算法 5.3 Kuhn-Munkreas 算法.

输入: 二分图 $G=(X,\Delta,Y)$, $|X|=|Y|$, 边权函数 $w:\Delta\to\mathbf{R}$.

输出: G 的最佳匹配 M.

(1) 选取 G 的一个可行顶标 l, 构造相等子图 G_l.

(2) 用匈牙利算法求 G_l 的最大匹配, 设为 M. 若 M 是 G_l 的完备匹配, 则 M 是 G 的最佳匹配, 算法停止; 否则, 转第 (3) 步.

(3) 设 u 是 G_l 中未被 M 许配的顶点, 不妨设 $u\in X$. 令

$$\begin{aligned}
&Z=\{v|v\in V(G_l),\ \text{且 } u,v \text{ 之间存在交错轨道}\},\\
&S=X\cap Z,\\
&T=Y\cap Z.
\end{aligned}$$

计算

$$\alpha_l=\min_{x_i\in S,y_j\notin T}\{l(x_i)+l(y_j)-w(x_iy_j)\}.$$

按如下公式修改可行顶标

$$\hat{l}(v)=\begin{cases} l(v)-\alpha_l, & v\in S,\\ l(v)+\alpha_l, & v\in T,\\ l(v). & \text{其他}.\end{cases}$$

令 $l\leftarrow\hat{l}$, 转第 (1) 步.

算法 5.3 中第 (3) 步中有一个很重要的步骤就是求 $Z=\{v|v\in V(G_l),\text{且}u,v\text{之间存在交错轨道}\}$ 集合。则实际上就可以通过求 u 的交错树来求解。u-交错树的定义为：对于 G 的子图 T，如果 T 是树，$u\in V(T)$，且，满足任意 $v\in V(T)$，T 中从 u 到 v 的轨道是交错轨道。

如图 5.20：

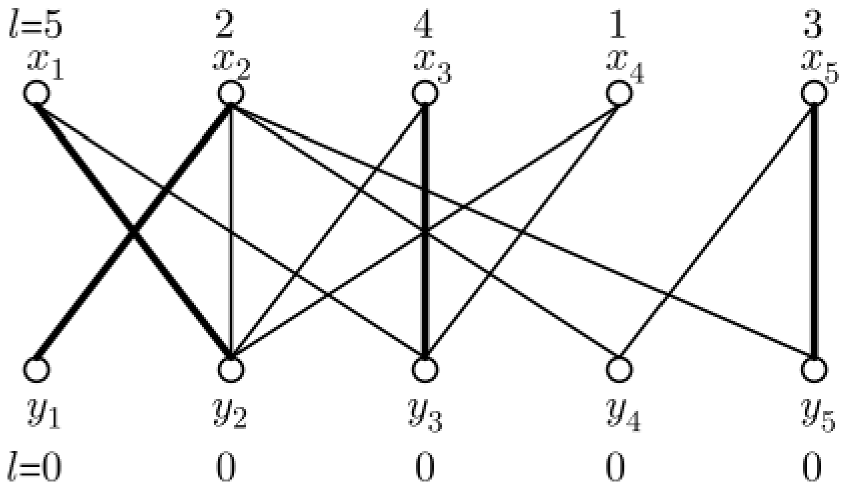

图 5.20 可行顶标与相等子图示例

若选取的 $u=x_4$ 时，可以得到 $x_4 \to y_3 \to x_3 \to y_2 \to x_1$ 这个交错轨道，因此 $Z=\{x_4,y_3,x_3,y_2,x_1\}$。

若选取的 $u=y_4$ 时，可以得到 $y_4 \to x_5 \to y_5 \to x_2 \to y_1$ 这个交错轨道，因此 $Z=\{y_4,x_5,y_5,x_2,y_1\}$。

如果觉得算法 5.1的交错树算法文字太多，看起来很复杂，则可以这样考虑（不妨设选取的顶点 $u \in X$）：再每一次找交错轨道时，从 X 向 Y 查找时，选取不属于匹配 M 的边，而从 Y 向 X 查找时，选取属于匹配 M 的边。（即在本例中，交错轨道从上往下的边都是未加粗的，而从下往上的边是加粗的。）

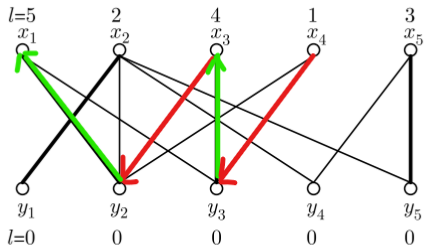

图 5.20　可行顶标与相等子图示例

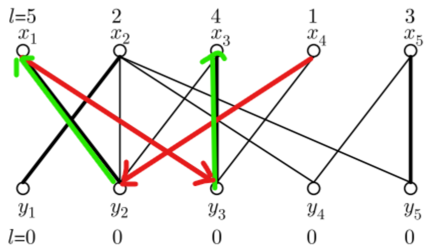

图 5.20　可行顶标与相等子图示例