## §7.6 一般图的最大权对集

### 一．线性规划与对偶问题回顾

Max z

对偶问题：

Min w

由前面的定理知，问题(P)和(D)的可行解满足：

问题(P)和(D)的最优解满足(7.10)式等式成立，从而有

(7.11)式和(7.12)式称为补松弛条件，它们经常表达为如下形式：（松弛条件与最优解）

### 二．最大权对集

1.问题：在赋权图G中找一个对集M，令，使得G不存在对集，满足。

2.赋权M可扩路

G的赋权M可扩路P可能有三种情形：

(a)P上E\M中的边比M中的边多，这样的P称为强可扩路；

(b) P上E\M中的边与M中的边一样多，这样的P称为中性可扩路；

(c) P上E\M中的边比M中的边少，这样的P称为弱可扩路。

\*注意：在情形(b)中，P还可能是一个M-交错圈。

定理7.5：G中存在赋权可扩路当且仅当M不是G中最大权对集。

证明：必要性）如果图中存在赋权可扩路P，则

是G中一个权比M更大的对集，因此，M不是G中权最大的对集。

充分性）假设M不是G中的最大权对集，而是G中最大权对集，有。令。则由一些M和交错路及M

和交错圈组成。由于，必有一个分支P，P上属于的边的权之和大于P上属于M的边的权之和，故P就是G中的赋权M可扩路。

### 三．线性规划问题：

设G是一个赋权图。M是G中一个对集，如果，则令变量，否则令。最大权对集问题可表示为线性规划问题如下：

Maximize

约束条件：

\*1.每条边(u,v)上的权为。

2.第1个约束条件表示跟每个顶点u关联的对集边至多有1条；

3.表示任意个顶点导出的子图，约束条件2表示任意 个顶点导出的子图中最多只有条对集边。

上述线性规划问题的对偶线性规划问题表示为：

Minimize

约束条件：

\*1.是对每个顶点v设的一个对偶变量，是对每个子结构设的一个对偶变量。

2.约束条件中对应的是的那一行中的项, 对应的是那一行中的项。和式

是对每一个包含(u,v)的子结构对应的那一行的求和。

上述原始问题和对偶问题的补松弛条件如下：

我们把上述3个松弛条件称为：边、点和奇子集松弛条件。若原始问题和对偶问题的一个解满足松弛条件，那么这个解就是最优解。

### 四．求解算法

初始时，对集为空对集，即对任意，对偶变量

初始时，除了顶点松弛条件不满足外，原始问题和对偶问题的约束条件都满足，其它松弛条件也都满足。在整个算法中，其它松弛条件保持满足，而顶点松弛条件最终对每个顶点满足，这时算法终止，所得的解即是最优解。

注意：如果某个顶点v不满足顶点松弛条件，则且v是M-非饱和顶点(由)。令

算法在相等子图中调用过程MAPS寻找中的M可扩路。如果可扩路找到，则新对集扩展到两个非饱和顶点r和s，它们满足

。则r和s变成满足松弛条件(II)。我们每次找强可扩路，虽然边上的权不一定增加，但每次扩展对集后，会有新顶点满足点松弛条件(II)，而其它松弛条件仍然满足。因为路上的边都属于，故边松弛条件(I)仍然满足。假设在中找不到强可扩路，这时对对偶变量的值作修改，使得新的边加入，一个伪点被扩展，或导致某个outer点的对偶变量变成0。在后一种情况下，如果该顶点是树根，则现在该顶点满足点松弛条件(II)；如果该顶点不是树根，那么该可扩路是从树根到该顶点的中性可扩路，作了扩展后，树根满足点松弛条件(II)。

注意这个扩展后，使得的点v变成不饱和，由于，它仍然满足点松弛条件(II)。如果是前两种情形(即边加入或伪点被扩展),

这时从同一个树根出发，查找可扩路的过程可以继续下去。最终这个查找将导致该树根满足顶点松弛条件(II)。

算法3：DVC (对偶变量改变过程)

1. FOR 所有最外层标记为outer的顶点u和所有包含在最外层的花中的顶点u，该花的伪点标记为outer DO

;

1. FOR 所有最外层标记为inner的顶点u和所有包含在最外层的花中的顶点u，该花的伪点标记为inner DO

;

1. FOR 所有最外层的花，该花的伪点标记为outer DO

;

1. FOR 所有最外层的花，该花的伪点标记为inner DO

.

\*u是最外层顶点表示u不包含在任何花中；u包含在最外层的花中，表示u包含在最外层的花中，但不包含在嵌套在里面的花中。

\*对偶变量修改的值是保证对偶问题仍有可行解的最大修改值。由过程DEV算出后，用值修改对偶变量，对偶约束条件和非负条件仍然成立。

\*在DVC第4步中，u和v不包含在同一个最外层花中；如果它们在同一个花中，

在修改后仍成立，因为各减(加)了一个, 而加(减)了一个。

算法4：DEV ()

1. ;

其中是最外层的花，它的伪点标记为inner;

1. ;

其中u是最外层标记为outer的顶点，或是包含在最外层花中的顶点，该花的伪点标记为outer;

1. ;

其中u是最外层标记为outer的顶点；或是包含在最外层花中的顶点，该花的伪点标记为outer；并且v是未标记的顶点，或是包含在最外层花中的顶点，该花的伪点未标记；

1. ;

其中u和v都或者是最外层标记为outer的顶点，或者是包含在不同最外层花中的顶点，这些花都标记为outer;

1. 。

\*

(a)如果，那么某个对偶变量变成0。

(b)如果，那么某个对偶变量变成0。

(C)如果，那么相关的边被加入到中，寻找可扩路的过程可以继续进行。

(d)如果，那么相关的边被加入到中。当寻找可扩路的过程继续，最终将导致发现一个奇圈。

算法5：HUT (匈牙利树过程)

DEV;

DVC;

IF THEN 扩展每一个标记为inner的最外层的伪点，它的z变量为0；

IF 并且树根T的y变量 THEN

BEGIN

5. 确定从根到某个y变量是0的顶点的可扩路P;

6. 交换P中的对集边和非对集边；

END;

7. IF 或 THEN 扩展;

8. IF THEN

BEGIN

9. 删除所有inner和outer标号；

10. 转 C;

END;

11. IF 或或 THEN 转 M.

算法6：最大权对集算法

;

FOR 所有 DO

C: 3. 选择一个的M非饱和顶点v；

IF 这样的顶点v不存在 THEN 转L;

标记v outer;

M: 4. MAPS();

B: 5. 确定一个花并收缩它；把该伪点标记为outer并赋给它的z变量0；转M;

H: 6. HUT;

A: 7. 确定该(强)可扩路；通过交换它的对集边和非对集边扩展对集；

删去所有outer和inner标记；转C;

L: 8. 在最终的图中扩展所有剩下的伪点，按发现它们的逆序来做；

在每个扩展的花中找到最大对集，加入整个图的对集。

### 五．例子：

见图7.15. 表格见图7.16.

1. 选择a (a标记为outer)

MAPS(G’): (a, b)被处理，b是非饱和点并且未标记

a b , 转出口A;

outer

A: (a, b)是(强)可扩路，因此(a, b)被加入M; a的标记outer取消；

转向出口C;

2. 选择c (c标记为outer)

MAPS(G’): (c, a)被处理：

c a b

outer inner outer

从b或c不能扩展新的边；

转向出口H；

HUT: DEV:

。因此，

DVC:

(b, c)加入，转向出口M;

MAPS(): (b,c)被处理；因为c是标记为outer，找到一个花，由边(c,a), (a,b)和(b,c)构成；转向出口B;

B: 该花被收缩成伪点B, 并标记为outer；(a,b)从M中删除；

转向出口M;

MAPS(G’): 从B不能扩展新的边；转向出口H;

HUT: DEV: (顶点b和c)，(边(d,b)和(c,e)),

。因此。

DVC: ,

(d,b)和(c,e)加入,

所有outer和inner标记删除；转向出口C;

3. 选择d (d标记为outer)

MAPS(G’): (d,b)被处理；b是非饱点，并且未标记；

d b

outer

转向出口A;

A: (d,b)是可扩路，因此(d,b)加入M; d上的标记outer删除；

转向出口C;

4. 选择e (e标记为outer)

MAPS(G’): (e,c)被处理；

e B d

outer inner outer

从d和e不能再扩展边；转向出口H;

HUT: DEV: (顶点B)，(顶点d和e)，

, ；因此;

DVC:

伪点B的花被扩展，花中偶长度的M交错路被加入T中，

(a,c)被加入M中；

e c a b d

outer inner outer inner outer

转向出口M;

MAPS(G’): 从e, a和d没有边扩展。

转向出口H；

HUT: DEV: (顶点d和e),

因此，

DVC:

,(T的根的y变量变成0)

所有inner和outer标记删除；

转向出口C;

5. 选择f (f标记为outer)

MAPS(G’): (f,a)被处理：

f a c , (c,e)被处理；

outer inner outer

f a c e

outer inner outer

e是非饱和点且未标记，转向出口A;

A: 可扩路是：((f,a), (a,c), (c, e)), 因此，(a,c)从M中删去，

同时将(f,a), (c,e)加入M中；所有inner和outer标记

删除，转出口C；

现在无M非饱和顶点，转出口L。不存在未扩展的伪点，因此算法终止，最大权对集为：。