山东师范大学 SHANDONG NORMAL UNIVERSITY

《算法设计与分析》
第07.2讲 分支限界法(2)
TSP问题的分支限界法

山东师范大学信息科学与工程学院
段会川
2014年12月

目录

TSP—问题定义

令 $G=(V,E)$ 是一个有向赋权图，顶点集为 $V=(v_0,v_1,\cdots,v_{n-1})$。货郎担问题可描述为：求从图中任一顶点 v_i 出发，经图中所有其他顶点一次且只有一次，最后回到同一顶点 v_i 的最短路径。这个问题，也就是求图的最短哈密尔顿回路问题。假定 c 为图的邻接矩阵，c_{ij} 表示顶点 v_i 到顶点 v_j 的关联边的长度（它可以是某种费用，例如城市 v_i 到城市 v_j 的交通费用或通信线路的花费等，因此又把 c 称为费用矩阵）。使用 8.1 节所叙述的第 2 种分支限界方法来求解这个问题。在此首先要确定选择哪一条边进行分支，以及怎样计算其下界。

郑宗汉 P247

目录

费用矩阵的特性及规约

假定 l 是图 G 的一条最短的哈密尔顿回路，$w(l)$ 是这条回路的费用。因为费用矩阵中的元素 c_{ij} 表示顶点 v_i 到顶点 v_j 的关联边的费用，根据哈密尔顿回路的性质，它和费用矩阵 c 中的元素有如下关系。

引理 8.1 令 $G=(V,E)$ 是一个有向赋权图，l 是图 G 的一条哈密尔顿回路，c 是图 G 的费用矩阵，则回路上的边对应于费用矩阵 c 中每行每列各一个元素。

郑宗汉 P247

费用矩阵的特性及规约

证明 假定图 G 有 n 个顶点，费用矩阵中的第 i 行元素表示顶点 v_i 到其他顶点的出边费用，第 i 列元素表示其他顶点到顶点 v_i 的入边费用。l 是图 G 的一条哈密尔顿回路。v_i 是回路中的任意一个顶点，$0 \le i \le n-1$。它在回路中只有一条出边，该出边对应于费用矩阵中第 i 行的一个元素。v_i 在回路中只出现一次，因此费用矩阵的第 i 行有且只有一个元素与其对应。其次，v_i 在回路中只有一条入边，因此，费用矩阵中的第 i 列也有且只有一个元素与其对应。回路中的顶点共有 n 个，对应于图 G 的 n 个顶点，所以费用矩阵的每一行每一列都有且只有一个元素与回路中的顶点的出边与入边一一对应。

郑宗汉 P247-8

费用矩阵的特性及规约

例如，如图 8.7（a）所示为一个 5 城市的货郎担问题的费用矩阵，令 $l = v_0v_3v_1v_4v_2v_0$ 是哈密尔顿回路，回路上的边对应于费用矩阵中的元素 $c_{03},c_{31},c_{14},c_{42},c_{20}$。可以看到，费用矩阵中的每一行每一列都有且只有一个元素与回路中的边相对应。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

郑宗汉 P248

费用矩阵的特性及规约

定义 8.1 费用矩阵 c 的第 i 行（或第 j 列）中的每个元素减去一个正常数 lh_i（或 ch_j），得到一个新的费用矩阵 $\bar{c}$，使得 $\bar{c}$ 中第 i 行（或第 j 列）中的最小元素为 0，称为费用矩阵的行归约(或列归约)。称 lh_i 为行归约常数，称 ch_j 为列归约常数。

例如，把图 8.7（a）中的每一行都进行行归约，第 0 行的每一个元素都减去 25，第 1 行的每一个元素都减去 5，第 2 行的每一个元素都减去 1，第 3 行的每一个元素都减去 6，第 4 行的每一个元素都减去 7，得到行归约常数 $lh_0 = 25, lh_1 = 5, lh_2 = 1, lh_3 = 6, lh_4 = 7$，所得结果如图 8.7（b）所示。把图 8.7（b）的第 3 列进行列归约，得到列归约常数 $ch_3 = 4$，所得结果如图 8.7（c）所示。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	$lh_0 = 25$
1	0	∞	13	26	21	$lh_1 = 5$
2	19	15	∞	6	0	$lh_2 = 1$
3	4	45	19	∞	0	$lh_3 = 6$
4	16	2	0	4	∞	$lh_4 = 7$

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

费用矩阵的特性及规约

定义 8.2 对费用矩阵 c 的每一行和每一列都进行行归约和列归约，得到一个新的费用矩阵 $\bar{c}$，使得 $\bar{c}$ 中每一行和每一列至少都有一个元素为 0，称为费用矩阵的归约。矩阵 $\bar{c}$ 称为费用矩阵 c 的归约矩阵。称常数 h

$$h = \sum_{i=0}^{n-1} lh_i + \sum_{i=0}^{n-1} ch_i \qquad (8.5.1)$$

为矩阵 c 的归约常数。

例如，对图 8.7（a）中的费用矩阵进行归约，得到图 8.7（c）所示的费用矩阵，把图 8.7（c）所示的费用矩阵，称为图 8.7（a）中的费用矩阵的归约矩阵。此时，归约常数 h 为

$$h = 25+5+1+6+7+4 = 48$$

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	$lh_0 = 25$
1	0	∞	13	26	21	$lh_1 = 5$
2	19	15	∞	6	0	$lh_2 = 1$
3	4	45	19	∞	0	$lh_3 = 6$
4	16	2	0	4	∞	$lh_4 = 7$

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

费用矩阵的特性及规约

定理 8.1 令 $G=(V,E)$ 是一个有向赋权图，l 是图 G 的一条哈密尔顿回路，c 是图 G 的费用矩阵，$w(l)$ 是以费用矩阵 c 计算的这条回路的费用。如果矩阵 $\bar{c}$ 是费用矩阵 c 的归约矩阵，归约常数为 h，$\bar{w}(l)$ 是以费用矩阵 $\bar{c}$ 计算的这条回路的费用，则有：

$$w(l) = \bar{w}(l) + h \qquad (8.5.2)$$

证明 假定 c_{ij} 是费用矩阵 c 的第 i 行第 j 列元素，$\bar{c}_{ij}$ 是费用矩阵 $\bar{c}$ 的第 i 行第 j 列元素。因为 $\bar{c}$ 是 c 的归约矩阵，因此对所有的 i, j，其中 $0 \le i, j \le n-1$，有

$$c_{ij} = \bar{c}_{ij} + lh_i + ch_j$$

$w(l)$ 是以费用矩阵 c 计算的哈密尔顿回路的费用，令

$$w(l) = \sum_{i,j \in l} c_{ij}$$

$\bar{w}(l)$ 是以费用矩阵 $\bar{c}$ 计算的同一条哈密尔顿回路的费用，令

$$\bar{w}(l) = \sum_{i,j \in l} \bar{c}_{ij}$$

由引理 8.1，回路上的边对应于费用矩阵 c 中每行每列各一个元素，则有

$$w(l) = \sum_{i,j \in l} c_{ij} = \sum_{i,j \in l} \bar{c}_{ij} + \sum_{i=0}^{n-1} lh_i + \sum_{j=0}^{n-1} ch_j = \bar{w}(l) + h$$

定理证毕。

费用矩阵的特性及规约

定理 8.2 令 $G=(V,E)$ 是一个有向赋权图，l 是图 G 的一条最短的哈密尔顿回路，c 是图 G 的费用矩阵，$\bar{c}$ 是费用矩阵 c 的归约矩阵，$\bar{G}$ 是与费用矩阵 $\bar{c}$ 相对应的图，$\bar{c}$ 是图 $\bar{G}$ 的邻接矩阵，则 l 也是图 $\bar{G}$ 的一条最短的哈密尔顿回路。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	$lh_0 = 25$
1	0	∞	13	26	21	$lh_1 = 5$
2	19	15	∞	6	0	$lh_2 = 1$
3	4	45	19	∞	0	$lh_3 = 6$
4	16	2	0	4	∞	$lh_4 = 7$

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

费用矩阵的特性及规约

证明 用反证法证明。若 l 不是图 $\bar{G}$ 的一条最短的哈密尔顿回路，则图 $\bar{G}$ 中必存在另一条回路 l^*，它是图 $\bar{G}$ 中最短的哈密尔顿回路；同时，它也是图 G 中的一条回路。令 $\bar{w}(l)$ 是以费用矩阵 $\bar{c}$ 计算的回路 l 的费用，$\bar{w}(l^*)$ 是以费用矩阵 $\bar{c}$ 计算的回路 l^* 的费用，因此必有：

$$\bar{w}(l) = \bar{w}(l^*) + \delta$$

其中，δ 是一个正数。又 l^* 也是图 G 中的一条回路，令 $w(l)$ 和 $w(l^*)$ 分别是以费用矩阵 c 计算的回路 l 和 l^* 的费用，由定理 8.1，有

$$w(l) = \bar{w}(l) + h$$

$$w(l^*) = \bar{w}(l^*) + h$$

其中，h 是费用矩阵 c 的归约常数。因此

$$\begin{aligned} w(l) &= \bar{w}(l) + h = \bar{w}(l^*) + \delta + h \\ &= w(l^*) + \delta \end{aligned}$$

则 l^* 是图 G 中比 l 更短的哈密尔顿回路，与定理的前提相矛盾。所以，l 也是图 $\bar{G}$ 的一条最短的哈密尔顿回路。

目录

TSP界限确定方法

按照定理 8.1 和定理 8.2，求解图 G 的最短哈密尔顿回路问题，可以先求图 G 费用矩阵 c 的归约矩阵 $\bar{c}$，得到归约常数 h 之后，再转换为求取与费用矩阵 $\bar{c}$ 相对应的图 $\bar{G}$ 的最短哈密尔顿回路问题。令 $w(l)$ 是图 G 的最短哈密尔顿回路的费用，$\bar{w}(l)$ 是图 $\bar{G}$ 的最短哈密尔顿回路的费用，由定理 8.1，有 $w(l)=\bar{w}(l)+h$。由此得出，图 G 的最短哈密尔顿回路的费用，最少不会少于归约常数 h。因此，图 G 的费用矩阵 c 的归约常数 h，便是货郎担问题状态空间树中根节点的下界。

例如，在图 8.7（a）所示的 5 城市货郎担问题中，图 8.7（c）所示的费用矩阵是其归约矩阵，归约常数 48 便是该问题的下界，说明该问题的最小费用不会少于 48。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	1h_0 = 25
1	0	∞	13	26	21	1h_1 = 5
2	19	15	∞	6	0	1h_2 = 1
3	4	45	19	∞	0	1h_3 = 6
4	16	2	0	4	∞	1h_4 = 7

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

TSP界限确定方法

假定使用 8.1 节所叙述的第 2 种分支限界法来解货郎担问题。选取沿着某一条边出发的路径，作为进行搜索的一个分支节点，把这个节点称为节点 Y；不沿该条边出发的其他所有路径集合，作为进行搜索的另一个分支节点，把这个节点称为节点 $\bar{Y}$。仍以图 8.7（a）及图 8.7（c）所示的 5 城市货郎担问题的费用矩阵及其归约矩阵为例。如果选取从顶点 v_1 出发，沿着 (v_1,v_0) 的边前进，则该回路的边必然包含费用矩阵中的 $\bar{c}_{10}$。根据引理 8.1，回路中恰好包含费用矩阵 $\bar{c}$ 中不同行不同列的元素各一个。因此，费用矩阵 $\bar{c}$ 中的第 1 行和第 0 列的所有元素，在今后的计算中将不再起作用，可以把它们删去。另外，回路中也肯定不会包含边 (v_0,v_1)，否则，将构成一个由边 (v_1,v_0) 和边 (v_0,v_1) 所组成的小回路，从而使所构成的回路不再成为哈密尔顿回路。因此，可以把边 (v_0,v_1) 断开，即把元素 $\bar{c}_{01}$ 置为∞。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	1h_0 = 25
1	0	∞	13	26	21	1h_1 = 5
2	19	15	∞	6	0	1h_2 = 1
3	4	45	19	∞	0	1h_3 = 6
4	16	2	0	4	∞	1h_4 = 7

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

TSP界限确定方法

经过上述处理后，图 8.7（c）中 5×5 的归约矩阵，可以降阶为图 8.8（b）所示的 4×4 的矩阵。对这个矩阵进一步进行归约，得到图 8.8（c）所示的归约矩阵，其归约常数为 5。而图 8.7（a）中的费用矩阵归约为图 8.7（c）中的费用矩阵时，归约常数为 48。根据定理 8.1 和定理 8.2，沿着边 (v_1,v_0) 出发的回路，其费用肯定不会小于 48+5。这样一来，就可以把这个数据作为节点 Y 的下界。它表明沿着顶点 v_1 出发，经边 (v_1,v_0) 的回路，其费用至少不会小于 48+5 = 53。

	0	1	2	3	4
0	∞	25	41	32	28
1	5	∞	18	31	26
2	20	16	∞	7	1
3	10	51	25	∞	6
4	23	9	7	11	∞

（a）

	0	1	2	3	4	
0	∞	0	16	7	3	1h_0 = 25
1	0	∞	13	26	21	1h_1 = 5
2	19	15	∞	6	0	1h_2 = 1
3	4	45	19	∞	0	1h_3 = 6
4	16	2	0	4	∞	1h_4 = 7

（b）

	0	1	2	3	4
0	∞	0	16	3	3
1	0	∞	13	22	21
2	19	15	∞	2	0
3	4	45	19	∞	0
4	16	2	0	0	∞

（c）

TSP界限确定方法

当搜索深度为 m，并选取沿着某一条边 v_iv_j 出发，作为进行搜索的一个分支节点时，在一般情况下必须进行如下处理。

（1）删去费用矩阵的第 i 行及第 j 列的所有元素，把原来 $n-m$ 阶的费用矩阵，降阶为 $n-m-1$ 阶。

（2）在费用矩阵中，把 c_{ji} 置为∞，因为今后不会经过边 v_jv_i。

在一般情况下，假定父亲节点为 X，$w(X)$ 是父亲节点的下界。现在，选择沿 v_iv_j 边向下搜索作为其一个分支节点，令该节点为 Y；沿其他非 v_iv_j 边向下搜索作为其另一个分支节点，令该节点为 $\bar{Y}$。经过上述步骤处理之后，费用矩阵被进一步降阶和归约，并得到降阶后的归约常数，设为 h，如图 8.8 所示。则节点 Y 的下界可由下式确定：

$$w(Y)=w(X)+h \tag{8.5.3}$$

TSP界限确定方法

降阶后的归约常数，设为 h，如图 8.8 所示。则节点 Y 的下界可由下式确定：

$$w(Y)=w(X)+h \tag{8.5.3}$$

因为 $\bar{Y}$ 节点是沿其他非 v_iv_j 边向下搜索的分支节点，则回路中不会包含 v_iv_j 边。这样，可以把 $\bar{Y}$ 节点相应的费用矩阵中的 c_{ij} 置为∞。同时，它必然包含费用矩阵中第 i 行的某个元素，以及第 j 列的某个元素。如果令 d_{ij} 为第 i 行中除 c_{ij} 外的最小元素与第 j 列中除 c_{ij} 外的最小元素之和，即

$$d_{ij}=\min_{0\le k\le n-1,k\ne j}\{c_{ik}\}+\min_{0\le k\le n-1,k\ne i}\{c_{kj}\} \tag{8.5.4}$$

则节点 $\bar{Y}$ 的下界可由下式确定：

$$w(\bar{Y})=w(X)+d_{ij} \tag{8.5.5}$$

例如，在图 8.7（a）中，如果把根节点作为父亲节点 X，则 $w(X)=48$。这时，如果选择边 (v_1,v_0) 向下搜索作为其一个分支节点，令该节点为 Y，则经过上述处理之后的费用矩阵和归约常数如图 8.8 所示。于是，节点为 Y 的下界为：

$$w(Y)=w(X)+h=48+5=53$$

而节点 $\bar{Y}$ 的下界为：

$$w(\bar{Y})=w(X)+d_{ij}=48+4+13=65$$

目录

- ☐ TSP问题定义
- ☐ 费用矩阵的特性及规约
- ☐ 界限确定方法
- ☐ 分支的选择
- ☐ 求解过程
- ☐ 示例
- ☐ 算法复杂度

TSP分支的选择

在明确了 Y 节点及 $\overline{Y}$ 节点下界的确定方法之后，现在考虑分支的选择方法。在从父亲节点处理完毕的归约矩阵 c 中，每行每列至少包含一个其值为 0 的元素。于是，分支的选择按照下面两个思路进行。

（1）沿 $c_{ij}=0$ 的方向选择，使所选择的路线尽可能短。

（2）沿 d_{ij} 最大的方向选择，使 $w(\overline{Y})$ 尽可能大。

第一点是显而易见的。第二点是考虑到 Y 节点有一个明确的选择方向，而 $\overline{Y}$ 节点尚没有明确的选择方向。如果能够使 $w(\overline{Y})\geq w(Y)$，使搜索方向尽可能沿着 Y 节点方向进行，将加快解题的速度。

因此，令 S 是费用矩阵中 $c_{ij}=0$ 的元素集合，D_{kl} 是 S 中使 d_{ij} 达最大的元素 d_{kl}，即：

$$D_{kl}=\max_{S}\{d_{ij}\} \tag{8.5.6}$$

则边 $v_k v_l$ 就是所要选择的分支方向。

TSP分支的选择

例如，在图 8.7（a）所示的 5 城市货郎担问题的费用矩阵中，当从根节点 X 开始向下搜索时，把图 8.7(a)所示费用矩阵归约为图 8.7(c)所示矩阵，得到根节点的下界 $w(X)=48$ 后，此时有 $c_{01}=c_{10}=c_{24}=c_{34}=c_{42}=c_{43}=0$，其搜索方向的选择如下：

$$d_{01}=3+2=5 \qquad d_{10}=13+4=17 \qquad d_{24}=2+0=2$$
$$d_{34}=4+0=4 \qquad d_{42}=0+13=13 \qquad d_{43}=0+2=2$$

则使 d_{ij} 达最大的元素是 $d_{10}=17$。因此，$D_{kl}=d_{10}=17$。由此确定所选择的方向为边 v_1v_0，并可据此建立两个分支节点 Y 和 $\overline{Y}$。此时，可以直接用式（8.5.7）来代替（8.5.5）：

$$w(\overline{Y})=w(X)+D_{kl} \tag{8.5.7}$$

目录

- ☐ TSP问题定义
- ☐ 费用矩阵的特性及规约
- ☐ 界限确定方法
- ☐ 分支的选择
- ☐ 求解过程
- ☐ 示例
- ☐ 算法复杂度

TSP分支限界的求解过程

使用分支限界法的求解过程中，将动态地生成很多节点，用节点表来存放动态生成的节点信息。因为必须按费用的下界来确定搜索的方向，因此可以用优先队列或堆来维护节点表。在此使用优先队列来维护节点表。至此，用分支限界法求解货郎担问题的求解过程可叙述如下。

（1）令当前可行解的最优下界 *bound* 为∞。

（2）建立父亲节点 X，令节点 X 的费用矩阵是 $X.c$，把费用矩阵 c 复制到 $X.c$，费用矩阵的阶数 $X.k$ 初始化为 n；归约 $X.c$，计算归约常数 h，令节点 X 的下界 $X.w=h$；初始化回路的顶点邻接表 $X.ad$。

（3）按式（8.5.4），由 $X.c$ 中所有 $c_{ij}=0$ 的元素 c_{ij}，计算 d_{ij}。

$$d_{ij}=\min_{0\leq k\leq n-1,k\neq j}\{c_{ik}\}+\min_{0\leq k\leq n-1,k\neq i}\{c_{kj}\} \tag{8.5.4}$$

（4）按式（8.5.6），选取使 d_{ij} 最大的元素 d_{kl} 作为 D_{kl}，选择边 v_kv_l 作为分支方向。

$$w(\overline{Y})=w(X)+d_{ij} \tag{8.5.5}$$

TSP分支限界的求解过程

（5）建立儿子节点 $\overline{Y}$，把 X 的费用矩阵 $X.c$ 复制到 $\overline{Y}.c$，把 X 的回路顶点邻接表 $X.ad$ 复制到 $\overline{Y}.ad$，$X.k$ 复制到 $\overline{Y}.k$；把 $\overline{Y}.c$ 中的元素 c_{kl} 置为∞，归约 $\overline{Y}.c$；按式（8.5.7）计算节点 $\overline{Y}$ 的下界 $\overline{Y}.w$；把节点 $\overline{Y}$ 的 $\overline{Y}.w$ 与 *bound* 进行比较，处理是否插入优先队列。

$$w(\overline{Y})=w(X)+D_{kl} \tag{8.5.7}$$

（6）建立儿子节点 Y，把 X 的费用矩阵 $X.c$ 复制到 $Y.c$，把 X 的回路顶点邻接表 $X.ad$ 复制到 $Y.ad$，$X.k$ 复制到 $Y.k$；c_{lk} 置为∞。

（7）删去 $Y.c$ 的第 k 行第 l 列元素，使 $Y.k$ 减 1，从而使费用矩阵 $Y.c$ 的阶数减 1；归约降阶后的费用矩阵 $Y.c$，按式（8.5.3）计算节点 Y 的下界 $Y.w$。

$$w(Y)=w(X)+h \tag{8.5.3}$$

（8）若 $Y.k=2$，直接判断最短回路的两条边，并登记于回路邻接表 $Y.ad$，使 $Y.k=0$。

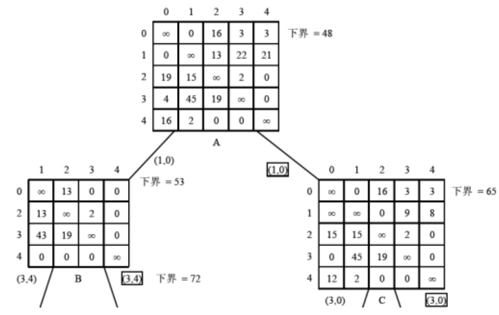
（9）把节点 Y 的 $Y.w$ 与 *bound* 进行比较，处理是否插入优先队列和更新 *bound*。

（10）取下优先队列元素作为节点 X，若 $X.k=0$，算法结束；否则，转步骤（3）。

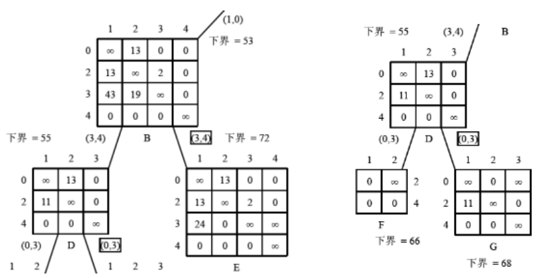
目录

- TSP问题定义
- 费用矩阵的特性及规约
- 界限确定方法
- 分支的选择
- 求解过程
- 示例
- 算法复杂度

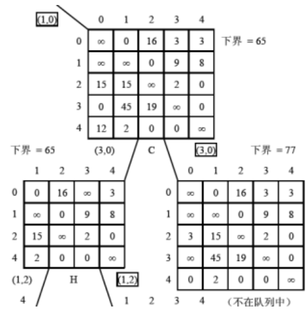
TSP分支限界求解示例



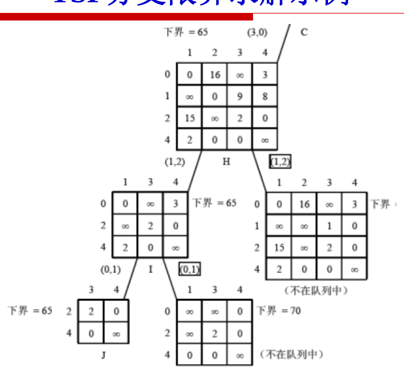
TSP分支限界求解示例



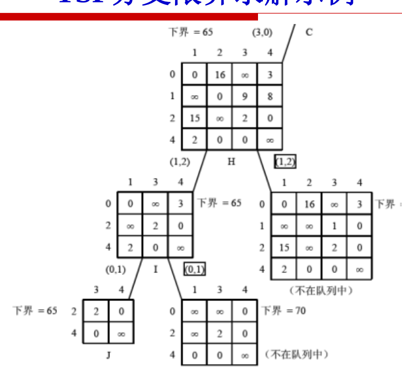
TSP分支限界求解示例



TSP分支限界求解示例



TSP分支限界求解示例



目录

- TSP问题定义
- 费用矩阵的特性及规约
- 界限确定方法
- 分支的选择
- 求解过程
- 示例
- 算法复杂度

TSP分支限界算法复杂度

该算法的时间花费估计如下：根据 8.5.4 节的结果，第 7 行初始化父亲节点，第 8 行归约父亲节点费用矩阵，都需 $O(n^2)$ 时间。第 9 行开始的 while 循环，循环体的执行次数取决于所搜索的节点个数，假定所搜索的节点数为 c。在 while 循环内部，第 10 行选择分支方向，需 $O(n^2)$ 时间。第 12 行把 x 节点数据复制到 z 节点（这里包括整个费用矩阵的复制工作），第 14 行归约 z 节点的费用矩阵，都需 $O(n^2)$ 时间。第 17 行把 z 节点插入优先队列，在最坏情况下需 $O(c)$ 时间。第 20 行，把 x 节点数据复制到 y 节点，同样需 $O(n^2)$ 时间。第 21 行登记回路邻接表，旁路有关的边，只需 $O(1)$ 时间。第 22 行删除 y 节点费用矩阵当前 vk 行 vl 列，第 23 行归约 y 节点费用矩阵，这些操作都需 $O(n^2)$ 时间。第 37 行把 y 节点插入队列，第 42 行删除队列首元素，都需 $O(c)$ 时间。其余的花费为 $O(1)$ 时间。因此，整个 while 循环在最坏情况下需 $O(cn^2)$ 时间。最后，在算法的尾部，第 45 行的 for 循环保存路线的顶点邻接表于数组 ad 作为算法的返回值，需 $O(n)$ 时间。第 48 行开始的 while 循环释放队列的缓冲区，在最坏情况下需 $O(c)$ 时间。所以，整个算法的运行时间为 $O(cn^2)$。

算法所需要的空间，主要花费在节点的存储空间。每个节点需要 $O(n^2)$ 空间存放费用矩阵，而存放费用矩阵的原始行、列号和当前行、列号的对应关系的映射表，以及回路的顶点邻接表仅需 $O(n)$ 空间。因此，每个节点相应需要 $O(n^2)$ 空间。所以，算法的空间复杂性也为 $O(cn^2)$ 空间。

目录

- TSP问题定义
- 费用矩阵的特性及规约
- 界限确定方法
- 分支的选择
- 求解过程
- 示例
- 算法复杂度

The End

Thanks !