WangluoSuanfa
Yu FuzaxingLilun

网络算法与复杂性理论

[第二版]

谢 政 著

国防科技大学出版社

网络算法与复杂性理论

(第二版)

谢 政 著

国防科技大学出版社
·湖南 长沙·

内容简介

本书全面系统地介绍了网络最优化中的基本问题和基本算法以及计算复杂性的基本内容和近似算法.取材恰当,叙述清晰,论证严谨,深入浅出.

全书共十二章,分为两部分:第一部分包括前十章,主要介绍最小树,最小树形图,最短路,最大流,最小费用流,最大匹配,最大权匹配和中国邮递员问题等基本问题的各种多项式算法,以及线性规划、整数线性规划的基本理论;第二部分包括后两章,讨论计算复杂性中的基本概念,NP 完全理论及重要的 NP 完全问题,还介绍了装箱问题,平行机排序问题,旅行商问题,背包问题等 NP 完全问题的近似算法.

本书可作为运筹学专业研究生教材,也可供应用数学、系统科学、管理科学、计算机科学和军事运筹学等有关专业的教师、研究生和大学高年级学生参考.

图书在版编目(CIP)数据

网络算法与复杂性理论/谢政著 .—长沙:国防科技大学出版社,2003.12
ISBN 7-81024-330-6

Ⅰ.网… Ⅱ.谢… Ⅲ.网络理论—算法—复杂性 Ⅳ.O157.5

国防科技大学出版社出版发行
电话:(0731)4572640 邮政编码:410073
E-mail:gfkdcbs@publich.cs.hn.cn
责任编辑:耿 筠 责任校对:唐卫葳
新华书店总店北京发行所经销
国防科技大学印刷厂印装

*

开本:787×960 1/16 印张:21 字数:388 千
2003 年 12 月第 2 版第 1 次印刷 印数:1-3000 册
ISBN 7-81024-330-6/O·41
定价:28.00 元

第二版序

《网络算法与复杂性理论》是《组合图论》(谢政、戴丽编著,国防科技大学出版社,2003)的姊妹篇,自1995年出版以来,被一些院校选作相关课程的教材.经过8年的教学实践,作者和出版社都有修订再版的愿望.在保留第一版的风格的基础上,第二版重新改写了大部分内容,既有增补也有删减,主要改动的地方有:

第一章增加了图与网络的两种表示形式——权矩阵和边(弧)目录,并改写了1.6节;

第二章增加了定理2.6的证明和一些算法的复杂性估计,并对一些定理的证明进行了改写;

第三章全部进行了改写,增加了两个著名的算法——广探法和深探法,及其相关的广探分枝、广探树形图、深探分枝、深探树形图等内容;

第四章把线性规划和对偶规划两节合并成一节,并重写了整数线性规划和全单位模矩阵的内容;

第五章除第2最短路这一节外,其余各节内容都有变化,修改了许多定理的叙述及证明,还增加了具有检测负回路功能的Ford算法、最小平均回路、网络的重心和网络的中心等内容;

第六章的内容改动较大,增加了Ford-Fulkerson算法失效的一个例子和预流推进算法,改写了循环流及带发点和收点的双容量网络等有关内容;

第七章全部重写,尤其是原始-对偶算法和状态算法这两节,并且增加了最小凸费用流和最小凹费用流一节;

第八章的内容改动较小,在匈牙利算法这一节中引进了交错树和

匈牙利树的概念，还增加了一些算法的复杂性估计；

第九章删去了种上树的概念及其有关结论，增加了交错树的定义，简化了内点和外点的定义（这不同于第一版中内点和外点的定义）及匈牙利树的定义；

第十章增加了赋权混合图上的邮递员问题一节；

第十一章和第十二章所有内容都进行了改写，统一了记号和术语，并且在第十一章中增加了 Co-NP 完全问题一节，在第十二章中，把装箱问题独立成一节，并增补了平行机排序问题的近似算法一节.

此外，对绝大部分算法的复杂性都重新进行了更为精细的估计，还增添了一些难度较高的习题作为对正文内容的扩充.

李建平副教授在本书第一版中所撰写的第十一章和第十二章为第二版这两章提供了很好的素材，研究生王芳、伍勇安、杨延飞、梁兆健曾协助我补充和查对了第十一章、第十二章的有关资料，在此一并致以谢意.

谢　政

2003 年 11 月

序

网络最优化的理论和方法已广泛地渗透于运筹学、信息论、控制论、管理科学和计算机科学等领域,并在工程技术、经济、军事等诸多方面都有着极为重要的应用.可以毫不夸张地说,现代社会在很大程度上是一个由通信网络、运输网络、能源和物资分配网络构成的巨大的复杂系统.网络最优化能为人们控制和管理这个系统提供一种有效的方法.由于大规模网络最优化问题的需要,研究各种有效算法一直是网络最优化研究的一个主旋律.但是,人们发现,网络最优化中的许多问题很难找到有效算法,也不知道它们是否存在有效算法,这就使得计算复杂性和近似算法的研究越来越受到普遍的重视.这两者的结合正是当前网络最优化研究的潮流.因此,网络算法与复杂性理论是高等院校有关专业的一门不可忽视的重要课程.

基于上述认识,我们在总结多年教学实践的基础上,参考国内外一些富有代表性的教材,并广泛查阅了有关文献,编写了这本教材.旨在系统地讲授网络最优化模型、算法与计算复杂性理论,目的是使读者不仅能系统地掌握网络最优化的建模思想和基本算法,而且学习和掌握计算复杂性的基本概念、基本理论和基本方法,提高分析和解决实际问题的能力.

本书全面地介绍了网络最优化中的基本问题和基本算法以及计算复杂性理论中的基本概念和一些常见的 NP 完全问题.全书共十二章,分为两部分:第一部分包括前十章,主要介绍网络最优化中的概念、模型和算法,同时,还注重对算法复杂性的分析;第二部分包括后两章,介绍 NP 完全理论和近似算法.这两部分虽然有一定的独立性,但我们始终强调它们之间的有机联系.另外,每章最后都精心配置了

一定数量的习题.

本书对基本概念的陈述力求准确简洁,对定理和算法的证明力求清晰严谨,对内容的编排力求系统完整.阅读本书只需要熟悉线性规划的一些基本内容.

由于学识有限,经验不足,书中肯定有不少的错误和疏漏,恳请广大读者批评指正.

在本书的写作过程中,一直得到国防科工委指挥技术学院陈庆华教授的关怀和鼓励,他仔细审阅了本书的原稿,并提出了许多宝贵建议;国防科技大学系统工程与数学系研究生陈浩光和汤泽滢两位同学为作者整理了部分资料.在此,一并表示感谢.

作　者

1994 年 11 月于长沙

目　　录

第九章　一般图的匹配

第十章　中国邮递员问题

第十一章　NP 完全理论

第十二章　近似算法

第一章 图与算法

从一定意义上讲,现代社会是一个由信息网络、通信网络、运输网络、能源和物资分配网络构成的巨大的复杂系统.网络最优化能为人们控制和管理这个网络系统提供一套有效的方法.

本章主要介绍图、网络、算法及算法复杂性的一些基本概念,给出八个典型的网络最优化问题及其相关的实际例子.这些内容是全书的基础.

1.1 图的基本概念

图(graph)G 是指由非空有限集合 $V(G)$和 $V(G)$中某些元素的无序对的集合 $E(G)$构成的二元组$(V(G),E(G))$. $V(G)$称为 G 的顶点集(vertex set),其中的元素称为 G 的顶点(vertex). $E(G)$称为 G 的边集(edge set),其中的元素称为 G 的边(edge).在不混淆的情况下,记 $V=V(G)$, $E=E(G)$, $G=(V, E)$.如果 $V=\{v_1, v_2, \cdots, v_n\}$,则 E 中元素 e 与 V 中某两个元素构成的无序对 $\{v_i, v_j\}$相对应,记 $e=v_iv_j$ 或 $e=v_jv_i$.

图可以用图形来表示,用小圆圈表示顶点,用小圆圈之间的连线表示边.例如,图 1.1 就表示图 $G=(V,E)$: $V=\{v_1, v_2, \cdots, v_5\}$, $E=\{e_1, e_2, \cdots, e_7\}$,其中 $e_1=v_1v_2$, $e_2=v_1v_2$, $e_3=v_2v_3$, $e_4=v_3v_4$, $e_5=v_4v_5$, $e_6=v_5v_2$, $e_7=v_4v_4$.

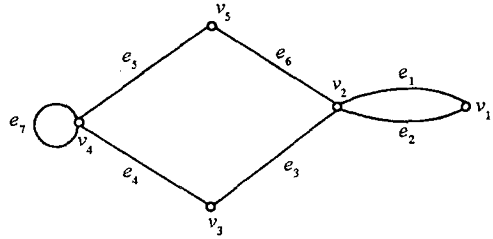

图 1.1 图的图形表示

设 $G=(V,E)$是一个图,若 $e=v_iv_j\in E$,则称顶点 v_i 和 v_j 是相邻的(adjacent),并称 v_i, v_j 为边 e 的端点(end),也称 e 与 v_i, v_j 关联(incident).若 $e_1, e_2 \in E$,且 e_1 和 e_2 有公共的端点,则称 e_1 与 e_2 是相邻的.我们把 G 中所有与顶点 v 相邻的顶点的集合称为 v 的邻域(neighbour),记为 $N_G(v)$ 或简记为 $N(v)$.

两个端点重合的边称为环(loop).如果有两条边的端点是同一对顶点,则称这两条边为重边(multiple edge).既没有环也没有重边的图称为简单图(simple graph).没有环的图称为无环图(unlooped graph).边集为空集的图称为空图(empty graph).至少有一条边的图称为非空图(nonempty graph).一个图的顶点数称为该图的阶(order).

图 G 中顶点 v 的度(degree)定义为和 v 关联的边的数目(与 v 关联的每个环算作两条边),记为 $d_G(v)$.称 $d_G(v)$ 是偶数的顶点 v 为偶点(even vertex),称 $d_G(v)$ 是奇数的顶点 v 为奇点(odd vertex).$d_G(v)=0$ 的顶点 v 称为孤立点(isolated vertex).$d(v)=1$ 的顶点 v 称为悬挂点(pendant vertex).容易得到下面的定理:

定理 1.1 设 $G=(V,E)$ 是一个图,则

$$\sum_{v \in V} d_G(v) = 2|E|. \qquad \square$$

如果图 G 的某些顶点和边可以排成非空的有限序列 $W = v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$,这里 $v_i \in V(G)(1 \leqslant i \leqslant k+1)$, $e_j \in E(G)(1 \leqslant j \leqslant k)$,并且 $e_i = v_i v_{i+1}$ $(1 \leqslant i \leqslant k)$,则称 W 为 G 的一条途径(walk).v_1 称为 W 的起点(origin),v_{k+1} 称为 W 的终点(terminus),$v_i (2 \leqslant i \leqslant k)$ 称为 W 的内部顶点(internal vertex),也称 W 为 G 的 (v_1, v_{k+1}) 途径.k 称为 W 的长(length).有时把途径 $v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$ 简单地记为 $v_1 v_2 \cdots v_k v_{k+1}$.值得注意的是,由于以顶点 v_i 和 v_{i+1} 为端点的边可能不止一条,因此 $v_1 v_2 \cdots v_k v_{k+1}$ 可能同时表示若干条不同的途径;但是简单图中的途径 $v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$ 由 $v_1 v_2 \cdots v_k v_{k+1}$ 完全确定.

途径 $W = v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$ 的节(section)是指由 W 中相继项构成的子序列 $v_i e_i v_{i+1} \cdots v_{j-1} e_{j-1} v_j$,它也是一条途径,这一子序列又称为 W 的 (v_i, v_j) 节.途径 $W = v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$ 的逆向途径(inversion walk)是指途径 $v_{k+1} e_k v_k \cdots v_2 e_1 v_1$,记为 W^{-1}.设 $W_1 = v_1 e_1 v_2 \cdots v_k e_k v_{k+1}$ 和 $W_2 = v_{k+1} e_{k+1} v_{k+2} \cdots v_l e_l v_{l+1}$ 是图 G 的两条途径,则称途径

$$W = v_1 e_1 v_2 \cdots v_k e_k v_{k+1} e_{k+1} v_{k+2} \cdots v_l e_l v_{l+1}$$

为 W_1 与 W_2 的衔接(concatenation),记作 $W = W_1 W_2$,也称 W 可以表示为 W_1 和 W_2 的并(union).

如果图 G 中途径 W 上的边互不相同,则称 W 为 G 的迹(trail).如果图 G

中的途径 W 上的顶点互不相同，则称 W 为 G 的链(chain). 易知，图 G 中的链必定是 G 的迹，但 G 中的迹不一定是 G 的链.

如果途径的长至少为 1，且起点和终点重合，则称该途径为闭途径(closed walk). 起点和终点重合且长至少为 1 的迹称为闭迹(closed trail). 起点、内部顶点互不相同的闭迹称为圈(cycle). 长为偶数的圈称为偶圈(even cycle)，长为奇数的圈称为奇圈(odd cycle).

如果图 G 和图 H 满足 $V(H)\subseteq V(G)$, $E(H)\subseteq E(G)$，则称 H 是 G 的子图(subgraph)，记为 $H\subseteq G$. 特别地，若 $V(H)=V(G)$, $E(H)=E(G)$，则记 $H=G$. 如果 $H\subseteq G$，且 $V(H)=V(G)$，则称 H 为 G 的支撑子图(spanning subgraph). 图 G 中的链和圈都可以看做是 G 的子图.

设 $V'\subseteq V(G)$, $V'\neq\emptyset$，令 $E'=\{e\in E(G)\mid e$ 的两个端点均属于 $V'\}$，则称 G 的子图 $G'=(V',E')$ 为由 V' 导出的子图(induced subgraph)，记为 $G'=G[V']$.

设 $E'\subseteq E(G)$, $E'\neq\emptyset$，令 $V'=\{v\in V(G)\mid v$ 是 E' 中某条边的端点$\}$，则称 G 的子图 $G'=(V',E')$ 为由 E' 导出的子图(edge-induced subgraph)，记为 $G'=G[E']$.

如果对于图 G 的任意两个顶点 v_i 和 v_j，G 中都存在 (v_i, v_j) 链，则称 G 是连通图(connected graph). 不连通的图称为非连通图(disconnected graph).

如果 $H\subseteq G$，且 H 是连通图，则称 H 是 G 的连通子图(connected subgraph). 图 H 称为图 G 的极大连通子图(maximal connected subgraph)是指 H 为 G 的连通子图，并且 G 中不存在连通子图 H' 使得 $H\subseteq H'$, $H'\neq H$. 图 G 的极大连通子图又称为 G 的连通分支(connected component). 由此可知，连通图恰有一个连通分支，而非连通图则有两个或两个以上的连通分支.

下面介绍图的几种运算.

若 $E_1\subseteq E(G)$，从图 G 中删去 E_1 的所有边得到的图记为 $G-E_1$，通常把 $G-\{e\}$ 简记为 $G-e$. 若 $H\subseteq G$, $E_2\subseteq E(G)\setminus E(H)$，且 E_2 中每条边的端点都属于 $V(H)$，在图 H 中添加 E_2 的所有边得到的图记为 $H+E_2$. 常常简单地用 $H+e$ 表示 $H+\{e\}$.

若 $V_1\subset V(G)$，则把从图 G 中删去 V_1 的所有顶点以及与 V_1 中顶点关联的边得到的图记为 $G-V_1$. $G-\{v\}$ 常常简记为 $G-v$. 容易知道 $G[V']=G-(V(G)\setminus V')$.

若 $H_1\subseteq G$, $H_2\subseteq G$，则把顶点集为 $V(H_1)\cup V(H_2)$、边集为 $E(H_1)\cup E(H_2)$ 的图称为 H_1 与 H_2 的并(union)，记为 $H_1\cup H_2$. 若 $E(H_1)\cap E(H_2)=$

$\emptyset$,则称 $H_1 \cup H_2$ 为 H_1 与 H_2 的和,记为 H_1+H_2.

设 e 是图 $G=(V,E)$ 的一条边,且 $e=v_iv_j$,我们从 G 中删去 e,并把顶点 v_i 和 v_j 重合为一个顶点 y,得到一个新的顶点集 $V'=(V\setminus\{v_i,v_j\})\cup\{y\}$,而把 G 中所有与 v_i 或与 v_j 关联的边都改为与 y 关联,G 中既不与 v_i 也不与 v_j 关联的边不变,得到一个新的边集 E',这样产生的新图 $G'=(V',E')$ 称为 G 关于边 e 的收缩(condensation).顶点 y 称为收缩 e 产生的人造顶点(artificial vertex).

设 $G=(V,E)$ 是一个图,S 为 V 的一个非空真子集,$\overline{S}=V\setminus S$,记

$$[S,\overline{S}]=\{v_iv_j\in E\mid v_i\in S,\ v_j\in\overline{S}\}.$$

如果 $[S,\overline{S}]\neq\emptyset$,则称 $[S,\overline{S}]$ 是 G 的一个边割(edge - cut).若 E' 是 G 的一个边割,但 E' 的任何真子集都不是 G 的边割,则称 E' 为 G 的极小边割(minimal edge-cut).G 的极小边割又称为 G 的补圈(cocycle).

一个图的边割与圈有如下的关系:

定理 1.2 设 C 和 $[S,\overline{S}]$ 分别是图 G 的圈和补圈,则

$$|E(C)\cap[S,\overline{S}]|\equiv 0 \pmod 2.$$

证明 当 $E(C)\cap[S,\overline{S}]=\emptyset$ 时,$|E(C)\cap[S,\overline{S}]|=0$.当 $E(C)\cap[S,\overline{S}]\neq\emptyset$ 时,设 $C=v_1e_1v_2e_2v_3\cdots v_pe_pv_1$,$v_1\in S$,且 $E(C)\cap[S,\overline{S}]=\{e_{i_1},e_{i_2},\cdots,e_{i_k}\}$,且 $1\leqslant i_1<i_2<\cdots i_k\leqslant p$.记 $\hat{V}=\{v_{i_1},v_{i_2},\cdots,v_{i_k}\}$,由 $v_1\in S$ 知,$v_{i_1}\in S$,$v_{i_k+1}\in S$.根据 $[S,\overline{S}]$ 的定义有

$$\{v_{i_j}\in\hat{V}\mid j \text{ 为奇数}\}\subseteq S,\quad \{v_{i_j}\in\hat{V}\mid j \text{ 为偶数}\}\subseteq\overline{S}.$$

因 $v_{i_k+1}\in S$,故 $v_{i_k}\in\overline{S}$,即知 k 为偶数.于是 $|E(C)\cap[S,\overline{S}]|$ 为偶数. □

1.2 有向图的基本概念

有向图(digraph)D 是指由一个非空的有限集合 $V(D)$ 和 $V(D)$ 中某些元素的有序对的集合 $A(D)$ 构成的二元组 $(V(D),A(D))$.$V(D)$ 称为 D 的顶点集,其中的元素称为 D 的顶点.$A(D)$ 称为 D 的弧集(arc set),其中的元素称为 D 的弧(arc).在不混淆时,记 $V=V(D)$,$A=A(D)$,$D=(D,A)$.

如果 $V(D)=\{v_1,v_2,\cdots,v_n\}$,那么 $A(D)$ 中元素 a 与 $V(D)$ 中某两个元素构成的有序对 $\{v_i,v_j\}$ 相对应,记 $a=(v_i,v_j)$,其中 v_i 称为 a 的尾(tail),v_j 称为 a 的头(head);a 称为 v_i 的出弧(out-arc),也称为 v_j 的入弧(in-arc);称 v_j 为 v_i 的出邻点,称 v_i 为 v_j 的入邻点.以 $N_D^+(v)$ 表示顶点 v 的所有出邻点的集

合,称为 v 的出邻域(out-neighbour);以 $N_D^-(v)$表示顶点 v 的所有入邻点的集合,称为 v 的入邻域(in-neighbour).顶点 v 在D 中出弧的数目记为$d_D^+(v)$,称之为 v 的出度(out-degree);顶点 v 在D 中入弧的数目记为$d_D^-(v)$,称之为 v 的入度(in-degree).显然有

定理 1.3　设 $D=(V,A)$是有向图,则

$$\sum_{v\in V} d_D^+(v) = \sum_{v\in V} d_D^-(v) = |A|. \qquad \square$$

有向图也可以用一个图形来表示.用小圆圈表示顶点,用小圆圈之间带有箭头的连线表示弧,连线的箭头由弧的尾指向弧的头.例如,图 1.2 就表示一个有向图 $D=(V,A)$: $V=\{v_1,v_2,\cdots,v_6\}$, $A=\{a_1,a_2,\cdots,a_{11}\}$,其中 $a_1=(v_1,v_1)$, $a_2=(v_1,v_2)$, $a_3=(v_3,v_2)$, $a_4=(v_3,v_4)$, $a_5=(v_4,v_3)$, $a_6=(v_2,v_4)$, $a_7=(v_5,v_4)$, $a_8=(v_4,v_6)$, $a_9=(v_6,v_5)$, $a_{10}=(v_6,v_2)$, $a_{11}=(v_6,v_2)$.

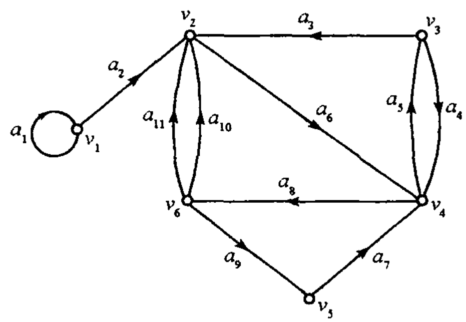

图 1.2　有向图的图形表示

尾和头重合的弧称为环.若两条弧有相同的头和相同的尾,则称这两条弧为重弧(multiple-arc).既没有环也没有重弧的有向图称为简单有向图(simple digraph).

把有向图 D 中每条弧上的箭头去掉,即把 D 的每条弧(v_i,v_j)用边 v_iv_j 代替,得到的图称为 D 的基础图(underlying graph).反之,给定一个图 G,如果把图 G 的每条边都规定一个方向,即把顶点的无序对按规定的方向改为有序对,得到的有向图称为 G 的定向图(oriented graph).

有向图中的许多概念可以通过它的基础图去描述.例如,设 G 是有向图D 的基础图,如果顶点 v_i 和 v_j 在G 中相邻,则称 v_i 和 v_j 在D 中相邻;若 G 是连通的,则称 D 是连通的;D 中的途径是指 D 中这样的一些顶点和弧构成的序列:使这些顶点与这些弧相对应的边构成 G 的途径.类似地定义有向图的迹、链、闭途径、闭迹和圈.

仿照图中子图、支撑子图和导出子图的定义，同样可以给出有向图的子图、支撑子图和导出子图的定义.另外，1.1 节介绍的图的一些运算也可以照搬到有向图上来.这些都留给读者去完成.

由于有向图中，“方向”是很重要的，因此我们要给出有向图中与方向有关的一些概念.

有向图 D 中的有向途径(directed walk)是指 D 中某些顶点和弧组成的非空有限序列 $W=v_1a_1v_2\cdots v_ka_kv_{k+1}$，其中 $v_i\in V(D)(1\leqslant i\leqslant k+1)$，$a_j\in A(D)(1\leqslant j\leqslant k)$，且 $a_i=(v_i,v_{i+1})(1\leqslant i\leqslant k)$. v_1 称为 W 的起点，v_{k+1}称为 W 的终点，W 上的其他顶点称为 W 的内部顶点，并称 W 为 D 的有向(v_1,v_{k+1})途径. k 称为 W 的长.有向途径 $v_1a_1v_2\cdots v_ka_kv_{k+1}$常常简单地用 $v_1v_2\cdots v_kv_{k+1}$来表示.

有向途径 $W=v_1a_1v_2\cdots v_ka_kv_{k+1}$的节是指由 W 中相继项构成的子序列 $v_ia_iv_{i+1}\cdots v_{j-1}a_{j-1}v_j$，这一子序列也称为 W 的(v_i,v_j)节.如果 W_1 和 W_2 都是有向图 D 中的有向途径，且 W_1 的终点正好是 W_2 的起点，则把 W_2 接在 W_1 的后面就得到 D 的一条新的有向途径 W，记作 $W=W_1W_2$，并称 W 为 W_1 与 W_2 的衔接，或称 W 可表示为 W_1 与 W_2 的并.

如果有向图 D 中的有向途径 W 上的弧互不相同，则称 W 为 D 的有向迹(directed trail).与图的链、闭途径、闭迹和圈的定义一样，完全类似地定义有向图中的有向链、有向闭途径、有向闭迹和有向圈.为方便计，我们常常把有向链称为路(path)，把有向圈称为回路(circuit).

如果对于有向图 D 中的任意两个顶点 v_i 和 v_j，D 中既存在(v_i,v_j)路，也存在(v_j,v_i)路，则称 D 是强连通的(strongly connected).易知，若有向图 D 是强连通的，则 D 是连通的，反之不然.同图的连通分支类似，可以定义有向图的强连通分支(strongly component).

可以用一个形象的比喻来解释连通有向图和强连通有向图的差异.设想有一个公路网连接若干个城镇，且每条公路只允许单向行驶，这样的公路网称为单行公路网.若把城镇看做顶点，单行公路看做弧，则单行公路网可看做一个有向图 D，D 连通相当于从任一城镇出发，不管公路规定的行驶方向，驱车可到达任一其他城镇；而 D 强连通则等同于从任一城镇驾车出发，严格按照规定的方向行驶，可以到达任一其他城镇.

图中边割的概念也可以推广到有向图上来.设 $D=(V,A)$是一个有向图，$\forall S,T\subseteq V$，定义

$$(S,T)=\{(v_i,v_j)\in A\mid v_i\in S,v_j\in T\}.$$