Golf-Sport: Managing Operations
A Case Study in Optimization

John D. Bulger & Matthew Hess, Valparaiso University

May 8, 2018

1 Introduction

This problem examines a manufacturing optimization case........

2 Background and Prior Work

Here is a good spot to insertour necessary references. We can evaluate solutions of similar problems

3 Description of Problem

Golf-Sport is a golfclub manufacturing company with three plants in Arizona. Each plant has an attached retail store and is treated as its only supplier. Each location has predicted minimum and maximum sales of each item type per period. The goal of this optimization is to maximize profit over a two-month period by determining the optimal number of each product to make at each location. In this problem, two types of constraints exist: local & global. Local constraints are unique to each plant/store combination, while the global constraints refer to company-wide resources.

4 Mathematical Formulation & Model

This problem is set up as a mixed integer programming problem, with a single objective function to be maximized (representing profit). The decision variable are set to be the production of each product item, indexed by location and period. In determining the maximum profit given the constraints, the solution will also yield the optimal production plan for Golf-Sport.

4.1 Assumptions

Two main assumptions were made in order to create an effective model for this problem. First, this model represents the company's production as if it were not a going-concern. The model is constructed to optimize profit over two months, thereby not building up any additional inventory for carryover into a third month. By default, the model will not have any inventory left at the end of the second month. If further information were to arise along with a modeling need for a third month, this could be accomplished by expanding the current model.

The second assumption used is the carryover of graphite stock between months. Advertising and graphite allotments are companywide in this problem, and it is clearly stated that the amount of money earmarked for advertising each month will not carryover into the next month. Graphite seems to have no such restriction, so it appears reasonable to incorporate the possibility of carryover of graphite stock.

4.2 Variables & Indices

Variable	**Description**
x	Production of Product i in Period j
e	Graphite Stock Carryover

Superscript	**Location**
C	Chandler
G	Glendale
T	Tucson

Index i	Product
s	Steel Shafts
g	Graphite Shafts
r	Forged Iron Heads
w	Metal Wood Heads
h	Titanium Insert Wood Heads
v	Set, Steel Shafts, Metal Heads
u	Set, Steel Shafts, Insert Heads
y	Set, Graphite Shafts, Metal Heads
z	Set, Graphite Shafts, Insert Heads

Index j	Period Description
1	Produced & Sold in Period 1
2	Produced & Sold in Period 2
3	Produced in Period 1 & Sold in Period 2

4.3 Objective

The objective of this model is to maximize profit, represented as the difference between sales revenue and cost. For the purposes of this function let *R, C & I* represent revenue, cost, and inventory cost, respectively. The function can be expressed as:

$$f(x) = \sum_{j=1}^{3} Rx_i - \sum_{j=1}^{3} Cx_i - \sum Ix_{i,3}$$

The total revenue from all products produced and sold, less the cost of manufacture and inventory, is represented in this function. While designed for a two-period optimization, the base function can be easily expanded to accomodate for more periods of production and sales.

4.4 Constraints

In construction of this model, the following constraints were constructed within the parameters set forth by the problem:

$$\sum c_i x_{i,1} + \sum c_i x_{i,3} \leq L_1 \tag{1}$$

$$\sum c_i x_{i,2} \leq L_2 \tag{2}$$

$$\sum c_i x_{i,1} + \sum c_i x_{i,3} \leq P_1 \tag{3}$$

$$\sum c_i x_{i,2} \leq P_2 \tag{4}$$

$$\sum c_i x_{i,1} + \sum c_i x_{i,3} \leq A \tag{5}$$

$$\sum c_i x_{i,2} \leq A \tag{6}$$

$$\sum c_i x_{i,1} + \sum c_i x_{i,3} + e \leq G_1, i = g, y, z \tag{7}$$

$$\sum c_i x_{i,2} \leq G_2 + e, i = g, y, z \tag{8}$$

$$\sum c_i x_{i,1} + \sum c_i x_{i,3} \leq U_1, i = v, u, y, z \tag{9}$$

$$\sum c_i x_{i,2} \leq U_2, i = v, u, y, z \tag{10}$$

$$Min_1 \leq x_{i,1} \leq Max_1 \tag{11}$$

$$Min_2 \leq x_{i,2} + x_{i,3} \leq Max_2 \tag{12}$$

$$x_{i,j} \geq 0, integer \tag{13}$$

$$e \geq 0 \tag{14}$$

Constraints 1-4 represent the labor and packing limitations by plant with cost by product. The first month's costs include the items manufactured and sold that month, as well as the items manufactured for

Item	Period 1	Period 2	Period 1 Carryover
Steel Shafts	0	0	251
Graphite Shafts	100	2000	0
Iron Heads	200	0	200
Metal Wood Heads	35	1	30
Titanium Insert Wood Heads	2000	1299	701
Set, Steel, Metal Heads	0	0	0
Set, Steel, Titanium Heads	0	0	0
Set, Graphite, Metal Heads	0	0	0
Set, Graphite, Titanium Heads	0	35	84

Table 1: Optimal Chandler Production

carryover into the second month. Constraints 5-8 represent the global constraints in this problem: advertising budget and graphite stock. The advertising budget is a fixed amount for each month, with no carryover. The graphite stock has a consistent amount per month as well, but excess from the first month can be carried over into the next month. This carryover amount is represented by e. Constraints 9 & 10 represent the unique assembly time for the sets of clubs, again varying by plant location.

The final constraints (11-14) represent variable bounds. Constraint 11 ensures that all items produced and sold in period 1 are within the minimum and maximum demand. Similarly, inequality 12 ensures that the sum of items produced in period 2 and items carryed over to period 2 are within that month's demand bounds. Constraints 13 & 14 further bound all items to be non-negative.

5 Implementation

5.1 Hardware & Software

The problem was coded and solved in MATLAB R2017b on a personal computer. The code does not appear to be backward-compatible due to the use of the *optimproblem* implementation. The small scale of this optimization does not necessitate a computer beyond average specifications.

5.2 Coding

Code

5.3 Algorithm

This problem was solved using the *intlinprog* function in MATLAB. This function conducts a linear relaxation on the problem, in order to set the upper bound. Then, the algorithm iterates through a branch-and-cut approach to solve for the optimal integer solution. By default, the function will iterate through 1,000,000 iterations prior to terminating. Additionally, the function applies basic heuristics in the form of Gomory cuts. Gomory cuts are an added constraint along a hyperplane to a vertex. They are commonly implemented in most commercial integer programming solvers as an initial heuristic. In optimization software implementations, these cuts tend to be accurate overall and invalid cuts are rarely made [1].

6 Solution

The model was successfully optimized to the same solution over several runs, yielding the following optimal solution with constraints as laid out in the original problem above:

7 Further Analysis

Conduct range sensitivity, examine extra problems in case study

8 Conclusions & Implications

8.1 Mathematical Approach

Evaluate our approach, suggest changes in future. EFFICIENCY

8.2 Company Approach

Discuss optimized solution, range sensitivity, business suggestions

References

[1] Gérard Cornuéjols, François Margot, & Giacomo Nannicini. (2013). On the safety of Gomory cut generators. *Mathematical Programming Computation, 5*(4), pp. 345-395.