第 14 章 最大流问题

在现代社会的基础设施与信息系统中，如何高效地调配和传输各种资源，成为保障城市运行、经济发展和社会服务的重要课题。无论是交通道路的车辆调度、供水管网的水流分配，还是通信网络中数据的传输，背后都蕴含着同一个核心问题——如何在有限的通道容量下，实现资源从源头到目标点的最大化流动。这一类问题在运筹学中被抽象为“最大流问题”（Maximum Flow Problem），是网络优化理论中的基石模型。

最大流问题不仅以其简洁的结构和严谨的数学性质成为经典的组合优化问题，更通过 Ford-Fulkerson、Edmonds-Karp、Dinic 等高效算法的不断发展，推动了图论、最优化和计算机科学等相关领域的进步。最大流理论的提出和完善，为交通规划、物流调度、通信网络设计、能源分配等实际工程提供了科学的决策依据和强大的分析工具。

随着社会系统的不断复杂化和网络规模的日益扩大，最大流问题的应用也从传统的单源单汇模型，拓展到多源多汇等更为复杂和多样的场景。通过本章的学习，读者将系统掌握最大流问题的基本概念、数学建模、核心算法及其理论基础，了解其在现实世界中的广泛应用，并为后续深入研究网络优化与运筹学的前沿知识奠定坚实基础。

14.1 最大流问题基本概念

在现代社会中，最大流问题是广泛应用于各种网络系统的一类经典问题。网络系统中包括管道网络、通信网络、交通网络、电力网络等，这些网络中每条边（即连接节点的通道）的最大通过能力（即容量）是有限的，实际流量不会超过这些边的容量。在不同的网络中，流量可以有不同的物理含义，例如车流、人流、物流、信息流等，统称为网络流。

最大流问题最早由 Ford 和 Fulkerson 在 1956 年提出，他们提出了标签算法（Labeling Algorithm）来求解该问题。20 世纪 50 年代，Ford 和 Fulkerson 建立了“网络流理论”，这标志着网络流理论成为网络应用领域的重要组成部分。此理

论广泛应用于交通网络（车流、人流、货物流）、供水网络（水流）、通信网络（信息流）以及金融系统（现金流）等领域。

最大流问题是一种组合最优化问题，其目标是最大化网络中的流量，同时不超过各边的容量限制。具体来说，该问题讨论如何在充分利用网络装置的传输能力的前提下，使得从源点（source）到汇点（sink）的流量达到最大，实现网络资源的最优分配。

最大流问题的提出，不仅丰富了运筹学理论，也为实际网络系统的优化提供了有力工具。为了更好地理解最大流问题的本质，需要进一步分析其基本结构和关键性质。最大流问题通常基于一个网络图，这个网络图由以下要素构成：**顶点**代表网络中的不同节点，如交通路口、供水站、通信设备等。**边**是连接顶点的通道，每条边都具有最大容量，表示该通道所能承载的最大流量。在实际应用中，流量不能超过这一容量限制。

在运筹学中，最大流问题是优化网络资源分配的经典模型，广泛应用于交通规划、通信网络设计和物流管理等领域。要深入理解这一问题，需要关注其几个关键性质，这些性质不仅勾勒出问题的结构轮廓，还为设计算法和分析结果提供了坚实的理论基础。

最大流问题的网络由一系列有向边构成，流量只能沿着边预设的方向从一个节点流向另一个节点，无法反向传递。这种单向性好比通信网络中的数据包，只能沿光纤的指定路径传输，限制了流量的传播路线，迫使人们在优化流量分配时必须严格遵循网络的拓扑结构。每一张这样的网络图都像一张精密的蓝图，指引着流量的方向与可能性。与此同时，网络中的每条边都被赋予一个固定的容量上限，代表其承载流量的最大能力。例如，一条互联网链路可能每秒传输 100 兆字节的数据。这种容量限制如同网络的“天花板”，不可突破，促使人们在规划流量时精打细算，既要充分利用高容量的关键路径，又要协调整个网络的资源分配，以确保整体流量的最大化。

在网络中，除了源点和汇点，其他所有节点都必须遵守流量守恒原则，即流入节点的总流量等于流出节点的总流量。可以用一个水管接头来类比：流入的水量必须全部流出，不能在接头处积聚或凭空消失。这种守恒性确保了流量的连续

性，让人们能专注于从源点到汇点的整体传输，而无需担心中间节点的不平衡。经典的最大流问题通常假设网络只有一个源点和一个汇点，就像从一座工厂向一个仓库运送货物。即使面对多起点或多终点的复杂场景，也可以通过引入虚拟节点，将问题转化为单源单汇形式，从而简化分析和求解。

最大流问题最引人注目的理论成果是最大流最小割定理。这一定理揭示了流量与网络结构的深刻联系。所谓“割”，是将网络的节点分成两组，源点在一组，汇点在另一组，割的容量是跨越这两组的边的容量之和。定理指出，源点到汇点的最大流量等于所有可能割的容量最小值。这一等价关系不仅为验证解的正确性提供了工具，还能帮助读者识别网络中的瓶颈。例如，如果找到一个割，其容量恰好等于计算出的最大流量，就可以确认已达到最优解，同时了解哪些边是限制流量的关键。

这些性质——有向边的限制、容量的上限、流量守恒、单源单汇的结构，以及最大流最小割的等价性——共同塑造了最大流问题的本质。它们相互关联，共同确保流量在约束下被优化，同时为算法设计提供了理论支撑。在实际应用中，这些性质帮助分析网络的传输极限，优化资源分配，并定位关键路径或薄弱环节。

14.2 网络与网络流

在理解了最大流问题的基本概念后，接下来需要从图论的角度，进一步认识描述网络流的基本工具——网络图。在运筹学中，最大流问题通常用网络图来表示。通过设计合适的算法，能够有效求解在网络图中从源点到汇点的最大可达流量，为实际应用中网络资源的优化配置提供理论基础和方法支持。

通过上述简要介绍，可以发现网络图对于实际问题建模的重要性。下面通过一个具体的例子，来进一步认识网络图。

如图 14.1 所示，假设有一个城市的眼镜公司，它需要将所有生产出来的眼镜通过公路分配给各个地区的销售点。这个分配的过程就可以用网络图表示。网络图由顶点和有向边组成，展示了不同节点之间的连接关系和运输能力。

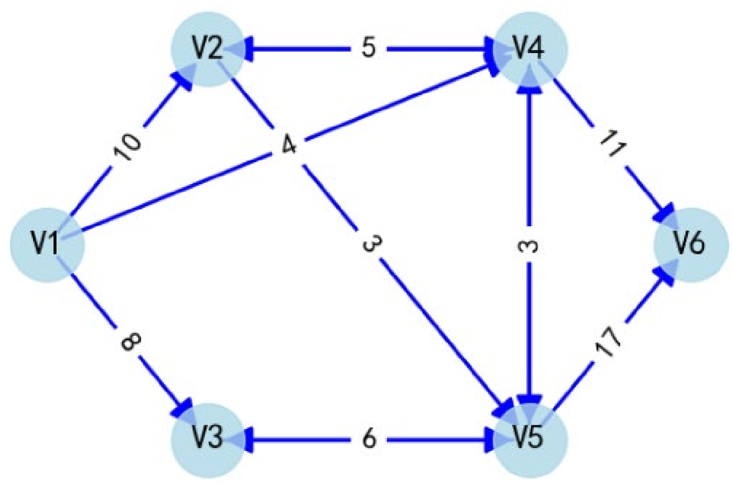

图 14.1 销售点的网络示意图

为了理解最大流问题的求解过程，可以首先从认识网络图的基本构成讲起。

（1）网络中的有向边和容量

在一个有向图中，一条有向边代表从节点v_i到节点v_j的运输路径。每条有向边旁边标注的数字通常有两种含义，容量表示有向边的最大通过能力。它是理想情况下，流量能够达到的上限。例如，在水流模型中，容量可能代表管道的最大流量。流量表示在实际条件下，当前有向边上流动的物理量。流量的数值不会超过容量。

在最大流问题中，目标是通过优化网络中的流量，使得从源点（源节点）到汇点（终点节点）的流量最大化，同时满足每条有向边的流量不超过其容量限制。

在上述例子中，眼镜生产公司就是网络图中的一个顶点，各个地区销售点是其他的顶点，公路是连接这些顶点的有向边。每条公路都有它的最大运输量（容量），比如一天可以运送多少副眼镜。公司每天通过公路运送的眼镜数量（流量）不能超过这条公路的最大运输量。眼镜生产公司想要知道一天最多能卖出多少副眼镜。眼镜公司是源节点，各个销售点是终点节点。需要优化网络中的流量，尽可能多的把眼镜从生产公司运到各个销售点。

（2）图的结构

网络图的结构可以有不同的分类，而这些分类帮助理解图的连接方式和网络

流动的复杂度。常见的图类型包括有向图、无向图等。

在有向图（Directed Graph）中，有向边具有方向性，表示流量的流动方向。每个有向边都有起点和终点，如(v_i,v_j)表示从v_i到v_j的有向边。无向图（Undirected Graph）与有向图不同，无向图中的边没有方向，表示连接是双向的，流量可以在任意方向流动。

这些图的结构不仅影响流量的传输，还直接影响求解最大流问题时的算法选择和计算效率。眼镜公司和销售点之间的公路网络可以用有向图来表示，因为运输是有方向性的，即只能从生产公司到销售点。但如果两个销售点之间也可以交换眼镜，那么就是无向图。

（3）连接性与图的连通性

在网络流问题中，图的连通性是分析和优化网络结构的基础，直接影响流量的传输效率和网络的可达性。连通性描述了图中顶点间是否存在路径，确保信息、物资或其他资源能够在网络中流动。根据连通性的特征，图可分为连通图和不连通图，并进一步分解为连通分图。

连通图是指无向图中任意两顶点间均存在至少一条路径。这种特性保证了网络中所有节点可以通过某些路径相互访问，例如在通信网络中，任意两个终端都能通过线路传输数据。连通图是网络流问题的基础，因为最大流等优化目标通常要求网络整体可达。

若图中存在某些顶点间无路径连接，则称为不连通图。不连通图无法支持全局的流传输，但在实际应用中，网络可能因故障或设计限制而呈现不连通特性。此时，分析不连通图的结构有助于识别孤立区域或瓶颈。

不连通图可分解为若干连通分图，每个连通分图是图的一个子图，包含一组彼此连通的顶点及其关联边。连通分图的概念在网络流问题中尤为重要，例如在最小割问题中，可通过识别连通分图来划分网络，分析流的分配或网络的脆弱性。此外，确定连通分图有助于优化路径选择，确保资源在局部连通区域内高效流动。

连通性确保了流量的传输通道是否畅通无阻。如果图中的某些部分不连通，

那么流量可能无法从源点流到汇点。眼镜公司在多种地方设有生产线，所有生产线生产的眼镜都必须通过物流路径运送到中央仓库，再从中央仓库配送到各个销售点。在这里，生产线可以被视为图中的顶点，物流路径可以被视为图中的有向边。公司希望保证所有生产线生产的眼镜都能够顺利运送到中央仓库，再送到销售点。在这种情况下，公司的物流网络需要是一个连通图。然而，并不是所有时候物流网络都是连通的，有时候可能存在一些地方，比如由于交通阻塞或自然灾害，导致一些顶点之间没有路径连接，那么网络就是不连通的。为了确保网络流畅运行，此时应该对不连通区域进行检测，分解成连通分图，然后通过增设新路径或优化路径选择来使这些连通分图相互连通。

（4）路径与回路的基本概念

在分析流量时，也需要理解链在图中的角色。图中的路径描述了从一个顶点到另一个顶点的可能流动路线。链（Chain）是由一系列顶点和边组成的路径，可能会有重复的顶点或边。

如果链中没有重复的顶点或边，那么它是简单链（Simple Chain）。简单链通常表示无环的最短路径。

初等链（Elementary Chain）是一种不包含重复顶点的路径，用来描述唯一的行进路线，通常适用于无环的最短路径算法。

在最大流问题中，路径的分类和图的结构分析是求解的关键。除了开放路径（如简单链），还需要关注闭合路径和相关概念，以全面理解流量的分配和优化。回路（Circuit）是一个起点和终点相同的闭合路径，可以包含重复的顶点和边。初等路（Elementary Path）是无重复顶点的路径，它用于描述唯一且无回返的行程，特别适用于分析流量网络中的关键路径。

这些路径的分类有助于更好地理解和分析网络结构，从而在最大流问题的求解中，合理选择增广路径，提高流量分配的效率。

（5）流量

流量是通过网络有向边的流动值，不超过其容量，记作f_{ij}，表示在v_i和v_j之

间的实际流动量。流量的方向与有向边方向一致。

在中间节点上，满足流入等于流出的原则。对于发点v_s和收点v_t，流入和流出不必相等。

$$\sum_{e\in N^-(v_k)} f(e) = \sum_{e\in N^+(v_k)} f(e)\,,\forall v_k \in V \setminus \{v_s, v_t\} \tag{14.1}$$

流量的净值即进入节点的流量与流出节点的流量之间的关系，发点净流量为负数（即输出流量较多），收点净流量为正数（即输入流量较多）。

在一个网络中，如果某一组流量分配满足以下两个条件，则称之为可行流（Feasible Flow）。

容量约束满足每条有向边上的流量f_{ij}必须小于或等于该有向边的容量c_{ij}。即，对于所有有向边(v_i,v_j)，满足 $0\le f_{ij}\le c_{ij}$。

除源点v_s和汇点v_t以外的所有节点，其流入流量和流出流量必须相等。这就是流量守恒。

(6)增广路径

在最大流问题中，增广路径的定义依赖于当前流量f的状态。为了更精确地描述增广路径，可以使用如下的定义和条件：

在一个源点为s、汇点为t的流网络中，给定当前流f，一条增广路径是指从源点s到汇点t的一条路径，该路径上的每条边要么是一条前向边（前向边表示原网络中还未完全利用的边，其残差容量为原容量减去当前流量），要么是一条后向边（后向边是原网络中已使用流量的逆向表示）。

形式化定义为：一条增广路径是一条从源点s到汇点t的简单路径 P，使得对于 P 中的每条前向边(u,v)，有$cf(u,v)=c(u,v)-f(u,v)>0$（即存在剩余容量），对于 P 中的每条后向边(u,v)，有$f(u,v)>0$（即存在可以回收的流量）。其中，$cf(u,v)$表示边(u,v)在当前流f下的剩余容量。

增广路径的核心意义在于它标识了网络中仍然可以增加总流量的路径。沿

着增广路径，可以将更多的流量从源点传送到汇点，从而使网络的总流量增加。当网络中不存在增广路径时，根据最大流最小割定理，当前流即为最大流。

找到一条增广路径后，可以执行增广操作：

确定增广路径上的瓶颈容量δ，即所有前向边的剩余容量和所有后向边的当前流量的最小值。对增广路径上的每条前向边(u,v)，增加流量$f(u,v)$←$f(u,v)$+δ。对增广路径上的每条后向边(u,v)，减少流量$f(u,v)$←$f(u,v)$-δ。通过这一操作，网络的总流量将增加δ单位。

路径μ中的每一条前向有向边(v_i,v_j)都是非饱和有向边，即其剩余容量：

$$r_{ij} = c_{ij} - f_{ij} > 0, \forall\left(v_i, v_j\right) \in \mu\text{的前向边} \tag{14.2}$$

$$r_{ij} > 0 \tag{14.3}$$

路径μ中的每一条反向有向边(v_j,v_i)都是非零流有向边，即反向流量f_{ji}>0。在满足这两个条件的情况下，可以沿着增广路径增加源点到汇点的总流量。具体的增量是链上所有有向边的剩余容量的最小值，这确保了在更新流量时不会超过任何一条有向边的容量限制。

在最大流算法中，通过不断寻找增广路径并沿着该链增加流量，可以使得网络中的总流量逐步接近最大值。每找到一条增广路径并更新流量后，网络中的可行流量增加。这个过程会反复进行，直到再也找不到增广路径为止，此时的流量即为最大流。

增广路径是 Ford-Fulkerson 算法和 Edmonds-Karp 算法等解决最大流问题的基础。这些算法重复寻找增广路径并进行增广，直到网络中不存在增广路径为止。

(7)单源单汇网络和多源多汇网络

单源单汇网络是指在一个有向图中，只存在一个源点和一个汇点的网络流模型。源点是指只有流出而没有流入的节点，汇点是指只有流入而没有流出的节点。在此网络中，流量从源点出发，经过网络中的边和中间节点，最终全部到达汇点。网络中的每条边都有一个容量限制，表示该边上最大可通过的流量。单源单汇网

络是最基本的网络流模型，是研究最大流、最小割等问题的基础。

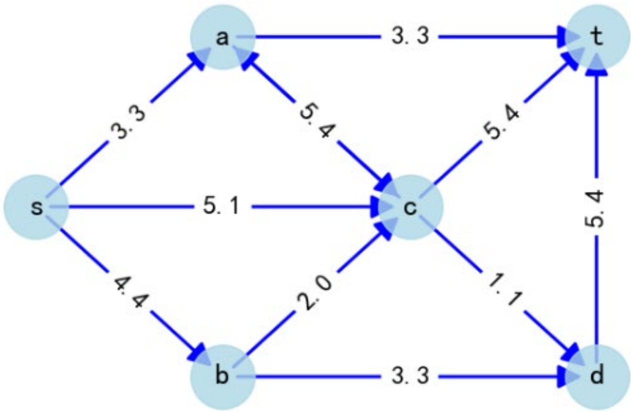

图 14.2 单源单汇网络

多源多汇网络是指在一个有向图中，存在多个源点和多个汇点的网络流模型。每个源点可以提供一定量的流量，每个汇点需要接收一定量的流量。网络中的流量需要从多个源点出发，经过各种可能的路径，最终分配到多个汇点，同时满足每条边的容量限制。多源多汇网络可以通过添加一个超级源点(连接到所有原始源点)和一个超级汇点(所有原始汇点连接到它)，转化为单源单汇问题进行求解。这种模型适用于更为复杂的资源分配和物流调度等现实问题。

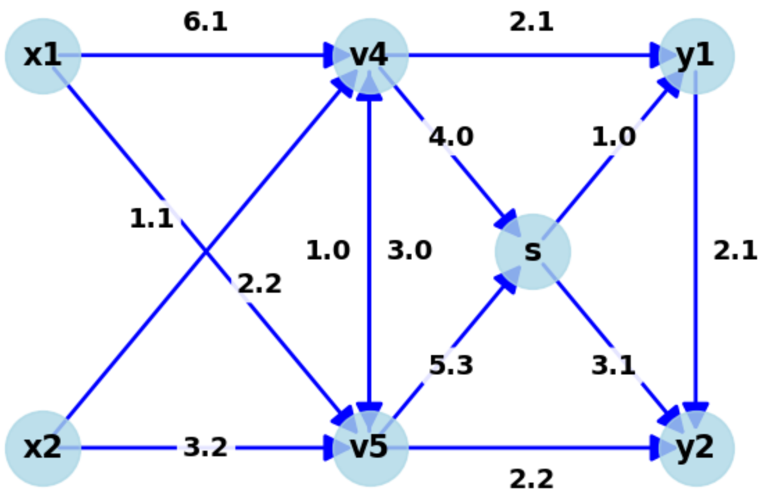

图 14.3 多源多汇网络

14.3 最大流问题的数学模型

在掌握了网络结构与流量基本概念后，可以将最大流问题进一步形式化，建立其数学模型，便于理论分析与实际求解。

(1)问题描述

考虑一个网络，可用有向图$G=(V,E)$表示，其中：V为节点集合，代表网络中的点位，E为有向边集合，代表连接点位的通道，每条有向边$(i,j)\in E$有一个非负容量c_{ij}，表示该通道的最大流量，网络中有特定的源点$s\in V$和汇点$t\in V$。确定从源点s到汇点t的最大可能流量，同时满足容量限制和流量平衡条件。

(2)数学建模

决策变量f_{ij}表示从节点i到节点j的流量，$(i,j)\in E$。目标函数是最大化从源点流出（或等价地，流入汇点）的总流量。完整线性规划模型如下：

$$\max\ Z=\sum_{j:(s,j)\in E} f_{sj} \tag{14.4}$$

$$s.t. \sum_{j(i,j)\in E} f_{ij}-\sum_{j(j,i)\in E} f_{ji}=0, \forall i\in V\setminus\{s,t\} \tag{14.5}$$

$$0\le f_{ij}\le c_{ij}, \forall(i,j)\in E \tag{14.6}$$

最大流问题的线性规划模型通过以源点流出总流量为目标函数，并对所有中间节点施加流量守恒约束，精确刻画了网络流的动态平衡本质。该模型将最大流问题转化为结构化的数学优化框架，为其理论分析（如对偶理论）及实际算法（如 Ford-Fulkerson 增广路径法）的高效实现提供了坚实基础。

例题 14.1 某物流公司需要优化其货物运输网络，从中央仓库（源点s）向区域配送中心（汇点t）运输货物。运输网络包含 4 个中转节点（A、B、C、D），各运输通道的容量限制如图 14.4 所示（单位：吨/小时）：

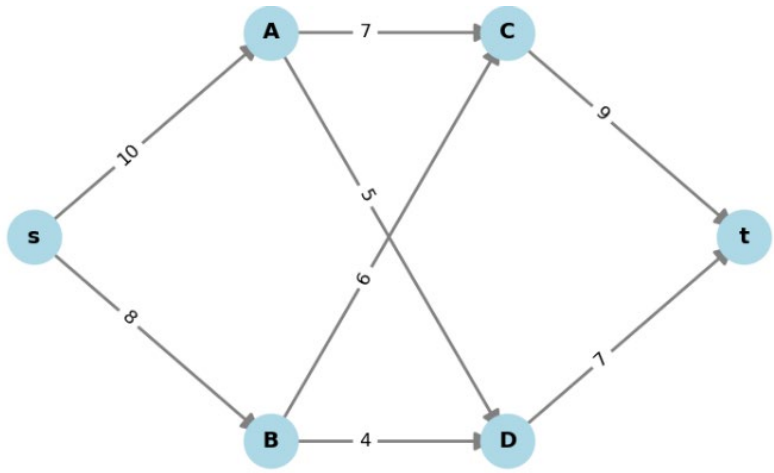

图 14.4 例题 14.1 中的货物运输的网络示意图

解 节点集合$V = \{s, A, B, C, D, t\}$；

有向边集合：$E = \{(s, A), (s, B), (A, C), (A, D), (B, C), (B, D), (C, t), (D, t)\}$；

容量约束：

$$c_{sA} = 10, c_{sB} = 8,$$

$$c_{AC} = 7, c_{AD} = 5,$$

$$c_{BC} = 6, c_{BD} = 4,$$

$$c_{Ct} = 9, c_{Dt} = 7.$$

决策变量f_{ij}是从节点i到节点j的实际流量，$(i, j) \in E$

为最大化总流量v，即源点流出量或汇点流入量：

$$\max v$$

s.t.

$$
\begin{gathered}
f_{sA}+f_{sB}=f_{Ct}+f_{Dt}, \\
f_{sA}=f_{AC}+f_{AD}, \\
f_{sB}=f_{BC}+f_{BD}, \\
f_{AC}+f_{BC}=f_{Ct}, \\
f_{AD}+f_{BD}=f_{Dt}, \\
f_{BD}\le 4, \\
f_{Ct}\le 9, \\
f_{sA}\le 10, \\
f_{sB}\le 8, \\
f_{BC}\le 6, \\
f_{Ac}\le 7, \\
f_{Dt}\le 7, \\
f_{AD}\le 5, \\
f_{ij}\ge 0,\forall (i,j)\in E.
\end{gathered}
$$

对于中间节点（非源点和汇点），进入该节点的流量等于离开该节点的流量。例如对于节点 A 来说，进入 A 的流量=离开 A 的流量，即$f_{sA}=f_{AC}+f_{AD}$。对于整个网络来说，s流出的总流量等于t流入的总流量，即$f_{sA}+f_{sB}=f_{Ct}+f_{Dt}$。容量限制反映实际运输通道的最大能力。流量不能为负，符合实际运输逻辑。

14.4 最大流问题的割与流量定理

在网络流问题中，最大流问题研究如何在有向网络中从源点到汇点传输最大可能的流量。网络的拓扑结构和有向边的容量共同决定了流量的上限，而割集、割量及最大流最小割定理为分析这一上限提供了理论框架。这些性质不仅揭示了流量与网络结构的数学联系，还为优化交通、通信、物流等领域的资源分配提供了科学依据。通过深入理解这些性质，可以有效识别网络瓶颈，设计高效的流量管理方案。

14.4.1 割集与割量

割集指的是在网络图$G=(V,E,C)$中，将顶点集V分为两个不相交的非空子集V_1和V_2，使得源点v_s位于V_1，而汇点v_t位于V_2。割集实际上包括从子集V_1到V_2的所有有向边的集合，通常记作(V_1,V_2)。这些有向边代表了从源点所属集合V_1流向汇点所属集合V_2的所有路径。

割集的核心作用在于，通过“割断”这些有向边，能够将源点和汇点有效隔离开。这意味着，如果割集中的所有有向边都被切断，则源点和汇点之间将没有任

何路径可以传递流量。因此，割集能够直观地反映出网络中的流量瓶颈，也就是限制流量从源点流向汇点的关键边。

在最大流问题中，最小割定理进一步揭示了割集与流量极限之间的重要联系。该定理指出，网络的最大流量实际上等于最小割集的容量。通过识别最小割集，可以确定网络中的瓶颈位置和瓶颈容量。

因此，割集和最小割定理不仅帮助理解网络中流量如何受到结构限制，还为算法设计（如 Ford-Fulkerson 算法）奠定了框架。

割集的意义在于它揭示了网络中哪些部分对流量具有关键的影响。当去掉割集中的所有有向边时，源点和汇点的路径就被完全隔断，这意味着无法再通过网络进行流动。

割量（cut capacity）是指割集(V_1,V_2)中所有有向边的容量之和，记作 C(V_1,V_2)。割量可以理解为一个割集对网络流量的限制能力。其计算方法如下：

$$C(V_1,V_2)=\sum_{(v_i,v_j)\in(V_1,V_2)} c_{ij} \tag{14.7}$$

其中，c_{ij}是有向边(v_i,v_j)的容量。

割量的作用在于它可以作为衡量网络瓶颈的指标。任何一个可行流f的流量$V(f)$都不能超过割量$C(V_1,V_2)$，因为割集中的有向边限制了流量的最大可能值。因此，割集从另一个角度解释了流量的极限在哪里，是网络中影响最大流的“瓶颈”。

对于任何一个网络中的可行流f，其流量$V(f)$不会超过任何一个割集的割量，即：

$$V(f)\le C(V_1,V_2) \tag{14.8}$$

这是因为割集将网络分成了源点和汇点两部分，所有从源点到汇点的流量必须通过割集中的有向边。割集的容量限制了可行流的最大值，因此它自然也限制了最大流的大小。

14.4.2 最大流与最小割定理

在掌握了最大流问题的基本结构和约束条件后，还需关注其最重要的理论基础——最大流最小割定理，这一理论深刻揭示了流量极限与网络结构之间的关系。

最大流最小割定理指出，在任意网络G中，从源点v_s到汇点v_t的最大流量等于v_s和v_t的分隔开所需割除的边权之和的最小值。换言之，最大流等于所有割断源点与汇点的割集中，割容量最小的那个割的容量。

这一定理可以通过“瓶颈”概念来直观理解：想象网络是一个水管系统，水从源点s流向汇点t。系统中最大可能的水流量受限于系统中最窄的“瓶颈”。任何从s到t的割都相当于切断系统中的一组管道，使得水无法从s流到t。最小割就是需要切断的管道容量总和最小的那个割集，代表了系统中最关键的瓶颈位置。

这个定理告诉人们，网络中的最大流量恰好等于这个最小瓶颈的容量。换言之，无论如何巧妙地设计流量方案，都无法突破最小割所代表的瓶颈限制。

在求解最大流问题时，通过 Ford-Fulkerson 算法不断寻找增广路径并最终达到没有增广路径的状态，此时网络中的流量达到了最大值。可以通过以下方法来确定

标记所有可以通过剩余网络到达的顶点集合，记为${V_1}^*$。剩余的未标记顶点构成集合${V_2}^*$。割集$({V_1}^*,{V_2}^*)$即为最小割集，其割量等于最大流的值。

最小割集的概念在实际应用中有重要意义，通过寻找最小割集，可以识别出网络中流量受到限制的部分，从而更好地进行优化。例如，在交通网络中，找到瓶颈道路，并增加其容量，可以大幅提升整体交通流量。

如果想增加网络的最大流量，应首先考虑增加最小割集中有向边的容量。这意味着通过优化最小割集的容量可以直接影响网络的最大流量。

最大流最小割定理表明了最大流量和网络瓶颈之间的密切联系。通过对网络的分析，找到那些限制网络流量的割集，可以帮助工程师和研究人员有针对性地改进网络结构，从而实现更高效的资源配置和流量管理。尤其是在通信、交通、供水等系统中，理解最大流和最小割的关系，有助于找到提高系统性能的最优策

略。

14.5 最大流问题的求解算法

有了理论基础后，实际应用中需要依靠高效的算法来求解最大流问题。下面将介绍几种经典的最大流算法及其原理。增广路算法是解决网络流问题的基础方法，也是最大流算法的核心思想。该算法基于残余网络和增广路的概念，通过迭代寻找可行路径来逐步增加网络总流量，直至达到最大流。

增广路算法的理论基础来自于最大流最小割定理。其基本原理是利用剩余容量不为零的边构建残余网络，然后在残余网络中寻找从源点到汇点的路径（增广路），并沿着这条路径增加流量。每次增广后，更新残余网络，重复这个过程直到找不到增广路为止。

在算法实现中，残余网络扮演着关键角色。对于原网络中的每条边(u,v)，如果当前流量为f，容量为c，则在残余网络中存在容量为$c-f$的正向边(u,v)和容量为f的反向边(v,u)。正向边表示还可以增加的流量，反向边表示可以撤销的流量。这种表示方法使得算法能够灵活调整已分配的流量，确保找到全局最优解。

寻找增广路的方法多种多样，可以使用深度优先搜索、广度优先搜索或其他图搜索算法。不同的搜索策略会影响算法的效率和收敛速度。例如，使用广度优先搜索寻找增广路通常能找到最短的增广路径，这有助于减少迭代次数，提高算法效率。

瓶颈容量是增广路上所有边的剩余容量中的最小值，它决定了此次增广可以增加的流量。在增广过程中，沿着增广路上的所有正向边增加瓶颈容量的流量，同时沿着所有反向边减少相同数量的流量。

增广路算法的一个重要特性是其结束条件：当残余网络中不存在从源点到汇点的路径时，算法终止。此时，可以证明当前流是最大流。通过最大流最小割定理，可以根据残余网络识别出最小割：所有从源点可达的顶点集合与其余顶点之间的边构成最小割。

尽管基本的增广路算法在最坏情况下性能不佳，但它为网络流问题提供了一

个清晰直观的解决框架。基于增广路思想，发展出了多种改进算法，如 Ford-Fulkerson 算法和 Dinic 算法等，这些算法通过不同的策略优化了寻找增广路的方式，提高了算法效率。

增广路算法的优雅之处在于它将复杂的网络流问题简化为一系列路径搜索问题，通过不断迭代逼近最优解。这种思想不仅在网络流算法中应用广泛，也为其他图论问题提供了解决思路。

14.5.1 Ford-Fulkerson 算法

Ford-Fulkerson 标签算法是一种迭代的网络流算法，用于逐步求解网络的最大流。它通过标号寻找增广路径，沿增广路径进行流量调整，直到网络中无法再找到增广路径为止。

Ford-Fulkerson 标签算法包括两大步骤。标号过程，用于寻找从源点到汇点的增广路径；调整过程，沿增广路径调整流量，以增加总流量。每次迭代都包含这两个步骤，直到找不到增广路径为止。

算法步骤

标号过程用于在剩余网络中寻找增广路径，从而找到可以增加流量的路径。

（1）对源点v_s进行标号，初始时对源点v_s标号为(0,+∞)，标号表示流量可传递的来源以及目前的可增广量。在标号时，v_s记为“标号未检查的点”。

（2）选择已标号未检查的顶点v_i。对于每一个已标号但未检查的顶点v_i，寻找其所有未标号的邻接点v_j，按以下规则进行标号：

前向有向边(v_i,v_j)：如果$f_{ij}<c_{ij}$，即有向边(v_i,v_j)还有剩余容量，则给v_j标号$(i,l(v_j))$，其中：

$$l(v_j) = \min(c_{ij} - f_{ij}, l(v_i))$$

代表目前可增量的最小值。

后向有向边(v_j,v_i)：如果$f_{ji}>0$，即可以通过后向有向边减少流量，则给v_j标

号$(-i,l(v_j))$

$$l(v_j)=\min(f_{ji},l(v_i))$$

标号限制：如果f_{ij}=c_{ij}，或者f_{ji}=0，则有向边无法传递流量，不标号。

（3）继续迭代，重复上述过程，直到汇点v_t被标号或者所有标号都检查过且再无可标号的顶点。

如果汇点被标号，说明找到了一条从源点到汇点的增广路径；否则，如果没有顶点可以继续标号，则当前的流量即为最大流。当找到增广路径后，需要沿增广路径调整流量，以增加总流量。

（4）找到增广路径，从汇点v_t开始逆向追踪，找到增广路径上的所有有向边。沿着增广路径从v_t到v_s，根据每个节点的标号追踪前驱节点v_k，直到源点v_s。通过标号过程中的前驱节点关系，可以形成从源点到汇点的完整路径μ。

（5）流量调整，确定调整量θ，该值为汇点标号中的$l(v_t)$，即增广路径上的最小剩余容量。沿增广路径调整流量，对于增广路径中的前向有向边(v_i,v_j)，增加流量：

$$f'_{ij}=f_{ij}+\theta$$

对于增广路径中的后向有向边(v_j,v_i)，减少流量：

$$f'_{ij}=f_{ij}-\theta$$

完成调整后，去掉所有标号，重新进行下一轮的标号过程。

（6）当所有标号都已检查完，而汇点v_t没有被标号时，说明无法找到增广路径。此时当前的流量f即为网络的最大流。

代码见电子资源

下面通过一个例题详细阐述如何使用标号法求解网络中的最大流。

例题 14.2 某通信公司需要优化其基站与数据中心之间的数据传输效率。在

网络高峰期，海量数据需从核心基站（源点v_s）传输至数据中心（汇点v_t），而中间经过多个中继基站（如图 14.5 所示）。公司希望计算当前网络拓扑下，单位时间内可通过的最大数据量（Mbps），以确定是否需要增加链路带宽或调整路由策略。

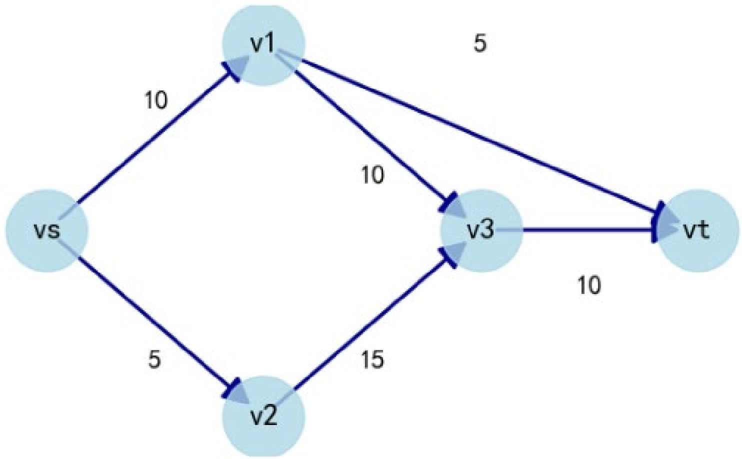

图 14.5 例题 14.2 中的基站与数据中心的网络示意图

解 （1）初始化

将所有边上的初始流量设为 0。

（2）第一次标号过程

对源点v_s标号(0,+∞)。

从源点v_s出发，检查v_1和v_2.对v_1：剩余容量为 10，标号为(v_s,10)。对v_2：剩余容量为 5，标号为(v_s,5)。

检查已标号的节点v_1和v_2，对于v_1，检查邻接点v_3：剩余容量为 10，标号为(v_1, min (10,10)=10)。对于v_2，检查邻接点v_3：剩余容量为 15，标号为(v_2,min(5,15)=5)。

对于节点v_3，可以到达汇点v_t，剩余容量为 10。从v_3标号汇点v_t为(v_3,min(10,10)=10)。

汇点v_t被标号，因此找到了一条增广路径$v_s \to v_1 \to v_3 \to v_t$。

（3）调整过程

确定增广量，增广路径v_s→v_1→v_3→v_t上最小剩余容量为 10，因此增广量θ=10。沿增广路径调整流量，增加流量：

$$f(v_s,v_1)=0+10=10$$

$$f(v_1,v_3)=0+10=10$$

$$f(v_3,v_t)=0+10=10$$

(4)第二次标号过程

重新初始化标号，源点v_s标号为(0,+∞)。

从v_s出发，检查邻接节点：

v_s→v_1：剩余容量 0，无法标号。

v_s→v_2：剩余容量 5，v_2标号为(v_s,5)。

检查已标号节点v_2的邻接节点：

v_2→v_3：剩余容量 15，v_3标号为(v_2,min(5,15)=5)。

检查v_3的邻接节点v_t：

v_3→v_t：剩余容量 0，无法标号。

沿反向边v_3→v_1（等效释放v_1→v_3的已用容量），给v_1标号(v_3,5)（可增广量 min(5,10)=5）。

沿边v_1→v_t（剩余容量 5），给v_t标号(v_1,5)（可增广量 min(5,5)=5）。

流量调整：

增广量θ=5，沿路径增加流量：$f(v_s,v_2)$=5，$f(v_2,v_3)$=5，$f(v_1,v_t)$=5；同时减少v_1→v_3的流量（等效反向边调整）。

剩余容量更新：v_s→v_2剩余容量 0，v_2→v_3剩余容量 10，v_1→v_t剩余容量 0。

(5)终止条件

源点v_s所有出边剩余容量为 0（无法再增广），算法结束。

总流量=$f(v_s,v_1)$+$f(v_s,v_2)$=10+5=15。

14.5.2 Dinic 算法

Dinic 算法是求解最大流问题的一种高效算法，尤其在处理稀疏图时表现出色。该算法由以色列数学家 Ephraim Dinic 于 1970 年提出，在稀疏图上运行较快。Dinic 算法是基于分层网络的增广路径算法，通过逐层分层并在分层网络上寻找阻塞流来加速寻找增广路径的过程。阻塞流是指在当前分层网络中，无法再通过增广路径增加流量的流，即每条从源点到汇点的路径上都至少有一条边已经饱和（流量等于容量）。

Dinic 算法基于分层网络的思想。它的主要过程包括以下几步。

构建分层网络（层次图）。在每次迭代中，从源点出发，对原图进行广度优先搜索，根据节点到达源点的最短距离对图进行分层，构造出一个层次图。在层次图中，所有从一个层到下一个层的边才会被保留。

寻找阻塞流。在分层网络上进行深度优先搜索，找到增广路径。若找到从源点到汇点的增广路径，就沿此路径尽量多地增加流量，并在分层网络中标记该路径上的边为满流状态，构成一个“阻塞流”。

迭代，删除已满流的边，重复以上步骤，重新构建分层网络和寻找阻塞流，直到无法再构造出新的分层网络为止。

Dinic 算法的步骤

(1)初始化

Dinic 算法首先从零流开始，即所有边上的初始流量均为 0。此时，所有边的剩余容量等于其初始容量，构建了一个初始的残差网络。残差网络包括所有当前可用的边，以及具有剩余容量的反向边。

残差网络是在最大流问题中，在当前流量分配状态下表示网络剩余可调整空间的辅助图结构；它包含两类边：前向边表示原网络中还可以增加的流量空间（容量减去当前流量），反向边表示已分配的可以撤销重新规划的流量；通过在残差

网络中不断寻找从源点到汇点的路径（增广路径）并调整流量，算法能够灵活地重新分配资源，既可增加新流量也可撤销前期的分配决策，直到找不到增广路径时，当前流即为最大流。

(2)构建分层图

在 Dinic 算法中，构建分层图是一个重要的步骤，用于在当前残差网络中对节点进行分层，使得后续增广路径的寻找更加高效。分层图的构建依赖广度优先搜索，具体步骤如下：将源点s标记为第 0 层，将其加入到队列中。随后通过广度优先搜索遍历网络。依次从队列中取出一个节点u，遍历从该节点出发的所有邻接节点v。如果节点v尚未被标记，并且边(u,v)在残差网络中有剩余容量，则标记v为u的下一层，同时将节点v加入队列中。此过程持续进行，如果汇点t被标记，则分层图构建完成；如果无法到达汇点t，则说明当前残差网络中不再存在从源点到汇点的路径，算法结束。

构建分层图的目的是将网络中的边按照到源点的最短距离进行划分，使得在寻找增广路径时可以尽可能沿着较短路径进行流量增广，从而提高求解效率。

(3)在分层图中寻找增广路径

在分层图中，Dinic 算法使用深度优先搜索寻找从源点s到汇点t的增广路径。增广路径的寻找过程如下：从源点 s 开始寻找增广路径。使用深度优先搜索从源点s出发，沿着未满流且符合分层条件的边向前搜索，直至到达汇点t。增广路径上的每一条边都必须是从当前层到下一层，且具有剩余容量。对于找到的增广路径，确定路径上所有边的最小剩余容量，称之为增广量。增广量表示可以在该路径上增加的最大流量。沿着增广路径将增广量添加到每条前向边的流量中，并在每条反向边上减去相应的增广量。这样做确保了后续的增广路径可以根据需要进行流量回退。不断在分层图中寻找增广路径，直至无法找到新的增广路径为止，此时的流量增广结束，形成阻塞流。

(4)阻塞流的定义与更新

阻塞流是在当前分层图中，无法再找到任何从源点到汇点的增广路径的状态。在找到阻塞流后，表明当前分层网络中的所有增广机会已经用尽。当找到阻塞流

后，需要重新构建分层图，更新残差网络中每条边的剩余容量，并重新通过广度优先搜索对网络进行层次划分。重新进行广度优先搜索，构建新的分层图，以便在新的分层网络中寻找更多的增广路径。

(5)重复过程直至没有新的增广路径

重复上述构建分层图和寻找增广路径的过程，直到无法从源点到达汇点，即当前残差网络中不再存在增广路径为止。当广度优先搜索无法从源点s到达汇点t时，算法终止。此时，当前的流量即为整个网络的最大流。

代码见电子资源

例题 14.3 某地区计划建设一个新的通信网络，将数据从源点 S 传输到目标节点 T。该网络中的各个节点代表不同的传输站点，边代表通信光纤链路，每条光纤链路有其最大带宽容量。要求使用 Dinic 算法求解从源点 S 到汇点 T 的最大数据传输量，并详细给出求解过程。

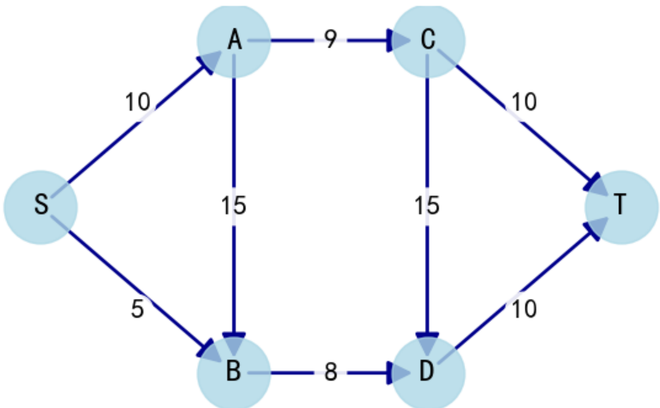

图 14. 6 例题 14. 3 中的通信网络的网络示意图

解 （1）初始设置

所有的流量初始值设为 0。构建网络图，记录所有边的容量和初始流量。

（2）构建分层图

从源点 S 出发，利用广度优先搜索给节点分层，分层的目的是在当前剩余网络中找到所有可行的增广路径。

第一轮广度优先搜索。S 被标记为第 0 层。从 S 出发，S->A(容量 10)：A 被

标记为第 1 层。S->B(容量 5)：B 被标记为第 1 层。

继续从第 1 层的节点扩展，A->C(容量 9)：C 被标记为第 2 层。A->B(容量 15)：但 B 已经在第 1 层，不需要标记。B->D(容量 8)：D 被标记为第 2 层。

从第 2 层的节点扩展，C->D(容量 15)：D 已经在第 2 层，不需要重新标记。C->T(容量 10)：T 被标记为第 3 层。D->T(容量 10)：T 已经在第 3 层，不需要重新标记。

分层图完成，层次结构分别为第 0 层 S，第 1 层 A、B，第 2 层：C、D，第 3 层：T。

（3）在分层网络中寻找增广路径

通过深度优先搜索，在分层图中寻找增广路径并计算阻塞流。增广路径 1：S->A->C->T。最小残余容量$min(10,9,10)=9$。沿该路径增加 9 的流量。更新残差网络，S->A 的容量减少为 10-9=1。A->C 的容量减少为 9-9=0。C->T 的容量减少为 10-9=1。

增广路径 2：S->B->D->T。最小残余容量$min(5,8,10)=5$。沿该路径增加 5 的流量。更新残差网络，S->B 的容量减少为 5-5=0。B->D 的容量减少为 8-5=3。D->T 的容量减少为 10-5=5。

（4）构建新的分层图

由于所有可能的增广路径已被找到，重新构建分层图。

从源点 S 出发，S->A(容量 1)：A 被标记为第 1 层。S->B(容量 0)：容量为 0，无法继续。从 A 扩展，A->B(容量 15)：B 被标记为第 2 层。A->C(容量 0)：容量为 0，无法继续。

从 B 扩展，B->D(容量 3)：D 被标记为第 3 层。从 D 扩展，D->T(容量 5)：T 被标记为第 4 层。

分层图更新为第 0 层 S，第 1 层 A，第 2 层 B，第 3 层 D，第 4 层 T。

（5）寻找增广路径

增广路径 3：S->A->B->D->T。最小残余容量$min(1,15,3,5)$=1。沿该路径增加 1 的流量。更新残差网络：S->A 的容量减少为 1-1=0。A->B 的容量减少为 15-1=14。B->D 的容量减少为 3-1=2。D->T 的容量减少为 5-1=4。

（6）构建新的分层图

从源点 S 出发，S->A(容量 0)：容量为 0，无法继续。S->B(容量 0)：容量为 0，无法继续。由于 S 无法标记任何其他节点，广度优先搜索结束，分层图中 T 无法被标记到，说明没有增广路径。

（7）计算最大流量

增广路径 1 流量为 9，增广路径 2 的流量为 5，增广路径 3 的流量为 1。总最大流量=9+5+1=15。

14.6 本章小结

在本章中，探讨了最大流问题的核心概念、基本性质以及求解方法。从最基础的流量守恒和容量约束，到 Ford-Fulkerson 以及 Dinic 等经典算法的详细介绍，看到了最大流问题在运筹学领域的重要性。通过引入增广路径、残差网络、分层图等工具，最大流问题得以在复杂网络中高效求解。此外，最大流与最小割定理的揭示进一步明确了最大流量与网络瓶颈之间的内在联系，这为理解网络系统中的瓶颈分析和资源优化提供了理论支持。

最大流问题不仅在数学研究中有重要地位，在实际生活中也有广泛的应用。例如，交通流管理、物流配送优化、电力网络调度和通信网络带宽分配等都是典型的最大流应用场景。通过使用最大流模型，能够提高系统的效率，优化资源配置，并减少不必要的阻塞与浪费。最大流问题的研究将继续受到诸多领域的关注。随着网络规模和复杂度的增加，传统算法在时间复杂度和计算效率方面面临挑战。因此，算法优化和新方法的探索将是未来的研究重点。尤其是在大规模稀疏图和动态网络环境中，更高效的增广路径寻找方法和并行计算策略可能会成为主流研究方向。此外，最大流问题的应用范围也在不断扩大，包括在人工智能、数据挖掘以及物联网领域的应用，这些新兴领域的需求将推动最大流理论与实践的发展。

总的来说，最大流问题是运筹学中一个经典且充满活力的研究方向。理解其基本原理、掌握各种求解算法，并将其应用于现实场景中，不仅有助于解决复杂的网络流问题，还为更广泛的系统优化和资源管理提供了强有力的工具。

研学互通

最大流问题作为网络优化理论的核心内容之一，自 20 世纪中叶被正式提出以来，已经发展成为运筹学、计算机科学、工程管理等多个领域的基础性研究方向。最大流问题不仅在理论研究中具有重要地位，其高效的求解算法和建模方法也广泛应用于交通运输、通信网络、供应链管理、图像分割、数据挖掘等实际场景。计算机技术和大规模数据分析的发展，对大规模、复杂网络中最大流问题的高效求解提出了更高要求，推动了相关算法的持续创新与理论完善。

在最大流问题的研究历程中，众多学者提出了具有里程碑意义的算法与理论成果，如 Ford-Fulkerson 算法、Dinic 算法、Edmonds-Karp 算法、Push-Relabel 算法等。这些算法不仅极大地提升了最大流问题的求解效率，也为最小割、多商品流等更复杂的网络优化问题提供了理论基础。此外，网络流理论与图论、线性规划、组合优化等学科的交叉融合，进一步拓展了其应用边界和研究深度。

为了帮助读者更好地理解最大流问题的理论基础、算法发展和应用前沿，以下精选了多篇经典论文和权威著作。这些文献系统梳理了最大流问题的起源、算法演进、复杂度分析及实际应用案例，是深入学习和拓展研究视野的重要参考资料。建议读者结合自身兴趣，循序渐进地阅读和思考，既夯实理论基础，也关注算法的工程实现与跨学科应用。

(1) Ford, L. R., & Fulkerson, D. R. (1956). Maximal flow through a network. Canadian Journal of Mathematics, 8(3), 399－404.

这篇开创性论文首次系统性地提出了最大流问题的数学模型和求解算法，被誉为网络流理论的奠基之作。作者提出的 Ford-Fulkerson 算法利用增广路径思想，通过不断寻找从源点到汇点的可行路径并沿路径增加流量，直到无法再增广为止，从而得到最大流。该算法不仅为最大流问题提供了理论基础，也为最小割问题等后续网络优化问题奠定了分析框架。适合希望了解网络流理论起源、算法设计思

想及其实际价值的读者。

(2) Dinic, E. A. (1970). Algorithm for solution of a problem of maximum flow in a network with power estimation. Soviet Math. Doklady, 11, 1277 – 1280.

Dinic 在这篇论文中提出了著名的 Dinic 算法，是最大流问题的高效解法之一。其核心思想在于采用分层图（Level Graph）和阻塞流（Blocking Flow），通过广度优先搜索构建层次网络，再用深度优先搜索在层次网络中寻找阻塞流。每轮增广都能显著推进流量，极大提升了算法效率。适合希望掌握最大流高效算法、理解分层增广方法的进阶读者。

(3) Edmonds, J., & Karp, R. M. (1972). Theoretical improvements in algorithmic efficiency for network flow problems. Journal of the ACM, 19(2), 248 – 264.

Edmonds-Karp 算法是 Ford-Fulkerson 算法的多项式时间实现。论文的创新点在于每次用广度优先搜索寻找最短增广路径，显著提升了算法的理论效率，避免了某些特殊情况下的非多项式行为。该论文不仅详细分析了算法的复杂度，还对网络流问题的理论边界进行了讨论。Edmonds-Karp 算法成为后续网络流算法复杂度分析的重要参照标准，被广泛用于理论研究和实际计算中，同时也促进了算法设计与图论、组合优化的进一步融合。

(4) Goldberg, A. V., & Tarjan, R. E. (1988). A new approach to the maximum flow problem. Journal of the ACM, 35(4), 921 – 940.

Goldberg 和 Tarjan 在这篇论文中提出了前推-重标（Push-Relabel）算法，也称预流推进（Preflow-Push）算法。与以往基于增广路径的方法不同，该算法引入了“预流”和“顶点高度”概念，通过“推送”和“重标”操作动态调整流量，显著提升了大规模网络的求解效率。Push-Relabel 算法在理论和工程应用中均表现出极高的性能，尤其适合处理顶点数和边数都很大的稠密网络。该论文还对算法的实现细节、复杂度分析、实际应用等进行了系统阐述，是网络流优化领域的重要里程碑。

(5) Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). Network flows: Theory, algorithms, and applications. Prentice Hall.

这本经典教材系统全面地介绍了网络流理论、算法实现与实际应用。书中不仅涵盖最大流、最短路径、多商品流、广义流等核心问题，还深入讲解了算法设计原理、复杂度分析、数据结构优化等内容。每章配有丰富的案例分析和习题，便于读者理论联系实际。作者还结合工业界和学术界的最新进展，介绍了网络流在交通、物流、通信、金融等领域的应用。无论是本科生、研究生还是工程技术人员，这本书都是学习和研究网络流优化不可或缺的权威参考。

思行经世：最大流问题驱动科技强国

最大流问题作为网络流理论的核心，研究在有向图中从源点到汇点传输的最大可能流量。该方法在交通调度、通信网络设计和能源分配等实际场景中具有广泛应用，为解决复杂系统优化问题提供了高效工具，助力我国在信息化和智能化领域实现技术突破与产业升级。

在现代物流体系中，最大流问题为物资调配提供了科学支撑。以京津冀协同发展为例，物流部门通过最大流模型优化区域间的物资运输网络，确定从北京到天津、河北的货物最大流量路径，确保医疗物资、粮食等关键资源的高效配送。模型通过分析道路容量和运输需求，计算最优流量分配，提升了配送效率。这一应用体现了“协同发展、服务民生”的区域精神，为京津冀一体化战略注入了科技动力。

在能源传输领域，最大流问题助力了电力网络的优化。以西电东送工程为例，国家电网通过最大流模型优化电力从西部水电站到东部城市的传输路径，最大化输送容量并减少损耗。Ford-Fulkerson 算法通过迭代增广路径，确保了电力资源的高效分配，满足了东部经济发展的用电需求。这一科学调度体现了“统筹全局、绿色发展”的能源精神，为我国能源安全和可持续发展提供了保障。

在信息通信领域，最大流问题为数据传输提供了高效方案。以中国电信的骨干网优化为例，运营商通过最大流模型优化数据流量分配，确保从核心服务器到终端用户的数据传输速率最大化。模型通过分析网络带宽和流量需求，找到最优路径，保障了视频会议、在线教育的流畅运行。这一技术支持体现了“科技为民、互联互通”的通信精神，为数字中国建设奠定了坚实基础。

最大流问题的每一次优化，都是资源高效利用与国家精神的交汇。京津冀物资调配的科学路径诠释了“协同发展”，西电东送的电力传输彰显了“绿色发展”，骨干网优化的数据流保障体现了“科技为民”。这些实践不仅提升了物流、能源和通信领域的效率，还激励读者在科技强国建设中以科学方法驱动未来。最大流问题作为网络流理论的基石，深刻融入我国的区域发展、能源调度和数字建设事业，为实现中华民族伟大复兴贡献了智慧与力量。

习题

习题 14.1 某城市计划改造道路，为评估主干道的通行能力，需计算从城西（源点 S）到城东（汇点 T）的最大交通流量。交通网络以路口为节点，道路通行能力（车辆/小时）为边的容量。给定网络包含节点 S、A、B、C、D、T，以及以下边及其容量：S→A(10)，S→B(15)，A→C(5)，A→D(10)，B→D(10)，C→T(10)，D→T(15)。请使用 Ford-Fulkerson 算法，逐步找出所有增广路径，并求解从 S 到 T 的最大交通流量.

习题 14.2 某小区的供水系统需要重新规划，以确保供水网络可以满足居民的日常需求。供水网络包括多个节点（如供水站和居民区连接点），每条水管的容量代表其最大输水能力。给定供水网络，节点包括：S（供水源头）、A、B、C、D、T（居民区接收端）。各节点之间水管的容量如表 14.1 所示，请使用 Dinic 算法求解从 S 到 T 的最大供水流量。

表 14.1 供水网络各节点间水管容量

起点	终点	水管容量（单位：m^3/h）
S	A	8
S	B	12
A	C	4
A	D	5
B	C	7
B	D	6
C	T	10

起点	终点	水管容量（单位：m^3/h）
D	T	8

习题 14.3 某区域发生自然灾害，需要从救援物资集散中心 S 向受灾最严重的城镇 T 紧急运送一批物资。现有若干中转站（A,B,C）和连接各点的道路。由于道路状况不同，每条道路每天能够通过的运输车辆数目是有限的。

集散中心 S 可以无限量供应物资。道路及日最大通行车辆数（辆/天）(这里假设单向)如下图 14.7 所示。

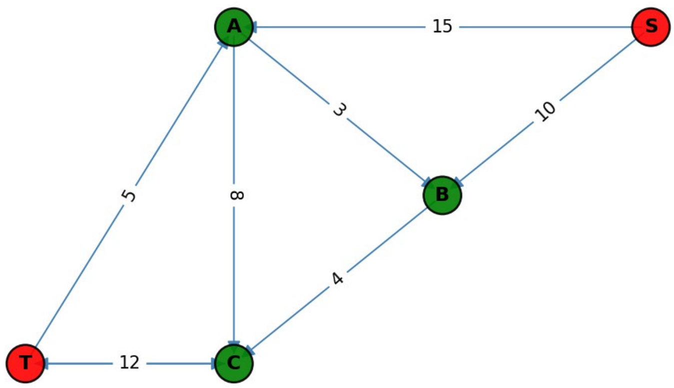

图 14.7 习题 14.3 中的救援物资运输的网络示意图

（1）将此紧急物资调配问题建模为一个最大流问题。

（2）计算每天最多能从集散中心 S 运送多少辆车的物资到达城镇 T？