traffic engineering, *IEEE ICN*, 2004.
- [164] H.I. Calvete, P.M. Mateo, An approach for the network flow problem with multiple objectives, *Computers and Operations Research*, 22(1995), 971-983.

- [165] A. Sedeño-Noda, C. González-Martín, The biobjective minimum cost flow problem, *European Journal of Operational Research*, 124(2000), 591-600.
- [166] A. Sedeño-Noda, C. González-Martín, An algorithm for the biobjective integer minimum cost flow problem, *Computers and Operations Research*, 28(2)(2000), 139-156.
- [167] A. Sedeño-Noda, C. González-Martín, J. Gutiérrez, The biobjective undirected two-commodity minimum cost flow problem, *European Journal on Operational Research*, 164(2005), 89-103.
- [168] D. Klingman, J. Mote, Solution approaches for network flow problems with multiple criteria, *Advances in Management Studies*, 1(1982), 1-30.
- [169] H. Lee, S. Pulat, Bicriteria network flow problems: Continuous case, *European Journal of Operational Research*, 51(1991), 119-126.
- [170] H. Lee, S. Pulat, Bicriteria network flow problems: integer case, *European Journal of Operational Research*, 66(1993), 148-157.
- [171] A. Mustafa, M. Goh, Finding integer efficient solutions for bicriteria and tricriteria network flow problems using Dinan, *Computers and Operations Research*, 25(1998), 139-157.
- [172] P. Pulat, F. Huang, H. Lee, Efficient solutions for the bicriteria network flow problem, *Computers and Operations Research*, 19(1992), 649-655.

### 非线性流

- [173] S. Kapoor, P.M. Vaidya, Fast algorithms for convex quadratic programming and multi-commodity flows, *Proceedings of 18<sup>th</sup> Annual ACM Symposium on Theory of Computing*, vol.18, (1986), 147-159.
- [174] A.V. Karzanov, S.T. McCormick, polynomial methods for separable convex optimization in totally unimodular linear spaces with applications to circulations and co-circulations in networks, *SIAM Journal on Computing*, 26(1997), 1245-1275.
- [175] A. Ouorou, P. Mahey, J. Ph. Vial, A Survey of Algorithms for Convex Multicommodity Flow Problems, *Management Science*, Vol. 46, No. 1 (Jan., 2000), pp. 126-147.
- [176] R.K. Ahuja, D. Hochbaum, J.B. Orlin, Solving the convex integer dual network flow problem, *Management Science*, 49(2003), 950-964.
- [177] M. Shigeno, Maximum network flows with concave gains, *Mathematical Programming*, 107(2006), 439-459.
- [178] D. B. M. M. Fontes and J. F. Gonçalves, Heuristic solutions for general concave minimum cost network flow problems, *Networks*, 50(1)(2007), 67-76.
- [179] T. Larsson, J. Marklund, C. Olsson, M. Patriksson, Convergent Lagrangian heuristics for nonlinear minimum cost network flows, *European Journal on Operational Research*, 189(2008), 324-346.
- [180] M. C. de Souza, P. Mahey, B. Gendron, Cycle-based algorithms for multicommodity

network flow problems with separable piecewise convex costs, *Networks*, 51(2)(2008), 133-141.

- [181] A. Ouorou, Implementing a proximal algorithm for some nonlinear multicommodity flow problems, *Networks*, 49(1)(2007), 18-27.

### 广义网络流

- [182] S. M. Murray, An interior point approach to the generalized flow problem with costs and related problems, *Ph.D Thesis*, Stanford University, 1993.
- [183] E. Cohen, N. Megiddo, New algorithms for generalized network flows, *Mathematical Programming*, 64(1994), 325-336.
- [184] K. D. Wayne, *Generalized Maximum Flow Algorithms*, PhD Thesis, Department of Operations Research and Industrial Engineering, Cornell University, 1999.
- [185] K. D. Wayne, A polynomial combinatorial algorithm for generalized minimum cost flow, STOC'99, Atlanta, 1999.
- [186] T. Radzik, Faster algorithms for the generalized network flow problem, *Mathematics of Operations Research*, 23(1998), 69-100.
- [187] T. Radzik, Improving time bounds on maximum generalized flow computations by contracting the network, *Theoretical Computer Science*, 312(2004), 75-97.
- [188] J. D. Oldham, Combinatorial approximation algorithms for generalized flow problems, *Journal of Algorithms*, 38(2001), 135-169.
- [189] A.V. Goldberg, S.A. Plotkin, É. Tardos, Combinatorial algorithms for the generalized circulation problem, *Mathematics of Operations Research*, 16(1991), 351-381.
- [190] D. Goldfarb, Z. Jin, A faster combinatorial algorithm for the generalized circulation problem, *Mathematics of Operations Research*, 21(1996), 529-539.
- [191] D. Goldfarb, Z. Jin, J. B. Orlin, Polynomial-time highest-gain augmenting path algorithms for the generalized circulation problem, *Mathematics of Operations Research*, 22(1997), 793-802.
- [192] D. Goldfarb, Z. Jin, Y. Lin, A polynomial dual simplex algorithm for the generalized circulation problem, *Mathematical Programming*, 91(2002), 271-288.
- [193] D. Goldfarb, Y. Lin, Combinatorial interior point methods for generalized network flow problems, *Mathematical Programming*, 93(2002), 227-246.
- [194] L. K. Fleisher, K.D. Wayne, Fast and simple approximation schemes for generalized flow, *Mathematical Programming*, 91(2002), 215-238.
- [195] A. Eguchi, S. Fujishige, T. Takabatake, A polynomial-time algorithm for the generalized independent-flow problem, *Journal of Operational Research Society of Japan*, 47(2004), 1-17.

## 子模流

- [196] C. Wallacher, U. Zimmermann, A polynomial cycle canceling algorithm for submodular flows, *Mathematical Programming*, 86(1999), 1-15.
- [197] S. Iwata, S. T. McCormick, M. Shigeno, A faster cost scaling algorithm for submodular flow, *Information Processing Letters*, 74(2000), 123-128.
- [198] S. Iwata, S.T. McCormick, M. Shigeno, Fast cycle canceling algorithms for minimum cost submodular flow, *Combinatorica*, 23(3)(2003), 503-525.
- [199] S. Iwata, S. T. McCormick, M. Shigeno, A strongly polynomial cut canceling algorithms for the submodular flow problem, *Proceedings of the 7<sup>th</sup> MPS conference on Integer Programming and Combinatorial Operation*, 1999, 259-272. Also see: *SIAM Journal on Discrete Mathematics*, 19(2005), 304-320.
- [200] S. Iwata, S. Moriguchi, K. Murota, A capacity scaling algorithm for M-convex submodular flow, *Mathematical Programming*, 103(2005), 181-202.

## 其他

- [201] R. K. Ahuja, J. B. Orlin, Combinatorial algorithms for inverse network flow problems, *Networks*, 40(4)(2002), 181-187.
- [202] R. K. Ahuja, H. Hamacher, A network flow algorithm to minimize beam-on time for unconstrained multileaf collimator problems in cancer radiation therapy, *Networks*, 45(2005), 36-41.
- [203] R. K. Ahuja, J. B. Orlin, P. Zuddas, G. Secki, Algorithms for the equal flow problem, *Management Science*, 45(1999), 1440-1455.
- [204] P. T. Sokkalingam, R. K. Ahuja, J. B. Orlin, Solving inverse spanning tree problems through network flow techniques, *Operations Research*, 47(1999), 291-300.
- [205] C. C. Aggarwal, R. K. Ahuja, J. Hao, J. B. Orlin, Diagnosing infeasibilities in network flow problems, *Mathematical Programming*, 81(1998), 263-280.
- [206] J. Hartline, A. Sharp, Incremental flow, *Networks*, 50(1)(2007), 77-85.
- [207] M. S. Daly, C. Alexopoulos, State-space partition techniques for multiterminal flows in stochastic networks, *Networks*, 48(2)(2006), 90-111.
- [208] F. Ordóñez, Jiamin Zhao, Robust capacity expansion of network flows, *Networks*, 50(2)(2007), 136-145.
- [209] R. D. Armstrong, Z. Jin, A new strongly polynomial dual network simplex algorithm, *Mathematical Programming*, 78(1997), 131-148.
- [210] G. Baier, E. Köhler, M. Skutella, The k-splittable flow problem, *Algorithmica*, 42(3-4)(2005), 231-248.
- [211] Y. Azar, O. Regev, Combinatorial algorithms for the unsplittable flow problem, *Algorithmica*, 44(1)(2006), 49-66.

- [212] A. Chakrabarti, C. Chekuri, A. Gupta, A. Kumar, Approximation algorithms for the unsplittable flow problem, *Algorithmica*, 47(1)(2007), 53-78.
- [213] L. Fleischer, J.B. Orlin, Optimal rounding of instantaneous fractional flow over time, *SIAM Journal on Discrete Mathematics*, 13(2)(2000), 145-153.
- [214] M. Shigeno, I. Takahasi, Y. Yamamoto, Minimum maximal flow problem: an optimization over the efficient set, *Journal of Global Optimization*, 25(2003), 425-443.

# 名词索引

---

## B

伴随网络, 315  
半径, 27  
胞腔, 230  
饱和, 295  
饱和的, 75  
闭包型条件, 108  
闭迹, 5  
闭途径, 5  
边  $k$  色可染的, 157  
边导出子图, 3, 91, 133  
边独立集, 130  
边独立数, 130  
边分离集, 65  
边分离数, 65  
边覆盖集, 132  
边覆盖数, 132  
边割集, 56  
边集, 1  
边界, 215  
边连通度, 57  
边色数, 158  
边正常  $k$  染色, 157  
并, 3  
并行弧, 260  
补图, 3  
不交并, 3, 175, 197  
不可免完备集, 237, 238  
不可平面图, 214, 220, 242

## C

Christofides 算法, 114  
出度, 261, 277  
出邻集, 261

## D

Dijkstra 算法, 11–15  
单连通, 264  
单源单汇网络, 292  
得分向量, 276  
底图, 261  
点  $k$  色可染的, 162  
点导出子图, 3  
点独立集, 125  
点覆盖, 129  
点覆盖数, 129  
点割集, 56  
点色数, 163  
叠加, 320  
顶点集, 1  
顶点集构造图, 181  
顶点正常  $k$  染色, 162  
定向图, 261  
独立数, 125  
度, 2, 261  
度型条件, 107  
度序列条件, 109  
对称差, 4, 68, 78  
对偶图, 220, 242  
多项式算法, 14

## E

Edmonds-Johnson 算法, 104  
Edmonds-Johnson 方法, 103  
Euler 闭迹, 97, 99, 101, 102, 112, 114, 158  
Euler 迹, 97, 99, 116  
Euler 图, 97, 99, 102, 104, 114, 116, 158,  
242, 269  
Euler 有向图, 268  
儿子, 278, 282  
二部图, 6, 80, 82, 83, 242  
二元前缀码, 280, 281

## F

Fleury 算法, 100, 116  
发点集, 292  
反向, 260  
反向弧, 296  
非饱和, 295  
费用函数, 319  
费用网络, 319  
分离集, 63  
分离数, 63  
分支点, 278  
分支权, 31  
辅助网络, 306  
负费用有向圈, 323

## G

改进, 159  
割点, 53  
根树, 278  
根子树, 286  
供需可行流, 318  
供需网络, 318  
关联, 2  
关联矩阵, 34, 43  
广义汇, 318  
广义源, 318

## H

H 图, 105  
H 路, 105

H 圈, 105, 113  
Hamilton 路, 105, 107, 113, 118  
Hamilton 圈, 92, 105, 128  
Hamilton 图, 105, 117, 128, 194, 222, 236  
Hamilton 有向路, 268, 270, 272, 288, 289  
Hamilton 有向圈, 268, 270  
Hamilton 有向图, 268-270  
好算法, 14  
和, 3  
厚度, 228  
后代, 278  
后向弧, 296  
后序遍历法, 286  
弧, 260  
弧集, 260, 294  
环边, 2  
环弧, 260  
汇点集, 292

## J

基础图, 261  
迹, 4, 268  
极大点独立, 125  
极大可平面图, 215  
极大可平面子图, 225  
极大平面图, 215  
极大团, 139  
极大外可平面图, 223  
极大外平面图, 223  
极小边覆盖, 132  
极小边割集, 56  
极小点覆盖, 129  
极小点割集, 56  
极小图, 237  
极小支配集, 122  
简单图, 2  
简单有向图, 260  
交叉数, 226, 247  
交错, 76  
近似比, 112  
距离, 5  
绝对性能, 112



**K**

$k$  边连通, 65  
 $k$  连通图, 65  
 $k$  色图, 163, 165, 170  
 $k$  色可染的, 233  
 $k$ -边连通, 57  
 $k$ -边割集, 56  
 $k$ -团, 139  
 $k$ -点割集, 56  
 $k$ -顶点割集, 56  
 $k$ -连通的, 56  
 Kruskal 算法, 20, 25, 41  
 Kuhn-Munkres 算法, 89  
 可  $k$  染色的, 162  
 可扩展路, 76, 88  
 可平面图, 214  
 可嵌入曲面, 214  
 可行顶点标号, 86  
 可行循环流, 314  
 可增路, 296  
 空图, 2  
 块, 60  
 亏格, 229, 230

**L**

离心率, 27  
 连通, 7  
 连通分支, 7  
 连通图, 7  
 临界  $k$  色图, 163  
 邻接矩阵, 34  
 邻近点法, 111  
 零图, 2  
 路, 4, 296  
 旅行商问题, 110

**M**

面  $k$  色可染, 233  
 面的度数, 215  
 面色数, 233  
 面正常  $k$  染色, 233

**N**

内部面, 215  
 内点, 278  
 逆向弧, 296, 327

**O**

偶圈, 5

**P**

Prim 算法, 23, 26  
 匹配, 75, 185  
 匹配数, 130  
 平凡图, 2, 91  
 平面嵌入, 214  
 平面图, 1, 214  
 剖分, 219  
 剖分图, 219

**Q**

奇分支, 76  
 奇圈, 5  
 前向弧, 296  
 前序遍历法, 285  
 前缀, 280  
 前缀码, 280  
 强连通, 264  
 强连通分支, 275  
 强完美图猜想, 182  
 区间图, 181  
 圈, 5, 262  
 圈流, 322

**R**

Ramsey 数, 122, 139, 140, 144  
 容量, 294  
 容量函数, 292  
 入度, 261  
 入邻集, 261  
 弱连通, 264

**S**

色多项式, 171

色临界图, 163  
色数, 163  
森林, 17  
上容量, 314, 316, 333  
上下容量网络, 316, 317  
生成树, 19, 123  
生成子图, 3, 66, 133, 234, 265  
始点, 260  
收点集, 292  
收缩, 219  
树, 17, 278  
树根, 278  
双向连通, 264  
顺向弧, 296  
四色猜想, 233, 259

#### T

同构, 9  
头, 260  
图, 1  
途径, 4  
团覆盖集, 180  
团覆盖数, 180  
团数, 139

#### W

外部面, 215  
外可平面图, 222  
外平面嵌入, 222  
外平面图, 222  
完美匹配, 75  
完美图, 175, 202  
完美图猜想, 177  
完全二部图, 6  
完全图, 2  
完全正则树, 278  
完全正则有序树, 278  
王, 271  
网络, 292  
尾, 260  
无限面, 215  
无向图, 261

五色定理, 235, 242

#### X

下容量, 314, 316  
弦图, 181  
相等子图, 87  
相对性能, 112  
相邻, 2  
相容流, 318  
兄弟, 278, 282  
匈牙利算法, 83

#### Y

严格有向图, 260  
一般 Ramsey 数, 146  
有限面, 215  
有向, 262  
有向闭, 268  
有向边, 260  
有向迹, 262, 268  
有向路, 262, 303  
有向生成树, 266  
有向树, 266  
有向图, 260  
有向途径, 262  
有效算法, 14  
有序树, 278  
右子, 278  
源点集, 292

#### Z

增广路, 76  
增量, 297  
增量网络, 302, 320  
正向弧, 296  
正则树, 278  
正则图, 2, 147  
正则有序, 278  
支, 122  
支配集, 122  
直边交叉数, 227, 251  
直径, 27

- 指派问题, 83, 95  
中国邮递员问题, 101  
中位点, 27  
中心, 27  
中心点, 237  
中序遍历法, 285  
终点, 260  
重边, 2  
子树, 278  
子图, 3, 41, 60, 220  
祖先, 278  
最大出度, 261  
最大点独立集, 125  
最大度, 2  
最大可平面子图, 225  
最大亏格, 232, 258  
最大流, 294  
最大匹配, 75  
最大团, 139  
最短路, 5, 44  
最佳边  $k$  染色, 159  
最佳前缀码, 281  
最小边覆盖, 132  
最小边割集, 56  
最小出, 261  
最小点覆盖, 129  
最小度, 2  
最小割, 294  
最小权匹配法, 114  
最小入度, 261  
最小生成树法, 112  
最小生成树问题, 19  
最小团覆盖集, 180  
最小支配集, 122  
最优 2 叉树, 281  
最优邮路, 102  
最优指派问题, 86  
左子树, 278