$$\boldsymbol{y} = \boldsymbol{B}^{\mathrm{T}}\boldsymbol{x}$$

is a vector of fictitious securities; one for each factor. The variance of the portfolio's return again involves only a diagonal matrix (or an almost diagonal matrix if $\boldsymbol{W}$ is not diagonal).

As shown in Jacobs, Levy, and Markowitz (2005, 2006), this diagonalization procedure applied to portfolios with both long and short positions gives

$$\mu_P = \begin{bmatrix} l \\ s \\ y \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} +\mu \\ -\mu \\ \phi \end{bmatrix}, \tag{3.30}$$

$$\sigma_P^2 = \begin{bmatrix} l \\ s \\ y \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} V & O & O \\ O & V & O \\ O & O & W \end{bmatrix} \begin{bmatrix} l \\ s \\ y \end{bmatrix} + \begin{bmatrix} l \\ s \\ y \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} O & -V & O \\ -V & O & O \\ O & O & O \end{bmatrix} \begin{bmatrix} l \\ s \\ y \end{bmatrix}, \tag{3.31}$$

with the artificial security vector defined as

$$\boldsymbol{y} = \boldsymbol{B}^{T}(\boldsymbol{l} - \boldsymbol{s}).$$

Although the second term on the right-hand side of Equation (3.31) distorts the diagonal form of the variance, for trimable models, as shown in Jacobs, Levy, and Markowitz (2005, 2006), one can effectively ignore it. One simply presents the appropriate fast algorithm with the $2n$ vector of expected security returns $[+\boldsymbol{\mu}^{\mathrm{T}}, -\boldsymbol{\mu}^{\mathrm{T}}]^{\mathrm{T}}$, the vector of expected factor returns $\boldsymbol{\Phi}$, the $2n$ vector of idiosyncratic variances $[+\boldsymbol{v}^{\mathrm{T}}, +\boldsymbol{v}^{\mathrm{T}}]^{\mathrm{T}}$, the covariance matrix of the factor returns $\boldsymbol{W}$, the $2n$ vector of betas $[+\boldsymbol{\beta}^{\mathrm{T}}, -\boldsymbol{\beta}^{\mathrm{T}}]$, and any constraints (including the constraint that couples the fictitious security vector to the real ones). Again, importantly, the dense covariance matrix of the real securities' returns is not presented directly to the algorithm. The fast algorithm will then find the correct efficient frontier.

3.3.3 Using the Critical Line Algorithm

Another method for optimizing EAE portfolios is to use the critical line algorithm (CLA) of Markowitz (1987) and Markowitz and Todd (2000). Unlike the fast algorithms described above, the CLA does not depend on the particular structure of the covariance matrix. In fact, a major advantage of the CLA is that it still maps out the entire, and correct, efficient frontier even when the covariance matrix is singular. It is, therefore, ideally suited to optimizing EAE portfolios.

In this section, we summarize the CLA with a presentation based on the use of 0-1 matrices (see, e.g., Magnus and Neudecker, 1986). This helps to clarify the structure of the algorithm, and to assist in its translation to matrix-oriented programming languages. In Section 3.3.3.1 we demonstrate how the standard CLA can be modified to apply to benchmark-sensitive portfolios such as EAE portfolios.

There is a perception among some researchers that the CLA cannot accommodate upper bounds on security holdings (see, e.g., Stein, Branke, and Schmeck, 2008, page 3949) without the addition of slack variables. Exercise 7.3 on page 179 of Markowitz and Todd (2000), however, shows how the algorithm can be modified so as to incorporate such bounds directly without the use of slack variables. In this section, we demonstrate the use of the CLA with upper bound modifications.

Our objective is to:

$$\begin{aligned} &\text{Find } \boldsymbol{x} \text{ to minimize } \mathcal{O} &&= \tfrac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Omega} \boldsymbol{x} - \tau\, \boldsymbol{\mu}^\top \boldsymbol{x} \\ &\text{subject to} && \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \\ & && \boldsymbol{x} \ \geq \boldsymbol{d}, \\ & && \boldsymbol{x} \ \leq \boldsymbol{u}. \end{aligned} \tag{3.32}$$

Notice that, with an obvious change in notation, the EAE problem in Equation (3.23) can be formulated in this way.

Define the following three sets: IN to represent the securities that are currently between their lower and upper limits (i.e., that are in the portfolio, or IN), UP to represent those securities that are currently at their upper limit (i.e., that are UP), and DN to represent those securities that are currently at their lower limit (i.e., that are DN). That is,

$$\mathrm{IN} = \{i \mid d_i < x_i < u_i\}, \quad \mathrm{UP} = \{i \mid x_i = u_i\}, \quad \mathrm{DN} = \{i \mid x_i = d_i\}. \tag{3.33}$$

We will use an IN, DN, or UP subscript on a vector to indicate that the vector contains only elements from the corresponding set.

Also define three operators $\mathbb{N}$, $\mathbb{D}$, and $\mathbb{U}$ corresponding to IN, DN, and UP. $\mathbb{N}$'s domain is the set of all portfolio vectors $\boldsymbol{x}$, and its range is the set of vectors whose dimension is the cardinality of IN, with $|\text{IN}| \leq n$. $\mathbb{N}\boldsymbol{x}$ is a vector that contains only the elements of $\boldsymbol{x}$ that are IN. $\mathbb{D}$ and $\mathbb{U}$ are defined analogously with respect to DN and UP, respectively. Thus, $\mathbb{N}\boldsymbol{x} = \boldsymbol{x}_{\text{IN}}$, $\mathbb{D}\boldsymbol{x} = \boldsymbol{x}_{\text{DN}}$, and $\mathbb{U}\boldsymbol{x} = \boldsymbol{x}_{\text{UP}}$. It can be useful to interpret the $\mathbb{N}$, $\mathbb{D}$, and $\mathbb{U}$ operators as multiplications by identity matrices that contain only the rows from their corresponding sets.[5] The matrix $\mathbb{N}$, for example, is generated using an identity matrix from which all rows have been removed except those that are in the set IN. By definition, $\mathbb{N}\boldsymbol{x} = \mathbb{N}\boldsymbol{d} = \boldsymbol{d}_{\text{DN}}$, and $\mathbb{U}\boldsymbol{x} = \mathbb{U}\boldsymbol{u} = \boldsymbol{u}_{\text{UP}}$.

While multiplication by $\mathbb{N}$, $\mathbb{D}$, or $\mathbb{U}$ removes elements not belonging to the corresponding sets, and therefore generally results in shortened vectors, multiplication by these matrices' transposes causes vector re-lengthening, with zeros inserted for all elements not belonging to the corresponding set. Using these properties, it is easy to show that $\boldsymbol{x}$ can be decomposed as follows:

$$\begin{aligned} \boldsymbol{x} &= \mathbb{N}^\top \boldsymbol{x}_{\text{IN}} + \mathbb{D}^\top \boldsymbol{d}_{\text{DN}} + \mathbb{U}^\top \boldsymbol{u}_{\text{UP}}, \\ &= \mathbb{N}^\top \boldsymbol{x}_{\text{IN}} + \mathbb{D}^\top \mathbb{D}\boldsymbol{d} + \mathbb{U}^\top \mathbb{U}\boldsymbol{u}, \\ &= \mathbb{N}^\top \boldsymbol{x}_{\text{IN}} + \boldsymbol{k}, \end{aligned} \tag{3.34}$$

where we have defined

$$\boldsymbol{k} = \mathbb{D}^\top \mathbb{D}\boldsymbol{d} + \mathbb{U}^\top \mathbb{U}\boldsymbol{u} \tag{3.35}$$

to be the vector whose elements are zero for IN securities, equal to the lower bound for DN securities, and equal to the upper bound for UP securities.

We now use the definitions of $\mathbb{N}$, $\mathbb{D}$, and $\mathbb{U}$ in the analysis of the Lagrangian in Equation (3.7), which applies to the problem in equation set (3.32). Let $\boldsymbol{\eta}$ be the gradient of the Lagrangian:

$$\boldsymbol{\eta} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{x}} = \boldsymbol{\Omega}\boldsymbol{x} - \boldsymbol{\mu}\tau + \boldsymbol{A}^\top \boldsymbol{\lambda}. \tag{3.36}$$

For securities that are between their upper and lower limits, i.e., for $i \in \text{IN}$, the gradient must be zero. Thus, premultiplying Equation (3.36) by $\mathbb{N}$ to select only the IN securities,

$$\mathbb{N}\boldsymbol{\Omega}\boldsymbol{x} - \mathbb{N}\boldsymbol{\mu}\tau + \mathbb{N}\boldsymbol{A}^\top \boldsymbol{\lambda} = 0$$

which, with the use of Equation (3.34), gives

$$\mathbb{N}\boldsymbol{\Omega}\mathbb{N}^{\top}\boldsymbol{x}_{\mathrm{IN}} + \mathbb{N}A^{\top}\boldsymbol{\lambda} = \mathbb{N}\boldsymbol{\mu}\tau - \mathbb{N}\boldsymbol{\Omega}\boldsymbol{k}. \tag{3.37}$$

Using Equation (3.34), the general linear constraint, Equation (3.6), becomes

$$A\mathbb{N}^{\top}\boldsymbol{x}_{\mathrm{IN}} = \boldsymbol{b} - A\boldsymbol{k}. \tag{3.38}$$

Finally, combining Equations (3.37) and (3.38) gives

$$\begin{bmatrix} \mathbb{N}\boldsymbol{\Omega}\mathbb{N}^{\top} & \mathbb{N}A^{\top} \\ A\mathbb{N}^{\top} & \mathrm{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{\mathrm{IN}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{b} \end{bmatrix} - \begin{bmatrix} \mathbb{N}\boldsymbol{\Omega} \\ A \end{bmatrix} \boldsymbol{k} + \begin{bmatrix} \mathbb{N}\boldsymbol{\mu} \\ 0 \end{bmatrix} \tau. \tag{3.39}$$

This is the equation that defines the straight-line segment along the efficient frontier for the given IN, UP, and DN sets. As τ changes, the solution reaches discrete critical points (i.e., corner portfolios) at which the compositions of these sets change. The critical line algorithm provides a computationally efficient way to step from corner portfolio to corner portfolio, there by tracing out the entire efficient frontier.

3.3.3.1 *CLA with a Benchmark*

In the discussion thus far, we have not been specific about the precise nature of the portfolio weights. The CLA as originally formulated is very general and can be applied to virtually any weight representation. However, for institutional portfolio management, it is useful to distinguish between total, benchmark, and active weights. More specifically, a portfolio's total weight vector $\boldsymbol{h}$ comprises the sum of two components: a benchmark or market component $\boldsymbol{m}$ and an active component $\boldsymbol{x}$. The skill of the active portfolio manager is focused on choosing the best active component. The Lagrangian in this case is

$$\mathcal{L}(\boldsymbol{h}) = \tfrac{1}{2}(\boldsymbol{h}-\boldsymbol{m})^{\top} Q (\boldsymbol{h}-\boldsymbol{m}) - \tau\boldsymbol{\theta}^{\top}(\boldsymbol{h}-\boldsymbol{m}) + \boldsymbol{\lambda}^{\top}(A\boldsymbol{h}-\boldsymbol{b}). \tag{3.40}$$

Repeating the process of optimizing this Lagrangian, but now choosing the variable of interest to be the total weight vector $\boldsymbol{h}$ rather than x, shows that all expressions remain identical except for the constant on the right-hand side of Equation (3.39). In particular, Equation (3.39) becomes

$$\begin{bmatrix} \mathbb{N}\boldsymbol{\Omega}\mathbb{N}^\top & \mathbb{N}A^\top \\ A\mathbb{N}^\top & \mathrm{O} \end{bmatrix} \begin{bmatrix} h_{\mathrm{IN}} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ b \end{bmatrix} - \begin{bmatrix} \mathbb{N}\boldsymbol{\Omega} \\ A \end{bmatrix} k + \begin{bmatrix} \mathbb{N}\boldsymbol{\Omega} \\ \mathrm{O} \end{bmatrix} m + \begin{bmatrix} \mathbb{N}\boldsymbol{\mu} \\ 0 \end{bmatrix} \tau. \quad (3.41)$$

3.3.3.2 *Algorithm Implementation*

Markowitz (1987) showed that the matrix (which we will call M_{IN}) on the left-hand side of Equations (3.39) and (3.41) is nonsingular, so these equations can be solved for $\boldsymbol{x}_{\mathrm{IN}}$ and $\boldsymbol{\lambda}$ to give[6]

$$\begin{bmatrix} h_{\mathrm{IN}} \\ \lambda \end{bmatrix} = \boldsymbol{\alpha} + \boldsymbol{\beta}\tau, \quad (3.42)$$

where, in our case,

$$\boldsymbol{\alpha} = M_{\mathrm{IN}}^{-1}\left[\begin{bmatrix} 0 \\ b \end{bmatrix} - \begin{bmatrix} \mathbb{N}\boldsymbol{\Omega} \\ A \end{bmatrix} k + \begin{bmatrix} \mathbb{N}\boldsymbol{\Omega} \\ \mathrm{O} \end{bmatrix} m\right], \quad (3.43)$$

$$\boldsymbol{\beta} = M_{\mathrm{IN}}^{-1}\begin{bmatrix} \mathbb{N}\boldsymbol{\mu} \\ 0 \end{bmatrix}. \quad (3.44)$$

Equations (3.43) and (3.44) can be substituted into Equation (3.36) to give

$$\boldsymbol{\eta} = \boldsymbol{\gamma} + \boldsymbol{\delta}\tau, \quad (3.45)$$

where, in our case,

$$\boldsymbol{\gamma} = G\boldsymbol{\alpha} + \boldsymbol{\Omega}(k - m), \quad (3.46)$$

and

$$\boldsymbol{\delta} = G\boldsymbol{\beta} - \boldsymbol{\mu}, \quad (3.47)$$

with

$$G = \begin{bmatrix} \boldsymbol{\Omega}\mathbb{N}^\top & A^\top \end{bmatrix}. \quad (3.48)$$

The CLA traces out the efficient set as risk tolerance τ is reduced in discrete steps from infinity down to zero.[7] That is, the algorithm steps from corner portfolio to corner portfolio. At a typical step, the algorithm has available the sets IN, UP, and DN. With these sets, it can solve for $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\delta}$. With this information it can compute which of the following events takes place at the next corner:

1. An IN security can reach its upper bound, and move from IN to UP. In this case, since the security is initially in IN, it must satisfy Equation (3.42), and as it attains the upper bound, it must satisfy $u_i = \alpha_i + \beta_i\tau$, from which we obtain that the corner portfolio occurs when τ reaches

$$\tau_1 = \frac{u_i - \alpha_i}{\beta_i}, \quad \beta_i < 0. \tag{3.49}$$

2. An IN security can reach its lower bound, and move from IN to DN. In this case, since the security is initially in IN, it must satisfy Equation (3.42), and as it attains the lower bound, it must satisfy $d_i = \alpha_i + \beta_i\tau$, from which we obtain that the corner portfolio occurs when τ reaches

$$\tau_2 = \frac{d_i - \alpha_i}{\beta_i}, \quad \beta_i > 0. \tag{3.50}$$

3. A security at its upper bound could move from UP to IN. In this case, for the security to enter IN, its gradient must increase to zero as τ decreases. From Equation (3.45), this occurs when τ reaches

$$\tau_3 = -\frac{\gamma_i}{\delta_i}, \quad \delta_i < 0. \tag{3.51}$$

4. A security at its lower bound could move from DN to IN. In this case, for the security to enter IN, its gradient must decrease to zero as τ decreases. From Equation (3.45), this occurs when τ reaches

$$\tau_4 = -\frac{\gamma_i}{\delta_i}, \quad \delta_i > 0. \tag{3.52}$$

The largest of $\{\tau_1, \tau_2, \tau_3, \tau_4\}$ determines the value of τ at the next corner, or if the algorithm terminates.

We now have all information necessary to implement the CLA for EAE portfolios with a constraint set that includes both lower and upper bounds.

3.4 EXAMPLE

To illustrate the use of the method described in Section 3.3.3, we found the efficient frontier for a portfolio of n = 100 securities whose excess returns were generated by the model

$$\boldsymbol{r} = \boldsymbol{\alpha} + \boldsymbol{\beta} r_m + \boldsymbol{\varepsilon},$$

where, for each realization, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ were normally distributed with means of zero and one, respectively, and standard deviations of 5% and 20%, respectively. The idiosyncratic return vector, $\boldsymbol{\varepsilon}$, was normally distributed with a mean of zero and a standard deviation of σ_ε = 30%. The market excess return, r_m, had a mean of μ_m = 5% and a standard deviation of σ_m = 15%. We formed the expected return vector using $\boldsymbol{\mu} = \boldsymbol{\alpha} + \boldsymbol{\beta}\mu_m$, and the covariance matrix using $\boldsymbol{\Omega} = \boldsymbol{\beta}\sigma_m^2\boldsymbol{\beta}^\top + \sigma_\varepsilon^2\boldsymbol{I}$.

To produce the efficient frontier for a long-only portfolio with a beta of one, we used a CLA program with $\boldsymbol{\mu}$ and $\boldsymbol{\Omega}$, together with the following constraint matrix and vector:

$$\boldsymbol{A} = \begin{bmatrix} \beta_1 & \cdots & \beta_n \\ 1 & \cdots & 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

We imposed lower and upper position constraints of zero and one, respectively.

To produce the efficient frontier for an EAE portfolio with a beta of one and enhancement of ε, we used the quantities defined above, together with the constraint set defined in Equation (3.20) in a CLA program using the $2n$ representation.

The resulting efficient frontiers for $\varepsilon \in$ {10%, 20%, 30%, 40%, 50%} are shown in Figure 3.1. As expected, the figure shows that, from high to moderate portfolio risks, a larger enhancement provides a larger expected return. As the portfolio risk reduces, the efficient frontiers converge to one another.

The optimal level of enhancement is a function of risk tolerance. Ignoring upper and lower bounds on portfolio holdings for simplicity, the Lagrangian for an EAE portfolio can be written in the form

$$\mathcal{L} = \tfrac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Omega}_k \boldsymbol{x} - \tau \boldsymbol{\mu}^\top \boldsymbol{x} + \boldsymbol{\lambda}^\top (A\boldsymbol{x} - \boldsymbol{b} - \boldsymbol{c}\varepsilon).$$

Regarding ***x*, *λ***, and ε as variables to be controlled, and rearranging the first-order conditions for optimality of the Lagrangian gives

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \\ \epsilon \end{bmatrix} = M^{-1} \begin{bmatrix} \boldsymbol{\mu} \\ 0 \\ 0 \end{bmatrix} \tau + M^{-1} \begin{bmatrix} 0 \\ \boldsymbol{b} \\ 0 \end{bmatrix},$$

where

$$M = \begin{bmatrix} \boldsymbol{\Omega}_k & A^\top & \mathrm{O} \\ A & \mathrm{O} & -\boldsymbol{c} \\ 0 & -\boldsymbol{c}^\top & 0 \end{bmatrix}.$$

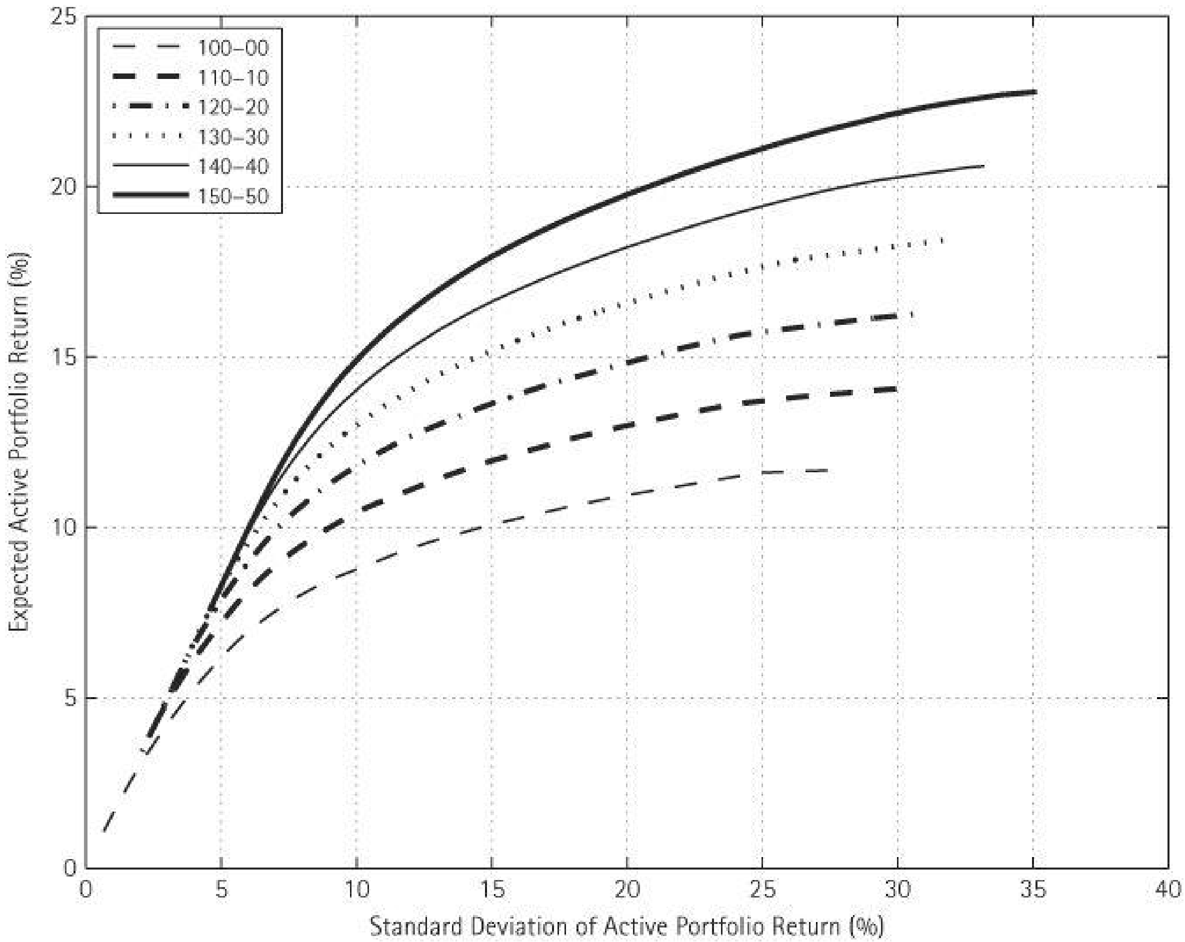

FIGURE 3.1 Efficient frontiers for EAE portfolios with 100 securities.

Therefore, in the simple case where bounds are ignored, enhancement ε is a linear function of risk tolerance τ. In the more realistic case where bounds are incorporated, ε will be a piecewise linear function of τ. As one's risk tolerance increases, so one's chosen enhancement should increase. Conversely, as one's risk tolerance decreases, so one's chosen enhancement should decrease.

At very low levels of risk tolerance ($\tau \approx 0$), the optimization approaches a variance minimization problem. For the long-only portfolio, the solution to this problem is to set the portfolio equal to the benchmark, at which point the active variance is zero. As higher levels of enhancement are imposed, the portfolio is forced away from the benchmark, and this introduces positive active variance.

3.5 CONCLUSION