INFORMS TutORials in Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

How to Influence and Improve Decisions Through Optimization Models

Jeffrey D. Camm

To cite this entry: Jeffrey D. Camm. How to Influence and Improve Decisions Through Optimization Models. *In* INFORMS TutORials in Operations Research. Published online: 19 Oct 2018; 1-19.
https://doi.org/10.1287/educ.2018.0180

informs
https://pubsonline.informs.org/series/educ/

TUTORIALS IN OPERATIONS RESEARCH
© 2018 INFORMS | ISBN 978-0-9906153-2-3
https://doi.org/10.1287/educ.2018.0180

How to Influence and Improve Decisions Through Optimization Models

Jeffrey D. Camm
Wake Forest University, School of Business, Winston-Salem, North Carolina 27109
Contact: cammjd@wfu.edu (JDC)

Abstract Industry's recent increased focus on data-driven decision making and the use of analytics in all sectors from sports to financial services to technology and healthcare has led to a resurgence in the interest of traditional operations research tools such as optimization, simulation, and decision analysis. As organizations mature analytically, it seems likely that we will see a further increase in interest in prescriptive analytics, including optimization modeling, which is the focus of this tutorial. With massive amounts of data being routinely collected in real time and an increased awareness on the part of management of the value of data, the availability of data are typically no longer the bottleneck in the optimization modeling process. Increased computing speed, improved algorithms, parallel processing, and cloud computing have increased the size of optimization problems we can solve to optimality. Considering better data availability and the dramatic increase in our ability to solve problems, what are the impediments keeping us from having significant influence and impact on decision making? Going forward, it is possible that our inability to (1) structure a messy decision problem into a useful optimization framework, (2) properly use the model to deliver valuable insights for management, and (3) communicate to management the value proposition of our insights will become the new reasons we might fail to have the impact we know is possible. In this tutorial, we review several types of optimization models and the art of modeling—that is, the process of going from mess to model. We discuss how to use an optimization model to provide not just "the answer" but also insights that will be useful to managers and influence their decision making. We discuss the importance of communication in influencing decisions and provide examples and best practices relevant for optimization.

Keywords optimization • math programming • modeling • influence • data-driven decision making

1. Introduction to Optimization Modeling

Mathematical optimization is the process of maximizing or minimizing some function, called the *objective function*, usually subject to a set of constraints. In operations research, we often refer to mathematical optimization as *mathematical programming*, because the act of optimizing results in an optimal program—that is, an optimal course of action to follow. In analytics, optimization falls into the category of *prescriptive analytics*, the set of analytical methodologies that prescribe a course of action.

An optimization *model* is a mathematical representation of a decision problem where the goal is to maximize (or minimize) the objective function, usually subject to constraints. Both the objective function and constraints involve mathematical functions of the *decision variables*. We use optimization algorithms such as the simplex algorithm, branch and bound, and generalized reduced gradient to solve the optimization model. By "solve," we mean find values of the decision variables that satisfy all the constraints of the problem and provide the maximum (or minimum) value of the objective function. In the vernacular of mathematical programming, any setting of the decision variables that satisfies all the constraints is called a *feasible solution*. Optimization algorithms therefore find a best feasible solution where best is measured by the objective function. Although fast and effective algorithms are necessary for solving optimization models, the focus of this tutorial is on the process of building optimization models and how to effectively use such models for improved decision making.

The applications of optimization models in industry are plentiful. Capital budgeting, financial portfolio optimization, supply chain network design, operations planning and scheduling, inventory optimization, marketing mix models, product design and product line management, and vehicle routing and facility location are but a few of the types of applications of optimization that have had significant impact on how businesses are run every day.

For the purposes of this tutorial, we define an optimization model as follows:

Maximize $f(x_1, x_2, x_3, \ldots, x_n)$
subject to

$$\begin{aligned} g_j(x_1, x_2, x_3, \ldots, x_n) &\le b_j \quad j = 1, 2, \ldots, k, \\ g_j(x_1, x_2, x_3, \ldots, x_n) &\ge b_j \quad j = k+1, k+2, \ldots, k+l, \\ g_j(x_1, x_2, x_3, \ldots, x_n) &= b_j \quad j = k+l+1, k+l+2, \ldots, k+l+m. \end{aligned}$$

Here, we have n decision variables and $k + l + m$ constraints with objective function f, constraint functions g_j, and constants b_j.

In Section 2, we discuss the process of modeling a decision problem as an optimization problem. We use examples of actual optimization problems from industry ("real-world" problems). In Section 3, we discuss how to generate alternative optima, and in Section 4, we discuss how to generate a family of solutions for your client to consider (even suboptimal solutions). We have found the approaches in Sections 3 and 4 to be incredibly useful in consulting. The importance of communication and implementation issues are discussed in Section 5. Although not often taught in the classroom, these "soft skills" are, in our experience, often the biggest impediment to having an impact on decision making. Section 6 is a discussion about benchmarking and measuring potential impact for your client. In Section 7, we discuss implementation issues. In Section 8, we discuss how to deal with the resistance to change that can come from using optimization models over time. We end with a summary and other lessons learned from 20 years of consulting in optimization with the humble understanding that we are still learning after all these years.

2. Building Optimization Models

Camm et al. [8] list five steps of the decision-making process:

Step 1. Identify and define the problem.
Step 2. Determine the criteria that will be used to evaluate possible solutions.
Step 3. Determine the set of possible solutions.
Step 4. Evaluate the possible solutions.
Step 5. Choose one of the possible solutions.

The first step is far from trivial and arguably the most difficult step. Managers typically do not describe problems; similar to a medical patient, they describe the symptoms of underlying problems. The job of the analyst is to listen, ask many questions, and diagnose/define the real problem to be solved. For more on the importance of defining the right problem, please see Cooper [10] and Weddell-Wedellsborg [21].

In the context of optimization modeling, the remaining steps involve developing the objective function (Step 2), determining the constraint set (Step 3), and solving the optimization model (Steps 4 and 5). The modeling process is the process of moving from the mess to a model, going from Step 1 to Steps 2 and 3.

Once a problem has been properly defined, there are several techniques that may be used to structure the modeling process, including influence diagrams (Powell and Baker [18]) and semantic networks (Evans and Camm [11]). If the problem is an optimization problem, we have found that answering the following list of simple questions is useful in model construction:

1. What am I trying to decide?
2. How am I restricted?
3. What is my metric of solution quality?

We use the Calhoun Textile Mill case (Camm et al. [6]) to demonstrate the modeling process as well as other key concepts in this tutorial.

Calhoun Textile Mill Case

The sales department at Calhoun Mills has confirmed orders for each of 15 fabrics. Management is concerned that there is not enough production capacity to meet demand.

Calhoun has two types of looms: dobby and regular. Dobby looms can be used to make all fabrics and are the only looms that can weave certain fabrics, such as plaids. Given in Table 1 are the demand for each fabric, the production rate for each fabric on each type of loom, the variable production cost for each fabric, and the outsource cost for each fabric. Note that if a fabric can be woven on each type of loom, then the production rates are equal, and so the variable cost per yard of producing in the mill is the same regardless of which type of loom is utilized. Also note that fabrics 1–4 cannot be made on a regular loom.

Calhoun has 90 regular looms and 15 dobby looms. The mill operates continuously during the quarter: 13 weeks, seven days a week, and 24 hours a day. Assume that the changeover cost for the looms is negligible.

For now, management would like to know how to allocate the looms to the fabrics and which fabrics to outsource in order to minimize the cost of meeting demand. How much of each fabric should be produced on each type of loom, and how much of each fabric should be outsourced?

What Am I Trying to Decide? We are trying to decide how much of each fabric to produce on a regular loom or a dobby loom, or how much to outsource.

How Am I Restricted? We have the following restrictions: (i) demand must be satisfied for each fabric, (ii) we cannot make fabrics 1–4 on a regular machine, (iii) we have a limited

Table 1. Data for Calhoun Textile Mills problem.

Fabric	Demand (yd)	Dobby rate (yd/hr)	Regular rate (yd/hr)	Mill cost ($/yd)	Outsource cost ($/yd)
1	16,500	4.653	x	0.66	0.80
2	52,000	4.653	x	0.56	0.70
3	45,000	4.653	x	0.66	0.85
4	22,000	4.653	x	0.55	0.70
5	76,500	5.194	5.194	0.61	0.75
6	110,000	3.809	3.309	0.62	0.75
7	122,000	4.185	4.135	0.65	0.80
8	62,000	5.232	5.232	0.49	0.60
9	7,500	5.232	5.232	0.50	0.70
10	69,000	5.232	5.232	0.44	0.60
11	70,000	3.733	3.733	0.64	0.80
12	82,000	4.135	4.135	0.57	0.75
13	10,000	4.439	4.439	0.50	0.65
14	380,000	5.232	5.232	0.31	0.45
15	62,000	4.185	4.135	0.50	0.70

Note. x means that the fabric cannot be made on a regular loom.

number of dobby looms and hence a limited number of dobby loom hours, and (iv) we have a limited number of regular looms and hence a limited number of regular loom hours.

What Is My Metric of Solution Quality? The measure of solution quality is the total cost of satisfying demand.

On the basis of the answer to the first question, we may define the following decision variables:

r_i = the amount in yards of fabric i to produce on a regular loom, $i = 1, 2, \ldots, 15$.
d_i = the amount in yards of fabric i to produce on a dobby loom, $i = 1, 2, \ldots, 15$.
o_i = the amount in yards of fabric i to outsource, $i = 1, 2, \ldots, 15$.

For ease of exposition, we may also define the follow data:

c_i = the cost per yard to produce fabric i, $i = 1, 2, \ldots, 15$.
s_i = the cost per yard to outsource fabric i, $i = 1, 2, \ldots, 15$.
p_i = production rate for fabric i in yards per loom hour, $i = 1, 2, \ldots, 15$.
$t_i = 1/p_i$, loom hours per yard required for producing fabric i, $i = 1, 2, \ldots, 15$.
dem_i = the demand for fabric i in yards, $i = 1, 2, \ldots, 15$.

Finally, we note that, ignoring any changeover time or down time, the number of loom hours available are approximately 32,760 dobby loom hours ($15 \times 24 \times 7 \times 13 = 32{,}760$) and 196,560 regular loom hours ($90 \times 24 \times 7 \times 13 = 196{,}560$). We may then use the following linear program to help management answer the question at hand:

$$\text{Minimize} \sum_{i=1}^{15} c_i\,(r_i + d_i) + \sum_{i=1}^{15} s_i\, o_i \tag{1}$$

subject to

$$\sum_{i=1}^{15} t_i\, r_i \le 196{,}560, \tag{2}$$

$$\sum_{i=1}^{15} t_i\, d_i \le 32{,}760, \tag{3}$$

$$r_i + d_i + o_i = dem_i \qquad i = 1, 2, \ldots, 15, \tag{4}$$

$$r_i = 0 \qquad i = 1, 2, 3, 4, \tag{5}$$

$$r_i,\, d_i,\, o_i \ge 0 \qquad i = 1, 2, \ldots, 15. \tag{6}$$

The objective function (1) is the production cost plus the outsource cost. Constraint (2) ensures that we do not exceed regular loom capacity. Constraint (3) ensures we do not exceed the dobby hours available. The constraint set (4) (there are 15 such constraints, one for each fabric) ensures that demand for each fabric is satisfied. Constraint set (5) ensures that we do not allow production for fabrics 1–4 on a regular loom. Finally, we refer to constraint set (6) as the nonnegativity constraints, as we cannot produce or outsource negative amounts of fabric.

The model (1)–(6) is a *linear program* because all functions are linear in the decision variables, the decision variables are nonnegative, and the decision variables are continuous (i.e., we allow fractional solutions). We may solve (1)–(6) using the simplex algorithm or an interior point algorithm.

A few observations are in order. First, this model is designed to help management understand how much of each fabric should be made on each type of loom or outsourced. The fundamental question really is, "Does anything need to be outsourced?" A closer look at the data in Table 1 shows that it is always cheaper to produce than to outsource. Hence, if we have enough capacity, we should produce everything. This tells us a lot about what to expect when we solve (1)–(6). If we solve our model and nothing is outsourced, it means we had enough capacity to do so. If we solve the model and some fabric is outsourced, we should see that all loom time is used.

Takeaway. *Always understand the data and anticipate model behavior—that is, the general solution characteristics. Understanding your data and your model is critical to being able to communicate recommendations to your client.*

The solution of (1)–(6) is shown in Table 2. All regular and dobby loom time is used. Only fabric 6 is outsourced, and it is split between regular looms and the outsourcing firm. Fabric 15 is also split, but between dobby and regular looms. The total cost is $599,108.16.

For completeness, we mention that inequality constraints that hold as equality are said to be *binding*. At optimality, the binding constraints form the bottleneck that prevents us from doing better. A by-product of solving a linear program is that we are provided a *shadow price* that indicates the change in the objective function for an increase of one on the right-hand side of a constraint. Each variable has a *reduced cost*, which is the shadow price for the nonnegativity constraint for that variable. In the Calhoun Mills case, regular and dobby loom hours are both binding constraints.

Now let us discuss some extensions of the basic model we have constructed. If in addition to nonnegativity we require that the decision variables take on only integer values—that is, a model consisting of (1)–(5) and (7) from below—then we have a *linear integer program:*

$$r_i, d_i, o_i \geq 0 \text{ and integer} \quad i = 1, 2, \ldots, 15. \tag{7}$$

Variables that are required to be integer but also between 0 and 1 inclusive are called *binary variables*. Binary variables allow us to model yes–no decisions. To illustrate the use of binary variables, let us consider another extension of (1)–(6). Suppose that we require that only one of the three options—regular loom, dobby loom, or outsource—may be used for each fabric.

We define the following binary variables:

$R_i = 1$ if fabric i is produced on regular looms and 0 if not; $i = 1, 2, \ldots, 15$.
$D_i = 1$ if fabric i is produced on dobby looms and 0 if not; $i = 1, 2, \ldots, 15$.
$O_i = 1$ if fabric i is outsourced and 0 if not; $i = 1, 2, \ldots, 15$.

Table 2. An optimal solution to the Calhoun Mills case.

Fabric	Dobby	Regular	Outsource	Total
1	16,500.00	0.00	0.0	16,500
2	52,000.00	0.00	0.0	52,000
3	45,000.00	0.00	0.0	45,000
4	220,00.00	0.00	0.0	22,000
5	0.00	76,500.00	0.00	76,500
6	0.00	6,871.51	103,128.49	110,000
7	0.00	122,000.00	0.00	122,000
3	0.00	62,000.00	0.00	62,000
9	7,500.00	0.00	0.00	7,500
10	0.00	69,000.00	0.00	69,000
11	0.00	70,000.00	0.00	70,000
12	0.00	82,000.00	0.00	82,000
13	0.00	10,000.00	0.00	10,000
14	0.00	380,000.00	0.00	380,000
15	9,230.09	52,769.91	0.00	62,000

The following constraints, when added to (1)–(6), will ensure that only one option is used for each fabric:

$$r_i \le dem_i R_i \qquad i = 1, 2, \ldots, 15, \tag{8}$$

$$d_i \le dem_i D_i \qquad i = 1, 2, \ldots, 15, \tag{9}$$

$$o_i \le dem_i O_i \qquad i = 1, 2, \ldots, 15, \tag{10}$$

$$R_i + D_i + O_i = 1 \qquad i = 1, 2, \ldots, 15, \tag{11}$$

$$R_i, D_i, O_i \in \{0, 1\} \qquad i = 1, 2, \ldots, 15. \tag{12}$$

Constraints (8)–(10) are sometimes referred to as *setup constraints*. Consider, for example, constraint (8). If $R_i = 0$, then $r_i = 0$. If $R_i = 1$, then we have $r_i \le dem_i$, but (11) implies $D_i = O_i = 0$, and so by (4), (9), and (11), we have $r_i = dem_i$. The previously mentioned constraint (11) ensures that only one option is used for each fabric, and (12) ensures that R_i, D_i, and O_i are binary.

The solution of the extended model is shown in Table 3. Note that the solution is quite different from the solution in Table 2. Fabrics 11 and 12 are outsourced. There are unused dobby and regular hours (2,205.5 hours and 9,065 hours, respectively). Also, note that because we added constraints to the original model, it cannot be that cost decreases. The total cost of \$611,262.75 is an increase of \$12,154.59 over the solution in Table 2. Also, the "all-or-nothing" nature of the revised model means that it is no longer true that all loom hours must be used before we outsource.

3. Alternative Optima

There are only four possible outcomes when solving an optimization problem: (i) there is a unique optimal solution, (ii) multiple optimal solutions exist, (iii) the problem is infeasible, and (iv) the problem is unbounded. Case (i) happens frequently and is often *assumed* to be the case. Case (iv) typically means you have a misspecified (underconstrained) model. Case (iii) can and does certainly happen in practice. For example, one way we might have found we do not have enough loom hours to meet demand in the Calhoun Mills problem is that we tried to run a related scheduling optimization model without an outsourcing option and found it to be

Table 3. The solution to Calhoun Mills with only one option per fabric.

Fabric	Dobby	Regular	Outsource	Total
1	16,500.00	0.00	0	16,500
2	52,000.00	0.00	0	52,000
3	45,000.00	0.00	0	45,000
4	22,000.00	0.00	0	22,000
5	0.00	76,500.00	0.00	76,500
6	0.00	110,000.00	0.00	110,000
7	0.00	122,000.00	0.00	122,000
3	0.00	62,000.00	0.00	62,000
9	7,500.00	0.00	0.00	7,500
10	0.00	69,000.00	0.00	69,000
11	0.00	0.00	70,000.00	70,000
12	0.00	0.00	82,000.00	82,000
13	0.00	10,000.00	0.00	10,000
14	0.00	380,000.00	0.00	380,000
15	0.00	62,000.00	0.00	62,000

infeasible. In this section we focus on case (ii), where there are multiple optimal solutions—that is, more than one best solution. It is hard to imagine that if alternative optimal solutions exist your client would not want to know about them. He or she would almost surely like to choose among the alternative solutions.

As noted by Williams [22] and many other texts, in a linear program, we can detect alternative optima by noticing at optimality that there is (i) a binding constraint with zero shadow price or (ii) a variable equal to zero with zero reduced cost. This classical teaching is fine, but what if we really want to explore optimal alternatives when they exist? We can construct a new linear program to find an alternative optimal solution with characteristics we want. As discussed by Camm et al. [8], we can use a revised model to find an alternative optimal solution that in an additive sense is most different from the solution in Table 2.

Define the following sets of indices of the variables that have a value of zero in the optimal solution (see Table 2):

$Z_D = \{5, 6, 7, 8, 10, 11, 12, 13, 14\}$,
$Z_R = \{1, 2, 3, 4, 9\}$,
$Z_O = \{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$.

We may use the following revised model to find an alternative optimal solution:

$$\text{Maximize} \left(\sum_{i \in Z_D} d_i + \sum_{i \in Z_R} r_i + \sum_{i \in Z_O} o_i \right) \tag{13}$$

subject to

$$\sum_{i=1}^{15} t_i \, r_i \le 196{,}560, \tag{14}$$

$$\sum_{i=1}^{15} t_i \, d_i \le 32{,}760, \tag{15}$$

$$r_i + d_i + o_i = dem_i \quad i = 1, 2, \ldots, 15, \tag{16}$$

$$r_i = 0 \quad i = 1, 2, 3, 4, \tag{17}$$

$$r_i, d_i, o_i \ge 0 \quad i = 1, 2, \ldots, 15, \tag{18}$$

$$\sum_{i=1}^{15} c_i (r_i + d_i) + \sum_{i=1}^{15} s_i \, o_i = 599{,}108.16. \tag{19}$$

From Table 2, the objective function (13) is the sum of the decision variables with value zero in the original solution. Constraints (14)–(18) are the same as in the original model. Constraint (19) is the objective function from the original model set equal to the optimal objective function value. This new constraint limits the set of feasible solutions to only those that are optimal in the original model (1)–(6). In this revised model, if the optimal objective function value is positive, we have found a solution that is, in an additive sense, maximally different from the original solution in that the sum of the variables that were zero in the original solution is as positive as possible. In summary, we have found a solution that meets the original constraint set and keeps the optimal objective function value to the original problem, but it has at least some positive value for variables that used to sum to zero (hence, the new solution is different but still optimal).

Solving the revised model results in an objective function value of 26,539.26. Table 4 shows the original and the alternative optimal solutions. In looking at the two optimal solutions, if for some reason it is deemed better to split the production of fabric 8 rather than fabric 15 over dobby and regular looms, then the alternative optimal solution would be preferred.

In a similar way, if there is interest in a single variable or set of variables, the objective function (13) can be replaced with that set. For example, if we want to find an alternative optimal solution with the largest value for the amount of fabric 11 made on a dobby loom, we would replace (13) with

$$\text{Maximize } d_{11}. \tag{20}$$

We can also find alternative optima when a model has binary variables. Balas and Jeroslow [1] provide a clever way of finding alternative optimal solutions for binary integer programs. After solving a binary integer program with variables X_j, we form two sets of indices:

$$O = \{j \mid X_j^* = 1\},$$
$$Z = \{j \mid X_j^* = 0\}.$$

Here, O is the set of indices of variables that were 1 in the optimal solution, and Z is the set of indices of variables that were 0 in the optimal solution. Consider adding the following constraint to the binary program:

$$\sum_{j \in O} X_j - \sum_{j \in Z} X_j \le |O| - 1, \tag{21}$$

where $|O|$ is the cardinality of the set O. By adding this constraint, we eliminate the previous solution but no other feasible solutions. If we append constraint (21) to the original problem, only one of three outcomes can occur: (i) we find a new solution, and it has the same optimal objective function value as our original solution; (ii) we find a new solution, but the objective function value has deteriorated; or (iii) the problem becomes infeasible. This procedure works iteratively; that is, we may apply constraints (21) as we find new solutions. This procedure provides an approach to finding alternative optimal solutions. Solve the original binary

Table 4. Two optimal solutions for the Calhoun Mills case.

Original optimal solution				Alternative optimal solution			
Fabric	Dobby	Regular	Outsource	Fabric	Dobby	Regular	Outsource
1	16,500	0	0	1	16,500	0	0
2	52,000	0	0	2	52,000	0	0
3	45,000	0	0	3	45,000	0	0
4	22,000	0	0	4	22,000	0	0
5	0	76,500	0	5	0	76,500	0
6	0	6,871.51	103,128.49	6	0	6,871.5103	103,128.49
7	0	122,000	0	7	0	122,000	0
8	**0**	**62,000**	0	8	**19,039.26**	**42,960.737**	0
9	**7,500**	**0**	0	9	**0**	**7,500**	0
10	0	69,000	0	10	0	69,000	0
11	0	70,000	0	11	0	70,000	0
12	0	82,000	0	12	0	82,000	0
13	0	10,000	0	13	0	10,000	0
14	0	380,000	0	14	0	380,000	0
15	**9,230.09**	**52,769.91**	0	15	**0**	**62,000**	0

Note. Bold indicates where the solutions are different.

program, apply constraint (21), and re-solve. If the objective function is the same, we have found an alternative optimal solution. Repeat the process (keeping all previously added constraints) and stop when either case (ii) or (iii) occurs.

To illustrate this, consider the Ohio Banking problem discussed by Sweeney et al. [19].

Ohio Banking Problem

Prior to 1979, banks in Ohio were only allowed to place branches in a county where the bank had a principal place of business. A new law in 1979 allowed banks to put branches in any county where the bank has a principal place of business and in any county adjacent to one in which it has a principal place of business. The question posed is, What is the minimum number of principal places of business, and in which counties should they be located to enable branches in all eighty-eight counties of Ohio? A county map of Ohio is shown in Figure 1.

The model for this problem is a classic *set cover* problem where the constraints ensure that for each of the 88 counties, the county has a principal place of business or a county adjacent to it has a principal place of business.

Let $x_j = 1$ if county j has a principal place of business and 0 if not; $j = 1, 2, \ldots, 88$.

Define data $a_{ij} = 1$ if county i and county j share a border and 0 if not (note that $a_{ii} = 1$). We need to solve the following binary program:

$$\text{Minimize} \sum_{j=1}^{88} x_j \tag{22}$$

subject to

$$\sum_{j=1}^{88} a_{ij} x_j \geq 1 \qquad i = 1, 2, \ldots, 88 \tag{23}$$

$$x_j \in \{0, 1\} \qquad j = 1, 2, \ldots, 88. \tag{24}$$

As an example of constraints (23), the first such constraint ensures that county 1 is covered (see Figure 1, in which county 1 shares borders with counties 1, 2, 18, and 19). Hence the first constraint is

$$x_1 + x_2 + x_{18} + x_{19} \geq 1.$$

Other constraints are formulated in the same manner.

An optimal solution to this problem is shown in Figure 2. The optimal objective function value is 15. The locations of the principal places of business are

$$O = \{8, 14, 17, 19, 29, 33, 35, 38, 51, 55, 59, 68, 75, 78, 87\}.$$

Applying a constraint using (21) and re-solving results in a second solution with an objective function value of 15:

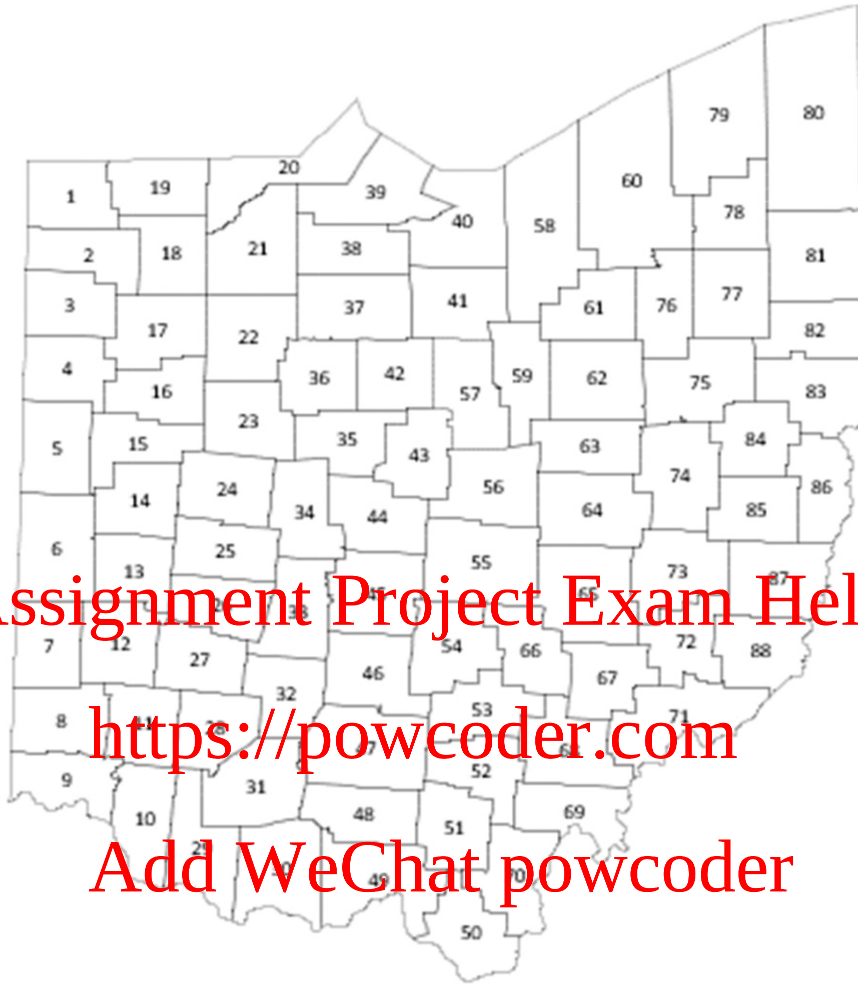
$$O = \{4, 8, 14, 18, 29, 33, 35, 38, 51, 55, 59, 68, 75, 78, 87\}.$$

Again, applying a constraint using (21) and re-solving results in a third solution with an objective function value of 15:

$$O = \{8, 14, 17, 18, 29, 33, 35, 38, 51, 55, 59, 68, 75, 78, 87\}.$$

Applying a constraint using (21) and re-solving results in an objective function value of 16. We know therefore that there are three alternative optima. The three solutions are shown in Figure 3 with the differences highlighted. Apart from the model, there might be good reasons for choosing one of these three solutions over the other two (e.g., land availability).

Figure 1. County map of the state of Ohio.



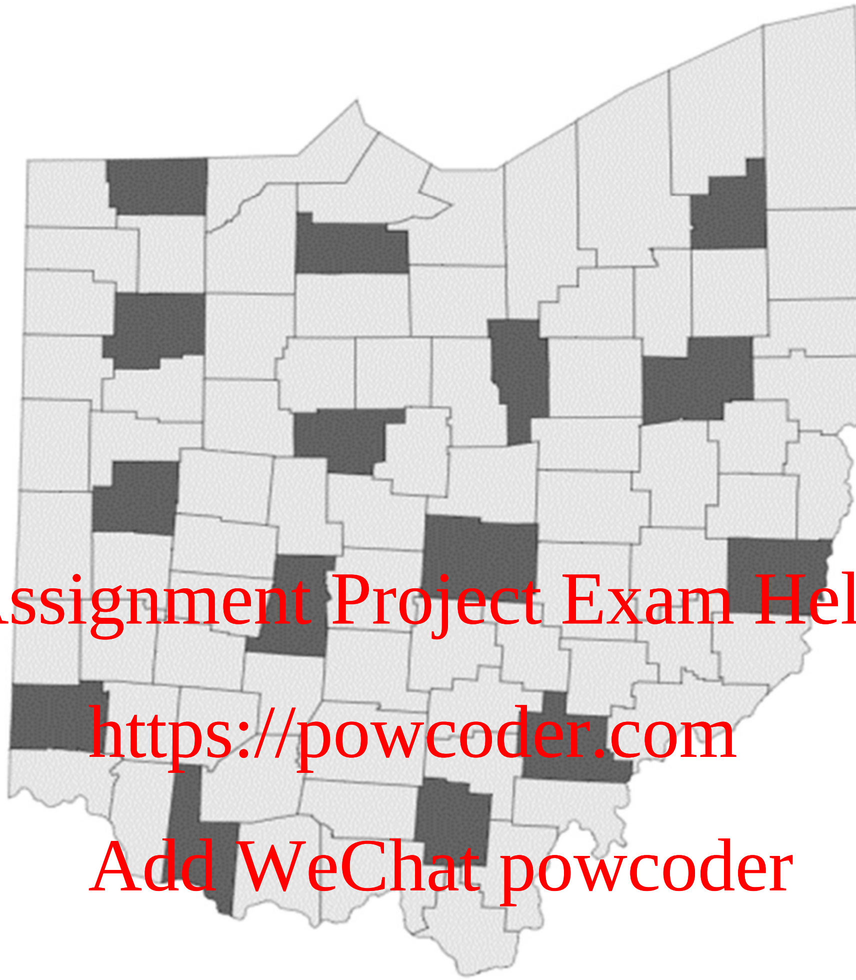
Takeaway. *Explore the model for alternative optima. Your client will likely want to know if alternative optimal solutions exist. There might be factors external to the model that make one of the alternatives much preferred over the others.*

In the next section, we discuss moving beyond alternative optima to generating useful suboptimal solutions.

4. Families of Solutions

Ultimately, managers are responsible for decisions, and so for many situations, the goal is not necessarily to provide "the answer" but rather to better inform the manager making the decision. The danger with optimization models is that by their very nature, they are designed to give "the answer." It is incumbent on analysts to use optimization models in a way that best serves the interest of the client/manager. This means using optimization to provide information, insights, and alternatives for the client to consider. We have just discussed alternative optima, but we can use the models we develop as a laboratory for experimenting and go beyond just finding alternative optima to also generate a family of solutions that contain some suboptimal solutions. A mathematically unique optimal solution might be optimal by millions of dollars or 10 dollars. Wouldn't that be good for a manager to know?

Figure 2. An optimal solution to the Ohio Banking problem.



And if there are other solutions very close to the optimal solution, wouldn't that be valuable information?

In Camm [4], I discuss an experience I had with a client for whom I was constructing a model for supply network design. The model was a mixed integer program. The model contained some real variables corresponding to the flow of goods from the fixed plants to distribution centers (DCs) and then from the DCs to the customers, as well as binary variables for indicating chosen DC locations. The objective was to find the cost-minimizing number and locations for the DCs to satisfy forecasted demand and service requirements. By solving several models for different network sizes (number of DCs), we determined that the number of DCs should be five. The projected savings was a little over 14%. The current network had five DCs, but three of the five current DCs were not in the optimal solution. Using Balas and Jeroslow's constraints on the binary portion of the problem, I generated a family of solutions of suboptimal solutions with five DCs (the optimal solution was unique). By exploring suboptimal solutions, we found that the second-best solution only increased cost by less than 0.5% but contained four of the five current DC locations. Opening and closing costs of facilities and the cost of disruption were not in the model. The client implemented the second-best solution, which required changing only one of the

Figure 3. The three alternative optimal solutions for the Ohio Banking problem.

existing DC locations, and retained almost all the cost savings of the mathematically optimal solution.

Takeaway. *Explore the model for families of solutions, including suboptimal solutions. Optimal solutions can be uniquely optimal by a lot or a little. Because models usually do not contain all the details of the real decision problem, a suboptimal solution to your model might be optimal for the client.*

For more reading on how to generate a diverse family of solutions, see Trapp and Konrad [20].

5. Communicating Results and Influencing Decisions

The best modeling and analysis, without effective influential communication to management, will go unused. This problem has been discussed for many years. See, for example, Woolsey [23], Lavasseur [15], Fellers [12], and Liberatore and Luo [16]. The recent increased interest in data-driven decision making brought on by analytics and big data has led to a new focus on data visualization and effective communication with data. An excellent reference on both storytelling and data visualization is the book by Knaflic [14]. Here, we provide a brief treatment based on Knaflic [14] and Camm et al. [7].

The first rule of effective presentation is to know your audience. Different audiences will require different types of messaging and delivery. An effective presentation for analysts will certainly be very different from an effective presentation for executives. A simple way to structure your presentation to your audience is to think of it as a three-act play. The first act (setup) is to establish an understanding of the problem/opportunity. The second act (conflict) focuses on the tension caused by the problem set up in the first act. The third act (resolution) resolves the tension through a proposed solution. Let us go back to the original version of the Calhoun Mills problem we discussed in Section 2 and model (1)–(6). We might structure our presentation to management as follows.

Act 1

Calhoun Mills has demand for 15 fabrics for the next quarter and needs to plan production. Revenue is fixed by contracts, so Calhoun seeks a way to meet the demand at minimal cost.

Act 2

It appears Calhoun does not have enough loom capacity to meet demand. Outsourcing some of the fabrics will be necessary, and outsourcing is always more expensive than internal production of the fabrics.

Act 3

Analysis shows that it is most cost effective to outsource some of the fabric 6 production. All other fabrics can be made in-house. The expected total cost is $599,108.16.

Using the three-act structure is a simplifying way of structuring the presentation to your client that forces you to tell a story, rather than spend too much time elaborating on what analysts like to talk about—namely, *how* you solved the problem.

In addition to a storytelling focus, good data visualization techniques can help you be successful in influencing the client to act. Camm et al. [7] provide the following simple guidelines for effective communication with data:

- *Consider design and layout:* The positioning of information and the type of chart used should draw attention to and clarify your message.
- *Avoid clutter*: Simpler is almost always better. For example, it is tempting to "jazz up" your presentation with three-dimensional pie and bar charts, but the third dimension typically adds no value and, in many cases, can obscure your message.
- *Use color purposefully and effectively*: Use color with a purpose—for example, to distinguish categories or draw attention to a part of your presentation.

Combining well-structured storytelling with good visualization will help you influence your client, but the story itself must also be compelling. This often means that you as an analyst must fully understand the solution you are recommending. There is a famous quote that has become known as the Woolsey/Swanson rule that states, "People would rather live with a problem they cannot solve rather than accept a solution they cannot understand" (Woolsey and Swanson [24], p. 169). The onus is on the analyst to convince the client that a proposed solution is the answer. To drive home this point, let us continue with the Calhoun Mills case.

Recall the solution is given in Table 2. The solution uses all dobby and regular loom hours, and a portion of fabric 6 production is outsourced. If this is the recommendation (outsource some of the fabric 6 production), any manager worth his or her salt will ask the question, why fabric 6? Figures 4–7 show an example set of slides that (a) follows the three-act structure, (b) uses simple graphics, and (c) gives a rationale for the solution. We could also provide alternative optima information as previously developed but, for brevity eliminated that here.

Takeaway. *Structure your presentation to your client as a three-act play: setup, conflict, and resolution. Use effective data visualization: consider the proper design and layout, avoid clutter, and use color purposefully and effectively. If possible, demystify your recommendation to avoid the black-box syndrome. Your client is going to want to know* why *you are recommending the solution you present. "Because this is the optimized solution" is not a good explanation.*

Figure 4. Sample slide: First act (setup).

Figure 5. Sample slide: Second act (conflict).

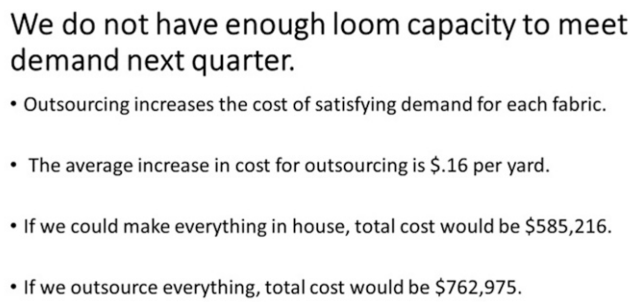


Figure 6. Sample slide: Third act (resolution).

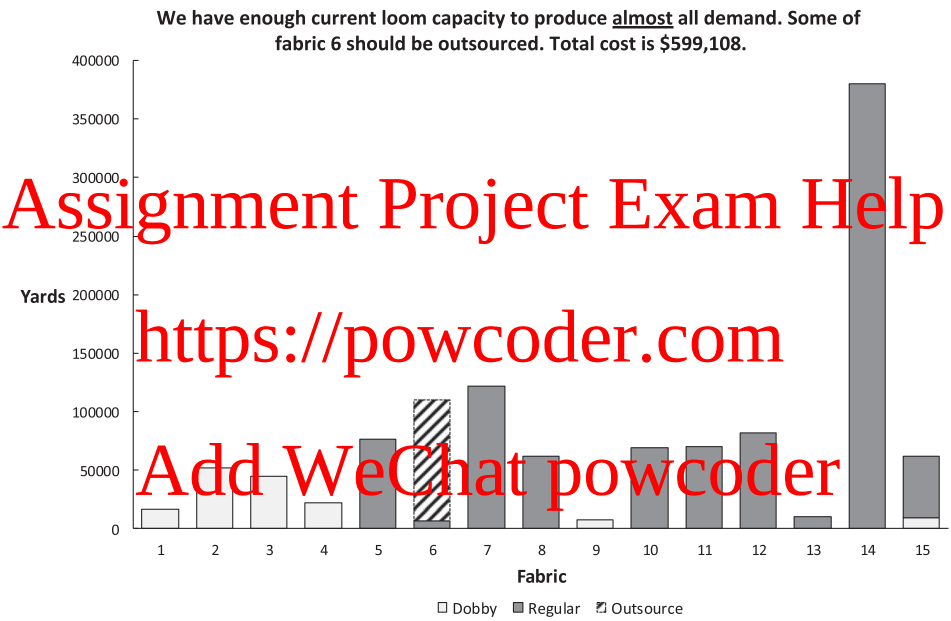
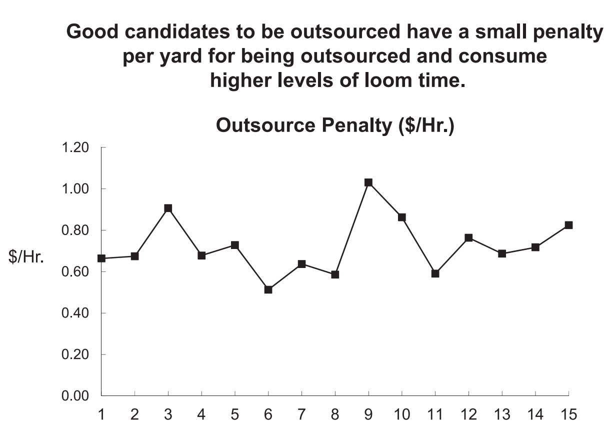


Figure 7. Sample slide: Rationale for the solution.



6. Benchmarking

For the Calhoun Mills solution discussed in the previous section, we were able to directly explain the solution based on the data. This is often possible for simpler models but might not always be the case for more complex models. In addition to quantifying the value of the recommendations, *benchmarking* can often help build a manager's confidence with a proposed solution. Benchmarking involves locking down the optimization model to the current plan before optimizing, essentially using the model to calculate a *baseline* value of the current plan. In a sense we obtained a benchmark for the Calhoun Mills case by deleting the loom capacity constraints so that all the demand could be produced in-house. That led to a cost of $585,216, which is not feasible, but does provide a lower bound on cost. The cost recommended from the model is $599,108, which is $13,892 higher than if we could produce everything in-house.

To illustrate a more typical use of benchmarking, let us consider a supply chain optimization problem. In this case, the product is produced outside the United States. The supply chain design problem is to choose ports of entry, the number of DCs and their locations, as well as the flow of product from port to DC and from DC to customers. The objective is to minimize total delivery cost subject to meeting demand and service delivery requirements. We have current annual demand by customer zone as well as forecasted demand for each customer zone five years into the future. We may use the following benchmarking procedure, outlined in the subsections below.

Current Baseline

Once the optimization model is constructed, we lock in via constraints the current ports, current DCs, and the current flow of product using current annual demand. Thus, we are using the model as a calculator to validate the data we have input to the model. In my experience, if the model's objective function value is within, say, ±10% of what the accountants say, the client will have faith in the model and be willing to move forward.

Optimized Current

In this case, we lock in the *structure* of the supply chain; that is, we lock in current ports and current DCs but allow the product flow to be optimized. This provides the client some immediate potential savings without the capital investment required for structural changes. The difference between the objective function values of the *current baseline* and the *optimized current* is the estimate of the quick achievable savings.

Future Baseline

The future baseline is the same model as the current baseline but uses future demand and allows for any extra needed capacity as a result of demand growth. That is, we assume the same ports and DCs as the current baseline and the same proportional delivery from each port to each DC and each DC to each customer zone as in the current operation.

Future Optimized

In this case, we truly optimize, allowing ports, DCs, and all flows to be chosen to minimize future costs and to satisfy future demand and service requirements. The difference between the objective function values of the *future baseline* and the *future optimized* is the value being provided to the client five years in the future.

A former supply chain project I participated in resulted in the following numbers for the objective function of cost:

- Current baseline = $213 million, optimized current: $210.8 million, current potential savings of $2.3 million.

- Future baseline = \$248.7 million, future optimized = \$230.2 million, future annual savings once fully implemented of \$18.5 million.

We note that it is possible to work as a consultant based on a percentage of realized savings contract. For a more in-depth discussion of this, see Camm and Tayur [5].

Takeaway. *Benchmarking to provide baseline values is important for building client acceptance of model results and for providing an estimate of the value of the optimization.*

7. Implementation

A colleague of mine recently said, "Insights without implementation are just fun and interesting facts" [17]. This is absolutely correct. If, at the end of the day, your client has not implemented anything from your project, then there is no value added and no impact. Therefore, it is imperative to provide your clients with a glide path from their current situation to the desired end state where projected benefits can be realized. We will use the Ohio Banking problem previously discussed as an example. Recall that we have three alternative optimal solutions to the set cover problem, as shown in Figure 3. Let us suppose that the first solution found is the solution chosen for implementation. How can we help the client with implementation?

If the client can implement 5 principal places of business in each of the next three years, then we should provide a plan for which of the 15 locations should be opened in each of the next three years. To provide a plan for implementation, we need a metric of the quality or contribution of each principal place of business location. The population of each county is available from census data, and we may use that as a proxy for banking business. We may use an extension of the set cover model to solve a related problem known as the maximal coverage problem (MCP) (Church and ReVelle [9]). We can use an MCP model to maximize the population reached, given a limit k on the number of principal places of business opened. The MCP is defined as follows:

Let $x_j = 1$ if we place a principal place of business in county j and 0 if not; $j = 1, 2, \ldots, 88$.
Let $y_i = 1$ if county i is covered and 0 if not; $i = 1, 2, \ldots, 88$.

Also, define data $a_{ij} = 1$ if county i and county j share a border and 0 if not (note that $a_{ii} = 1$). Define pop_i to be the population of county i. We have the following MCP model:

$$\text{Maximize} \quad \sum_{i=1}^{88} pop_i y_i \tag{25}$$

subject to

$$\sum_{j=1}^{88} a_{ij} x_j \geq y_i \quad i = 1, 2, ..., 88, \tag{26}$$

$$\sum_{j=1}^{88} x_j \leq k, \tag{27}$$

$$x_j \in \{0, 1\} \qquad j = 1, 2, ..., 88, \tag{28}$$

$$y_i \in \{0, 1\} \qquad i = 1, 2, ..., 88. \tag{29}$$

The objective function (25) maximizes the population reached. Note that by virtue of y_i being binary, the population of each county will be counted only once. The constraint set (26) ensures that a county i is covered only if it or an adjacent county is selected as a principal place of business. Constraint set (27) ensures that no more than k principal places of business are selected.

We may use (25)–(29) to generate a glide path to the end state of 15 principal places of business over the three-year implementation period. First, we set all variables not in the solution to zero; that is, $x_j = 0$ for all $j \notin O = \{8, 14, 17, 19, 29, 33, 35, 38, 51, 55, 59, 68, 75, 78, 87\}$. The year 1 solution for implementation is found by setting $k = 5$ and solving. The locations from the year 1 solution are constrained to one, and then the model is solved again for the next five by setting $k = 10$ and resolving. Finally, the chosen second set of five is locked down to one. The remaining set of five is implemented in year 3. In this way, we have found the three sets of five principal places of business to open in each of the first three years so that the maximal reachable population is achieved as early as possible.

Implementation (or transition) plans can usually be found by using an extension or restriction of the original model used to make the recommendation.

Takeaway. *If your work does not lead to implementation of your recommendations, then it is unlikely to have provided value or impact. Your recommendation should include an implementation plan for the accepted solution. Your model is useful in developing such a plan.*

8. Persistence

Persistence in optimization has to do with the extent to which a previously calculated optimal solution persists into the future (Brown et al. [3]). In situations where a model is solved over time, there will likely be strong managerial resistance if the solution changes radically from one time period to the next. Dramatic changes in the optimal solution over time can result in too much disruption that is not worth the effort. As previously discussed, a suboptimal solution might be very close to optimal. So, to the extent that the previously implemented solution is close to optimal for the next period, it might be good enough to continue to use going forward, either partially or completely, in the future. In fact, the implementation plan derived in the previous section used persistence. In solving for year 2 of the implementation plan, had we not locked in the principal places of business found for year 1 with $k = 5$, we could have found a completely different set of locations when solving with $k = 10$ for year 2. Instead, we made sure that the year 1 locations persisted in the year 2 solution.

Brown et al. [3] discuss different ways of implementing this idea of persistence, including, for example, that some subset of the previous solution persists whereas the other parts are free to change.

Takeaway. *In situations where an optimization model is solved and re-solved over time, dramatic changes in the solution from one run to the next can lead to unnecessary disruption and managerial resistance to change. Enforcing persistence in a solution through constraints can mitigate the disruption and accompanying resistance.*

9. Conclusion

In this tutorial, we have provided lessons learned from 20 years of applying optimization models for clients. Modeling is both an art and a science. Understanding and properly defining the problem are the key first steps. Similar to a good physician, a good modeler needs to be a good listener and ask questions to get beyond symptoms to the root of the underlying condition to be modeled. This can be an exhausting exercise that requires patience and endurance both on your part as the analyst and on the part of your client as well. When asked what he would do if he had an hour to save the world, Albert Einstein replied, "I would spend 55 minutes defining the problem and then five minutes solving it." Professor Einstein is spot on.

In my experience, solving the model is easy. The challenge is to use the model in creative ways to provide valuable information to your client. Especially with an optimization model, we tend to think we have found "the answer." That is a dangerous and risky way to operate.

Rather, use your optimization model to learn and provide valuable alternatives for your client. I highly recommend the classic paper by Geoffrion [13].

Another challenge is to effectively influence the decision maker. Strong data visualization and storytelling skills are key, along with a detailed implementation plan. You need to put yourself in your client's position. What would you need to know to implement recommendations and have an impact? Have you provided that to your client? For more reading on how to drive change through optimization, I recommend Brown and Rosenthal [2].

In closing, over the last decade, the increased availability of data and interest in analytics has led to an increased use of operations research and optimization. Interest in data science and artificial intelligence is now also increasing dramatically. The ability to properly define problems and solve them with optimization models will undoubtedly continue to grow in importance. Opportunities for you, as an optimization analyst, have never been greater.

Acknowledgments

The author's knowledge and practice of optimization in the real world has been enhanced by interaction over the years with some very smart and pragmatic colleagues, who shared with him a passion for real-world optimization. The author acknowledges the influence of Jerry Brown (Naval Postgraduate School), Ted Newton (Procter & Gamble, retired), Dennis Sweeney (Professor Emeritus, University of Cincinnati), the late Rick Rosenthal (Naval Postgraduate School), and the late Gene Woolsey (Colorado School of Mines).

References

[1] E. Balas and R. Jeroslow. Canonical cuts on the unit hypercube. *SIAM Journal on Applied Mathematics* 23(1):61–69, 1972.

[2] G. Brown and R. Rosenthal. Optimization tradecraft: Hard-won insights from real-world decision support. *Interfaces* 38(5):356–366, 2008.

[3] G. Brown, R. Dell, and R. Wood. Optimization and persistence. *Interfaces* 27(5):15–37, 1997.

[4] J. Camm. A (very) short course on suboptimization. *Interfaces* 44(4):428–431, 2014.

[5] J. Camm and S. Taylor. Editorial: How to monetize the value of OR. *Interfaces* 40(6):446–450, 2010.

[6] J. Camm, P. M. Dearing, and S. Tadisina. The Calhoun Textile Mill Case: An exercise on the significance of linear programming model formulation. *IIE Transactions* 19(1):23–28, 1987.

[7] J. Camm, M. Fry, and J. Shafer. A practitioner's guide to best practices in data visualization. *Interfaces* 47(6):473–488, 2017.

[8] J. Camm, J. Cochran, M. Fry, J. Ohlmann, D. Anderson, D. Sweeney, and T. Williams. *Business Analytics*, 3rd ed. Cengage, Cincinnati, 2019.

[9] R. Church and C. ReVelle. The maximal covering location problem. *Papers of the Regional Science Association* 32:101–118, 1974.

[10] M. Cooper. Defining problems: The most important business skill you've never been taught. Entrepreneur (September 26), https://www.entrepreneur.com/article/237668, 2014.

[11] J. Evans and J. Camm. Using pictorial representations in teaching linear programming modeling. *IIE Transactions* 22(2):191–195, 1990.

[12] J. Fellers. Using the cooperative learning model to teach students "people skills." *Interfaces* 26(5): 42–49, 1996.

[13] A. Geoffrion. The purpose of mathematical programming is insight, not numbers. *Interfaces* 7(1): 81–92, 1976.

[14] C. Knaflic. *Storytelling with Data*. John Wiley & Sons, New York, 2015.

[15] R. Lavasseur. Effective communication—A critical skill for MS/OR professionals. *Interfaces* 21(2): 22–24, 1991.

[16] M. Liberatore and W. Luo. A comparison of technical and soft skill requirements for analytics and OR professionals. *Interfaces* 43(2):194–197, 2013.

[17] B. Martin. Personal communication, July 18, 2018.

[18] S. Powell and K. Baker. *Business Analytics: The Art of Modeling with Spreadsheets*, 5th ed. John Wiley & Sons, New York, 2016.

[19] D. Sweeney, R. Mairose, and R. K. Martin. Strategic planning in bank location. *Proceedings of the 1979 American Institute for Decision Sciences Conference*, Vol. 2, 74–76, 1979.

[20] A. Trapp and R. Konrad. Finding diverse optima and near-optima to binary integer programs. *IIE Transactions* 47(11):1300–1312, 2015.

[21] T. Weddell-Wedellsborg. Are you solving the right problems? *Harvard Business Review* 95(1): 76–83, 2017.

[22] H. P. Williams. *Model Building in Mathematical Programming*, 5th ed. John Wiley & Sons, West Sussex, UK, 2013.

[23] G. Woolsey. An exemplary essay on communication or corporate style, corporate substance, & the sting. *Interfaces* 9(3):10–12, 1979.

[24] R. E. D. Woolsey and H. Swanson. *Operations Research for Immediate Application: A Quick and Dirty Manual*. HarperCollins, New York, 1975.