Two Dimensional Cutting

1 Problem Model

A large rectangle $A_0 = (L_0, W_0)$ of length L_0 and width W_0 is to be cut into m smaller rectangular pieces; piece i has size (L_i, W_i) and value v_i. Let P_i and Q_i be the minimum and maximum number of pieces of type i that can be cut from $A_0 (0 \leq P_i \leq Q_i$ for $i = 1, \cdots, m)$.

Notations used in this analysis is summarized in Table 1.

notation	meaning
A_0	a large rectangle
L_0	length of the large rectangle
W_0	width of the large rectangle
L_i	length of type i rectangle
W_i	width of type i rectangle
v_i	value of type i rectangle
P_i	minimum number of pieces of type i rectangle
Q_i	maximum number of pieces of type i rectangle

2 Solution

We define q_{ipqr} and x_{ipq} in the following

$$a_{ipqrs} = \begin{cases} 1 & \text{if a piece of type i, when cut its bottom left-hand corner at (p,q), cuts out the point (r,s)} \\ 0 & \text{otherwise} \end{cases}$$

To prevent double counting when two pieces are cut adjacent to one another, we define

$$a_{ipqrs} = \begin{cases} 1 & \text{if } 0 \leq p \leq r \leq p + L_i - 1 \leq L_0 - 1 \text{ and } 0 \leq q \leq s \leq q + W_i - 1 \leq W_0 - 1 \\ 0 & \text{otherwise} \end{cases}$$

We define x_{ipq}

$a_{ipqrs} =$

$$\begin{cases} 1 & \text{if a piece of type i is cut with its bottom left-hand corner at (p,q) where } 0 \leq p \leq L_0 - L_i \text{ and } 0 \leq q \leq W_o - w_i \\ 0 & \text{otherwise} \end{cases}$$

Then the program is

$$\text{maximize } \sum_{i=1}^{m}\sum_{p\in L}\sum_{q\in W} v_i x_{ipq} \tag{1}$$

$$\text{subject to } \sum_{i=1}^{m}\sum_{p\in L}\sum_{q\in W} a_{ipqrs} x_{ipq} \leq 1, \forall r \in L, s \in W \tag{2}$$

$$P_i \leq \sum_{p\in L}\sum_{q\in W} x_{ipq} \leq Q_i, i = 1, ..., m \tag{3}$$

$$x_{ipq} \in (0, 1), i = 1, ..., m, \forall p \in L, q \in W \tag{4}$$

The first constraint ensures that any point is cut out by at most one pieces;
The second constraint ensure that the number of cut pieces of any type lies within the required range;
The third constraint is the integrality constraint.