

## PREFACE

The subject called “optimization” is concerned with maximization and minimization. More precisely, the purpose of optimization is to find the values of *variables* that either maximize or minimize the value of a given *function*. In many cases, especially among those studied in this book, the variables are required to satisfy side conditions such as equations or inequalities in which case the term *constrained optimization* is appropriately used. When no such side conditions are imposed, the optimization problem is said to be *unconstrained*.

In an effort to convey the importance of their subject, some contemporary writers on optimization have joyfully quoted a line published in 1744 by Leonhard Euler [60], arguably the most prolific mathematician of all time. In a work on elastic curves, Euler proclaimed: *Nihil omnino in mundo contingit, in quo non maximi minimive ratio quæpiam eluceat*. That is: *Nothing at all takes place in the universe in which some rule of maximum or minimum does not appear*. In addition to being a great mathematician, Euler was a profoundly religious man. The quotation reproduced here should be construed as a pious view in line with what is called Leibnizian optimism. One sees this by considering the entire sentence from which Euler’s statement is excerpted and the sentence that comes after it. As translated by Oldfather, Ellis, and Brown [153, pp. 76–77], Euler wrote:

*For since the fabric of the universe is most perfect, and is the work of a most wise Creator, nothing at all takes place in the universe in which some rule of maximum or minimum does not appear. Wherefore there is absolutely no doubt that every effect in the universe can be explained as satisfactorily from final causes, by the aid of maxima and minima, as it can from the effective causes themselves.*

This outlook brings to mind the character Dr. Pangloss in Voltaire's *Candide* who insisted that this is "the best of all possible worlds."

Having strong connections with human activity, both practical and intellectual, optimization has been studied since classical antiquity. Yet because of these connections, optimization is as modern as today. In their handbook on optimization [148], the editors, Nemhauser, Rinnooy Kan, and Todd, declare that

No other class of techniques is so central to operations research and management science, both as a solution tool and as a modeling device. To translate a problem into a mathematical optimization model is characteristic of the discipline, located as it is at the crossroads of economics, mathematics and engineering.

Around the middle of the 20th century, some kinds of optimization came to be known as "mathematical programming." *Linear programming* came first; soon thereafter came *nonlinear programming*, *dynamic programming*, and several other types of programming. Together, these subjects and others related to them were subsumed under the title *mathematical programming*. The term "programming" was originally inspired by work on practical planning problems. Programming in this sense simply means the process of finding the levels and timing of activities. This work came to entail mathematical modeling, the development and implementation of solution techniques, and eventually the scientific study of model properties and algorithms for their solution.

The terms "mathematical programming" and "computer programming" came into existence at about the same time.<sup>1</sup> Whereas the latter has become a household word, the same cannot be said of the former. When "mathematical programming" turns up in everyday language, it is typically either not understood at all or confused with "computer programming." Adding to the confusion is the fact that computers and computer programming are used in the implementation of solution methods for mathematical programming problems. The contemporary use of "optimization" as a synonym for "mathematical programming" reflects the wide-spread desire among profes-

---

<sup>1</sup>For a book on classical and "modern" optimization problems, see Nahin [146]. The author invokes the Euler quotation given above, but it must be said that this publication gives a little more of the context in which it was stated as well as a reference to Oldfather et al. [153], an annotated English translation of Euler's tract.

sionals in the field to avoid the confusion between these different senses of the word “programming.” It also helps to anchor the subject historically.

In 1971, the scientific journal *Mathematical Programming* came into existence and along with it, the Mathematical Programming Society (MPS) which undertook the sponsorship of a series of triennial “International Symposia on Mathematical Programming” that had actually begun in 1949. Today there are many journals dealing solely with mathematical programming (optimization), and there are many more that cover it along with other subjects. Likewise there are dozens of conferences every year that feature this topic. In 2010 the MPS membership voted to change the name of their organization to *Mathematical Optimization Society* (MOS). For the sake of archival continuity, however, the name of the society’s journal was not changed.

Faced with this linguistic transition from “mathematical programming” to “optimization,” we use both terms. But when it comes to talking about linear programming, for instance, we feel there is too much current literature and lore to abandon the name altogether in favor of “linear optimization.” Furthermore, it is traditional—even natural—to refer to a linear programming problem as a “linear program.” (Analogous usage is applied to nonlinear programming problems.) We are unaware of any suitable replacement for this terminology. Accordingly, we use it frequently, without reservation.

The remarkable growth of optimization in the last half century is attributable to many factors, not least of which is the effort exerted around the world in the name of national defense. This brought forth a great upsurge of activity in problem solving which, in turn, called for problem formulation, analysis, and computational methods. These are aspects of optimization, just as they are for many science and engineering disciplines.

The recognition of a “problem” is an all-important element of practical work, especially in management science. This can take many forms. For example, one is the realization that something needs to be created or put in place in order to get some job done. Today we are bombarded by advertizing for products described as “solutions,” for instance, “network solutions.” Implicitly, these are responses to problems. Oftentimes, a problem of getting something done requires *choices* whose consequences must satisfy given *conditions*. There may even be a question of whether the given conditions can be satisfied by the allowable choices. These are called *feasibility issues*. A set of choices that result in the conditions being satisfied is called a *feasible*

*solution*, or in everyday language, a *solution*. The problem may be to find a feasible solution that in some sense is “better” than all others. Depending on how goodness is judged, it may be necessary to find a feasible solution for which the corresponding measure is cheapest, most profitable, closest, most (or least) spacious, most (or least) numerous, etc. These goals suggest maximization or minimization of an appropriate measure of goodness based on feasible solutions. This, of course, means *optimization*, and a feasible solution that maximizes or minimizes the measure of goodness in question is called an *optimal solution*.

## Modeling

After the problem is identified, there arises the question of how to represent it.<sup>2</sup> Some representations are graphic, others are verbal. The representations of interest in optimization are mathematical, most often algebraic. The choices mentioned above are represented by *variables* (having suitable units), the conditions imposed on the variables (either to restrict the values that are allowable or to describe what results the choices must achieve) are represented by *relations*, *equations*, or *inequalities* involving *mathematical functions*. These are called *constraints*. When a measure of goodness is present, it is expressed by a mathematical function called the *objective function* or simply the *objective*. This mathematical representation of the (optimization) problem is called a *mathematical model*.

Linear and nonlinear optimization models are so named according to the nature of functions used as objective and constraints. In linear optimization, *all* the functions used in the mathematical model must be linear (or strictly speaking, affine), whereas in nonlinear optimization, *at least one* of them must be nonlinear. An optimization model with a linear objective functions would be classified as nonlinear as long as it had a nonlinear function among its constraints. Some nonlinear optimization models have only linear constraints, but then of course, their objective functions are nonlinear.

For example,  $c(x, y) = 3x + 4y$  is a linear function, and  $c(x, y) \leq 5$ , or equivalently  $3x + 4y \leq 5$ , is a linear inequality constraint. On the other hand the function  $f(x, y) = x^2 + y^2$  is a nonlinear function, and  $f(x, y) \leq 2$ , or equivalently  $x^2 + y^2 \leq 2$ , is a nonlinear inequality constraint.

---

<sup>2</sup>For two books on modeling, see [4] and [191].

In general, mathematical models of problems have several advantages over mere verbal representations. Among these are *precision*, *clarity*, and *brevity*. In effect, models get down to the essence of a real situation and facilitate a precise way of discussing the problem. Some other advantages of mathematical models are their *predictive powers* and *economy*. Using mathematical models to represent real-world situations, it is possible to perform experiments (simulations) that might otherwise be expensive and time-consuming to implement in a real physical sense.

Of course, the value of a mathematical model has much to do with its accuracy. It is customary to distinguish between the *form* of a model and its *data*. In building a mathematical model, one chooses mathematical functions by which to represent some aspect of reality. In an optimization problem, there is an objective function, and there may be constraints. The representation of reality is usually achieved only by making some *approximations* or *assumptions* that affect the form of the model. This leaves room for questioning its adequacy or appropriateness. Mathematical models typically require data, also known as *parameter values*. Here the question of *accuracy* of the parameters is important as it can affect the results (the solution). If optimization is to yield useful information, the matter of acquiring sufficient, accurate data cannot be overlooked.

## Computation (algorithms and software)

After the formulation of an optimization model and the collection of data, there comes a desire to “get the answer,” that is, to find an optimal solution. As a rule, this cannot be done by inspection or by trial and error. It requires specialized computational methods called *algorithms*. You should have encountered maximization and minimization problems in the study of calculus. These are usually not difficult to solve: you somehow find a root of  $f'(x) = 0$  or, in the multivariate case,  $\nabla f(x) = 0$ . Yet even this can require some work. In mathematical programming, the task is ordinarily more challenging due to the usually *much greater* number of variables and the presence of constraints.

Typically, optimization algorithms are constructed so as to generate a sequence of “trial solutions” (called *iterates*) that *converge* to an “optimal solution.” In some instances (problem types), an algorithm may be guaranteed to terminate after a finite number of iterations, either with an optimal

solution or with evidence that no optimal solution exists. This is true of the Simplex Algorithm for (nondegenerate) linear programming problem.

Some of the terms used in the previous paragraph, and later in this book, warrant further discussion. In particular, when speaking of a “finite algorithm” one means an algorithm that generates a finite sequence of iterates terminating in a resolution of the problem (usually a solution or evidence that there is no solution). By contrast, when speaking of a “convergent algorithm” one means an algorithm that produces a convergent sequence of iterates. In this case it is often said that the algorithm converges (even though it is actually the sequence of iterates that converges).

A major factor in the development of optimization (and many other fields) has been the computer. To put this briefly, the steady growth of computing power since the mid-twentieth century has strongly influenced the kinds of models that can be solved. New advances in computer hardware and software have made it possible to consider new problems and methods for solving them. These, in turn, contribute to the motivation for building even more powerful computers and the relevant mathematical apparatus.

It should go without saying that algorithms are normally implemented in computer software. The best of these are eventually distributed internationally for research purposes and in many cases are made into commercially available products for use on real-world problems.

## Analysis (theory)

Optimization cannot proceed without a clear and tractable definition of what is meant by an optimal solution to a problem. In everyday language (especially in advertising) we encounter the word *optimal* (and many related words such as *optimize*, *minimize*, *maximize*, *minimum*, *maximum*, *minimal*, *maximal*) used rather loosely. In management science, when we speak of optimization, we mean something stronger than “improving the status quo” or “making things better.” The definition needs to be precise, and it needs to be verifiable. For example, in a minimization problem, if the objective function is  $f(x)$  and  $x \in S$ , where  $S$  is the set of all allowable (feasible) values for  $x$ , then

$$x^* \in S \quad \text{and} \quad f(x^*) \leq f(x) \quad \text{for all } x \in S$$

is a clear definition of what we mean by saying  $x^*$  is a (global) minimizer of  $f$  on  $S$ . But in almost all instances, this definition by itself is not useful because it cannot always be checked (especially when the set  $S$  has a very large number of elements).

It is the job of analysis to define what an optimal solution is and to develop tractable tests for deciding when a proposed “solution” is “optimal.” This is done by first exhibiting nontrivial properties that optimal solutions must have. As a prototype of this idea, we can point to the familiar equation  $f'(x) = 0$  as a condition that must be satisfied by a local (relative) minimizer<sup>3</sup>  $x = x^*$  of a differentiable function on an open interval of the real line. We call this a *necessary condition* of local minimality. But, as we know, a value  $\bar{x}$  that satisfies  $f'(x) = 0$  is not guaranteed to be a local minimizer of  $f$ . It could also be a local maximizer or a point of inflection. We need a stronger *sufficient condition* to help us distinguish between first-order stationary points, that is, solutions of  $f'(x) = 0$ .

We use mathematical analysis (theory) to develop conditions such as those described above. In doing so, we need to take account of the properties of the functions and sets involved in specifying the optimization model. We also need to take account of what is computable so that the conditions we propose are useful. But optimization theory is concerned with much more than just characterization of optimal solutions. It studies *properties* of the types of functions and sets encountered in optimization models. Ordinarily, optimization problems are classified according to such properties. Examples are linear programming, nonlinear programming, quadratic programming, convex programming, . . . . Optimization theory also studies questions pertaining to the *existence* and *uniqueness* (or lack of it) of solutions to classes of optimization problems.

The analysis of the *behavior* of algorithms (both in practice and in the abstract) is a big part of optimization theory. Work of this kind contributes to the creation of efficient algorithms for solving optimization models.

## Synergy

Modeling, computing, and theory influence each other in many ways. In practice, deeper understanding of real-world problems comes from the com-

---

<sup>3</sup>The same would be true for a local maximizer.

combination of mathematical modeling, computation, and analysis. It very often happens that the results of computation and analysis lead to the redefinition of a model or even an entirely new formulation of the problem. Sometimes the creation of a new formulation spurs the development of algorithms for its solution. This in turn can engender the need for the analysis of the model and the algorithmic proposals. Ultimately, the aim is to solve problems having a business, social, or scientific purpose.

## About this book

This book is based on lecture notes we have used in numerous optimization courses we have taught at Stanford University. It is an introduction to linear and nonlinear optimization intended primarily for master's degree students; the book is also suitable for qualified undergraduates and doctoral students. It emphasizes modeling and numerical algorithms for optimization with continuous (not integer) variables. The discussion presents the underlying theory without always focusing on formal mathematical proofs (which can be found in cited references). Another feature of this book is its emphasis on cultural and historical matters, most often appearing among the footnotes.

Reading this book requires no prior course in optimization, but it does require some knowledge of linear algebra and multivariate differential calculus. This means that readers should be familiar with the concept of a finite-dimensional vector space, most importantly  $R^n$  (real  $n$ -space), the algebraic manipulation of vectors and matrices, the property of linear independence of vectors, elimination methods for solving systems of linear equations in many variables, the elementary handling of inequalities, and a good grasp of such analytic concepts as continuity, differentiability, the gradient vector, and the Hessian matrix. These and other topics are discussed in the Appendix.

## Other sources of information

The bibliography at the end of the book contains an extensive list of textbooks and papers on optimization. Among these are Bertsimas and Freund [14], Hillier and Lieberman [96], Luenberger and Ye [125], Bradley et al. [20], Murty [144], and Nash and Sofer [147]. After consulting only a few other



references, you will discover that there are many approaches to the topics we study in this book and quite a few different notational schemes in use. The same can be said of many fields of study. Some people find this downright confusing or even disturbing. But this conceptual and notational diversity is a fact of life we eventually learn to accept. It is part of the price we pay for gaining professional sophistication.

## **Acknowledgments**

Finally, we want to express sincere thanks to Professor Frederick S. Hillier who—for many years—persistently encouraged us to complete the writing of this book for the Springer International Series in Operations Research & Management Science of which he was once the series editor. No less deserving of our gratitude are Suzanne Cottle and Radhika Thapa, our supportive wives.

Linear and Nonlinear Optimization

Cottle, R.W.; Thapa, M.N.

2017, XXXI, 614 p. 58 illus., Hardcover

ISBN: 978-1-4939-7053-7