第 13 章 最小生成树问题

在当今信息化与智能化迅猛发展的时代，如何以最优成本高效地连接分散的节点，已成为交通、通信、电力、物流等众多领域的核心问题。最小生成树（Minimum Spanning Tree）作为图论与运筹学中的基础模型，正是解决此类网络优化难题的有力工具。无论是在城市电网的布局、5G 基站的连接，还是在智慧物流与应急救援网络的设计中，最小生成树模型都能为决策者提供科学的资源配置方案，最大程度地节约建设成本、提升系统效率。

最小生成树问题不仅以其简洁的定义和优雅的数学性质成为算法设计的经典范例，更通过 Kruskal、Prim 等贪心算法的实现，展现了运筹学在实际工程中的应用价值。随着网络规模和复杂性的不断提升，最小生成树问题也衍生出多种变种，如带约束的最小生成树、二次最小生成树和多目标优化等，进一步拓展了其理论边界和实际应用场景。

本章将系统介绍最小生成树的基本概念、数学模型和经典算法，剖析其在现实世界中的广泛应用，并探讨其变种问题与前沿发展。通过理论与案例的结合，帮助读者理解最小生成树在优化决策中的核心地位，为后续深入学习网络优化与图算法奠定坚实基础。

13.1 最小生成树问题介绍与基本概念

13.1.1 最小生成树问题的背景介绍

想象你是一名城市规划师，面对一张标有多个居民区的地图，每个居民区之间可以修建道路，但不同道路的建设成本差异巨大。你的任务是设计一个方案，使得所有居民区都能相互到达，同时总建设成本最低——这本质上就是在求解一张图的最小生成树问题。

最小生成树问题是运筹学与图论中的经典问题，它在通信网络设计、电力传输优化、物流管道铺设等领域有着广泛的应用。其核心目标是在一个带权连通无向图中，选择一组边连接所有顶点，并保证这些边的总权重最小。这个问题看似简单，但其解决过程中蕴含的贪心策略与数学证明思想，是算法设计与优化的精

髓之一。

实际上，最小生成树问题的研究可以追溯到 20 世纪初，最早由捷克数学家 Otakar Borůvka 在 1926 年提出，用于解决电力网络的最优布线问题。此后，随着计算机科学和运筹学的发展，最小生成树问题逐渐成为网络优化领域的基础模型，并催生了大量高效算法，如 Kruskal 算法、Prim 算法等。

13.1.2 基本概念

在运筹学的学习中，许多优化问题都依赖于图论中的核心概念，这些概念不仅提供了分析复杂网络的工具，还在实际应用中发挥着重要作用，比如通信网络设计、物流配送路径规划等。接下来，将深入探讨一些基本概念，包括树、带权图、生成树和最小生成树，通过清晰的引出和详细的描述，帮助读者建立直观的理解。

先从“树”这一基础结构开始探讨。想象一棵现实中的大树，枝干从根部延伸，层层分叉，覆盖所有叶片，却从不交错形成闭合的环。在图论中，树是一种特殊的图结构，它以简洁的方式连接节点，满足两个关键条件，一是连通性，即树中的任意两个节点之间都存在一条路径，确保整个结构是一个整体；二是无圈性，即树中没有任何闭合路径，沿着任何路径都不会回到起点。这两个条件赋予了树独特的简洁性。一个重要的性质是，对于一棵包含n个节点的树，它的边数总是$n-1$。例如，一棵有 11 个节点的树，会有 10 条边，节点通过这些边形成一个无冗余、无循环的结构。这种性质的背后逻辑是，如果边数不足，节点无法全部连通；如果边数过多，就会形成圈，破坏树的定义。树的这种简洁特性使其在文件系统、网络拓扑设计等场景中广泛应用，也为后续更复杂的结构奠定了基础。

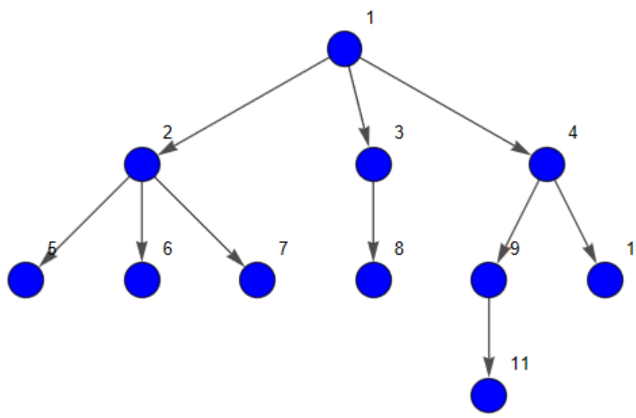

图 13.1 树的结构

接下来，将目光转向带权图，这是一个在优化问题中更为常见的结构。如果说普通的图是通过点和边来描述节点之间的关系，那么带权图则为这些关系增添了“代价”或“意义”。具体而言，带权图是一种图，其中每条边都被赋予了一个非负的权值，这个权值可能代表距离、成本、时间或其他与应用场景相关的量。例如，在一张城市地图中，节点代表路口，边代表道路，而边的权值可能是道路的长度或通行时间。带权图可以是有向的（边的连接为单向）或无向的（边的连接为双向），其灵活性使其广泛应用于路径规划、物流优化等领域。在带权图中，每条边e的权值通常记为$w(e)$，可以是整数、浮点数或其他数值类型。一个重要的应用是计算子图的权值总和，比如一条路径或一棵树的总权值，这在寻找最短路径或最小成本网络时尤为关键。带权图的引入，能够从单纯的拓扑关系转向更实际的量化分析，为后续的优化问题奠定了基础。

在理解了树和带权图后，可以进一步探讨生成树这一重要概念。生成树是从一个连通图中提取出的子图，它以最简洁的方式连接所有节点，形同一个网络的“骨架”。具体来说，对于一张包含n个节点的连通图，生成树通过选取$n-1$ 条边，确保所有节点连通，同时避免形成任何圈。这与树的性质一致，因为生成树本质上就是图中的一棵树，只不过它是从更复杂的图结构中精心挑选出来的。例如，想象一张城市交通网络，节点是路口，边是道路，生成树的任务是选择最少的道路连接所有路口，同时避免形成循环路线。在实际应用中，生成树常用于设计电力网络或通信网络，确保所有节点互联，同时消除冗余连接。生成树的边数为何固定为$n-1$？这是因为它继承了树的连通性和无圈性，任何偏离这一边数的选择都会导致不连通或形成圈。生成树的独特价值在于，它提供了一种从复杂网络中提炼核心结构的方法，为后续优化问题，如最小生成树的计算，铺平了道路。

聚焦于最小生成树，这是在带权连通图中一个极为重要的概念。不同的生成树可能包含不同的边，而每条边都有其权值，因此生成树的总权值（即所有边权值之和）可能不同。最小生成树的目标是找到一棵生成树，使得其总权值在所有可能的生成树中最小。换句话说，它是在保证所有节点连通的前提下，以最低的“成本”构建网络的方案。形式化地描述，考虑一张无向图G=(V,E)，其中V是顶点

集合，E是边集合，每条边$e(u,v)$具有一个权值$w(u,v)$。最小生成树问题要求找出一个边集合$T \subseteq E$，满足两个条件：一是T形成的子图连接了所有顶点且无圈，二是T中所有边的权值之和最小。这样的边集合T就构成了图G的最小生成树。在实际应用中，最小生成树广泛用于网络设计，比如在铺设光纤网络时，选择最短的电缆长度连接所有节点，从而降低建设成本。

通过对树、带权图、生成树和最小生成树的探讨，可以看到这些概念如何从简单的拓扑结构逐步过渡到复杂的优化问题。它们不仅是图论的基石，也是运筹学中解决实际问题的重要工具。在后续章节中，将进一步探讨如何利用算法（如 Kruskal 算法或 Prim 算法）高效地构造最小生成树，以及这些概念在更广泛的应用场景中的作用。

满足以上条件的T称为图G的最小生成树。（图 13.2 中红线所连接的部分即为最小生成树）。

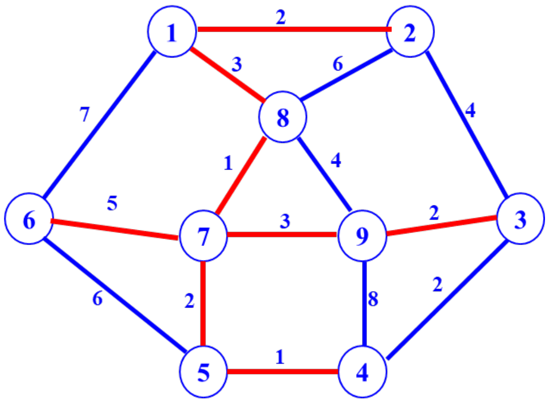

图 13.2 最小生成树结构

13.1.3 最小生成树问题的数学模型

在最小生成树问题中，希望在一个带权连通图$G=(V,E)$中找到连接所有顶点且权值总和最小的生成树。对于该问题，可以用数学优化的形式来描述，具体如下，定义目标函数为所有选择的边权重的总和，表示为：

$$\min\ z = \sum_{i=1}^{m} w_i x_i \tag{13.1}$$

其中，x_i是一个 0-1 变量，用于表示边e_i是否被选择。

$s.t.$

$$x_i = \begin{cases} 1, & \text{若选择边} e_i \\ 0, & \text{否则}. \end{cases} \tag{13.2}$$

$$\forall i \in \{1,2,\dots,m\}.$$

$$x \in T.$$

约束条件是生成树必须连通所有的节点，因此选择的边集x必须属于图G的生成树集合T，即$x \in T$。每条边的选择只能是 0 或 1，即$x_i \in \{0,1\}$。

其中T为图G中所有生成树的集合。w_i为边e_i的权重，m为图中边的总数。

最小生成树的特点

(1)最小生成树问题是一类典型的网络问题，涉及如何在一个带权连通图中找到最优路径以连接所有节点。该问题的求解对许多网络设计、交通布局和电力传输等领域具有重要应用。

(2)最小生成树不一定唯一。如果图中存在多条权值相同的边，可能会有多个不同的最小生成树。

(3)对于任何一个带权连通图，最小生成树的边权值总和小于或等于图中任何其他生成树的边权值总和。

13.2 最小生成树问题常见求解方法

本章重点讲解两种经典的贪心算法——Prim 算法与 Kruskal 算法。这两种算法理论严谨性高、应用广泛，且是理解组合优化与贪心策略的范本，其他方法（如启发式算法）将在后文简要提及。

13.2.1 Prim 算法

Prim 算法（1957 年）是由 RobertC.Prim 提出，通过从一个顶点开始逐步扩展生成树的方式找到最小生成树。

Prim 算法的核心思想是贪心法，每次选择当前生成树中连接权值最小的边，

使得生成树的权值和逐步趋于最小。该算法保证生成树的连通性和无环性，最终得到一个权值和最小的生成树。

算法步骤

(1)初始化

从图中任意选择一个节点作为生成树的起始节点，将其加入生成树的顶点集U。初始化一个空的边集T，用于存储生成树中的边。

(2)选择边

找到与生成树中节点集U连接的所有边中权值最小的边(u,v)，其中$u \in \mathrm{U}$且$v \in V - U$（即v还未加入生成树）。将边(u,v)加入生成树的边集T，并将顶点v加入U。

(3)重复选择

重复步骤 2，直到生成树的顶点集U包含图中的所有节点，即$U = V$。

(4)输出结果

得到包含$n-1$条边的生成树，输出生成树的边集T和总权值。

代码见电子资源

例题 13.1 某通信公司需在 5 个基站之间铺设光缆，确保所有基站互联。每条潜在光缆路线的铺设成本（单位：万元）如图 13.3 所示：

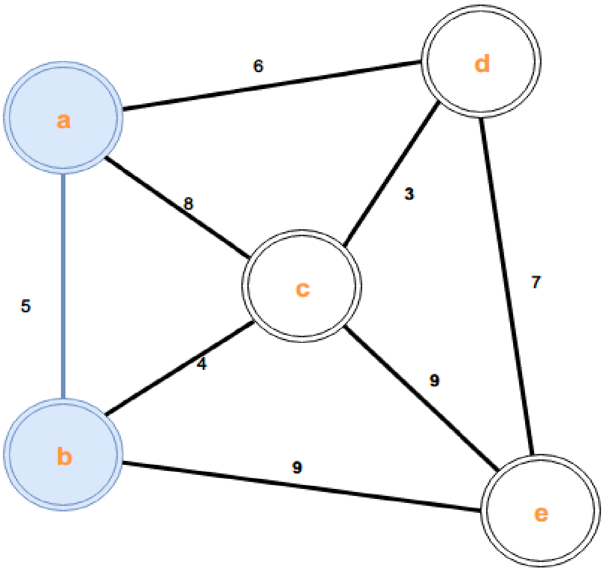

图 13.3 例题 13.1 中的铺设成本的网络图

解 采取 Prim 算法求解该最小生成树问题。

1. 初始化

    起始节点 a

    当前生成树的顶点集合：{a}

    当前生成树的边集合：∅。

2. 从节点 a 出发，选择最小权重的边：

    可选边为(a,b,5)、(a,d,6)、(a,c,8)

    选择(a,b,5)

    更新生成树的顶点集合：{a,b}

    更新生成树的边集合：{(a,b,5)}。

3. 从{a,b}中选择权重最小的边：

    可选边为(a,d,6)、(a,c,8)、(b,c,4)、(b,e,9)

    选择(b,c,4)

更新生成树的顶点集合：{a,b,c}

更新生成树的边集合：{(a,b,5),(b,c,4)}。

4. 从{a,b,c}中选择权重最小的边：

   可选边为(a,d,6)、(c,d,3)、(b,e,9)、(c,e,9)

   选择(c,d,3)

   更新生成树的顶点集合：{a,b,c,d}

   更新生成树的边集合：{(a,b,5),(b,c,4),(c,d,3)}。

5. 从{a,b,c,d}中选择权重最小的边：

   可选边为(d,e,7)、(b,e,9)、(c,e,9)

   选择(d,e,7)

   更新生成树的顶点集合：{a,b,c,d,e}

   更新生成树的边集合：{(a,b,5),(b,c,4),(c,d,3),(d,e,7)}。

6. 生成树已包含所有顶点，算法结束。

最小生成树结果

最小生成树的边：{(a,b,5),(b,c,4),(c,d,3),(d,e,7)}。最小生成树的总权重：5+4+3+7=19。

13.2.2 Kruskal 算法

Kruskal 算法（1956 年）是由 Joseph Kruskal 提出，通过边排序和选择最小边的方式构建最小生成树。

Kruskal 算法以边为单位构建生成树，每次选择当前最小权值的边，将其加入生成树，同时使用并查集来确保边的加入不会产生环。最终，Kruskal 算法能保证得到一棵权值总和最小的生成树。

并查集（Union-Find）是一种高效的数据结构，用于解决动态连通性问题，

特别是在图论中判断两个节点是否属于同一连通分量，并支持快速合并连通分量。它通过两个核心操作实现：查找（Find）用于定位某个节点所属集合的代表元素，合并（Union）用于将两个集合合并为一个。

Kruskal 算法步骤

(1)将图中所有的边按权值升序排序。初始化一个空集合T用于存储生成树的边集。为每个顶点创建一个单独的集合（使用并查集管理）。

(2)按照升序依次遍历（1）中所得到边集中的元素。对每条边(u,v)，检查它的两个端点是否属于不同的集合（即是否属于不同连通分量），如果属于不同集合，将边(u,v)加入生成树集合T，并将两个顶点的集合合并（即使用并查集的合并操作）；如果属于同一集合，跳过该边（因为加入该边会形成环）。

(3)重复步骤 2，直到生成树包含$n-1$ 条边（n是图的顶点数）。

(4)输出生成树的边集T及其总权值。Kruskal 算法通常使用并查集（Union-Find）数据结构来管理节点的连通性。通过路径压缩和按秩合并，并查集可以高效地判断两个顶点是否属于同一集合，并在合并集合时优化性能。

在查找操作中，通过路径压缩将节点直接连接到其所在集合的根节点，从而加速后续的查找。在合并两个集合时，将节点数较少的集合合并到节点数较多的集合，从而减少树的高度，提升效率。并查集的查找和合并操作的时间复杂度接近常数，为$O(\alpha(V))$，其中α是阿克曼函数的反函数。

代码见电子资源

例题 13.2 某地区需在 7 个变电站之间架设输电线路，确保所有变电站互联。每条线路的架设成本（单位：百万元）如图 13.4 所示：

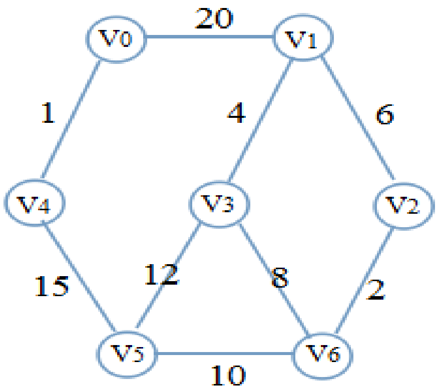

图 13.4 例题 13.2 中的架设成本的网络图

解 （1)首先按边权重从小到大排序，以便从最小权重的边开始构建生成树。排序后的边列表如下。

$[(V_0, V_4, 1), (V_6, V_2, 2), (V_1, V_3, 4), (V_2, V_1, 6), (V_3, V_6, 8), (V_5, V_6, 10), (V_5, V_3, 12),$

$(V_4, V_5, 15), (V_0, V_1, 20)]$。建立一个空的最小生成树边集合 MST=[]，并初始化总权重 weight=0。使用并查集来追踪连通性，以避免环路的形成。

（2）逐步添加边。选择边(V_0,V_4,1),检查V_0和V_4是否属于同一连通分量。由于它们不属于同一分量，因此加入这条边。更新并查集，使V_0和V_4处于同一分量。更新 MST=[(V_0,V_4,1)]，weight=1。

选择边(V_6,V_2,2),检查V_6和V_2是否属于同一连通分量。它们不属于同一分量，因此加入这条边。更新并查集，使V_6和V_2处于同一分量。更新 MST=[(V_0,V_4,1),(V_6,V_2,2)]，weight=3。

选择边(V_1,V_3,4),检查V_1和V_3是否属于同一连通分量。它们不属于同一分量，因此加入这条边。更新并查集，使V_1和V_3处于同一分量。更新 MST=[(V_0,V_4,1),(V_6,V_2,2),(V_1,V_3,4)]，weight=7。

选择边(V_2,V_1,6),检查V_2和V_1是否属于同一连通分量。它们不属于同一分量，因此加入这条边。更新并查集，使V_2和V_1（连同V_3和V_6）处于同一分量。更新 MST=[(V_0,V_4,1),(V_6,V_2,2),(V_1,V_3,4),(V_2,V_1,6)]，weight=13。

选择边(V_3,V_6,8),检查V_3和V_6是否属于同一连通分量。由于它们已经处于同一

分量（通过(V_1,V_3)和(V_2,V_1)连接到一起），跳过该边，以避免形成环。

选择边$(V_5,V_6,10)$,检查V_5和V_6是否属于同一连通分量。它们不属于同一分量，因此加入这条边。更新并查集，使V_5和V_6（连同V_2、V_1、V_3）处于同一分量。更新 MST=[$(V_0,V_4,1)$,$(V_6,V_2,2)$,$(V_1,V_3,4)$,$(V_2,V_1,6)$,$(V_5,V_6,10)$]，weight=23。边$(V_5,V_3,12)$中的V_5和V_3已经处于同一分量，跳过该边。

选择边$(V_4,V_5,15)$,检查V_5和V_4是否属于同一连通分量。它们不属于同一分量，因此加入这条边。更新并查集，使V_5和V_4处于同一分量。更新 MST=[$(V_0,V_4,1)$,$(V_6,V_2,2)$,$(V_1,V_3,4)$,$(V_2,V_1,6)$,$(V_5,V_6,10)$，$(V_4,V_5,15)$]，weight=38。

(3)检查终止条件。现在最小生成树中已经有$n-1=6$条边，所有节点均已连通，算法终止。最终结果为最小生成树的边为：[$(V_0,V_4,1)$,$(V_6,V_2,2)$,$(V_1,V_3,4)$,$(V_2,V_1,6)$,$(V_5,V_6,10)$,$(V_4,V_5,15)$]。最小生成树的总权重为38。

13.3 最小生成树问题的变种问题

在运筹学和图论的应用中，最小生成树问题提供了连接所有节点的最低成本方案。然而，现实中的网络设计、物流规划或通信系统往往面临更为复杂的约束，单纯追求权值总和最小已不足以满足需求。这些额外的限制条件催生了最小生成树的变种问题，其中约束最小生成树问题和度约束最小生成树问题是两个重要的分支。这些变种问题不仅丰富了最小生成树的理论体系，还在实际工程中展现了强大的应用价值。以下，将逐一探讨这些变种问题的定义、约束类型、数学模型及求解方法，帮助读者深入理解其理论与实践意义。

13.3.1 约束最小生成树问题

在实际的网络设计中，最小生成树的求解往往需要考虑额外的限制条件。例如，设计一张通信网络时，不仅希望总成本最低，还可能要求某些节点（如服务器）的连接数不能过多，或者某些路径的延迟不能超过规定值。这些附加条件使得问题变得更加复杂，形成了约束最小生成树问题（Constrained Minimum Spanning Tree Problem）。与经典最小生成树不同，约束最小生成树需要在满足基本连通性和无圈性的同时，遵守特定的约束，从而在网络设计、物流配送、城市

交通规划等领域找到更贴合实际的解决方案。

约束最小生成树的核心在于平衡成本优化与约束满足。常见的约束类型包括以下几种，每种都与特定的应用场景密切相关。首先是度数约束，它限制每个节点在生成树中的最大连接边数。例如，在电力网络中，变电站的连接数可能因设备容量限制而不能过多，度数约束能有效避免节点过载，提高系统的稳定性。其次是距离或延迟约束，这要求生成树中某些路径的长度或延迟不超过给定阈值。在物流配送中，运输路线需要确保货物在规定时间内送达，距离约束能优化配送效率并提升客户满意度。第三是预算约束，为生成树的总成本设定上限，这在基础设施建设中尤为常见，确保项目在经济可行的范围内完成。此外，分层结构或分区约束适用于通信网络或数据中心设计，要求生成树满足层次化的功能需求，例如核心层与接入层节点在带宽或处理能力上的差异。最后，可靠性约束关注网络的容错能力，通过增加冗余路径或保护关键节点，确保在部分节点或边故障时，网络仍能保持连通。这种约束在电力或通信等关键基础设施中至关重要，避免因故障导致的服务中断。

为了形式化地描述约束最小生成树问题，可以将其建模为一个优化问题。假设有一个带权连通图G=(V,E)，其中V是节点集，E是边集，每条边i∈E具有权值w_i。定义决策变量x_i，当边i被选入生成树时，x_i=1，否则x_i=0。约束最小生成树问题的数学模型如下：

$$\min f(x) = \sum_{i=1}^{m} w_i x_i \tag{13.3}$$

s.t.

$$\mathbf{x} \in T, \tag{13.4}$$

$$x_i \in \{0,1\}, \forall i \in \{1,2,\ldots,m\}, \tag{13.5}$$

$$\deg(v) \leq d, \forall v \in V, \tag{13.6}$$

$$\sum_{i=1}^{m} w_i x_i \leq B，\ \forall i \in \{1,2,\ldots,m\}. \tag{13.7}$$

公式(13.4)表示基本生成树约束，要求$x \in T$，其中T是所有生成树的集合。

(2)二进制约束，要求$x_i \in \{0,1\}$，表示边的选择状态。

(3)其他附加条件。不同的附加约束可表示为，度数约束，对于每个节点v，满足$deg(v) \leq d$，其中$deg(v)$表示节点v的度数（与该节点相连的边的数量），d为最大度数限制；预算约束，$\sum_{i=1}^{m} w_i x_i \leq B$，其中$B$为预算上限。

由于附加约束的存在，约束最小生成树问题通常是 NP 难的，经典的贪心算法（如 Prim 或 Kruskal）无法直接适用。为此，研究者开发了多种求解方法。首先是拉格朗日松弛法，它通过将复杂约束（如度数约束）松弛到目标函数中，引入拉格朗日乘子，将问题转化为较易求解的子问题，通过迭代调整乘子获得近似最优解。其次是启发式算法或智能优化算法，如模拟退火等，这些方法通过启发式规则快速找到近似解，适合大规模问题的快速求解。第三是分支定界法，通过系统地探索解空间并剪枝，适合小规模问题的高精度求解。混合整数线性规划（MILP）将问题建模为线性规划形式，利用商用求解器（如 CPLEX 或 Gurobi）求解，适用于中小规模问题。

13.3.2 度约束最小生成树问题

在众多约束最小生成树变种问题中，度约束最小生成树问题（Degree-Constrained Minimum Spanning Tree, DCMST）因其广泛的应用而备受关注。在网络设计中，节点的连接数（即度数）往往受到物理或功能限制。例如，通信网络中的服务器可能因带宽或处理能力限制而只能连接有限的节点，度约束最小生成树问题通过限制每个节点的度数，确保生成树在成本最低的同时满足这些物理约束。

度约束最小生成树问题是在一个带权连通图中寻找一棵生成树，使得生成树中所有边的权值和最小，并且每个节点的度数不超过预设的限制。

度约束最小生成树问题定义在一个带权连通无向图G=(V,E)上，其中V是包含n个节点的节点集，E是包含m条边的边集，每条边e∈E具有非负权值$w(e)$每个节点v∈V有一个度数上限d_v，表示其在生成树中最多连接的边数。$\delta(v)$表示与节点v直接相连的边的集合。是找到一棵生成树T，使得总权值$\sum_{e \in T} w(e)$最小，且

每个节点v的度数$\sum_{e\in\delta(v)} x_e \le d_v$。选择边的决策变量$x_e$：

$$x_e = \begin{cases} 1\text{，如果 e 被选择} \\ 0\text{，否则} \end{cases}$$

$$\forall e \in E.$$

基于以上定义，度约束最小生成树的数学模型可以表示如下：

$$\min\ Z = \sum_{e\in E} w(e) \cdot x_e \tag{13.8}$$

$$\text{s.t.} \sum_{e\in T} x_e = n - 1 \tag{13.9}$$

$$\sum_{e\in\delta(S)} x_e \ge 1, \forall S \subset V, S \ne \emptyset, S \ne V \tag{13.10}$$

$$\sum_{e\in\delta(v)} x_e \le d_v, \forall v \in V \tag{13.11}$$

$$x_e \in \{0,1\}, \forall e \in E. \tag{13.12}$$

其中目标函数表示生成树的总权值最小化。生成树连通性约束和割约束保证生成树是连通的。度约束限制每个节点的度不超过预设的度限制d_v。0-1 变量约束(13.12)$x_e \in \{0,1\}$表示边的选取状态。

求解度约束最小生成树问题同样具有较高的复杂性和挑战性。为此，学者们提出了多种有效的求解方法，其中包括割平面法、拉格朗日松弛法以及各类启发式算法。割平面法通过逐步引入割约束，不仅能够保证生成树的连通性，还能有效地处理度数限制，从而逐步逼近最优解。拉格朗日松弛法则以其对复杂约束的灵活处理能力著称，通过引入拉格朗日乘子，将度约束“松弛”为目标函数的一部分，使原本难以直接求解的问题转化为一系列更易处理的子问题，在度约束最小生成树问题中表现出较强的实用性。与此同时，启发式算法凭借其全局搜索能力和较高的计算效率，能够在大规模实例中快速获得高质量的近似解。综合来看，这些方法各具优势，通常需要根据具体问题规模和实际需求灵活选择与组合，以

获得理想的求解效果。

例题 13.3 某城市计划部署一个光纤网络，连接市内的所有主要区域。网络需要满足以下条件：所有区域必须通过光纤直接或间接连通。由于预算限制，布线总成本需要最小化。医院要求至少连接两条光纤线路，以确保冗余和可靠性。网络不能形成环路。

城市有 6 个区域（编号为v_1,v_2,v_3,v_4,v_5,v_6），其中v_1是医院。各区域之间可能的连接路径及成本（单位：万元）如表 13.1 所示：

表 13.1 例题 13.3 中的区域的路径成本

连接路径	(v_1,v_2)	(v_1,v_3)	(v_1,v_4)	(v_2,v_3)	(v_2,v_5)	(v_3,v_4)	(v_3,v_6)	(v_4,v_5)	(v_4,v_6)	(v_5,v_6)
成本	4	7	6	3	5	2	8	4	5	6

解 找到一个光纤网络布线方案（生成树），使得：

（1）总成本最小。

$$min \sum_{e \in E} w(e) x_e$$

（2）满足生成树的基本约束（连通、无环、覆盖所有顶点）。

（3）v_1的度数至少为 2

$$\sum_{e \in \delta(v_1)} x_e \geq 2$$

将度约束松弛到目标函数中，引入拉格朗日乘子λ_1≥0（对应v_1的度约束）。松弛后的目标函数为：

$$L(x, \lambda_1) = \sum_{e \in E} w(e) x_e + \lambda_1 (2 - \sum_{e \in \delta(v_1)} x_e)$$

展开后得到：

$$L(x,\lambda_1)=\sum_{e\notin\delta(v_1)} w(e)x_e+\sum_{e\in\delta(v_1)}(w(e)-\lambda_1)\,x_e+2\lambda_1$$

$$\forall e\in E.$$

其中：对于$e\in\delta(v_1)$(即$(v_1,v_2),(v_1,v_3),(v_1,v_4)$)，边的调整权重为$w(e)-\lambda_1$；对于$e\notin\delta(v_1)$，边的权重保持$w(e)$不变；对于给定的$\lambda_1$，需要最小化$L(x,\lambda_1)$。这等价于在调整后的图上求一个最小生成树。

如果$w(e)-\lambda_1<0$(对于$e\in\delta(v_1)$)，则优先选择这些边。使用 Kruskal 等算法在调整后的权重下求解。

13.3.3 二次最小生成树问题

在最小生成树问题的研究中，二次最小生成树问题（Quadratic Minimum Spanning Tree Problem，简称 QMST 问题）是一个较为复杂的变种问题。与经典最小生成树问题不同的是，二次最小生成树问题不仅考虑每条边的权值，还考虑了边与边之间的相互影响，即边之间的权值。这使得二次最小生成树问题更贴近某些实际应用场景，例如在设计网络时，不仅要考虑单条连接的成本，还要考虑两条连接之间的关联成本。

（1）二次最小生成树问题的定义

给定一个带权连通图$G=(V,E)$，其中V是节点集，E是边集。对于每条边$i\in E$，有一个与之相关的权值w_i。在二次最小生成树问题中，除了单条边的权值w_i，还考虑了边对(i,j)的二次交互权值q_{ij}，表示在生成树中同时包含边i和j所需的额外成本或收益。

二次最小生成树问题可以表述为，在图G中，找出一棵生成树T，使得生成树的权值和边对之间的二次交互权值之和最小化。

（2）数学模型

$$\min\ f(x)=\sum_{i\in E}w_ix_i+\sum_{i\in E}\ \sum_{j\in E,j>i}q_{ij}x_ix_j \tag{13.15}$$

s.t.

$$x\in T, \tag{13.16}$$

$$x_i\in\{0,1\},\forall i\in E,\forall j\in E. \tag{13.17}$$

该模型中生成树约束，要求$x\in T$，其中T是所有生成树的集合。$x_i\in\{0,1\}$，其中x_i=1 表示边i被选择为生成树的一部分，x_i=0 则表示边i未被选择。边对交互项q_{ij}表示边i和边j的二次交互权值，当且仅当x_i=1 且x_j=1 时才会计入总成本。

这个模型不仅包含了单边权值项w_i，还包含了边对之间的交互权值项$q_{ij}x_ix_j$，增加了问题的复杂性。

为应对二次最小生成树问题的复杂性，研究者开发了多种求解方法，适用于不同规模和精度要求的问题。拉格朗日松弛法通过将二次项松弛到目标函数，引入乘子，将问题分解为标准最小生成树子问题，可用贪心算法求解。迭代调整乘子后，可获得高质量近似解，适合大规模问题，如无线传感器网络优化，但解质量依赖乘子调整。启发式和元启发式算法，如遗传算法、模拟退火和禁忌搜索，通过迭代搜索快速找到近似解。遗传算法模拟进化过程，通过交叉和变异优化生成树；模拟退火允许接受次优解以跳出局部最优。这些方法适合大规模问题（如城市交通网络），但解质量可能不稳定。

综上，二次最小生成树问题的求解方法需根据问题规模和精度要求选择；实际应用中，可结合多种方法，例如用启发式算法生成初始解，再用拉格朗日松弛法优化，以兼顾效率和质量。这些方法共同推动了二次最小生成树问题在网络设计和优化中的应用，为复杂实际问题提供了灵活的解决方案。

例题 13.4 某城市计划在发生地震、洪水等紧急灾害时，快速部署一个临时的应急通信网络，连接市内的 5 个关键避难点（编号为 A、B、C、D、E）。由于灾害可能导致部分基础设施受损，通信网络需要通过便携式无线中继设备建立连接。每条连接的部署成本（单边权值）取决于距离和地形难度。此外，若某些连

接线路之间过于靠近，会因电磁干扰增加额外的维护成本(边对交互权值，表 13.2 仅列出非零值，其他未列出的为 0)。目标是设计一棵总成本（包括部署成本和干扰成本）最小的连通生成树。

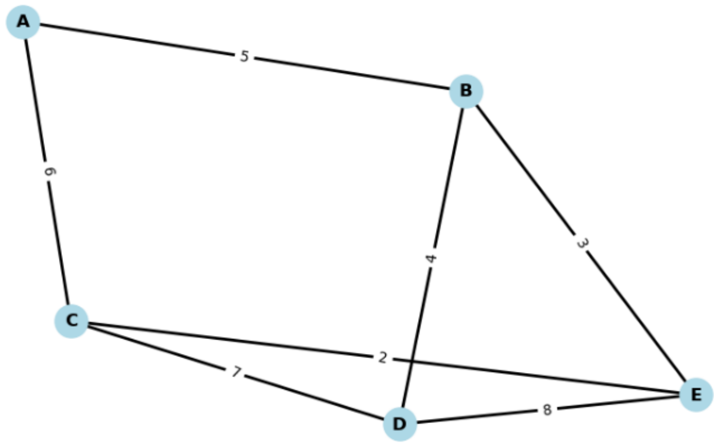

图 13.5 例题 13.4 中的设备的部署成本图

表 13.2 例题 13.4 中的设备的干扰维护成本

边对	干扰成本（千元）
AB 和 BD	3
AC 和 CE	2
BE 和 DE	1

解 给定一个带权连通图$G=(V,E)$，其中，$V=\{A,B,C,D,E\}$表示 5 个关键避难点。E表示避难点之间可能的无线连接边，边的权值w_i表示部署单条连接的成本（单位：千元）。q_{ij}表示边i和边j同时选择时因电磁干扰产生的额外成本（单位：千元）。

边的权值和边对交互权值如下：$w_{AB}=5, w_{AC}=6, w_{BD}=4, w_{BE}=3, w_{CD}=7, w_{CE}=2, w_{DE}=8$。

边对交互权值如下：

$$q_{AB,BD}=3(AB\text{和}BD\text{线路靠近，产生干扰})$$

$$q_{AC,CE}=2(AC\text{和}CE\text{线路靠近，产生干扰})$$

$$q_{BE,DE}=1(BE\text{和}DE\text{线路靠近，产生干扰})$$

为该临时应急通信网络设计一棵二次最小生成树T，使得总成本最小化。

$$minZ=\sum_{e\in E}w_e\cdot x_e+\sum_{\substack{e,f\in E\\ e<f}}q_{e,f}\cdot x_e\cdot x_f$$

s.t.

$$\sum_{e\in T}x_e=4$$

$$T\text{连通所有节点}\left(\text{无环，由边数约束隐含}\right)$$

$$x_e\in\{0,1\},e\in E$$

总成本由边部署成本和边对干扰成本两部分组成,部署成本：选中边的部署费总和。干扰成本：同时选中的边对的干扰费总和。约束条件包括生成树基本约束和变量取值约束等。

13.3.4 双目标最小生成树问题

在实际应用中，优化问题往往需要同时考虑多个目标，例如在网络设计中，不仅要关注连接成本，还要考虑网络的稳定性或效率。这种情况下，双目标最小生成树问题（Bicriteria Minimum Spanning Tree Problem, BMST）是一个重要的研究方向。

（1）双目标最小生成树问题的定义

双目标最小生成树问题是在带权连通图中寻找一棵生成树，以同时优化两个目标。每条边不仅有一个成本权值，还有一个第二目标的权值（例如时间、可靠性等）。双目标最小生成树问题的目的是在满足生成树的基础上，找到权值和最优的组合，以平衡或优化这两个不同的目标。

（2）问题描述

给定一个带权连通图$G=(V,E)$，其中，节点集V表示图中的所有节点。边集

E表示节点间的连接，每条边$e \in T$有两个非负权值，$c(e)$表示边e的第一目标权值（如成本）。$t(e)$表示边e的第二目标权值（如时间、可靠性等）。

双目标最小生成树问题的目标是找到一棵生成树，使得以下两个目标同时达到最优：

最小化第一目标权值的总和：

$$\min \sum_{e \in T} c(e)$$

最小化第二目标权值的总和：

$$\min \sum_{e \in T} t(e)$$

由于两个目标往往相互冲突，双目标最小生成树问题一般通过求解一组非支配解（Pareto Optimal Solutions）来获得一组平衡的解集合。

（3）数学模型

双目标最小生成树问题的数学模型可以表示为：

$$\min\ Z = \left(\sum_{e \in T} c(e), \sum_{e \in T} t(e) \right) \tag{13.18}$$

$$\text{s.t.} \sum_{e \in T} x_e = n - 1 \tag{13.19}$$

$$\sum_{e \in \delta(S)} x_e \geq 1, \forall S \subset V, S \neq \emptyset, S \neq V \tag{13.20}$$

$$x_e \in \{0,1\}, \forall e \in T. \tag{13.21}$$

其中决策变量x_e是 0-1 变量，表示边e是否被选入生成树：

$$x_e = \begin{cases} 1\text{，如果}e\text{被选择} \\ 0\text{，否则} \end{cases}$$

目标函数要求同时最小化两种权值的总和。生成树的连通性和无环约束保证所选边形成一个连通且无环的生成树。

双目标最小生成树问题是一个多目标优化问题，求解时需要生成一个非支配解集。以下是常用的求解方法。

加权和法，通过引入权重α和$1-\alpha$来将两个目标加权合并为一个单一目标：

$$\min\left(\alpha\sum_{e\in T}c(e)+(1-\alpha)\sum_{e\in T}t(e)\right) \tag{13.22}$$

$$\forall e\in T,\alpha\in[0,1].$$

通过改变α的值（例如$\alpha\in[0,1]$），可以得到不同权衡下的解。

简单来说，加权和法通过引入权重将两目标合并为单一目标函数。通过调整α的值，可生成多个生成树，每个生成树对应一种成本与延迟的权衡。针对每个α，问题转化为标准的最小生成树问题，可通过 Prim 或 Kruskal 算法高效求解。该方法计算简单，适合小规模网络，但可能无法覆盖所有 Pareto 最优解。

ε-约束法会优化一个目标，同时将另一个目标作为约束条件。例如，最小化$\sum_{e\in T}c(e)$的同时，约束$\sum_{e\in T}t(e)\le\epsilon$其中$\epsilon$是用户设定的上限。

改变ϵ的值可以得到不同的 Pareto 解。通过逐步调整ϵ的值，从最小延迟到较大延迟，多次求解生成树，可获得一组 Pareto 最优解。该方法能较全面地探索 Pareto 前沿，适合需要精确解的场景。

对于大规模网络，元启发式算法如遗传算法更具优势。这些算法通过编码生成树，在成本和延迟的二维目标空间中搜索 Pareto 最优解。从一组随机生成树开始，通过迭代优化保留非支配解，最终形成近似的 Pareto 解集。模拟退火或禁忌搜索也可用于快速求解复杂问题。这些方法计算效率高，适合大规模或高复杂度网络。

例题 13.5 某城市计划建设一个绿色交通网络，连接 5 个主要区域（编号为 A、B、C、D、E），以推广低碳出行。网络通过自行车道或步行道实现，每条路径有两个关键指标需要优化：建设成本（单位：万元，表示铺设路径所需的资金）和碳排放量（单位：吨，表示建设过程中产生的碳排放，例如材料运输和施工排

放）。城市希望设计一个交通网络（生成树），在尽量降低建设成本的同时，减少碳排放，以实现经济和环保的双重目标。

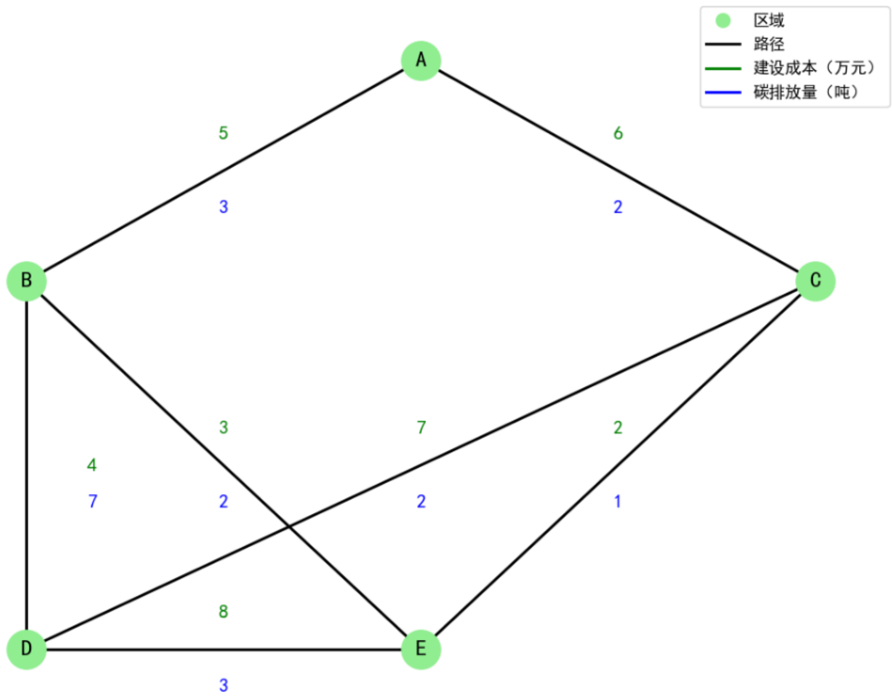

图 13.6 例题 13.5 中的建设成本与碳排放量的网络图

解 上述带权连通图$G=(V,E)$，其中：$V=\{A,B,C,D,E\}$表示 5 个主要区域。E 表示区域之间可能的路径，每条边$e\in E$有两个权值：$c(e)$是建设成本（单位：万元）；$t(e)$是碳排放量（单位：吨）

双目标最小生成树问题的目标是找到一棵生成树$T\subseteq E$，同时优化两个目标。数学模型如下：

$$\min\left(\sum_{e\in T}c(e),\sum_{e\in T}t(e)\right)$$

s.t.

$$\sum_{e\in T}x_e=4,$$

$$x_e\in\{0,1\},\forall e\in E.$$

其中，$x_e=1$ 表示边e被选入生成树T，$x_e=0$ 表示未被选择，边数约束保

证生成树连通且无环。目标是同时最小化总成本 $\sum_{e\in T} c(e)$ 和总碳排放 $\sum_{e\in T} t(e)$。

13.4 本章小结

本节全面探讨了最小生成树问题及其几个重要的变种。首先定义了最小生成树，并介绍了两种经典的求解算法：Prim 算法和 Kruskal 算法。这些贪心算法有效地找到了给定加权无向图的最小生成树。

深入探讨了最小生成树问题的更复杂变种，文章讨论了度约束最小生成树，其中对每个节点施加最大边数，以及其他约束，如预算或距离限制。

文章介绍了双目标最小生成树，其中需要同时优化两个目标函数（例如成本和时间）。帕累托最优解作为求解方法。

最复杂变种，二次最小生成树考虑边对之间的相互作用，在目标函数中引入了二次项。讨论的求解方法包括拉格朗日松弛和元启发式算法等。

强调不同方法对不同问题规模和复杂性的适用性。对于较小的问题，精确方法如分支定界法等是可行的。对于较大的问题，由于计算复杂性，启发式算法和元启发式算法是必不可少的。

研学互通

最小生成树作为运筹学和图论中的核心概念，不仅在理论层面体现了贪心算法的精妙设计，还在交通网络规划、通信系统优化和基础设施建设等领域有着广泛的应用。本章主要介绍了最小生成树的基本定义、Kruskal 算法和 Prim 算法的实现，以及一些简单的应用案例。然而，最小生成树的理论发展和实际应用远不止于此。为了帮助读者更全面地理解这一主题的学术背景、算法演变和现实意义，特别在章节结尾设置了拓展阅读部分。通过阅读经典教材和原始文献，读者可以深入掌握算法的数学推导、复杂度分析及其在更复杂场景中的应用，进而培养独立研究的能力。这些参考资料不仅能深化对最小生成树的理解，还能为后续学习图算法、网络优化等高级主题打下坚实基础。

如果读者对最小生成树的内容感兴趣，并希望进一步探索其理论基础与应用场景，不妨阅读以下经典文献。

(1) Even, S. (2011). Graph algorithms (2nd ed.). Cambridge University Press.

本书由以色列计算机科学家 ShimonEven 撰写，是图算法领域的经典教材之一，2011 年第二版对内容进行了全面更新，系统介绍了图算法的理论基础与实现方法。书中涵盖了最短路径、最小生成树、网络流和匹配问题等核心主题，特别适合希望从理论到实践全面掌握图算法的读者。在最小生成树部分，Even 详细讲解了 Kruskal 算法和 Prim 算法的步骤、伪代码及其复杂度分析。此外，书中提供了大量实际案例，如通信网络设计、电路布局优化和交通路线规划，展示了最小生成树在工程中的应用价值。对于运筹学或计算机科学专业的本科高年级学生和研究生而言，这本书是深入学习图算法的理想选择。

(2) Gross, J. L., & Yellen, J. (2005). Graph theory and its applications (2nd ed.). CRC Press.

由 JonathanL.Gross 和 JayYellen 合著的 GraphTheoryandItsApplications 是图论领域的权威教材，2005 年第二版内容全面且深入浅出，适合初学者和进阶读者。书中从图的基本定义和性质入手，逐步介绍连通性、树、最小生成树、最短路径和图着色等主题，并扩展到网络优化等高级应用。在最小生成树章节，作者详细比较了 Kruskal 和 Prim 算法的实现步骤与适用场景，并通过交通网络设计、电力系统优化和分布式计算等案例，展示了最小生成树的实际意义。此外，书中提供了大量例题、习题及答案，书末还附有丰富的参考文献，便于读者自学或课堂使用。对于运筹学初学者或希望通过实际案例加深理解的读者，这本书是不可多得的资源，尤其适合本科生和教师作为补充教材。

(3) Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. Proceedings of the American Mathematical Society, 7(1), 48–50.

JosephB.Kruskal 在 1956 年发表的这篇论文是最小生成树领域的奠基性文献，首次提出了著名的 Kruskal 算法，标志着贪心策略在图论中的经典应用。论文聚

焦于如何在加权无向图中通过边权排序构建最小生成树，Kruskal 详细阐述了算法的核心思想——按权值从小到大选择边并避免回路，并通过数学证明确保了算法的正确性。文中还探讨了最小生成树与旅行商问题（TSP）的潜在联系，为后续研究提供了重要启发。尽管论文篇幅较短，但其简洁的语言和直观的示例充分展现了贪心算法设计的数学美感。对于希望了解最小生成树算法历史背景的读者，这篇论文是必读之作，尤其适合研究生或研究人员深入研究算法起源，需具备一定的数学基础以理解证明细节。

(4) Prim, R. C. (1957). Shortest connection networks and some generalizations. Bell System Technical Journal, 36(6), 1389–1401.

RobertC.Prim 在 1957 年发表的这篇论文提出了 Prim 算法，成为求解加权无向图最小生成树的另一经典方法，最初为优化电话网络设计而开发，后来广泛应用于网络优化领域。Prim 算法以从单个节点出发、逐步扩展的方式构建生成树，每次选择与当前树相连的最小权值边，类似于最短路径算法的设计思路。论文详细描述了算法步骤，并通过示例验证了其有效性，同时探讨了算法在动态网络中的推广应用，如如何适应网络拓扑变化。Prim 的工作不仅奠定了最小生成树算法的多样性基础，还为分布式算法的研究提供了启发。这篇论文语言简洁、逻辑严谨，适合希望深入理解 Prim 算法推导及其在通信网络中应用的读者，尤其是研究生或从事网络优化研究的专业人士。

思行经世：最小生成树连接科技强国

最小生成树作为图论中的经典算法，通过在连通无向图中选择权值最小的边，构建一个包含所有顶点且无环的树，实现网络连接的最优化。其核心在于采用贪心策略，如 Kruskal 算法按权值升序选择边，或 Prim 算法从初始节点逐步扩展，确保总成本最低。这一方法在资源有限的复杂系统中展现了高效规划能力，在我国科技强国建设中发挥了重要作用，体现了中国特色社会主义制度下科技创新与实践的智慧结晶。

在智慧城市建设中，最小生成树算法为交通网络优化提供了科学支持。以深圳智慧交通系统为例，城市管理者通过 Kruskal 算法规划公交线路，选择最短的

道路连接各站点，最大化网络覆盖的同时减少建设成本。这一优化显著提升了公共交通效率，缓解了城市拥堵，体现了“创新驱动、智慧为民”的城市精神，为“智慧中国”战略注入了活力。同样，在5G网络布局中，最小生成树算法助力了通信网络的优化。以中国移动的5G建设为例，工程师通过Prim算法优化基站连接，选择最经济的线路布局，确保信号覆盖最大化，降低了建设成本并提升了通信效率，彰显了“科技为民、服务大局”的通信精神，为我国5G技术的全球领先奠定了基础。

在灾后重建中，最小生成树算法为基础设施恢复提供了保障。以汶川地震后的重建为例，救援团队利用最小生成树算法优化临时通信网络，选择最短路径连接各救援点，快速恢复通信能力，确保救援工作高效推进。这一过程体现了“众志成城、共克时艰”的抗震精神，最小生成树的精准计算为灾后重建注入了科技力量。最小生成树的每一次优化，都是资源节约与国家精神的交汇。智慧交通的科学规划诠释了“创新驱动”，5G 网络的优化布局彰显了“科技为民”，灾后重建的快速恢复体现了“众志成城”。这些实践不仅推动了基础设施现代化，还激励读者在科技强国建设中以科学方法连接未来。最小生成树算法作为图论的瑰宝，深刻融入我国的交通、通信和灾后重建事业，为实现中华民族伟大复兴贡献了智慧与力量。

习题

习题 13.1 一名电力工程师，负责为五个偏远村庄（A、B、C、D、E）设计一个低成本的电力网络。目标是以最低的电缆铺设成本连接所有村庄，确保每个村庄都能通电（即形成一棵生成树）。每对村庄之间的电缆铺设成本（权重）已通过地形勘测确定，单位为十万元，边的权重如下：(A,B)=4，(A,C)=8，(B,C)=2，(B,D)=7，(C,D)=5，(C,E)=6，(D,E)=3。使用Prim算法，从顶点A开始，找到最小生成树及其总权重。展示算法的每个步骤。

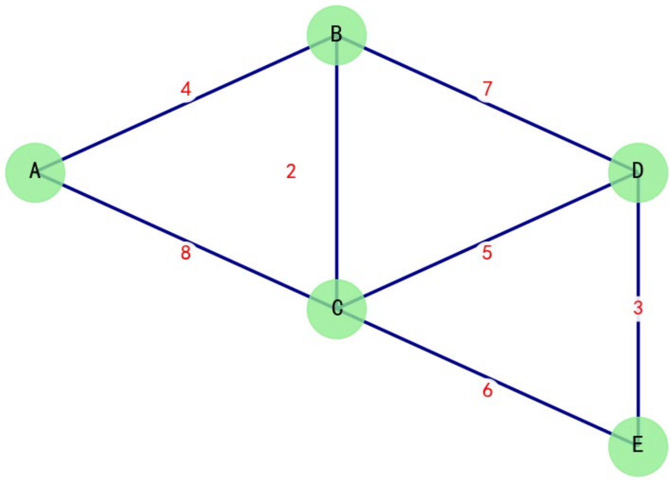

图 13.7 习题 13.1 中的电缆铺设成本的网络图

习题 13.2 对于习题 13.1 中的同一张图，使用克鲁斯卡尔算法找到最小生成树。显示排序后的边，并解释在每个步骤中并查集数据结构的使用。

习题 13.3 将习题 13.1 中图的最小生成树问题表述为整数线性规划。定义决策变量、目标函数和约束条件。（不需要求解，只需建立 ILP 模型）。

习题 13.4 假设习题 13.1 中每个节点的最大度数约束为 2。解释此约束如何改变问题以及可以使用哪些方法来找到解决方案（至少提及两种）。

习题 13.5 为习题 13.1 中的每条边添加第二个权重，表示“时间”（例如，(A,B)=4（成本），2（时间））。说明你将使用哪种方法找到双目标最小生成树（最小化成本和时间）的最优解。