

Linear Programming Notes I: Introduction and Problem Formulation

1 Introduction to Operations Research

Economics 172 is a two quarter sequence in Operations Research. Management Science majors are required to take the course. I do not know what Management Science is. Most of you picked the major. I assume that you either know what it is or do not care. You may not know what Operations Research is. I am going to tell you, but it will leave you disappointed.

Operations Research is research into operations. The field began during the Second World War. The military needed to solve a lot of different kinds of resource allocation problems. A prototypical problem was a form of the transportation problem that we'll study later in the course. In this problem the military had supplies available in several different locations (ammunition factories), it had several different locations that needed the supplies (battle fronts), it knew how much it cost to ship supplies from any factory to any front. It knew how much was produced at each factory and how much was needed in each front. It wanted to figure out how to minimize the cost of shipping the supplies to the various locations while meeting two types of resource availability constraints (that you do not send more ammunition from a factory than is available and that you send as much as necessary to each battle front). Many other resource allocation problems arose in the planning of military operations. Operations Research was a field of study that tried to come up with practical solutions to these problems.

People need to allocate resources even in peacetime. Economics is a discipline devoted to the study of methods to allocate scarce resources. It is natural to study the methods of Operations Research in an economics class. Some of the methods developed have direct relevance to decision making. Courses in Operations Research are therefore traditional parts of undergraduate and graduate business programs.

Operations Research as a discipline involves several different things. First, there is the identification of real world situations that lend themselves to formulation as mathematical optimization problems. Second, there is a process of translating these problems into mathematical language. Third, there is the development of mathematics that explains the general structure of the mathematical problems that arise in the second stage. Fourth, there is the development of methods for solving these problems.

The Operations Research sequence introduces some of the basic mathematical techniques for describing and solving problems (steps 3 and 4 above). It provides practice in the formulation of problems (steps 1 and 2 above).

A mathematical programming problem is an optimization problem subject to constraints. In the general problem, you are given a function f and a set S . You are asked to find a solution to the problem:

$$\max f(x) \text{ subject to } x \in S. \tag{1}$$

A linear programming problem is a mathematical programming problem in which the function f is linear and the set S is described using linear inequalities or equations. It turns out that lots of interesting problems can be described as linear programming problems. It turns out that there is an efficient algorithm that solves linear programming problems efficiently and exactly. It turns out that the solutions to linear programming problems provide interesting economic information. Economics 172A concentrates on these problems.

Economics 172B primarily studies non-linear programming. That is, problems in which the function f is non-linear and the set S is described using non-linear inequalities or equations. This theory uses calculus techniques.

In the Economics 172 sequence, the word “programming” has nothing to do with computer programming (although it is true that there are computer programs that can be used to solve mathematical programming problems). This terminology is confusing, but it is standard.

2 Introduction to Linear Programming

Economics 172A studies linear programming. So you need to know what a linear function is. The function f of n variables $x = (x_1, \dots, x_n)$ is linear if there are constants a_1, \dots, a_n such that

$$f(x) = a_1x_1 + \dots + a_nx_n. \tag{2}$$

This expression is also written:

$$f(x) = \sum_{i=1}^n a_i x_i = a \cdot x, \tag{3}$$

where $a = (a_1, \dots, a_n)$.

Two properties characterize linear functions: additivity and constant returns to scale. Additivity means that $f(x + y) = f(x) + f(y)$. Constant returns to scale means that $f(cx) = cf(x)$ for any constant c . These properties make sense sometimes. Other times they are silly. Suppose $f(z)$ is how much it costs you to buy z . Remember, z has n components, so you can think of z as a list. The i th entry in the list, z_i , tells you the amount of good i that you are buying. Additivity says that if you first buy x and then buy y , it costs the same amount if you bought $x + y$ at one time. Constant returns to scale says that buying half as much costs half as much and buying twice as much costs twice as much. Provided that there are no specials (“buy two, get one free”), most of what you buy at the grocery store satisfies these properties. It certainly describes how you pay for gasoline at the pump. On the other hand, linearity does not hold in

milk prices: a gallon container costs less than two half gallon containers. Still, one of the most basic linear functions that we deal with is the one that assigns value to lists of goods. If p_i is the price per unit of good i , then the linear function $f(x) = p \cdot x$ gives the cost of buying the ‘bundle’ x consisting of x_1 units of good 1, x_2 units of good 2, and so on.

Linearity is usually not a very good assumption for utility functions. If $f(z)$ represents the utility (loosely, happiness) you get from having z , then both additivity and constant returns to scale are likely to fail. For example, if x represents having a CD player (and nothing else), while y represents having a CD of your favorite music, then presumably $f(x + y) > f(x) + f(y)$ and also $2f(x+y) > f(2(x+y))$. The first inequality says that having both the CD player and the CD is better than the sum of utilities available from having exactly one. (You could argue that having only the CD player or only the CD is worthless.) The second inequality says that having twice the utility from both CD player and CD is better than having the utility of two CD players and two copies of the CD. (You could argue that the second copy of the disk and the second player is useless.) In economics it is typical to assume diminishing marginal utility. In our context that is just a fancy way of saying that doubling what you have does less than double your utility (the second \$100,000,000 does not generate as much additional utility as the first \$100,000,000.)

The linearity assumption does not apply to production processes that have fixed costs (the first unit costs much more than subsequent units) or capacity constraints. It does not apply to situations in which ‘units’ are not perfectly divisible (that is, the components of x theoretically measure continuous quantities not numbers of people). Divisibility is a standard simplifying assumption.

The point is that linearity is an assumption. You should reflect on whether it is a reasonable assumption in the applications that arise during the quarter.

Now return to (1). It is time to get a better understanding of what a mathematical programming problem is. The next few paragraphs will contain several really important definitions. You’ll hear them over and over again.

S is your **constraint set** or **feasible set**. Maybe it is the different combinations of things that you can afford to buy. Maybe it is the different combination of things that you have the available raw materials to manufacture. In any event, it is what keeps you from doing whatever you want. The function f is your **objective function**. It is what you are trying to optimize (optimize means either minimize or maximize). It is possible that the set S is empty. If this is true, then your problem is **infeasible**. You can’t solve it. This is a perfectly reasonable mathematical possibility. Economically, it means that your constraints are inconsistent. You will see examples soon enough. If S is not empty, then the problem is **feasible**.

What does it mean to solve a mathematical programming problem? A **solution** to (1) is a special value x^* that has two properties:

1. **Feasibility.** $x^* \in S$.
2. **Optimality.** If $x \in S$, then $f(x^*) \geq f(x)$.

That is, a solution must satisfy the constraints of the problem and, among all things that satisfy the constraints, yield the highest objective function value. If x^* is a solution to (1), then $f(x^*)$ is called the **optimal value** (or sometimes just value) of the problem. Not all problems have solutions (for example, infeasible problems have no solution). Problems may have more than one solution. (There may be two different ways to solve the problem.) If a problem has a solution, then the value must be unique (otherwise the lower number can't be the value).

Our problems will turn out to fall into one of three categories. They will either be infeasible or they will have a solution or they will be **unbounded**. A problem is unbounded if it is possible to make the objective function arbitrarily large. In symbols, this means that for any M , there exists an $x \in S$ such that $f(x) > M$. In words, a problem is unbounded if for any target value of the objective function (M) it is possible to find a way to make f even bigger than M using a feasible point x .

These definitions apply to any problem like (1). The course restricts attention to linear programming problems. A linear programming problem is a mathematical programming problem in which f is linear and the set S is described by linear inequalities or equations. There is a standard form for writing linear programming problems (LPs).

$$\max c \cdot x \text{ subject to } Ax \leq b \text{ and } x \geq 0. \quad (4)$$

In this formulation, $c = (c_1, \dots, c_n)$, $b = (b_1, \dots, b_m)$, 0 denotes an n dimensional list of zeros, and A is a matrix with m rows and n columns (an $m \times n$ matrix); the entry of A in row i and column j is a_{ij} . In this basic problem, the given data are c (the coefficients of the variables in the objective function), b (the resources constraints), and A (the technology). In order to formulate the problem, you must know these things. The problem that I have described has n variables (the components of x) and $m + n$ constraints. The first m constraints come from the set of inequalities summarized by $Ax \leq b$. The remaining n constraints are the **non-negativity** constraints on the components of x . The notation $Ax \leq b$ is short hand for the system of m inequalities. A representative inequality (the i th inequality) takes the form

$$\sum_{j=1}^n a_{ij}x_j \leq b_i.$$

The objective function and the constraints in the problem are all linear. In principle, the objective in a linear programming problem can be to maximize or to minimize; the constraints can be written in the form of equations or inequalities of either direction, and inequality constraints need be present for some (or none) of the variables. It turns out that any linear programming problem can be written in the standard form above. I'll say more about that later. At this point, note only that (4) describes the set of problems we will study.

Now I can comment on the contents of the course outline. The first topic is problem formulation. This is the process of taking a situation described in words and translating it into a mathematical problem in the form (4). This process probably represents the most likely application you might make of the techniques of the class in the “real world.” The classroom is not the real world. You will see rather contrived examples. During the first week of the class, I will describe possible linear programming problem and formulate a couple slowly. There will be formulations throughout the class. My experience is that students have trouble formulating problems. You might find that the first topic is the most challenging part of the course.

When there are only two variables, it is possible to solve linear programming problems graphically. The second topic shows you how to do this. Graphical solution is easy and illustrates most of the basic ideas about solutions of linear programming problems. The problem is that most problems involve more than two variables and graphical methods do not apply.

Algorithms exist that can solve any linear programming problem. These algorithms are widely used in industry. The oldest and still most widely used algorithm is the simplex algorithm. Versions of the algorithms are available as part of common spreadsheet programs. Since your computer already knows the algorithm and can do computations more easily than you can, it makes no sense to teach you the entire procedure. Still, the essentials of the simplex algorithm are straightforward and instructive. Knowing how the algorithm works is useful on its own and also helps you interpret solutions provided by computers. I will spend some time teaching you a bit about the algorithm in Topic 3.

The fourth topic is the heart of the course. It turns out that when you solve a linear programming problem you automatically solve another linear programming problem (called the **dual** of the original problem). The theory of duality is beautiful and interesting (to the mathematically inclined). It also provides truly important economic information about solutions to linear programming problems. Sensitivity Analysis refers to the study of what happens to the solution to a linear programming problem when one changes the problem (by varying the objective function or the resource constraints). There is a lot to say here. We will say some things theoretically. Other things we will illustrate using solutions to problems obtained by the computer.

Game Theory is a big topic (there is an entire undergraduate course devoted to it). It is a mathematical theory of strategic interaction. Zero-sum games are a special class of game that includes most of the things called games by normal people (chess, poker, tic-tac-toe) and generally situations where players have completely opposed interests. It turns out that there is an intimate relationship between zero-sum games and linear programming. I will tell you about it. (I should warn you that game theory rarely provides practical advice on how to play a game.)

The final topic covers a special class of linear programming problem. This problem has special structure. It provides a useful way to introduce integer linear programming (that is, linear programming problems with the additional restriction that all variables must be whole numbers).

3 Introduction to Problem Formulations

Problem formulation is the most important part of a Operations Research course for a Management Science major. When you are the boss, you'll hire a geeky engineer to do some basic math and write software. You'll earn big bucks by identifying the important problems and translating them from a verbal identification to a mathematical form. The engineer will then solve the mathematical problem. You will interpret the solution and put it into practice. It is important for you to know enough about the basic mathematics for you to be able to frame questions that the engineer might be able to answer and to be able to judge whether the answers provided are sensible. Formulation, however, is key.

Unfortunately, I have little useful to say on the topic. In order to formulate problems, you need to be able to understand symbols, you need common sense, and you need practice. I am not aware of a mechanical series of steps you can take in order to complete a formulation.

Now I will go through a particular (and standard) linear programming problem and formulate it.

The problem is called the Diet Problem. Here is the story.

You run a small institution (prison, junior high school, third world country). People work in your institution and you must feed them. Your job is to meet their basic requirements for nutrients at minimum cost. In order to do this, you need to know several things. You must know what foods are available and the cost of each food. You must also know which nutrients are necessary and the nutritional content of each of the foods. With this information you can figure out how much any combination of food costs and you can figure out the nutritional content of any combination of food. You can decide which combinations of food are sufficient to meet the nutritional requirements and then pick the cheapest combination that meets the nutritional requirement. (Perhaps the story makes more sense if you imagine that your job is to feed the animals on your farm.)

So that is the basic verbal story. It has a surface plausibility. That is, you can imagine someone wanting to find cheap ways to feed people. It is a bit bizarre because it contains no mention of what the foods taste like. The story does not place restrictions on food (for example, not too much salt, sugar, or fat; or no meat), although these restrictions can be included without much trouble.

In order to formulate the problem as a linear programming problem, we need notation to describe the given data. This information typically is given to you in the statement of a formulation problem. Assume that there are n different kinds of food. The price per unit of the j th food is p_j . Assume that there are m different nutrients. The nutritional requirement of nutrient i is c_i . Finally, let a_{ij} be the amount of the i th nutrient in one unit of the j th food.

Let me repeat this information less abstractly. The n different foods could be things like lettuce, hamburger, potatoes, oranges, pizza, and so on. When I talk about the j th food, I mean one of these (maybe I list all available foods in alphabetical order and number them 1 through n). p_j is the unit price of food j . So, p_1 might be the price of a head of lettuce; p_2 might be the price of a pound of hamburger; and so on. The m different nutrients could be things like vitamin

C, iron, niacin, and so on. c_i is the daily minimum requirement of nutrient i . The units of these things are weird (I think that Vitamin E is measured in “International Units,” other nutrients are measured in grams). This does not matter, as long as you can figure out how much of each nutrient you can find in each food. That is where the a_{ij} comes in. Suppose that there are .5 grams of niacin in a head of lettuce. If lettuce is food 1 and niacin is nutrient 5, then this means that $a_{51} = .5$.

Now we have a description of the problem in words and a description of the basic data of the problem. Notice that you (as the manager of the institution) could find out the data. You look up food prices at the grocery store. You consult a nutritionist to figure out the entries in the matrix A . You check government standards to figure out the nutritional requirements. Your problem is to figure out what to buy. In order to formulate this as a mathematical problem, you need to invent a name for what you are looking for.

Step 1: Identify Variables.

You are looking for amounts of food. Therefore, your variables are quantities of each of the n foods. These are unknowns and need names. Let x_j be the number of units of food j purchased. You want to find $x = (x_1, \dots, x_n)$.

Now you need to use this notation to figure out the objective function and the constraints of the problem.

Step 2: Write Down the Objective Function.

The objective is to minimize the cost of the food that you buy. If you buy x how much will it cost? Break it down. Buying x means that you buy x_1 units of the first food, x_2 units of the second food, and so on. How much do you spend on the first food? It costs p_1 per unit. Therefore you spend p_1x_1 on the first food. How much do you spend in total? You just add up what you spend on each of the foods. This quantity is:

$$p_1x_1 + \dots + p_jx_j + \dots + p_nx_n = \sum_{j=1}^n p_jx_j = p \cdot x. \tag{5}$$

(5) is the objective function. That is, you want to find x to $\min p \cdot x$.

I invoked linearity assumptions to write the objective function. I assumed constant returns to scale when I asserted that if p_j is the price of one unit, then p_jx_j is the price of x_j units. This is an assumption. Maybe it is impossible to buy goods in tiny quantities. Maybe it is possible to get large purchases at lower costs per unit. If so, then the linearity assumption is not appropriate (although it may be a reasonable approximation). I also invoked additivity when I claimed that the cost of the entire purchase is just the sum of the amount spent on each food. This assumption is reasonable, but you can imagine settings where people get discounts for buying large quantities.

If the problem was simply to minimize costs, then the answer would be easy. Buy no food. After all, that costs you nothing. The problem with that is that the people in your institution will die. You want to minimize expenditures, but

only after you have met the nutritional requirements. You need a way to decide whether the food you buy actually satisfies nutritional requirements.

Step 3: Write Down the Constraints.

The constraints are that you satisfy nutritional requirements. You need to buy enough food to supply all nutrients in (at least) the recommended amounts. How much nutrient i do you need? c_i . How much of this nutrient is supplied when you have x ? Again, take it one food at a time. You have x_1 units of the first food. This means that you obtain $a_{i1}x_1$ units of the i th nutrient coming from the first food. (The units of x_1 might be pounds (of hamburger); the units of a_{ij} might be grams (of iron) per pound (of hamburger)). Hence the product gives you a quantity of grams (of iron). How much nutrient i do you get from x ? Add up the amount of nutrient i you get from each food.

$$a_{i1}x_1 + \cdots + a_{ij}x_j + \cdots + a_{in}x_n = \sum_{j=1}^n a_{ij}x_j. \quad (6)$$

Therefore, to supply enough of nutrient i you must satisfy the constraint that (6) be greater than or equal to c_i . The constraint:

$$a_{i1}x_1 + \cdots + a_{ij}x_j + \cdots + a_{in}x_n = \sum_{j=1}^n a_{ij}x_j \geq c_i \quad (7)$$

(7) describes the i th nutritional constraint. The entire problem imposes such a constraint for each nutrient. That is, (7) must hold for $i = 1, \dots, m$.

I can lump the constraints together using matrix notation: the m constraints described by (7) are: $Ax \geq c$. Once again notice that I made linearity assumptions to formulate the constraints. If the Vitamin C you get from oranges detracts from the Vitamin C you get from kiwis, then additivity fails. If your body cannot process more than 3 potatoes in a day (causing them to pass from your system without supplying nutrients), then the constant returns to scale assumption fails.

It is also natural to add the restriction that you cannot buy negative quantities of food. In symbols: $x \geq 0$.

Step 4: Write Down the Entire Problem.

The work is over. Now just summarize it. The problem is to find x to solve:

$$\min p \cdot x \text{ subject to } Ax \geq c \text{ and } x \geq 0.$$

In practice, you will be given values for the parameters of the problem (A , p , and c) and then would go ahead and try to find a numerical solution. In fact, you can amuse yourself by going to:

<http://www-fp.mcs.anl.gov/otc/Guide/CaseStudies/diet/>

and finding out the costs of sample diets (I am not sure where the prices or requirements came from). This program allows you to select the foods that you

are willing to eat. I selected about thirty foods and was told that I should limit my diet to carrots, peanut butter, potatoes, and skim milk. With these, I could meet my nutritional requirements for 99 cents per day. This diet was heavy on the peanut butter. I decided that maybe I didn't want to survive by spreading it on carrots so I ruled out peanut butter. When I did, my optimal diet cost \$4.32 and involved nine different foods. I am grateful that my enormous university salary provides me the luxury of spending even more than this on food every day.

During the formulation problem, it is useful to think about what the solution of the problem might look like. Would you expect the problem to have a solution? In theory two things can go wrong. Maybe the problem is infeasible. That would mean that it is impossible to find any foods that would satisfy the nutritional requirements. This could happen if the government decided that everyone needed to consume positive quantities of Vitamin X , but there was no food that contained Vitamin X . (My son eats chicken soup, pasta, corn bread, chicken nuggets, and chocolate desserts. It is possible that this would not be enough to satisfy nutritional requirements without vitamin supplements.) On the other hand, if you could find every nutrient in some food, then (by buying enough) you could guarantee that you satisfy all of the requirements. That is, it is sensible to assume that the problem is feasible. Could it be unbounded? For a minimization problem, this would mean that you could make the cost of the optimal diet arbitrarily small - not close to zero, but smaller than any number. It does not make sense that the diet would cost less than, say, $-\$100$. (I would interpret this as meaning that the store paid you \$100 to take the food.) Indeed, if you assume that prices are all non-negative (true everywhere but in Mom's kitchen), then any bundle of food you purchase will cost a non-negative amount, so the cost of diets are bounded below. In summary, it is sensible to assume that the diet problem has a solution.

Before leaving the diet problem, I want to describe another problem that is based on precisely the same data as the diet problem. Here is the story. You still run the institution. Someone approaches you and says: "Why bother with food? All you care about is that your animals get nutrients. I sell pills (one kind of pill for each nutrient). I have set my prices so that you can get nutrients more cheaply from me than through food. I will sell you exactly the nutrients you need and you will be better off." You think about this and decide that it sounds reasonable. The new problem is to figure out how the pill seller should behave. Her problem is to set prices of pills that maximize the amount she can get selling you the necessary nutrients subject to the constraint that the pills provide nutrients more cheaply than food.

Step 1: Variables.

The pill seller wants to find prices for each nutrient pill. That is, she is looking for $y = (y_1, \dots, y_i, \dots, y_m)$, where y_i is the price charged for a pill that supplies one unit of nutrient i .

Step 2: Objective.

The pill seller wants to maximize her profit. She sells c . If she can charge the prices y , then she earns $c \cdot y$.

Step 3: Constraints.

What does it mean for the pills to be cheaper than food? Consider the first food. You don't care what it looks like or what it tastes like. You only care what nutrients it provides. Suppose you try to replace food one with pills. What kinds of pill would you need? Food one supplies (in theory) amounts of all m nutrients. If you wanted to replace the nutrient i found in one unit of food one with nutrient i pills, you would need a_{i1} pills. This means that replacing the nutrient i in food one with pills would cost $a_{i1}y_i$. The total amount you would need to replace the nutrients in food one with pills is therefore $\sum_{i=1}^m a_{i1}y_i$.

In order for the (nutrients in the) pills to be cheaper than (the nutrients in) food one, it must be that

$$\sum_{i=1}^m a_{i1}y_i \leq p_1.$$

I want to impose this kind of constraint for each food. That is, each food is at least as expensive as the cost of its nutrients. This leads to, for each $j = 1, \dots, n$,

$$\sum_{i=1}^m a_{ij}y_i \leq p_j.$$

In concise notation, this becomes $A^t y \leq p$ (where A^t is the transpose of A : the matrix you get when you interchange rows and columns). Throughout this course I will write this expression $yA \leq p$. Those comfortable with linear algebra will know that this notation confuses row vectors with column vectors, but it is convenient and should lead to no confusion.

Also, I add a non-negativity constraint (that states that the pill seller does not give people money to take her pills): $y \geq 0$.

Step 4: Conclusion.

Put the constraints together and we have the pill seller's problem:

Find $y = (y_1, \dots, y_m)$ to solve:

$$\max c \cdot y \text{ subject to } yA \leq p \text{ and } y \geq 0.$$

On one hand, the pill seller's problem is just a contrived way to practice problem formulation. It turns out, however, that it illustrates an important idea that will appear later in the course. At this stage, I want to point out several things.

Both the diet problem and the pill seller's problem use the same basic data (A , c , and p). I constructed the pill seller's problem so that there will be a relationship between its value and the value of the diet problem. Specifically, the optimal value of the diet problem (the minimum cost) will be greater than or equal to the optimal value of the pill problem (the maximum earnings of the seller). Why? The constraints in the pill problem guarantee that pills are cheaper than food. What the pill seller earns is what you would need to pay to buy all of the necessary nutrients. Since these nutrients cost less when

purchased in pill form (by the construction of the prices) than when purchased in food form, it must be that the cost of the pills is cheaper than the cost of the food. You can prove this. Suppose that x is feasible for the diet problem and y is feasible for the pill problem. That means that x satisfies $Ax \geq c$ and $x \geq 0$ and y satisfies $yA \leq p$ and $y \geq 0$. It follows that

$$yAx \geq y \cdot c$$

(This follows because $Ax \geq c$ and $y \geq 0$. All you are doing is multiplying m separate inequalities by a non-negative numbers and then adding them up. Note that yAx and $y \cdot c$ are both numbers.) and also

$$yAx \leq p \cdot x.$$

Combining these two inequalities yields $y \cdot c \leq yAx \leq p \cdot x$ and, in particular, $y \cdot c \leq p \cdot x$. This inequality says in symbols what I said in words earlier: The cost of the pills (priced so that pills are cheaper than food) is no greater than the cost of a feasible diet.

The general property of linear programming problems that you'll learn is that when you actually solve these problems, the values are equal. That is, when you find the minimum cost diet, it will cost exactly the same amount as you would pay a profit-maximizing pill seller for the pills. This relationship (and consequences of it) allow us to interpret the prices obtained when you solve the pill seller's problems as interesting economic quantities. It turns out that they actually provide the economic value of nutrients as seen by you (in your role as institutional menu planner). These prices give simple ways to answer questions of the form: how much extra would it cost to satisfy the diet problem if the nutritional requirement of the first nutrient went up by one unit.