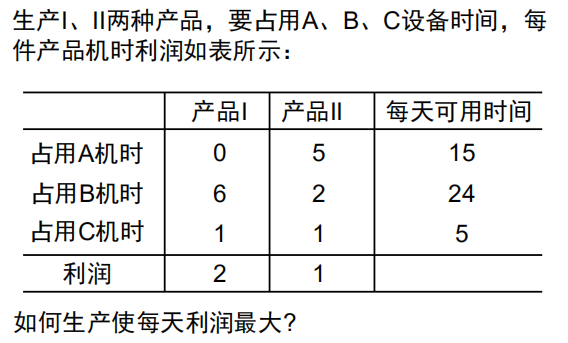
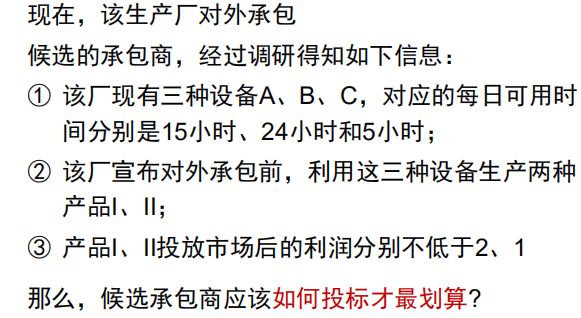
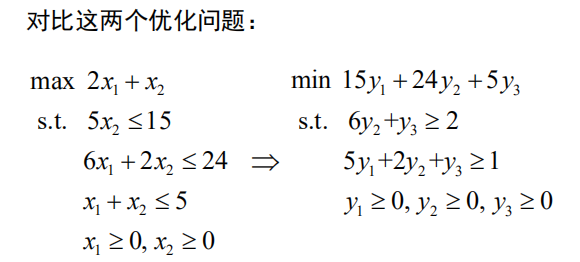
每个线性规划问题都有一个与之对应的对偶问题。

简单考虑如下的生产分配问题



我们有下面的对偶问题：





该问题的任意一个可行解对应的目标函数值都不小于原问题的目标函数值，但是两个问题的最优目标函数值（有限）相同。

一般而言：

1、每个对偶变量对应原问题的一个约束条件

2、原问题是等式约束则对偶变量无不等式约束（非负约束）

3、原问题是不等式约束则对偶变量有不等式约束

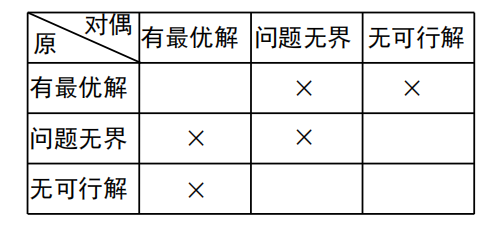
4、原问题变量和对偶问题约束条件同样具有如上规律

任何原问题和对偶问题之间都存在下述相互关系：

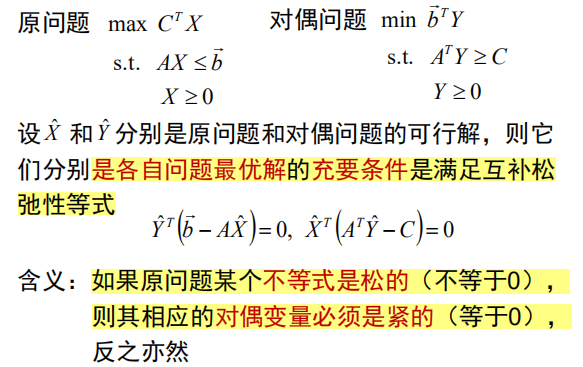
弱对偶性：原对偶问题任何可行解的目标值都是另一问题最优目标值的界（推论：原对偶问题目标值相等的一对可行解是各自的最优解）

强对偶性：原对偶问题只要有一个有最优解，另一个就有最优解，并且最优目标值相等

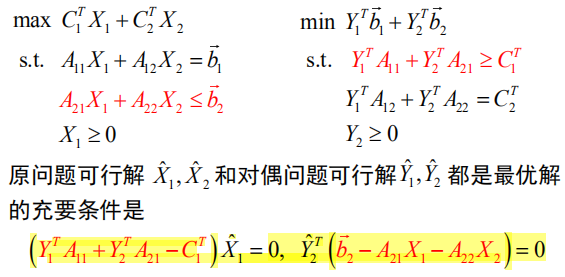
互为对偶的线性规划问题解之间关系有如下四种：



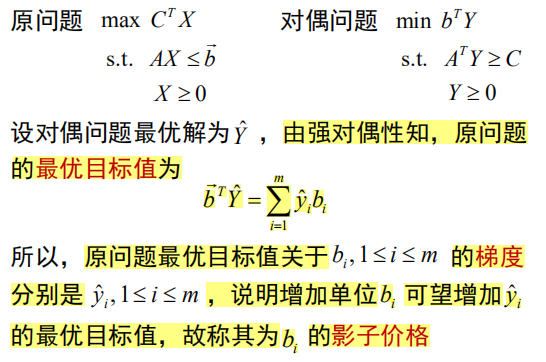
原问题与对偶问题之间存在互补松弛性：



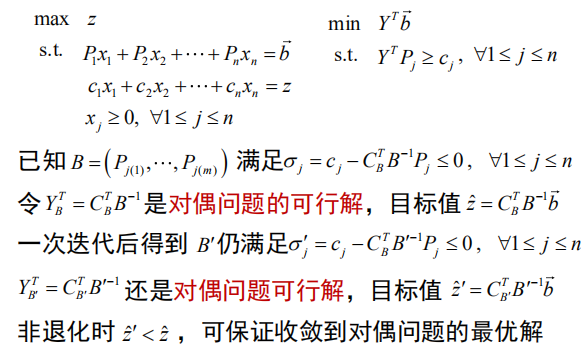
一般形式的线性规划互补松弛定理：



经济学中有所谓影子价格的概念：如果增加某些 约束条件的数值，原问题的最优目标值应该增加，增加单位约束使得原问题最优值的增加量为该约束条件的影子价格。影子价格可以由对偶线性规划问题清楚地描述：



对偶单纯形法：



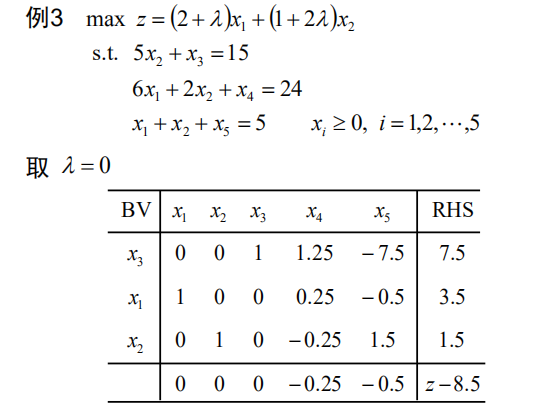
当线性规划问题中地某个约束条件或价值变量中含有参数时，原问题称之为参数线性规划，它有如下的处理方法：

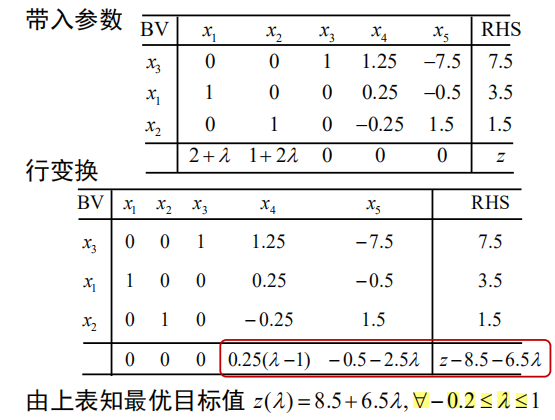
1）固定λ的数值解线性规划问题

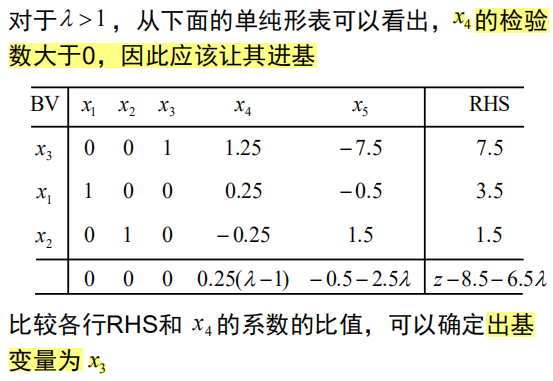
2）确定保持当前最优基不变的λ的区间

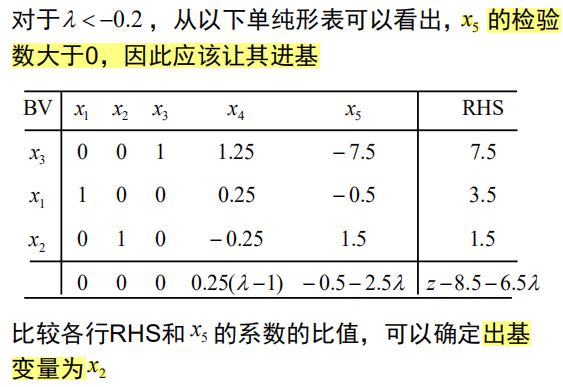
3）确定λ在上述区间附近的最优基，回2）

如以下问题：





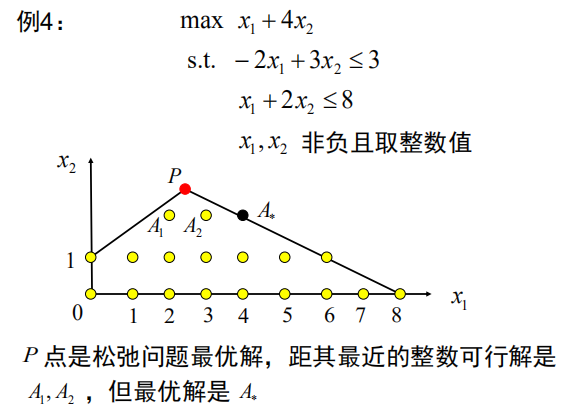




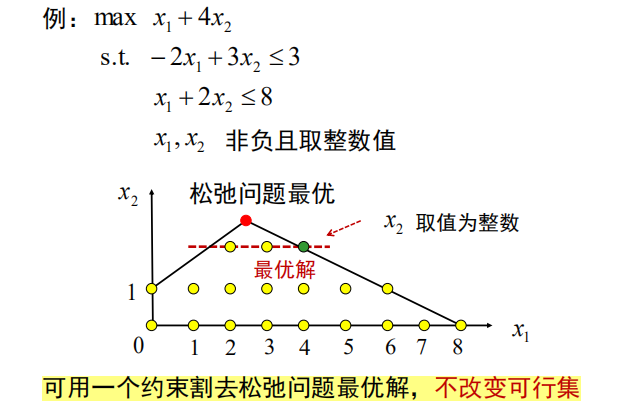
在实际问题中，许多变量以及它们的约束条件往往是离散的，或者说限定在整数域上，这便引入了整数线性规划的概念。

具体而言，整数线性规划包含纯整数线性规划（所有变量是整数变量）、混合整数线性规划（同时包含整数和非整数变量）、0-1型整数线性规划（变量等于0或1）

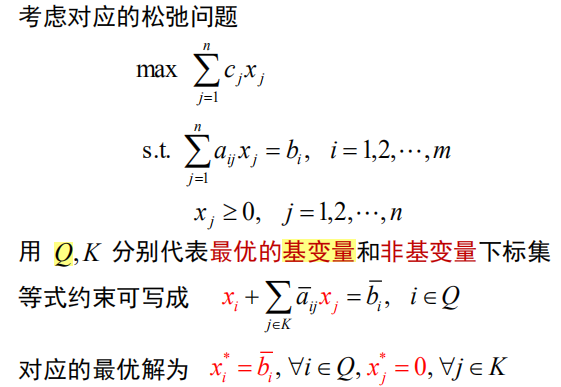
去除整数规划的整数约束后的问题称为其松弛问题。一般情况，原问题的解并不一定是其松弛问题的最优解附近的整数解，例如：

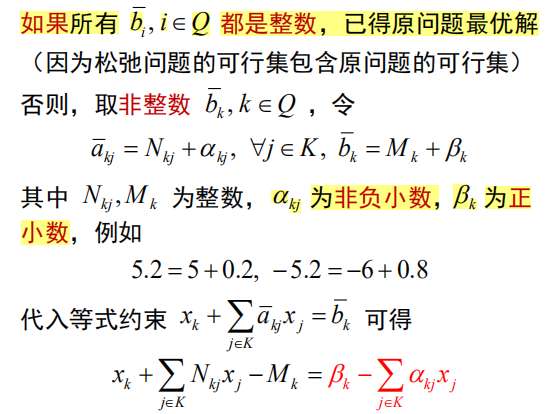


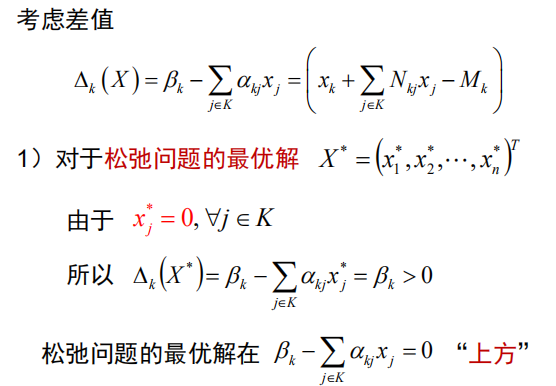
通常的解决办法是在松弛问题的基础上出发，不断地引入整数的约束条件，从而求出整数规划的解。

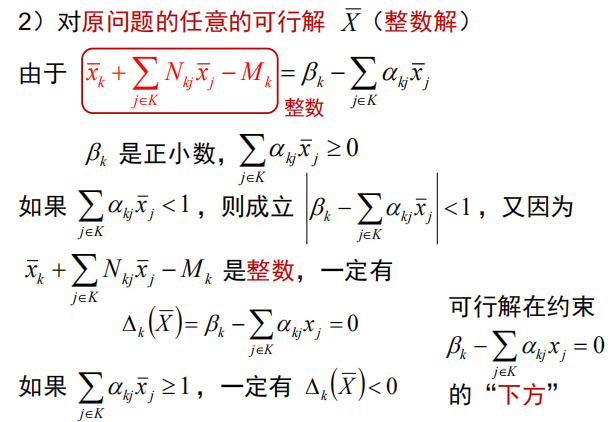


现在，问题便转化为找到合适的约束条件（割平面），以实现不改变原问题可行域，将松弛问题的最优解“割去”。

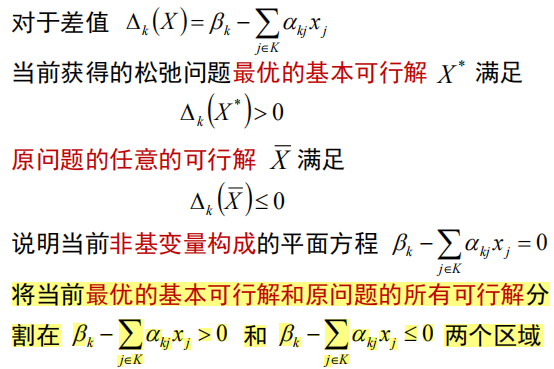




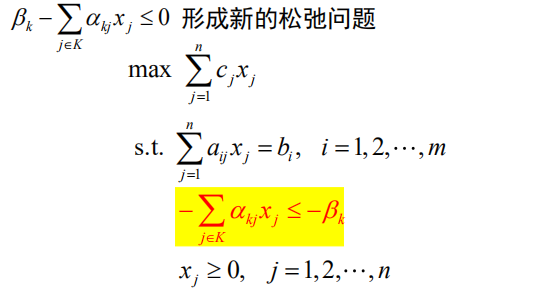




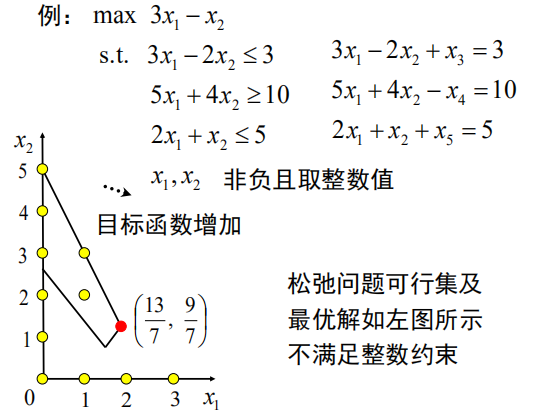
总结前面的讨论可知：

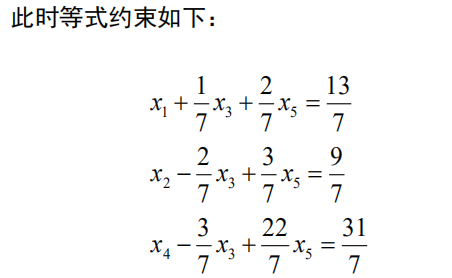


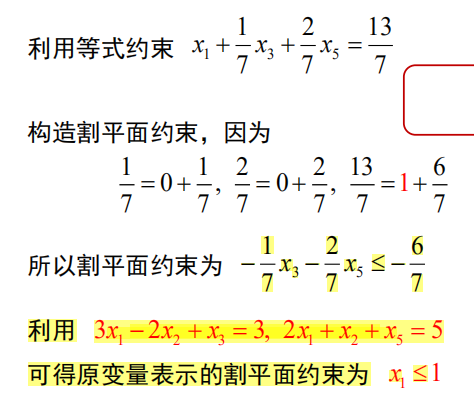
根据前面的讨论，若对松弛问题增加不等式约束：

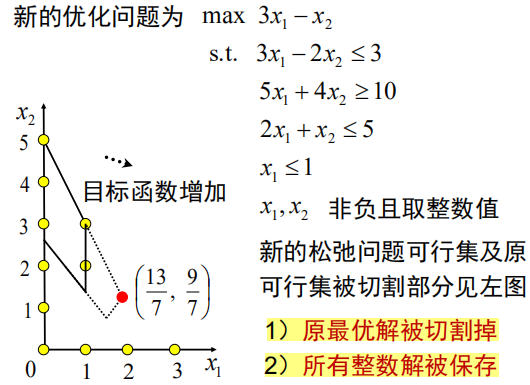


那么当前不满足整数约束的最优解将被切割掉，而原问题的所有的可行解都仍然包含在新的可行集中。





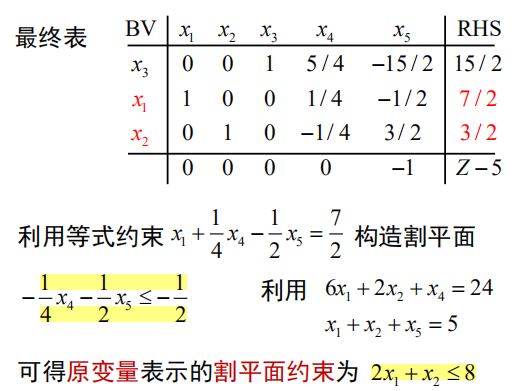




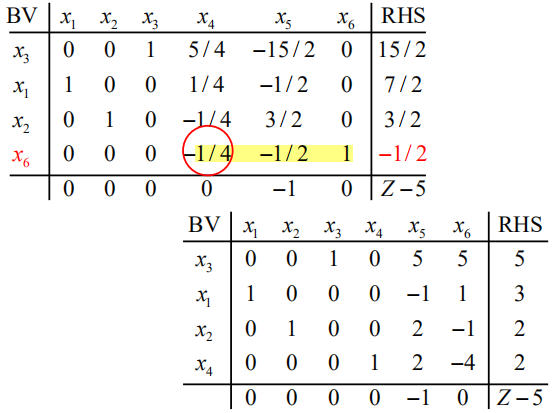
可以观察得到，每次增加一个不等式约束后，可以用新的不等式约束的松弛变量做新增加的基变量，从而上一个松弛问题的非基变量都没有改变，因此其检验数

也不改变，每次增加一个不等式约束后，可以在上一个松弛问题的最后的单纯型表的基础上用对偶单纯型法求解新的松弛问题。

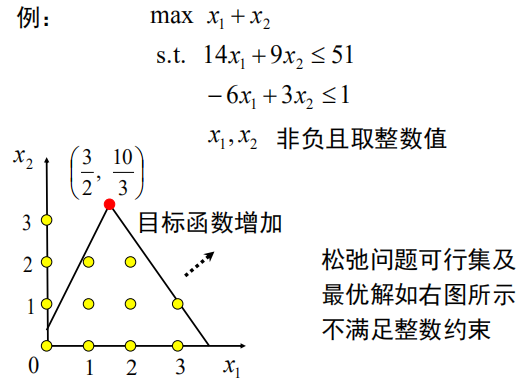
比如某次松弛问题的优化过程已经得到以下的最优单纯形表：



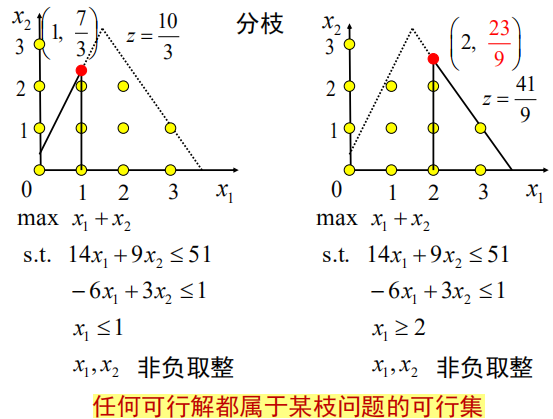
引入变量x6随后的优化过程如下

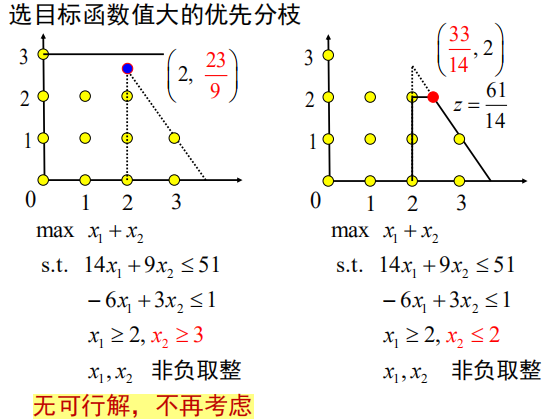


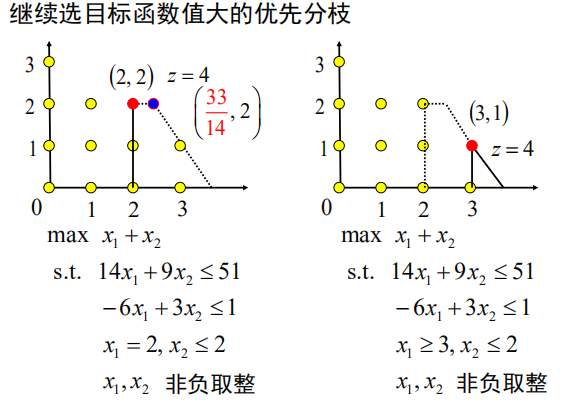
上面的“割平面”方法在实际使用的过程中依然有很多不便之处，所以我们引入一种新的分枝定界法。



依据松弛问题最优解进行如下的分支：

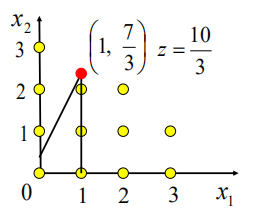






所谓的定界是指，对于上面两张图所示的可行集，已经找到最优解，最优目标函数值等于 4 ，由此确定了该问题最优目标函数的一个下界，如果某个分枝的松弛问题的最优值小于这个界，由于整数最优目标值更小，所以可断定该枝不含最优解，不用再分枝。

回到尚未确定最优解的一枝，如下图所示，由于其松弛问题的最优值小于前面确定的下界4，因此可断定该枝不含最优解，因此不用再分枝，从而确定了该整数规划问题的最优解



0-1型整数线性规划在一般的线性规划问题中有特殊的用途，一般而言，我们用0-1变量统一互相排斥的约束条件，比方说下面的问题：

