be accurately solved recursively going backwards in time. The situation starts to change dramatically with increase of the number of variables on which the value functions depend, like in the example discussed in the next section. The discretization approach may still work with several state variables, but it quickly becomes impractical when the dimension of the state vector increases. This is called the "curse of dimensionality." As we shall see it later, stochastic programming approaches the problem in a different way, by exploring convexity of the underlying problem, and thus attempting to solve problems with a state vector of high dimension. This is achieved by means of discretization of the random process D_t in a form of a scenario tree, which may also become prohibitively large.

1.3 Multi-Product Assembly

1.3.1 Two Stage Model

Consider a situation where a manufacturer produces n products. There are in total m different parts (or subassemblies) which have to be ordered from third party suppliers. A unit of product i requires a_{ij} units of part j, where $i = 1, \dots, n$ and $j = 1, \dots, m$. Of course, a_{ij} may be zero for some combinations of i and j. The demand for the products is modeled as a random vector $D = (D_1, \dots, D_n)$. Before the demand is known, the manufacturer may pre-order the parts from outside suppliers, at the cost of c_j per unit of part j. After the demand D is observed, the manufacturer may decide which portion of the demand is to be satisfied, so that the available numbers of parts are not exceeded. It costs additionally l_i to satisfy a unit of demand for product i, and the unit selling price of this product is q_i. The parts which are not used are assessed salvage values $s_j < c_j$. The unsatisfied demand is lost.

Suppose the numbers of parts ordered are equal to x_j, $j = 1, \dots, m$. After the demand D becomes known, we need to determine how much of each product to make. Let us denote the numbers of units produced by z_i, $i = 1, \dots, n$, and the numbers of parts left in inventory by y_j, $j = 1, \dots, m$. For an observed value (a realization) $d = (d_1, ..., d_n)$ of the random demand vector D, we can find the best production plan by solving the following linear programming problem

$$
\begin{aligned}
\operatorname*{Min}_{z,y} \ & \sum_{i=1}^{n} (l_i - q_i) z_i - \sum_{j=1}^{n} s_j y_j \\
\text{s.t. } & y_j = x_j - \sum_{i=1}^{n} a_{ij} z_i, \quad j = 1, \dots, m, \\
& 0 \le z_i \le d_i, \quad i = 1, \dots, n, \quad y_j \ge 0, \quad j = 1, \dots, m.
\end{aligned}
$$

Introducing the matrix A with entries a_{ij}, where $i = 1, \dots, n$ and $j = 1, \dots, m$, we can write this problem compactly as follows:

$$
\begin{aligned}
\operatorname*{Min}_{z,y} \ & (l - q)^{\mathsf{T}} z - s^{\mathsf{T}} y \\
\text{s.t. } & y = x - A^{\mathsf{T}} z, \\
& 0 \le z \le d, \quad y \ge 0.
\end{aligned}
\tag{1.23}
$$

Observe that the solution of this problem, that is, the vectors z and y, depend on realization d of the demand vector D as well as on x. Let $Q(x, d)$ denote the optimal value of problem (1.23). The quantities x_j of parts to be ordered can be determined from the following optimization problem

$$\operatorname*{Min}_{x \geq 0} \; c^{\mathsf{T}} x + \mathbb{E}[Q(x, D)], \tag{1.24}$$

where the expectation is taken with respect to the probability distribution of the random demand vector D. The first part of the objective function represents the ordering cost, while the second part represents the expected cost of the optimal production plan, given ordered quantities x. Clearly, for realistic data with $q_i > l_i$, the second part will be negative, so that some profit will be expected.

Problem (1.23)–(1.24) is an example of a *two stage stochastic programming problem*, with (1.23) called the *second stage problem*, and (1.24) called the *first stage problem*. As the second stage problem contains random data (random demand D), its optimal value $Q(x, D)$ is a random variable. The distribution of this random variable depends on the first stage decisions x, and therefore the first stage problem cannot be solved without understanding of the properties of the second stage problem.

In the special case of finitely many demand scenarios $d^1, \dots, d^K$ occurring with positive probabilities $p_1, \dots, p_K$, with $\sum_{k=1}^{K} p_k = 1$, the two stage problem (1.23)–(1.24) can be written as one large scale linear programming problem:

$$\begin{aligned}
\text{Min } & c^{\mathsf{T}} x + \sum_{k=1}^{K} p_k \left[(l - q)^{\mathsf{T}} z^k - s^{\mathsf{T}} y^k \right] \\
\text{s.t. } & y^k = x - A^{\mathsf{T}} z^k, \quad k = 1, \dots, K, \\
& 0 \leq z^k \leq d^k, \quad y^k \geq 0, \quad k = 1, \dots, K, \\
& x \geq 0,
\end{aligned} \tag{1.25}$$

where the minimization is performed over vector variables x and z^k, y^k, $k = 1, ..., K$. We have integrated the second stage problem (1.23) into this formulation, but we had to allow for its solution (z^k, y^k) to depend on the scenario k, because the demand realization d^k is different in each scenario. Because of that, problem (1.25) has the numbers of variables and constraints roughly proportional to the number of scenarios K.

It is worthwhile to notice the following. There are three types of decision variables here. Namely, the numbers of ordered parts (vector x), the numbers of produced units (vector z) and the numbers of parts left in the inventory (vector y). These decision variables are naturally classified as the *first* and the *second* stage decision variables. That is, the first stage decisions x should be made *before* a realization of the random data becomes available and hence should be independent of the random data, while the second stage decision variables z and y are made *after* observing the random data and are functions of the data. The first stage decision variables are often referred to as *"here-and-now"* decisions (solution), and second stage decisions are referred to as *"wait-and-see"* decisions (solution). It can also be noticed that the second stage problem (1.23) is feasible for every possible realization of the random data; for example, take $z = 0$ and $y = x$. In such a situation we say that the problem has *relatively complete recourse*.