

# Decomposition-Based Pricing for Integer Programs

John F. Raffensperger, PhD  
Department of Management,  
University of Canterbury  
Christchurch, New Zealand  
[j.raffensperger@mang.canterbury.ac.nz](mailto:j.raffensperger@mang.canterbury.ac.nz)

---

## Abstract

Mamer & McBride recently introduced a new decomposition algorithm for linear programming. They call this decomposition algorithm decomposition-based pricing. Our paper applies decomposition-based pricing to integer programs. For integer programs, we show that decomposition-based pricing produces tighter bounds than other common decomposition algorithms. We use the classic one-dimension cutting stock problem as an example. The subproblem is the same as that in the classic Gilmore-Gomory algorithm. The master corresponds to the cutting stock model developed by Dyckhoff. The new algorithm suggests that the typical formulation of the decomposition master can be improved.

---

## 1 Two old cutting stock algorithms – Gilmore-Gomory and Dyckhoff

The venerable cutting stock problem has been studied since at least 1961, notably by Gilmore and Gomory (1961) and Dyckhoff (1981). Gilmore and Gomory developed a column generation algorithm, with a knapsack subproblem. However, their algorithm can require many iterations, and the integer solution is not always optimal. Dyckhoff observed that the knapsack subproblem could be modeled as a shortest path linear program, which allowed reformulation of the original problem. The reformulated original problem could be solved directly with linear programming without column generation, and the integer solution is optimal. But with Dyckhoff's formulation, branch and bound can take a long time, and some models are too large to solve.

In this paper, we show how to solve Dyckhoff's formulation with column generation. The algorithm used is decomposition-based pricing (Mamer & McBride 2000). When decomposition-based pricing is applied to the integer programming formulation, we show that we can improve the bound given by the Gilmore-Gomory model, while greatly reducing the number of iterations required.

### 1.1 The Gilmore-Gomory cutting stock algorithm

Gilmore and Gomory developed the classic column generation algorithm for the cutting stock problem. Their column generation algorithm uses the following master model.

Indices:  $i$  is a product length demanded.  
 $j$  is a stock cutting pattern.

Parameters:  $d_i$  is the demand for product of length  $i$ .  
 $a_{ij}$  is the number of pieces of length  $i$  in pattern  $j$ .

Variables:  $x_j$  is the number to make of pattern  $j$ .

Model *GGM*: 1) Min  $\sum_j x_j$ ,

- 2)  $\sum_j a_{ij}x_j \geq d_i$  for each demand  $i$ ,
- 3)  $x_j$  integer.

Explanation: 1) Minimise the number of stock lengths required.  
 2) Satisfy demand for each product length  $i$ .  
 3) Only whole lengths can be cut.

In the Gilmore-Gomory algorithm, model  $GGM$  is first solved as a linear program, in order to obtain dual prices from constraint set 2, thereby allowing us to find a new pattern by solution of the following knapsack subproblem.

Indices:  $i$  is the product length demanded.  
 Parameters:  $d_i$  is the demand for product of length  $i$ ,  
 $l_i$  is the length of product  $i$ .  
 $L$  is the length of the uncut stock material.  
 Variables:  $a_i$  is the number of pieces of product length  $i$  to put in the pattern.

Model  $S$ :  
 4)  $\text{Min } 1 + \sum_i \pi_i a_i$   
 5)  $\sum_i l_i a_i \leq L$ ,  
 6)  $a_i$  integer.

Explanation: 4) Minimise the reduced cost of the new pattern.  
 5) The new pattern must be no longer than the stock material.  
 6) Only whole pieces of product can be put in the new pattern.

The Gilmore-Gomory method starts with some reasonable feasible solution in the master  $GGM$ , and solves  $GGM$  with constraint set 3 (integrality) relaxed. The dual prices  $\pi_i$  from constraint set 2 are then used as the objective coefficients for iteration  $j$  in model  $S$ . Model  $S$  is solved, producing a solution  $a_i$ , which becomes a new column  $a_{ij}$  in model  $GGM$ . When the objective value of model  $S$  becomes nonnegative, the LP relaxation of model  $GGM$  has been solved. For an integer solution in  $GGM$ , constraint set 3 (integrality) is then imposed, and a good integer feasible solution can be found.

Unfortunately, this algorithm can take many iterations, even hundreds. The integer solution at the end has no guarantee of optimality. Indeed, it is easy to construct trivial solutions where the integer solution to model  $GGM$  is not optimal.

The subproblem  $S$  can be solved with integer programming or dynamic programming.

## 1.2 Dyckhoff's method

Dyckhoff (1981) observed that since model  $S$  can be solved with dynamic programming, and therefore as a shortest path network program, model  $S$  can be reformulated to solve the original knapsack problem in one go, without the need for column generation. Dyckhoff's model is as follows.

Indices:  $h, i, j$  are measure points where the stock material is to be cut.  
 $k$  is product.  
 Parameters:  $d_k$  demand for product of length  $k$ ,  
 $l_k$  is the length of product  $k$ .  
 $L$  is the length of the uncut stock material.

Variables:  $y_{i,j}$  = pieces of length  $j - i$ , cut at  $i$  inches from the end.  
 $w_{i,L}$  = pieces of waste of length  $L - i$ , cut at  $i$  inches from the end.

Model  $DM$ : 7) Minimise  $\sum_j w_{j,L}$   
 8)  $\sum_h y_{h,i} - \sum_j y_{i,j} = 0$ , for all  $i \leq L - l_K$ ,  
 9)  $\sum_h y_{h,i} - \sum_j y_{i,j} - w_{i,L} = 0$ , all  $L - l_K < i \leq L$ ,  
 10)  $\sum_{i,j: j-i=l_k} y_{i,j} \geq d_k$ , for each product  $k$ ,  
 11)  $y_{i,j}$  integer.

Explanation: 7) Minimise total waste.

- 8) Total material to the left of cut point  $i$  equals total material to the right of cut point  $i$ .
- 9) Total usable material to the left of cut point  $i$  equals total waste material to the right of cut point  $i$ .
- 10) Total material cut of length  $l_k=j-i$  satisfies demand for product  $k$ .

The optimal solution to model  $DM$  gives the optimal integer solution to the cutting stock problem. Unfortunately, the size of this model depends on the magnitude of the data. Larger stock lengths and smaller products result in many variables and rows.

The contribution of this paper is to combine the Gilmore-Gomory column generation procedure with Dyckhoff's formulation to reduce the disadvantages of both of them, while still improving the bounds on the Gilmore-Gomory procedure.

## 2 Decomposition based pricing for the cutting stock problem

### 2.1 What is decomposition based pricing (DBP)?

Mamer & McBride (2000) published a fundamentally new decomposition algorithm for linear programming. Briefly, here is a description of their method.

Suppose we wish to solve

$$P: \{\min cx \mid Ax \geq b \ (\pi), Dx \geq f, x \geq 0\}.$$

To solve this with DBP, we will create a subproblem and a master, similar to Dantzig-Wolfe decomposition. The subproblem is the same as in Dantzig-Wolfe,

$$S^k: \{\min cx + \pi(b - Ax), Dx \geq f, x \geq 0\}.$$

At each iteration  $k$ , we solve  $S^k$ , then we note which variables are positive in the solution to  $S^k$ , and we record those variables' indices in the set  $I^k = \{j \mid x_j > 0 \in S^k \text{ for any } l \leq k\}$ . Since  $I^k$  is a set of variable indices,  $|I^k| \leq n$  for all  $k$ , where  $n$  is the number of variables in  $P$ . Hopefully  $|I^k| \ll n$ .

The DBP master is not the same as the Dantzig-Wolfe master. The master is the original model  $P$ , but restricted to variables that are positive in the subproblem  $S$ .

Algebraically, model  $P^{k+1}$  is restricted to the set of columns  $I^k$ , so  $P^k: \{\min cx \mid Ax \geq b \ (\pi), Dx \geq f, x \geq 0, x_j = 0 \text{ for all } j \notin I^k\}$ .

At each iteration  $k$ , subproblem  $S^k$  provides columns to the master  $P^{k+1}$ . The master  $P^{k+1}$  supplies dual prices  $\pi$  for subproblem  $S^{k+1}$ . As with Dantzig-Wolfe, the stopping condition is  $v(S^k) = v(P^k)$ .

### 2.2 DBP and the cutting stock problem – the LP relaxation.

To apply DBP to the cutting stock problem, we start with Dyckhoff's model  $DM$ . However, we drop all variables except some reasonable feasible solution.

Define arc set  $\alpha$  to contain pairs of indices  $i, j$  that correspond to variables in Dyckhoff's model. For example, if a 10 meter stock length were cut to two 5 meter lengths, then the arcs in  $\alpha$  would be  $(0,5)$  and  $(5,10)$ .

A feasible solution to the subproblem must be a path in the network. It is convenient if the cut lengths are ordered from largest to smallest. For example, if a 10-meter stock length were cut in lengths of 2, 5, and 3, then the arcs in  $\alpha$  would be  $(0,5)$ ,  $(5,8)$ , and  $(8,10)$ . The new model is then:

- Model  $DM'$ :
- 7) Minimise  $\sum_j w_{j,L}$
  - 8)  $\sum_h y_{h,i} - \sum_j y_{i,j} = 0$ , for all  $i \leq L - l_K$ ,
  - 9)  $\sum_h y_{h,i} - \sum_j y_{i,j} - w_{i,L} = 0$ , all  $L - l_K < i \leq L$ ,
  - 10)  $\sum_{i,j: j-i=l_k} y_{i,j} \geq d_k$ , for each product  $k$ ,
  - 11)  $y_{i,j}$  integer for  $(i,j) \in \alpha$ .
  - 12)  $y_{i,j} = 0$  for  $(i,j) \notin \alpha$ .

The subproblem is as before:

- Model  $S$ :
- 4) Min  $1 + \sum_i \pi_i a_i$
  - 5)  $\sum_i l_i a_i \leq L$ ,
  - 6)  $a_i$  integer.

After solving  $S$ , we sort the solution lengths decreasing, create arcs for and add them to model  $DM'$ , then solve  $DM'$  with constraint set 11 (integrality) relaxed. Then we can retrieve dual prices  $\pi_i$  from constraint set 10, to update the objective function in model  $S$ . The algorithm ceases with the optimal linear programming solution when  $1 + \sum_i \pi_i a_i \geq 0$ . The optimal value of model  $DM'$  will be exactly equal to the optimal values of models  $DM$  and  $GGM$ .

We will present numerical results that show that this algorithm converges in fewer iterations than the Gilmore-Gomory method. Furthermore, this new method also improves on model  $GGM$ , when we need integer solutions.

### 2.3 Finding integer solutions to the cutting stock problem with DBP

With model  $DM'$ , we can obtain an integer solution after the LP optimum has been found. All we need to do is impose constraint set 11 (integrality). It may seem that this is the same as imposing integrality in the Gilmore-Gomory master  $GGM$ , but in fact model  $GG'$  tends to produce better solutions.

These models are easily coded in AMPL and solved with Cplex. AMPL's "arc" and "node" notation allows Cplex to use a fast network solver. We will present examples where the integer solutions to  $GG'$  is strictly better than the integer solutions to  $GGM$ .

## 3 References

Dyckhoff, Harald, "A New Linear Programming Approach to the Cutting Stock Problem," *Operations Research*, **29**:6, Nov-Dec. 1981.

Gilmore, P.C. and R.E. Gomory, "A Linear Programming Approach to the Cutting Stock Problem," *Operations Research*, **9**, 1961, pp. 849-859.

Mamer, John W., and McBride, Richard D., "A Decomposition-Based Pricing Procedure for Large-Scale Linear Programs: An Application to the Linear Multicommodity Flow Problem," *Management Science*, **46**:5, May 2000.