1 Overview of Optimization Models

Optimization is the process of finding the *best* way of making decisions that satisfy a set of constraints. In mathematical terms, an optimization model is a problem of the form

$$\begin{array}{ll} \min_{\mathbf{x}} & f(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in \mathcal{X}, \end{array} \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $\mathcal{X} \subseteq \mathbb{R}^n$.

Model (1.1) has three main components, namely the vector of *decision variables* $\mathbf{x} := \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^\mathsf{T} \in \mathbb{R}^n$; the *objective function* $f(\mathbf{x})$; and the *constraint set* or *feasible region* $\mathcal{X}$. The constraint set is often expressed in terms of equalities and inequalities involving additional functions. More precisely, the constraint set $\mathcal{X}$ is often of the form

$$\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n : g_i(\mathbf{x}) = b_i, \text{ for } i = 1, \dots, m, \text{ and } h_j(\mathbf{x}) \leq d_j, \text{ for } j = 1, \dots, p\}, \tag{1.2}$$

for some $g_i, h_j : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \dots, m$, $j = 1, \dots, p$. When this is the case, the optimization problem (1.1) is usually written in the form

$$\begin{array}{lll} \min_{\mathbf{x}} & f(\mathbf{x}) & \\ \text{s.t.} & g_i(\mathbf{x}) = b_i, & \text{for } i = 1, \dots, m \\ & h_j(\mathbf{x}) \leq d_j, & \text{for } j = 1, \dots, p, \end{array}$$

or in the more concise form

$$\begin{array}{ll} \min_{\mathbf{x}} & f(\mathbf{x}) \\ \text{s.t.} & \mathbf{g}(\mathbf{x}) = \mathbf{b} \\ & \mathbf{h}(\mathbf{x}) \leq \mathbf{d}. \end{array}$$

We will use the following terminology. A *feasible point* or *feasible solution* to (1.1) is a point in the constraint set $\mathcal{X}$. An *optimal solution* to (1.1) is a feasible point that attains the best possible objective value; that is, a point $\mathbf{x}^* \in \mathcal{X}$ such that $f(\mathbf{x}^*) \leq f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}$. The *optimal value* of (1.1) is the value of the objective function at an optimal solution; that is, $f(\mathbf{x}^*)$ where $\mathbf{x}^*$ is an optimal solution to (1.1). If the feasible region $\mathcal{X}$ is of the form (1.2) and $\mathbf{x} \in \mathcal{X}$, the *binding constraints* at $\mathbf{x}$ are the equality constraints and those inequality constraints that hold with equality at $\mathbf{x}$. The term *active constraint* is also often used in lieu of "binding constraint". The problem (1.1) is *infeasible* if $\mathcal{X} = \emptyset$. On

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

the other hand, (1.1) is *unbounded* if there exist $\mathbf{x}_k \in \mathcal{X},\ k = 1, 2, \ldots,$ such that $f(\mathbf{x}_k) \to -\infty$.

1.1 Types of Optimization Models

For optimization models to be of practical interest, their computational tractability, that is, the ability to find the optimal solution efficiently, is a critical issue. Particular structural assumptions on the objective and constraints of the problem give rise to different classes of optimization models with various degrees of computational difficulty. We should note that the following is only a partial classification based on the current generic tractability of various types of optimization models. However, what is "tractable" in some specific context may be more nuanced. Furthermore, tractability evolves as new algorithms and technologies are developed.

Convex optimization: These are problems where the objective $f(\mathbf{x})$ is a convex function and the constraint set $\mathcal{X}$ is a convex set. This class of optimization models is tractable most of the time. By this we mean that a user can expect any of these models to be amenable to an efficient algorithm. We will emphasize this class of optimization models throughout the book.

Mixed integer optimization: These are problems where some of the variables are restricted to take integer values. This restriction makes the constraint set $\mathcal{X}$ non-convex. This class of optimization models is somewhat tractable a fair portion of the time. By this we mean that a model of this class may be solvable provided the user does some judicious modeling and has access to high computational power.

Stochastic and dynamic optimization: These are problems involving random and time-dependent features. This class of optimization models is tractable only in some special cases. By this we mean that, unless some specific structure and assumptions hold, a model of this class would typically be insoluble with any realistic amount of computational power at our disposal. Current research is expected to enrich the class of tractable models in this area.

The modeling of time and uncertainty is pervasive in almost every financial problem. The various types of optimization problems that we will discuss are based on how they deal with these two issues. Generally speaking, *static models* are associated with simple single-period models where the future is modeled as a single stage. By contrast, in *multi-period* models the future is modeled as a sequence, or possibly as a continuum, of stages. With regard to uncertainty, *deterministic models* are those where all the defining data are assumed to be known with certainty. By contrast, *stochastic models* are ones that incorporate probabilistic or other types of uncertainty in the data.

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

A good portion of the models that we will present in this book will be convex optimization models due to their favorable mathematical and computational properties. There are two special types of convex optimization problems that we will use particularly often: *linear* and *quadratic* programming, the latter being an extension of the former. These two types of optimization models will be discussed in more detail in Chapters 2 and 5. We now present a high-level description of four major classes of optimization models: linear programming, quadratic programming, mixed integer programming, and stochastic optimization.

Linear Programming

A linear programming model is an optimization problem where the objective is a linear function and the constraint set is defined by finitely many linear equalities and linear inequalities. In other words, a linear program is a problem of the form

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{D}\mathbf{x} \geq \mathbf{d} \end{array}$$

for some vectors $\mathbf{c} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^m, \mathbf{d} \in \mathbb{R}^p$ and matrices $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{D} \in \mathbb{R}^{p\times n}$.

The term *linear optimization* is sometimes used in place of linear programming. The wide popularity of linear programming is due in good part to the availability of very efficient algorithms. The two best known and most successful methods for solving linear programs are the *simplex method* and *interior-point methods.* We briefly discuss these algorithms in Chapter 2.

Quadratic Programming

Quadratic programming, also known as quadratic optimization, is an extension of linear programming where the objective function includes a quadratic term. In other words, a quadratic program is a problem of the form

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{Q}\mathbf{x} + \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{D}\mathbf{x} \geq \mathbf{d} \end{array}$$

for some vectors and matrices $\mathbf{Q} \in \mathbb{R}^{n\times n}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{d} \in \mathbb{R}^p$, $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{D} \in \mathbb{R}^{p\times n}$. It is customary to assume that the matrix $\mathbf{Q}$ is symmetric. This assumption can be made without loss of generality since

$$\mathbf{x}^{\mathsf{T}}\mathbf{Q}\mathbf{x} = \mathbf{x}^{\mathsf{T}}\tilde{\mathbf{Q}}\mathbf{x}$$

where $\tilde{\mathbf{Q}} = \frac{1}{2}(\mathbf{Q} + \mathbf{Q}^{\mathsf{T}})$, which is clearly a symmetric matrix.

We note that a quadratic function $\frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{Q}\mathbf{x} + \mathbf{c}^{\mathsf{T}}\mathbf{x}$ is convex if and only if the matrix $\mathbf{Q}$ is positive semidefinite ($\mathbf{x}^{\mathsf{T}}\mathbf{Q}\mathbf{x} \geq 0$ for all $x \in \mathbb{R}^n$). In this case the above quadratic program is a convex optimization problem and can be solved

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

efficiently. The two best known methods for solving convex quadratic programs are *active-set methods* and *interior-point methods*. We briefly discuss these algorithms in Chapter 5.

Mixed Integer Programming

A mixed-integer program is an optimization problem that restricts some or all of the decision variables to take integer values. In particular, a mixed integer linear programming model is a problem of the form

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^{\mathsf{T}}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{D}\mathbf{x} \geq \mathbf{d} \\ & x_j \in \mathbb{Z},\ j \in J \end{array}$$

for some vectors and matrices $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{d} \in \mathbb{R}^p$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{D} \in \mathbb{R}^{p \times n}$ and some $J \subseteq \{1, \ldots, n\}$.

An important case occurs when the model includes *binary* variables, that is, variables that are restricted to take values 0 or 1. As we will see, the inclusion of this type of constraint increases the modeling power but comes at a cost in terms of computational tractability. It is noteworthy that the computational and algorithmic machinery for solving mixed integer programs has vastly improved during the last couple of decades. The main classes of methods for solving mixed integer programs are *branch and bound*, *cutting planes*, and a combination of these two approaches known as *branch and cut*. We briefly discuss these algorithms in Chapter 8.

Stochastic Optimization

Stochastic optimization models are optimization problems that account for randomness in their objective or constraints. The following formulation illustrates a generic type of stochastic optimization problem

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbb{E}(F(\mathbf{x}, \omega)) \\ & \mathbf{x} \in \mathcal{X}. \end{array}$$

In this problem the set of decisions $\mathbf{x}$ must be made before a random outcome ω occurs. The goal is to optimize the expectation of some function that depends on both the decision vector $\mathbf{x}$ and the random outcome ω. A variation of this formulation, that has led to important developments, is to replace the expectation by some kind of *risk measure* ϱ in the objective:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \varrho(F(\mathbf{x}, \omega)) \\ & \mathbf{x} \in \mathcal{X}. \end{array}$$

There are numerous refinements and variants of the above two formulations. In particular, the class of *two-stage stochastic optimization with recourse* has been

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

widely studied in the stochastic programming community. In this setting a set of decisions $\mathbf{x}$ must be made in stage one. Between stage one and stage two a random outcome ω occurs. At stage two we have the opportunity to make some second-stage *recourse* decisions $\mathbf{y}(\omega)$ that may depend on the random outcome ω.

The two-stage stochastic optimization problem with recourse can be formally stated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) + \mathbb{E}[Q(\mathbf{x},\omega)] \\ & \mathbf{x} \in \mathcal{X}. \end{aligned}$$

The *recourse* term $Q(\mathbf{x},\omega)$ depends on the first-stage decisions $\mathbf{x}$ and the random outcome ω. It is of the form

$$\begin{aligned} Q(\mathbf{x},\omega) := \min_{\mathbf{y}(\omega)} \quad & g(\mathbf{y}(\omega),\omega) \\ & \mathbf{y}(\omega) \in \mathcal{Y}(\mathbf{x},\omega). \end{aligned}$$

The second-stage decisions $\mathbf{y}(\omega)$ are *adaptive* to the random outcome ω because they are made after ω is revealed. The objective function in a two-stage stochastic optimization problem contains a term for the stage-one decisions and a term for the stage-two decisions where the latter term involves an expectation over the random outcomes. The intuition of this objective function is that the stage-one decisions should be made considering what is to be expected in stage two.

The above two-stage setting generalizes to a multi-stage context where the random outcome is revealed over time and decisions are made dynamically at multiple stages and can adapt to the information revealed up to their stage.

1.2 Solution to Optimization Problems

The solution to an optimization problem can often be characterized in terms of a set of *optimality conditions.* Optimality conditions are derived from the mathematical relationship between the objective and constraints in the problem. Subsequent chapters discuss optimality conditions for various types of optimization problems. In special cases, these optimality conditions can be solved analytically and used to infer properties about the optimal solution. However, in many cases we rely on numerical solvers to obtain the solution to the optimization models.

There are numerous software vendors that provide solvers for optimization problems. Throughout this book we will illustrate examples with two popular solvers, namely Excel `Solver` and the MATLAB®-based optimization modeling framework `CVX`. Excel and MATLAB files for the examples and exercises in the book are available at:

`www.andrew.cmu.edu/user/jfp/OIFbook/`

Both Excel `Solver` and `CVX` enable us to solve small to medium-sized problems and are fairly easy to use. There are far more sophisticated solvers such as the

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

commercial solvers IBM®-ILOG® CPLEX®, Gurobi, FICO® Xpress, and the ones available via the open-source projects COIN-OR or SCIP.

Optimization problems can be formulated using modeling languages such as AMPL, GAMS, MOSEL, or OPL. The need for these modeling languages arises when the size of the formulation is large. A modeling language lets people use common notation and familiar concepts to formulate optimization models and examine solutions. Most importantly, large problems can be formulated in a compact way. Once the problem has been formulated using a modeling language, it can be solved using any number of solvers. A user can switch between solvers with a single command and select options that may improve solver performance.

1.3 Financial Optimization Models

In this book we will focus on the use of optimization models for financial problems such as portfolio management, risk management, asset and liability management, trade execution, and dynamic asset management. Optimization models are also widely used in other areas of business, science, and engineering, but this will not be the subject of our discussion.

Portfolio Management

One of the best known optimization models in finance is the portfolio selection model of Markowitz (1952). Markowitz's mean–variance approach led to major developments in financial economics including Tobin's mutual fund theorem (Tobin, 1958) and the capital asset pricing model of Treynor[1], Sharpe (1964), Lintner (1965), and Mossin (1966). Markowitz was awarded the Nobel Prize in Economics in 1990 for the enormous influence of his work in financial theory and practice. The gist of this model is to formalize the principle of diversification when selecting a portfolio in a universe of risky assets. As we discuss in detail in Chapter 6, Markowitz's mean–variance model and a wide range of its variations can be stated as a quadratic programming problem of the form

$$\begin{array}{rl} \min\limits_{\mathbf{x}} & \frac{1}{2}\gamma \cdot \mathbf{x}^{\mathsf{T}}\mathbf{V}\mathbf{x} - \boldsymbol{\mu}^{\mathsf{T}}\mathbf{x} \\ & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{D}\mathbf{x} \geq \mathbf{d}. \end{array} \tag{1.3}$$

The vector of decision variables $\mathbf{x}$ in model (1.3) represents the portfolio holdings. These holdings typically represent the percentages invested in each asset and thus are often subject to the full investment constraint $\mathbf{1}^{\mathsf{T}}\mathbf{x} = 1$. Other common constraints include the long-only constraint $\mathbf{x} \geq \mathbf{0}$, as well as restrictions related to sector or industry composition, turnover, etc. The terms $\mathbf{x}^{\mathsf{T}}\mathbf{V}\mathbf{x}$ and $\boldsymbol{\mu}^{\mathsf{T}}\mathbf{x}$ in the objective function are respectively the variance, which is a measure of risk,

[1] "Toward a theory of market value of risky assets". Unpublished manuscript, 1961.

https://doi.org/10.1017/9781107297340.002 Published online by Cambridge University Press

and the expected return of the portfolio defined by $\mathbf{x}$. The risk-aversion constant $\gamma > 0$ in the objective determines the tradeoff between risk and return of the portfolio.

Risk Management

Risk is inherent in most economic activities. This is especially true of financial activities where results of decisions made today may have many possible different outcomes depending on future events. Since companies cannot usually insure themselves completely against risk, they have to manage it. This is a hard task even with the support of advanced mathematical techniques. Poor risk management led to several spectacular failures in the financial industry in the 1990s (e.g., Barings Bank, Long Term Capital Management, Orange County). It was also responsible for failures and bailouts of a number of institutions (e.g., Lehman Brothers, Bear Stearns, AIG) during the far more severe global financial crisis of 2007–2008. Regulations, such as those prescribed by the Basel Accord (see Basel Committee on Banking Supervision, 2011), mandate that financial institutions control their risk via a variety of measurable requirements. The modeling of regulatory constraints as well as other risk-related constraints that the firm wishes to impose to prevent vulnerabilities can often be stated as a set of constraints

$$\mathbf{RM}(\mathbf{x}) \leq \mathbf{b}. \tag{1.4}$$

The vector $\mathbf{x}$ in (1.4) represents the holdings in a set of risky securities. The entries of the vector-valued function $\mathbf{RM}(\mathbf{x})$ represent one or more measures of risk and the vector $\mathbf{b}$ represents the acceptable upper limits on these measures. The set of risk management constraints (1.4) may be embedded in a more elaborate model that aims to optimize some kind of performance measure such as expected investment return.

In Chapter 2 we discuss a linear programming model for optimal bank planning under Basel III regulations. In this case the components of the function $\mathbf{RM}(\mathbf{x})$ are linear functions of $\mathbf{x}$. In Chapter 11 we discuss more sophisticated risk measures such as value at risk and conditional value at risk that typically make $\mathbf{RM}(\mathbf{x})$ a nonlinear function of $\mathbf{x}$.

Asset and Liability Management

How should a financial institution manage its assets and liabilities? A static model, such as the Markowitz mean–variance portfolio selection model, fails to incorporate the multi-period nature of typical liabilities faced by financial institutions. Furthermore, it penalizes returns both above and below the mean. A multi-period model that emphasizes the need to meet liabilities in each period for a finite (or possibly infinite) horizon is often more appropriate. Since liabilities and asset returns usually have random components, their optimal management requires techniques to optimize under uncertainty such as stochastic optimization.

We discuss several asset and liability management models in Chapters 3, 16, and 17. A generic asset and liability management model can often be formulated as a stochastic programming problem of the form

$$\begin{array}{ll} \max_{\mathbf{x}} & \mathbb{E}(U(\mathbf{x})) \\ & \mathbf{F}\mathbf{x} = \mathbf{L} \\ & \mathbf{D}\mathbf{x} \geq \mathbf{0}. \end{array} \tag{1.5}$$

The vector $\mathbf{x}$ in (1.5) represents the investment decisions for the available assets at the dates in the planning horizon. The vector $\mathbf{L}$ in (1.5) represents the liabilities that the institution faces at the dates in the planning horizon. The constraints $\mathbf{F}\mathbf{x} = \mathbf{L}$, $\mathbf{D}\mathbf{x} \geq \mathbf{0}$ represent the cash flow rules and restrictions applicable to the assets during the planning horizon. The term $U(\mathbf{x})$ in the objective function is some appropriate measure of utility. For instance, it could be the value of terminal wealth at the end of the planning horizon. In general, the components $\mathbf{F}, \mathbf{L}, \mathbf{D}$ are discrete-time random processes and thus (1.5) is a multi-stage stochastic programming model with recourse. In Chapter 3 we discuss some special cases of (1.5) with no randomness.

1.4 Notes

George Dantzig was the inventor of linear programming and author of many related articles as well as a classical reference on the subject (Dantzig, 1963). A particularly colorful and entertaining description of the diet problem, a classical linear programming model, can be found in Dantzig (1990).

Boyd and Vandenberghe (2004) give an excellent exposition of convex optimization appropriate for senior or first-year graduate students in engineering. This book is freely available at:

`www.stanford.edu/~boyd/cvxbook/`

Ragsdale (2007) gives a practical exposition of optimization and related spreadsheet models that circumvent most technical issues. It is appropriate for senior or Master's students in business.