Solving Optimization Problems Using The Simplex Algorithm

By Jason Anderson

Prof. Phillips

CSC 349

6 / 2 / 06

Introduction

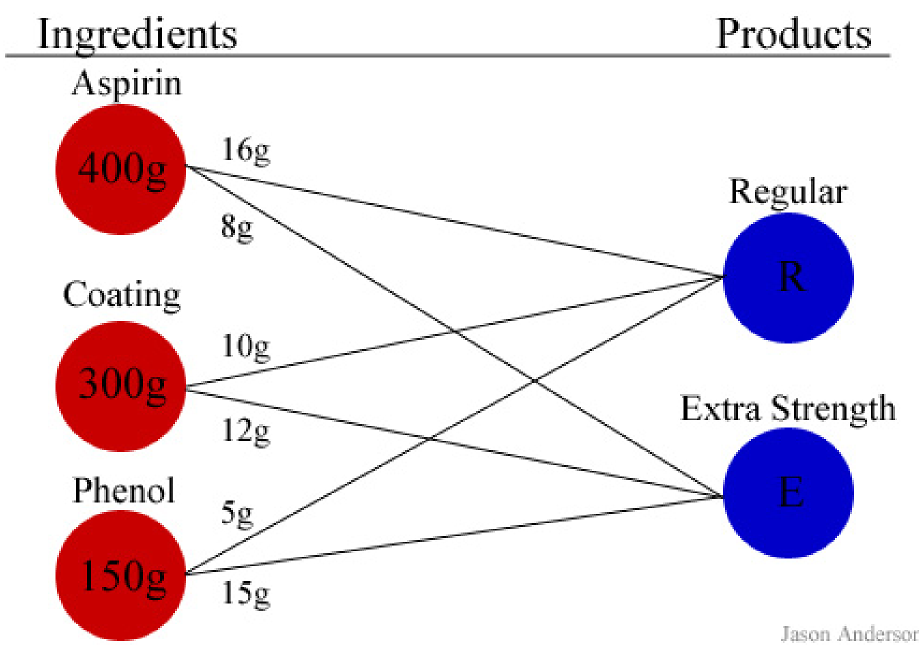
Optimization is one of the most critical problems of today. Everything from maximizing the workflow in a company to minimizing production costs is considered an optimization problem. For instance, if a circuit board company makes 100,000 boards a day and can reduce the time to produce a board by half a second, the company is able to create 70,000 additional boards a day. With this tremendous increase in productivity, businesses are incredibly interested in optimization solutions. Another example of an optimization problem is computer networks. Routers determine the optimal way to send information, which helps to maintain a fast (possibly internet) connection for all its users. As the role of technology escalates, router optimization will become increasingly more important.

One of the most famous optimization problems is the transportation problem. The essential challenge is how to efficiently move resources while minimizing transportation costs. Currently Oracle provides an inventory database program to many large companies which solves these transportation problems every time a location needs supplies. Many great thinkers have worked to find answers to these problems and one popular solution is the Simplex algorithm.

The Simplex algorithm was developed by George Dantzig in 1947 to distribute Air Force resources effectively. He realized his allocation dilemma could be modeled by a linear program but lacked a method to solve it. After months of work, Dantzig eventually combined geometry and the principle of duality to create a solution. Not long after his discovery did the Simplex algorithm become famous and widely used.

Explanation/Example of the Simplex Algorithm

The Simplex algorithm is a simple process which optimizes a function based on a number of constraints. In the following example, the Simplex algorithm will be used to solve an optimization problem for a fictional drug company. Suppose a drug company has a limited supply of ingredients and wishes to maximize their profit while selling two different strengths of a pain reliever. Each medication is made up of three primary ingredients: aspirin, coating, and phenol. Below is a diagram depicting the total amount of each ingredient available and the amounts needed to make one box of each medication which are shown on the edges.



A diagram depicting the supply of ingredients and the necessary amounts to create each product

R represents the number of boxes of regular strength pain reliever and E represents the number of extra strength. The drug company makes a profit of \$2 for every box of regular strength and \$5 on every box of extra strength. Since this company wishes to maximize their profit, the profit equation is required to calculate the profit for every value of R and E. This is the *objective function* which will be maximized for valid values of R and E. In this example, the objective function is as follows.

The Objective Function

Profit(P) = (2 * R) + (5 * E)

This objective function is subject to the following constraints which are needed to ensure that the algorithm produces a real and optimal solution.

Constraints

Word Definition	Mathematical Definition
The amount of Aspirin used cannot exceed the supply	(16 * R) + (8 * E) $\leq$ 400g
The amount of Coating used cannot exceed the supply	(10 * R) + (12 * E) $\leq$ 300g
The amount of Phenol used cannot exceed the supply	(5 * R) + (15 * E) $\leq$ 150g
A negative number of boxes cannot be produced	R, E $\geq$ 0

Now that the objective function and constraints are mathematically defined, the Simplex algorithm can be applied. It starts by converting the constraints into equalities by introducing slack variables. For example,

(16 * R) + (8 * E) $\leq$ 400 becomes S_1 = 400 - (16 * R) - (8 * E),

where the value of S_1 is the difference between 400 and (16 * R) + (8 * E). In other words, S_1 is the amount of slack between the two expressions. Changing the constraints to slack form yields the following equations.

Initial to Slack Form of Constraints

Initial Form	Slack Form
(16 * R) + (8 * E) $\leq$ 400	S_1 = 400 - (16 * R) - (8 * E)
(10 * R) + (12 * E) $\leq$ 300	S_2 = 300 - (10 * R) - (12 * E)
(5 * R) + (15 * E) $\leq$ 150	S_3 = 150 - (5 * R) - (15 * E)

The non-negativity constraints now include S_1- S_3 but can be left in the initial inequality form as the Simplex algorithm assumes no values are negative. The Simplex algorithm will try different solutions until it finds the greatest *basic feasible solution* (BFS). The

BFS is obtained by setting all *non-basic variables* in the maximization function to zero. For example in the initial state,

Initial Simplex Algorithm State

$$
\begin{aligned}
P &= (2 * R) + (5 * E) \\
S_1 &= 400 - (16 * R) - (8 * E) \\
S_2 &= 300 - (10 * R) - (12 * E) \\
S_3 &= 150 - (5 * R) - (15 * E)
\end{aligned}
$$

Basic Variables

Non-Basic Variables

the BFS is zero because P = (2 * 0) + (5 * 0) = 0. One can test if a BFS is a maximum by checking if there are any negative coefficients in the maximization equation. Because both 2 and 5 are positive, zero is not a maximum and an iteration of the Simplex algorithm must occur. To do this, choose a *leaving variable* and an *entering variable*. The leaving variable is the variable with the greatest positive coefficient in the maximization function. If two variables have the same greatest positive coefficient, either one can be chosen. In the example, E is the leaving variable because five is the greatest positive coefficient in the maximization function. The entering variable is the basic variable which limits the leaving variable the most. To determine how much an entering variable limits, set all variables except the leaving variable to zero and solve for the equation. In the example, finding how much S_1 limits E, corresponds to solving the equation

$$0 = 400 - (8 * E) \rightarrow E = 50.$$

The solution states that the possible leaving variable, S_1, limits E to fifty. After computing each limiting value of the basic variables, S_3 is found to limit E the most with a maximum value of 5. Now the leaving variable, E, and the entering, S_3, can be used to rewrite the third equation from

$$S_3 = 150 - (5 * R) - (15 * E) \quad \text{to} \quad E = 10 - (1/3 * R) - (1/15 * S_3).$$

Substituting this new equation into every other equation where an E is present yields the following state.

After First Iteration of the Simplex Algorithm

$$
\begin{aligned}
P &= 50 + (1/3 * R) - (1/3 * S_3) \\
S_1 &= 320 - (40/3 * R) + (8/15 * S_3) \\
S_2 &= 180 - (6 * R) + (4/5 * S_3) \\
E &= 10 - (1/3 * R) - (1/15 * S_3)
\end{aligned}
$$

The cycle can now repeat. First, check if the current BFS, fifty, is optimal. Because there is still a positive coefficient in the maximization function, fifty is not optimal and another leaving and an entering variable must be chosen. R has the greatest positive coefficient so that is the leaving variable. Next, S_2 is found to limit R the most with a minimum value of twenty-four. Then solve for R and substitute the new equation into the others. This produces the following result.

After Second Iteration of the Simplex Algorithm

$$
\begin{aligned}
P &= 58 - (1/40 * S_1) - (24/75 * S_3) \\
R &= 24 - (3/40 * S_1) + (1/25 * S_3) \\
S_2 &= 36 + (9/20 * S_1) - (14/25 * S_3) \\
E &= 2 - (1/40 * S_1) - (2/25 * S_3)
\end{aligned}
$$

Again check to see if there is another positive coefficient, but there are none left. This signifies that the current BFS, fifty-eight, is the optimal profit and the Simplex algorithm terminates. To extract the values of R and E, set all non-basic variables to zero to determine

$$R = 24 - (3/40 * 0) + (1/25 * 0) = 24 \quad \text{and} \quad E = 2 - (1/40 * 0) - (2/25 * 0) = 2.$$

Therefore the drug company must make twenty-four boxes of regular strength pain reliever and two boxes of extra strength to maximize their profits.

Steps of the Simplex Algorithm

After completing an example of the Simplex algorithm, the steps used can be codified.

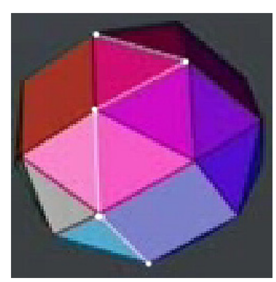
Parameters: Objective Function, Constraint Equations

1) Place the constraints into slack form by introducing slack variables for each inequality constraint.
2) If there are no positive coefficients in the maximization function, terminate and calculate the basic feasible solution which is optimal. Else proceed.
3) Choose a leaving variable and an entering variable using the largest negative coefficient/most limiting strategy.
4) Solve for the leaving variable in terms of the entering variable and substitute the new equation into all others.
5) Return to step 2.

This completes the high level pseudocode for the Simplex algorithm.

Geometric Explanation of the Simplex Algorithm

The Simplex algorithm can be thought of geometrically as well. If the constraints are graphed in $\Re^{\#\text{ of variables}}$, then the resulting region is a polytope containing all feasible solutions. The Simplex algorithm exploits the fact that the optimal solution lies at a vertex (or edge if there are infinitely many solutions) of this polytope. This is how the Simplex algorithm acquired its name. Geometrically, the Simplex algorithm starts at an arbitrary vertex and follows the adjacent edge which offers the greatest improvement of the optimization function. This repeats until the current vertex is greater than all adjacent vertices. At this point, the algorithm has found the optimal



A depiction of a polytope and the simplex algorithm's path.

solution and terminates. Currently researchers are working to improve the Simplex algorithm by converting the edges of the polytope into a two dimensional graph which is easier to navigate. The effort promises adjustments which will improve the Simplex algorithm from worse-case exponential time to worse-case polynomial time. Further reading on their progress can be found in the further reading section.

Efficiency of the Simplex Algorithm

As previously mentioned, the Simplex algorithm executes in worse-case exponential time. While this is true, it is highly unlikely to happen with real-world problems. Its average-case is polynomial time which keeps the Simplex algorithm still competitive today. There are worse-case polynomial time optimization algorithms such as the Ellipsoid algorithm, however these are mostly gimmicks and far worse than the Simplex algorithm in practice. The one rival of the Simplex algorithm is the Interior Point method which does not work with the edges of the polytope, but rather moves inside the feasible region to find the optimum. These two algorithms work well in the average case and most optimization problems use one or the other.

Conclusion

The Simplex algorithm is a method currently used to solve all types of optimization problems such as network flow and the transportation problem. The drug company example showed how the algorithm works with a real-world problem to produce an optimal solution. While the Simplex algorithm might appear as just a set of steps, the computer actually is strategically walking from vertex to vertex on a multidimensional polytope. This has the potential to take exponential time, however this is highly unlikely with the average case executing in polynomial time. The Simplex algorithm not only offers an efficient way to solve optimization problems, it also is a way for companies to save employee time and resources.

References & Further Reading

Descriptions and Applications of the Simplex Algorithm:

Brown, A. R. Optimum Packing and Depletion. London: Macdonald, 1971.

Casti, John L. 5 Golden Rules. New York: MJF Books, 1996.

Greenberg, Michael R. Applied Linear Programming. New York: Academic Press, 1978.

Wu, Nesa. Linear Programming and Extensions. New York: McGraw-Hill, 1981.

Devising a worse-case polynomial Simplex algorithm:

Koltun, Vladlen. "The Arrangment Method for Linear Programming."
<http://theory.stanford.edu/~vladlen/lp.pdf>

Notes: (All examples and charts are of original creation)