Portfolio Optimization
Part 2 – Constrained Portfolios

John Norstad

j-norstad@northwestern.edu
http://www.norstad.org

March 28, 2005
Updated: November 3, 2011

Abstract

We develop the critical-line algorithm for solving portfolio optimization problems with low and high bound asset constraints.

Contents

List of Figures

1 Introduction

In [8] we developed the classical mean-variance optimization theory for unconstrained portfolios. In this theory, all possible combinations of asset weights are permitted, even negative weights (short-selling) and weights greater than 100% (leverage). The only restriction is the budget constraint, which says that the sum of the asset weights must equal 1.

In many situations additional external constraints are imposed on investors. For example, typical 401(k) and 403(b) plans do not make it possible to sell short or use leverage. In these kinds of cases we often wish to constrain all the asset weights to the range 0-100%. In other cases we may wish to impose other kinds of lower and upper bounds on individual assets.

In this paper we generalize the theory to handle these kinds of lower and upper bound constraints. Each asset may be unconstrained, have a lower bound but not an upper bound, have an upper bound but not a lower bound, or have both lower and upper bounds.

As in [8], we closely follow Sharpe's presentation in [10]. Our development of the algorithm, especially the proofs, is also guided by Markowitz [6, 7].

We use the notation of [8]. We have n assets. We are given a vector x of instantaneous expected returns α_i, a symmetric positive semidefinite matrix V of covariances $\rho_{i,j}$, and an iso-elastic coefficient of relative risk aversion A. The decision variable is the vector w of asset weights w_i. The problem is to maximize expected utility:

$$f(w) = w'x - \frac{1}{2}Aw'Vw$$

subject to the constraints:

$$\sum_{i=1}^{n} w_i = 1$$

$$L_i \leq w_i \leq H_i \quad \text{for } i = 1 \dots n$$

We require $L_i \leq H_i$, and we permit $L_i = -\infty$ and/or $H_i = +\infty$, which means that there is no low or high constraint respectively on the asset weight w_i.

Definition 1.1 *The* feasible set F *is the set of all vectors of asset weights* w *which satisfy the constraints:*

$$F = \left\{ w \mid \sum_{i=1}^{n} w_i = 1 \text{ and } L_i \leq w_i \leq H_i \right\}$$

We can state the problem more succinctly in terms of the feasible set F as follows:

$$\text{maximize } f(w) = w'x - \frac{1}{2}Aw'Vw \text{ over } w \in F$$

or even more simply:

$$\text{maximize } f \text{ over } F$$

Note that the objective function f is parameterized by the coefficient of relative risk aversion A. So we are really solving a whole family of maximization problems parameterized by A. Thus, when demanded by the context, we will often say:

$$\text{maximize } f \text{ over } F \text{ for } A$$

As in [8], we make the important assumption that the assets are linearly independent and that there are no arbitrage opportunities.

1.1 Outline of the Solution

In the unconstrained problem, as the coefficient of relative risk aversion A changes, asset weights for the efficient portfolio change smoothly. Some asset weights increase as A increases, and some decrease. We review the basic properties of the solution to the unconstrained problem in section 2.

In the constrained problem, as A changes, asset weights also change, and as an asset weight changes it may encounter its low or high bound. At this point, the asset weight becomes "pinned" to its low or high bound, and ceases to change as A changes. The reverse can also happen. As A changes, an asset weight that was formerly pinned to a bound may come alive and start to change.

For a particular efficient portfolio for a given value of A, each asset is in one of three states: It is pinned to its low bound, it is pinned to its high bound, or it is in between its low and high bounds. Following Sharpe [10], we call these three states *down*, *up*, and *in*.

The *down*, *in* and *up* states have a direct relationship with the first partial derivatives of the objective function $f(w)$ at the efficient portfolios which are solutions to the problem. This relationship is expressed by the critical *Kuhn-Tucker* conditions, which we discuss in detail in section 3.

A *segment* is a range of values of A over which the assets do not change their states. We define this notion in section 4 and show how to compute the efficient portfolios within a segment and the endpoints of a segment.

As A changes, when we encounter the end of a segment, at least one asset changes state. If we know the states of all of the assets in the first segment, we can compute what the states will be in the next segment. We show how to do this in section 5. This is called the *critical-line algorithm*. It was first discovered by Harry Markowitz.

If we know any one of the segments in a constrained solution, we can use the critical-line algorithm to find all the other segments and completely solve the problem for all values of A. Thus all that remains is to find some way to discover

an initial segment. We discuss how to do this in section 5.4, by solving a convex quadratic programming problem to find the minimum variance portfolio.

1.2 Notation

We will sometimes use the gradiant notation as a compact way to express equations involving partial derivatives of functions. The ∇ symbol ("nabla") is used to do this. It gathers all of the partial derivatives of a function into a vector. For example, suppose $f(w)$ is a function of n variables $w_1 \dots w_n$. Then:

$$\nabla f = \left(\frac{\partial f}{\partial w_1}, \dots, \frac{\partial f}{\partial w_n} \right)$$

and:

$$\nabla f(w) = \left(\frac{\partial f}{\partial w_1}(w), \dots, \frac{\partial f}{\partial w_n}(w) \right)$$

As an example, the following equation is a compact way of saying that all the first partial derivatives of f evaluated at some point w are greater than or equal to 0:

$$\nabla f(w) \geq 0$$

We take the liberty of sometimes using this notation to represent a column vector, and sometimes a row vector. The meaning will always be clear from the context.

We also use boldface $\mathbf{1}$ and $\mathbf{0}$ symbols to represent column or row vectors all of whose elements are 1 or 0 respectively. For example, if λ is a number, then $\lambda\mathbf{1}$ is a vector with each of its elements equal to λ, and if x is a vector, then $x\mathbf{1}$ is the sum of all the elements of x. When the context demands more clarity, we use c and r subscripts to denote column and row vector versions. E.g., $\mathbf{1}_c$ and $\mathbf{1}_r$.

As an example of these two notations combined, the following equation says that all the first partial derivatives of f evaluated at w are equal to λ:

$$\nabla f(w) = \lambda\mathbf{1}$$

We use the symbol $\Rightarrow$ for logical implication. $A \Rightarrow B$ means "if A then B."

Finally, we take the liberty of mildly overloading the prime symbol ($'$). In some contexts, we use the symbol to take the transpose of a vector, as in the definition of our objective function $f(w) = w'x - \frac{1}{2}Aw'Vw$. In other contexts, we use it to talk about pairs of vectors or numbers, as in "let s and s' be two state vectors."

2 Review of the Unconstrained Problem

Recall the solution to the unconstrained problem in [8].

The problem is to maximize expected utility:

$$f(w) = w'x - \frac{1}{2}Aw'Vw$$

subject to the budget constraint:

$$\sum_{i=1}^{n} w_i = 1$$

The first order partial derivatives of f are:

$$\frac{\partial f}{\partial w_i} = \alpha_i - A\sum_{j=1}^{n} \rho_{i,j} w_j \tag{1}$$

or, using gradiant notation:

$$\nabla f = x - AVw \tag{2}$$

To deal with the budget constraint, we introduce a Lagrange multiplier λ and a new objective function $\hat{f}$:

$$\hat{f}(w, \lambda) = f(w) + \lambda \left(1 - \sum_{i=1}^{n} w_i\right) \tag{3}$$

To solve the problem, we take the $n+1$ partial derivatives of $\hat{f}$ and set them equal to 0.

$$\frac{\partial \hat{f}}{\partial w_i} = \alpha_i - A\sum_{j=1}^{n} \rho_{i,j} w_j - \lambda = 0 \tag{4}$$

$$\frac{\partial \hat{f}}{\partial \lambda} = 1 - \sum_{i=1}^{n} w_i = 0 \tag{5}$$

Rewrite these equations as:

$$\sum_{j=1}^{n} \rho_{i,j} w_j + \lambda/A = \alpha_i/A \tag{6}$$

$$\sum_{i=1}^{n} w_i = 1 \tag{7}$$

This is a set of $n+1$ linear equations in $n+1$ unknowns which we can solve using linear algebra. Define vectors and matrices as follows:

$$\hat{V} = \begin{pmatrix} V & \mathbf{1}_c \\ \mathbf{1}_r & 0 \end{pmatrix} \quad \hat{w} = \begin{pmatrix} w \\ \lambda/A \end{pmatrix} \quad \hat{x} = \begin{pmatrix} x \\ 0 \end{pmatrix} \quad \hat{y} = \begin{pmatrix} \mathbf{0}_c \\ 1 \end{pmatrix}$$

Then equations (6) and (7) become:

$$\hat{V}\hat{w} = \frac{1}{A}\hat{x} + \hat{y} \tag{8}$$

Let:

$$\hat{c} = \hat{V}^{-1}\hat{x} \tag{9}$$

$$\hat{d} = \hat{V}^{-1}\hat{y} \tag{10}$$

Our solution is:

$$\hat{w} = \frac{1}{A}\hat{c} + \hat{d} \tag{11}$$

The solution vector w is the first n elements of $\hat{w}$. The Lagrange multiplier λ is:

$$\lambda = \hat{c}_{n+1} + A\hat{d}_{n+1} \tag{12}$$

We can arrive at this solution only if the enhanced covariance matrix $\hat{V}$ is non-singular, so that we can use its inverse $\hat{V}^{-1}$ in equations (9) and (10). In [8] we showed that this matrix is non-singular if there are no arbitrage opportunities or linearly dependent assets. We make this assumption throughout this paper.

Note that at a solution w, equation (4) says that:

$$\nabla f(w) = \lambda \mathbf{1} \tag{13}$$

The economic meaning of this equation is that efficient portfolios are always in equilibrium in the sense that the marginal utility of each asset is the same. The Lagrange multiplier λ is a measure of that marginal utility.

3 Kuhn-Tucker Conditions

3.1 Asset States and Slope Conditions

The objective function f we wish to maximize is a quadratic function of n variables. We have a number of constraints, including both the budget constraint and low and high bound constraints on each variable.

To help develop some insight into the solution, we begin by drastically simplifying the problem. Consider the trivial optimization problem where we have just one variable x, a quadratic objective function $f(x)$ we wish to maximize, no "budget constraint," and a single pair of low and high bound constraints L and H on the variable.

This problem is easy to solve. Figure 1 illustrates the solution. There are three cases.

The graph of the objective function is a simple parabola with a global maximum value at some point.

If the global maximum is less than the low bound, the solution is the low bound. This is the *down* case, the top graph in Figure 1. The slope of the parabola at the solution is negative in this case.

If the global maximum is between the low and high bounds, it is the solution. This is the *in* case, the middle graph in Figure 1. The slope of the parabola at the solution is zero in this case.

If the global maximum is greater than the high bound, the solution is the high bound. This is the *up* case, the bottom graph in Figure 1. The slope of the parabola at the solution is positive in this case.

The slope of the parabola is the first derivative of the objective function. In our simple problem we have the following conditions on this derivative at the solution point x:

$$f'(x) \begin{cases} \leq 0 & \text{if the state is down} \\ = 0 & \text{if the state is in} \\ \geq 0 & \text{if the state is up} \end{cases}$$

The intuition behind these conditions should be clear. If our solution is at the high bound, f must be increasing in value as we approach the high bound from below. If our solution is at the low bound, f must be increasing in value as we approach the low bound from above, which means it is decreasing in value if we approach the low bound from below. If our solution is in the middle somewhere, the slope must be zero.

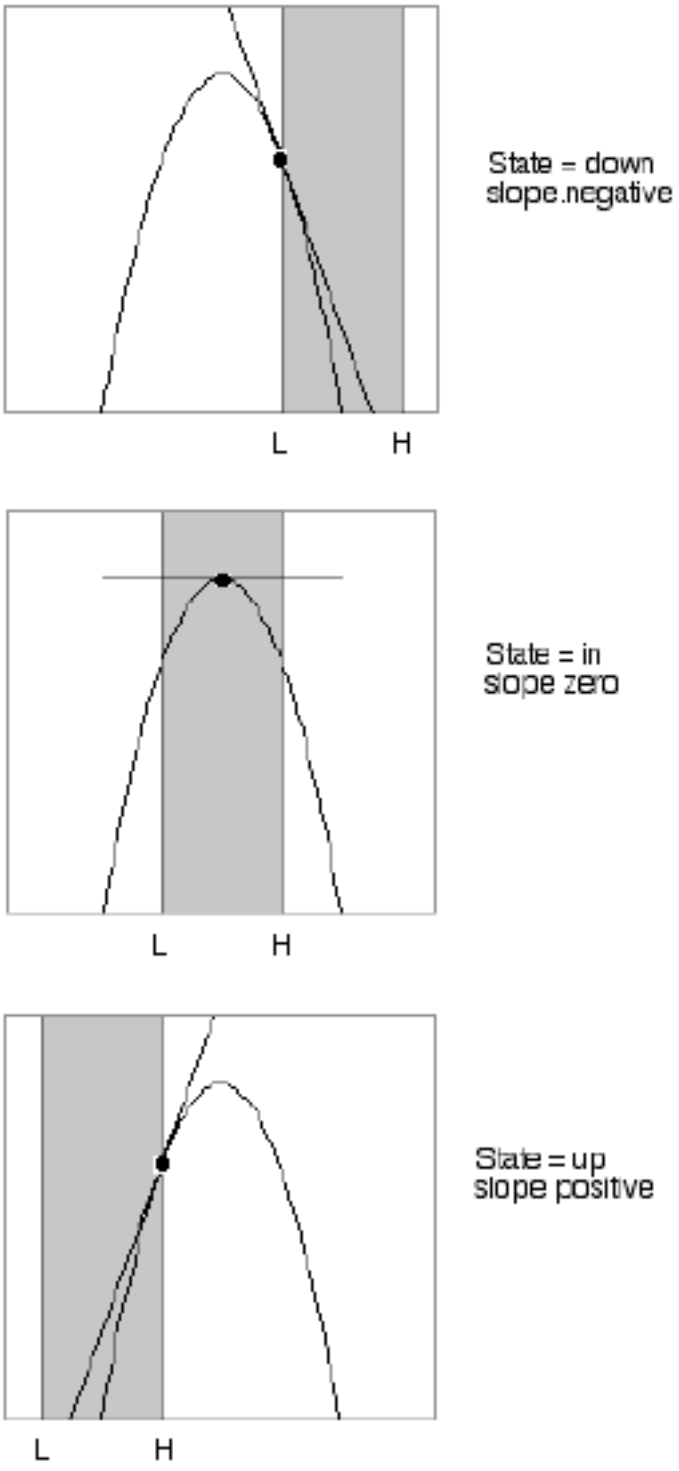


Figure 1: Asset States and Slope Conditions

For our full problem, we can expect that there will be some kind of similar conditions on the first order partial derivatives of our objective function which are related to the states of the individual assets. For example, our intuition tells us that if the solution has asset i at its high bound, the value of our objective function f must be increasing (or at least non-decreasing) as asset i approaches its high bound from below, which implies that the first partial derivative of f with respect to asset i must be non-negative, and so on.

In the next section we work out all the mathematical details to show that this is indeed the case, with a correction involving a little bit of extra complexity due to the budget constraint and the Lagrange multiplier λ we use to deal with that constraint.

3.2 Definitions and Theorems

Definition 3.1 *A* state vector *s is a vector of dimension n where each element of the vector has one of the three possible values $s_i = down$, up or in. Given such a state vector, define:*

$$\begin{aligned} D_s &= \{i \mid s_i = down\} \\ I_s &= \{i \mid s_i = in\} \\ U_s &= \{i \mid s_i = up\} \end{aligned}$$

We will omit the subscripts on D, I and U when the context is clear.

Definition 3.2 *Let s be a state vector, w a vector of asset weights, λ a Lagrange multiplier, and A a coefficient of relative risk aversion. The* Kuhn-Tucker *conditions for (s, w, λ, A) are:*

$$w \in F$$

$$w_i = \begin{cases} L_i & for\ all\ i \in D \\ H_i & for\ all\ i \in U \end{cases}$$

$$\frac{\partial f}{\partial w_i}(w) \begin{cases} \leq \lambda & for\ all\ i \in D \\ = \lambda & for\ all\ i \in I \\ \geq \lambda & for\ all\ i \in U \end{cases}$$

We will prove in a moment that the solutions to our maximization problem satisfy the Kuhn-Tucker conditions, and vice-versa. This is the multidimensional version of the intuition we developed in the previous section about slopes (first derivatives) and solutions to quadratic maximization problems with low and high bound constraints.

Note the clear economic meaning behind these conditions, similar to the economic meaning behind equation (13) in the unconstrained solution. For constrained efficient portfolios, the *in* assets all have the same marginal utility and

are in equilibrium, as in the unconstrained case. The *up* and *down* assets are not, however, in equilibrium. Assets that are *up* and pinned at their high bounds have greater or equal marginal utility, which means that we might be able to improve our portfolio (increase its utility) if the bound were not present. Similarly, *down* assets have lesser or equal marginal utility, which means that we might be able to improve our portfolio if we could go below the lower bounds.

We will need the following definition momentarily:

Definition 3.3 *A function f is* convex *if for all u and v in the domain of f and $0 \leq \alpha \leq 1$:*

$$f(\alpha u + (1-\alpha)v) \leq \alpha f(u) + (1-\alpha)f(v)$$

In order to prove our theorem, we need the fundamental *Kuhn-Tucker Theorem* from mathematical programming theory. This theorem generalizes the classical Lagrange multiplier theory to handle inequality constraints as well as equality constraints. We state the theorem here without proof:[1]

Theorem 3.1 *(Kuhn-Tucker) Let f be a convex continuously differentiable function, $h = (h_1 \dots h_m)$ a vector of linear functions, and $g = (g_1 \dots g_r)$ a vector of linear functions. Consider the problem:*

$$\textit{minimize } f(w) \textit{ subject to the constraints:}$$

$$h(w) = 0$$
$$g(w) \leq 0$$

w is a solution to the problem if and only if there exist Lagrange multipliers $\lambda = (\lambda_1 \dots \lambda_m)$ and $\mu = (\mu_1 \dots \mu_r)$ which satisfy:

$$\nabla f(w) + \lambda \nabla h(w) + \mu \nabla g(w) = 0$$
$$\mu \geq 0$$
$$\mu g(w) = 0$$

This theorem says a great deal using only a few symbols.

First note that there is no restriction on the signs of the Lagrange multipliers λ for the equality constraints, but all of the Lagrange multipliers μ for the inequality constraints are non-negative.

[1] The version of the Kuhn-Tucker theorem we state here is just one simplified form of the full theorem, a form that is sufficient for our needs. For more information on this important theorem and proofs of its various versions, see any good textbook on mathematical programming.

When we write the first two equations out in full we get:

$$\frac{\partial f}{\partial w_i}(w) + \sum_{j=1}^{m} \lambda_j \frac{\partial h_j}{\partial w_i}(w) + \sum_{k=1}^{r} \mu_k \frac{\partial g_k}{\partial w_i}(w) = 0 \quad \text{(for all } i = 1 \dots n)$$

$$\mu_k \geq 0 \quad \text{(for all } k = 1 \dots r)$$

Because $\mu_k \geq 0$ and $g_k(w) \leq 0$ for all k, the last equation $\mu g(w) = 0$ is a terse way of saying the following:

$$g_k(w) < 0 \Rightarrow \mu_k = 0 \quad \text{(for all } k = 1 \dots r)$$

In other words, we always know that for each inequality constraint k we have $\mu_k \geq 0$. In addition, if the solution w is not "at the constraint boundary" ($g_k(w) < 0$), then we also know that $\mu_k = 0$.

In our problem we are maximizing our objective function f. The Kuhn-Tucker theorem is stated in terms of minimizing functions. Maximizing f is clearly equivalent to minimizing $-f$.

This version of the Kuhn-Tucker theorem we are using requires that the objective function being minimized must be convex. This convexity condition is what makes it possible to state the conclusion with the "if and only if" clause. Without convexity, the "only if" part of the theorem still holds, but not the "if" part.

It is easy to show that $g = -f$ is convex, because the covariance matrix V is positive semidefinite. Suppose u and v are vectors and $0 \leq \alpha \leq 1$. Then:

$$\begin{aligned}
& \alpha g(u) + (1-\alpha) g(v) - g(\alpha u + (1-\alpha)v) \\
= \;& -\alpha f(u) - (1-\alpha) f(v) + f(\alpha u + (1-\alpha)v) \\
= \;& -\alpha \left[u'x - \frac{1}{2} A u'Vu \right] - (1-\alpha) \left[v'x - \frac{1}{2} A v'Vv \right] + \\
& (\alpha u + (1-\alpha)v)'x - \frac{1}{2} A (\alpha u + (1-\alpha)v)'V(\alpha u + (1-\alpha)v) \\
= \;& \alpha \frac{1}{2} A u'Vu + (1-\alpha) \frac{1}{2} A v'Vv - \frac{1}{2} A (\alpha u + (1-\alpha)v)'V(\alpha u + (1-\alpha)v) \\
= \;& \frac{1}{2} A \left[\alpha u'Vu + (1-\alpha) v'Vv - \alpha^2 u'Vu - (1-\alpha)^2 v'Vv - 2\alpha(1-\alpha) u'Vv \right] \\
= \;& \frac{1}{2} A \alpha \left[u'Vu + v'Vv - \alpha u'Vu - \alpha v'Vv - 2(1-\alpha) u'Vv \right] \\
= \;& \frac{1}{2} A \alpha (1-\alpha) \left[u'Vu + v'Vv - 2u'Vv \right] \\
= \;& \frac{1}{2} A \alpha (1-\alpha) \left[(u-v)'V(u-v) \right] \\
\geq \;& 0 \quad \text{because } A > 0,\ \alpha \geq 0,\ (1-\alpha) \geq 0 \text{ and } V \text{ is positive semidefinite}
\end{aligned}$$

We can now prove the main theorem of the paper. This theorem is what makes the critical-line algorithm work.

Theorem 3.2 *w maximizes f over F for A iff there is a state vector s and a Lagrange multiplier λ such that (s, w, λ, A) satisfies the Kuhn-Tucker conditions.*

Proof:

First note that w maximizes f over F for A iff w minimizes $-f$ for A subject to the following constraints:

$$\begin{aligned} h(w) &= \sum_{i=1}^{n} w_i - 1 = 0 \\ g_k^L(w) &= L_k - w_k \leq 0 \quad (\text{for } k = 1 \dots n,\ L_k \neq -\infty) \\ g_k^H(w) &= w_k - H_k \leq 0 \quad (\text{for } k = 1 \dots n,\ H_k \neq +\infty) \end{aligned}$$

By the Kuhn-Tucker theorem, w maximizes f over F for A iff there exist Lagrange multipliers λ, $\mu_k^L \geq 0$ and $\mu_k^H \geq 0$ which satisfy:

$$-\frac{\partial f}{\partial w_i}(w) + \lambda \frac{\partial h}{\partial w_i}(w) + \sum_{k=1}^{n} \mu_k^L \frac{\partial g_k^L}{\partial w_i}(w) + \sum_{k=1}^{n} \mu_k^H \frac{\partial g_k^H}{\partial w_i}(w) = 0 \quad (\text{for all } i = 1 \dots n)$$

$$\begin{aligned} g_k^L(w) < 0 &\Rightarrow \mu_k^L = 0 \\ g_k^H(w) < 0 &\Rightarrow \mu_k^H = 0 \end{aligned}$$

Evaluate the partial derivatives of h, g_k^L and g_k^H and rearrange and simplify. Because our constraint functions are so simple, most of the partial derivatives are 0, so their terms drop out and we get:

$$\begin{aligned} \frac{\partial f}{\partial w_i}(w) - \lambda &= \mu_i^H - \mu_i^L \quad (\text{for all } i = 1 \dots n) \\ L_i < w_i &\Rightarrow \mu_i^L = 0 \\ w_i < H_i &\Rightarrow \mu_i^H = 0 \end{aligned} \tag{14}$$

Note that using gradiant notation we can write the first equation above as:

$$\nabla f(w) - \lambda \mathbf{1} = \mu^H - \mu^L$$

Suppose w maximizes f over F for A. Then by equations (14) we have:

$$\frac{\partial f}{\partial w_i}(w) - \lambda = \begin{cases} -\mu_i^L \leq 0 & \text{if } L_i = w_i < H_i \\ 0 & \text{if } L_i < w_i < H_i \\ \mu_i^H \geq 0 & \text{if } L_i < w_i = H_i \\ \mu_i^H - \mu_i^L & \text{if } L_i = w_i = H_i \end{cases}$$

Define the state vector s as follows:

$$s_i = \begin{cases} down & \text{if } L_i = w_i < H_i \\ down & \text{if } L_i = w_i = H_i \text{ and } \mu_i^H \leq \mu_i^L \\ in & \text{if } L_i < w_i < H_i \\ up & \text{if } L_i < w_i = H_i \\ up & \text{if } L_i = w_i = H_i \text{ and } \mu_i^H > \mu_i^l \end{cases}$$

Then (s, w, λ, A) satisfies the Kuhn-Tucker conditions.

Conversely, suppose (s, w, λ, A) satisfies the Kuhn-Tucker conditions. Define:

$$\mu_i^L = \begin{cases} \lambda - \dfrac{\partial f}{\partial w_i}(w) & \text{if } i \in D \\ 0 & \text{otherwise} \end{cases}$$

$$\mu_i^H = \begin{cases} \dfrac{\partial f}{\partial w_i}(w) - \lambda & \text{if } i \in U \\ 0 & \text{otherwise} \end{cases}$$

Then $\mu^L \geq 0$, $\mu^H \geq 0$ and equations (14) are satisfied, so w maximizes f over F for A.

For computational reasons that will become clear in the following sections, we need to work with state vectors that always have at least one *in* state. So we prove the following simple extension of Theorem 3.2.

Theorem 3.3 *w maximizes f over F for A iff there is a state vector s with at least one* in *state and a Lagrange multiplier λ such that (s, w, λ, A) satisfies the Kuhn-Tucker conditions.*

Proof:

The "if" direction is immediate. To prove the "only if" direction, start with the state vector s and Lagrange multiplier λ we derived in the proof of Theorem 3.2. Suppose that s has only *down* and *up* elements. We know that:

$$\frac{\partial f}{\partial w_i}(w) \begin{cases} \leq \lambda & \text{if } i \in D \\ \geq \lambda & \text{if } i \in U \end{cases}$$

Define:

$$\lambda' = \begin{cases} \max\limits_{i \in D} \left[\dfrac{\partial f}{\partial w_i}(w) \right] & \text{if } D \neq \{\} \\ \min\limits_{i \in U} \left[\dfrac{\partial f}{\partial w_i}(w) \right] & \text{if } D = \{\} \end{cases}$$

$$s'_i = \begin{cases} s_i & \text{if } \dfrac{\partial f}{\partial w_i}(w) \neq \lambda' \\ in & \text{if } \dfrac{\partial f}{\partial w_i}(w) = \lambda' \end{cases}$$

Then (s', w, λ', A) satisfies the Kuhn-Tucker conditions, and s' has at least one *in* state, and our proof is complete.

Note that in everything we have done so far, the coefficient of relative risk aversion A has been held constant. We want to find a solution for the whole family of problems, not just one of them, so we will soon start to vary A over its domain $(0, \infty)$. In particular, we will be looking at what happens to the Kuhn-Tucker conditions as A varies. For this purpose, the following definitions will prove useful.

Definition 3.4 *A state vector s is* valid for *a coefficient of relative risk aversion A if it contains at least one* in *asset and there is some $w \in F$ and λ for which (s, w, λ, A) satisfies the Kuhn-Tucker conditions.*

Definition 3.5 *A state vector s is* valid *if it is valid for at least one coefficient of relative risk aversion A.*

4 Solution Segments

4.1 Computing a Solution

Suppose a state vector s is valid for a coefficient of relative risk aversion A. Then it contains at least one *in* asset, and there is a vector of asset weights $w \in F$ and a Lagrange multiplier λ for which the Kuhn-Tucker conditions are satisfied for A. w is the efficient portfolio for A.

The Kuhn-Tucker condition for $i \in I$ is:

$$\frac{\partial f}{\partial w_i} = \lambda$$

Substituting equation (1) for $\frac{\partial f}{\partial w_i}$ gives:

$$\alpha_i - A \sum_{j=1}^{n} \rho_{i,j} w_j = \lambda$$

or after rearranging slightly:

$$\sum_{j=1}^{n} \rho_{i,j} w_j + \lambda / A = \alpha_i / A$$

Thus the Kuhn-Tucker conditions imply that w must be a solution to the following set of linear equations:

$$\begin{aligned} &\text{For } i \in D: && w_i = L_i \\ &\text{For } i \in I: && \sum_{j=1}^{n} \rho_{i,j} w_j + \lambda / A = \alpha_i / A \\ &\text{For } i \in U: && w_i = H_i \\ & && \sum_{i=1}^{n} w_i = 1 \end{aligned} \tag{15}$$

Consider this problem in reverse. Suppose we are given the state vector s with at least one *in* state and a value of A. We can easily solve the simultaneous linear equations above to get the values of w and λ and check the Kuhn-Tucker conditions to see if we have a solution (that is, to see if s is valid for A).

We use matrix algebra. The computations are similar to the ones for the unconstrained problem in section 2.

We start by building an enhanced covariance matrix $\tilde{V}$. This is the same as the $\hat{V}$ matrix for unconstrained problems except for the rows corresponding to

assets in the *down* and *up* states.

$$\tilde{V} = \begin{pmatrix} \tilde{v}_{1,1} & \cdots & \tilde{v}_{1,n} & \tilde{v}_{1,n+1} \\ \vdots & & \vdots & \vdots \\ \tilde{v}_{n,1} & \cdots & \tilde{v}_{n,n} & \tilde{v}_{n,n+1} \\ 1 & \cdots & 1 & 0 \end{pmatrix}$$

For assets i in the *in* state, we set row i to the covariances for asset i, plus a 1 in the last column, as in $\hat{V}$. For assets i in the *down* and *up* states, we set row i to a vector with 1 in column i and 0 everywhere else:

$$\tilde{v}_{i,j} = \begin{cases} \rho_{i,j} & \text{if } i \in I \text{ and } j \leq n \\ 1 & \text{if } i \in I \text{ and } j = n+1 \\ 0 & \text{if } i \in D \cup U \text{ and } i \neq j \\ 1 & \text{if } i \in D \cup U \text{ and } i = j \end{cases}$$

Now define the three column vectors $\tilde{w}$, $\tilde{x}$, and $\tilde{y}$:

$$\tilde{w} = \begin{pmatrix} w_1 \\ \vdots \\ w_n \\ \lambda/A \end{pmatrix}$$

$$\tilde{x}_i = \begin{cases} \alpha_i & \text{if } i \in I,\ i \leq n \\ 0 & \text{if } i \in D \cup U,\ i \leq n \\ 0 & \text{if } i = n+1 \end{cases}$$

$$\tilde{y}_i = \begin{cases} 0 & \text{if } i \in I,\ i \leq n \\ L_i & \text{if } i \in D,\ i \leq n \\ H_i & \text{if } i \in U,\ i \leq n \\ 1 & \text{if } i = n+1 \end{cases}$$

The linear equations become:

$$\tilde{V}\tilde{w} = \frac{1}{A}\tilde{x} + \tilde{y} \tag{16}$$

with the solution:

$$w_i = \frac{1}{A}\tilde{c}_i + \tilde{d}_i \tag{17}$$

$$\lambda = \tilde{c}_{n+1} + A\tilde{d}_{n+1} \tag{18}$$

where:[2]

$$\tilde{c} = \tilde{V}^{-1}\tilde{x} \tag{19}$$

$$\tilde{d} = \tilde{V}^{-1}\tilde{y} \tag{20}$$

[2]We show in the next section that $\tilde{V}$ is non-singular, so it has an inverse $\tilde{V}^{-1}$.

In order to be a solution to the problem we must have $w \in F$. That is, w must satisfy all of the constraints. w satisfies the budget constraint because that is one of the equations we solved. For a *down* or *up* asset, w also satisfies the low and high bound constraints, because our equations pin the asset weight to the low or high bound. For an *in* asset, we can inspect the solution vector w to see if the asset weight is indeed between the low and high bounds.

To check the Kuhn-Tucker conditions on the first order partial derivatives we start with equation (1):

$$\frac{\partial f}{\partial w_i} = \alpha_i - A \sum_{j=1}^{n} \rho_{i,j} w_j \tag{21}$$

Substitute equation (17) into equation (21) to get:

$$\begin{aligned} \frac{\partial f}{\partial w_i} &= \alpha_i - A \sum_{j=1}^{n} \rho_{i,j} \left(\frac{1}{A} \tilde{c}_j + \tilde{d}_j \right) && (22) \\ &= \alpha_i - \sum_{j=1}^{n} \rho_{i,j} \tilde{c}_j - A \sum_{j-1}^{n} \rho_{i,j} \tilde{d}_j && (23) \end{aligned}$$

Subtract λ from both sides and substitute equation (18) to get:

$$\begin{aligned} \frac{\partial f}{\partial w_i} - \lambda &= \alpha_i - \sum_{j=1}^{n} \rho_{i,j} \tilde{c}_j - A \sum_{j-1}^{n} \rho_{i,j} \tilde{d}_j - \lambda && (24) \\ &= \left(\alpha_i - \sum_{j=1}^{n} \rho_{i,j} \tilde{c}_j - \tilde{c}_{n+1} \right) - A \left(\sum_{j-1}^{n} \rho_{i,j} \tilde{d}_j + \tilde{d}_{n+1} \right) && (25) \end{aligned}$$

Compute new vectors $\tilde{e}$ and $\tilde{f}$ as follows:

$$\begin{aligned} \tilde{e}_i &= \alpha_i - \sum_{j=1}^{n} \rho_{i,j} \tilde{c}_j - \tilde{c}_{n+1} && (26) \\ \tilde{f}_i &= \sum_{j=1}^{n} \rho_{i,j} \tilde{d}_j + \tilde{d}_{n+1} && (27) \end{aligned}$$

Then:

$$\frac{\partial f}{\partial w_i} - \lambda = \tilde{e}_i - A \tilde{f}_i \tag{28}$$

and the Kuhn-Tucker conditions on the first-order partial derivatives become:

$$\tilde{e}_i - A \tilde{f}_i \quad \begin{cases} \leq 0 & \text{for all } i \in D \\ = 0 & \text{for all } i \in I \\ \geq 0 & \text{for all } i \in U \end{cases} \tag{29}$$

Equations (19) and (20) can be written as:

$$\begin{aligned} \tilde{V}\tilde{c} &= \tilde{x} \\ \tilde{V}\tilde{d} &= \tilde{y} \end{aligned}$$

Multiplying out row i of these equations for $i \leq n$ shows that:

$$\begin{aligned} &\text{For } i \in D: \quad \tilde{c}_i = 0 \text{ and } \tilde{d}_i = L_i \\ &\text{For } i \in I: \quad \tilde{e}_i = 0 \text{ and } \tilde{f}_i = 0 \\ &\text{For } i \in U: \quad \tilde{c}_i = 0 \text{ and } \tilde{d}_i = H_i \end{aligned}$$

Note that for $i \in I$, the equations guarantee that $\tilde{e}_i - A\tilde{f}_i = 0$, so we only need to check the conditions for $i \in D \cup U$ in equation (29).

The vectors $\tilde{e}$ and $\tilde{f}$ can be computed using matrix algebra using the unconstrained enhanced covariance matrix $\hat{V}$ and the unconstrained enhanced expected return vector $\hat{x}$ that we defined in section 2:

$$\tilde{e} = \hat{x} - \hat{V}\tilde{c}$$

$$\tilde{f} = \hat{V}\tilde{d}$$

Using gradiant notation, we have:

$$\begin{aligned} \nabla f - \lambda \mathbf{1} &= \tilde{e} - A\tilde{f} \\ &= \hat{x} - \hat{V}\tilde{c} - A\hat{V}\tilde{d} \\ &= \hat{x} - A\hat{V}\left(\frac{1}{A}\tilde{c} + \tilde{d}\right) \\ &= \hat{x} - A\hat{V}\tilde{w} \end{aligned}$$

The following theorem summarizes the results of this section.

Theorem 4.1 *If (w, λ) is a solution to the simultaneous linear equations (15), with the vectors $\tilde{c}$, $\tilde{d}$, $\tilde{e}$, $\tilde{f}$ computed by the equations above, then w maximizes f over F for A iff the following subset of the Kuhn-Tucker conditions are satisfied:*

$$\begin{aligned} &\textit{For } i \in D: \quad \tilde{e}_i - A\tilde{f}_i \leq 0 \\ &\textit{For } i \in I: \quad L_i \leq \frac{1}{A}\tilde{c}_i + \tilde{d}_i \leq H_i \\ &\textit{For } i \in U: \quad \tilde{e}_i - A\tilde{f} \geq 0 \end{aligned}$$

Proof:

We showed above that the simultaneous linear equations force all of the other Kuhn-Tucker conditions to be satisfied.

4.2 Non-Singularity and Uniqueness

We always work with state vectors that have at least one *in* asset. This restriction is required to make the enhanced covariance matrix $\tilde{V}$ non-singular. It there are only *down* and *up* assets, the linear equations are all of the form $w_i = L_i$ or $w_i = H_i$, plus the budget constraint equation. None of these equations involve λ, and there is either no solution to the equations (if the budget constraint is not met), or there are an infinite number of solutions where λ can be assigned any value at all (if the budget constraint is met).

Under the assumption that the full enhanced covariance matrix $\hat{V}$ is non-singular and that a state vector s contains at least one *in* asset, it is easy to show that $\tilde{V}$ must be non-singular, and hence that the equations have a unique solution.

Suppose $\tilde{V}$ is singular. Then there is a vector $\tilde{w} \neq 0$ with $\tilde{V}\tilde{w} = 0$. By the way we construct $\tilde{V}$, we must have $\tilde{w}_i = 0$ for all $i \in D \cup U$. But then it is easy to see that we must also have $\hat{V}\tilde{w} = 0$ for the same vector $\tilde{w} \neq 0$. This violates our assumption that $\hat{V}$ is non-singular.

The economic meaning of this argument is clear. If $\tilde{V}$ is singular, the *in* assets of s are either linearly dependent or can be used to form a zero-budget arbitrage portfolio. The same set of assets are obviously linearly dependent or can form a zero-budget arbitrage portfolio for the full unconstrained problem.

Because $\tilde{V}$ is non-singular, the simultaneous linear equations (15) always have a unique solution.

4.3 Computing the Endpoints of a Segment

Given a value of A and a state vector s, we now know how to compute the constrained solution for those values and check it for validity using the Kuhn-Tucker conditions.

Suppose we have a valid solution that satisfies the Kuhn-Tucker conditions for some A. For what other values of A is it valid? There is some range of values around A which define the full segment over which the solution has the same asset states. We need to be able to compute the endpoints of that segment.

Consider what happens as we let A decrease or increase in value and the asset weights in the efficient portfolio solution vector $\tilde{w}$ change. One of two things may happen (or neither of them may happen). First, the efficient portfolio may contain some asset in the *in* state which reaches its lower or upper bound. If this happens we have reached an endpoint, because the constraints prohibit us from permitting an asset to go below its lower bound or above its upper bound. Second, the Kuhn-Tucker conditions may become invalid for one of the assets in the *down* or *up* state—that is, for some $i \in D \cup U$, the equation (29) may change sign. If this happens we have also reached an endpoint, because we cannot go

past the point where the Kuhn-Tucker conditions are no longer satisfied.

It is laborious but not difficult to compute the values of the endpoints.

We restrict our attention to one asset at a time and compute two new vectors p and q. For each i, p_i and q_i are the minimum and maximum values of A for which the Kuhn-Tucker conditions are satisfied for asset i. Thus the interval $[p_i, q_i]$ gives the range of values for A over which all the conditions for asset i remain satisfied.

We use our two equations (17) and (28) for this purpose:

$$w_i = \frac{1}{A}\tilde{c}_i + \tilde{d}_i \tag{30}$$

$$\frac{\partial f}{\partial w_i} - \lambda = \tilde{e}_i - A\tilde{f}_i \tag{31}$$

First suppose that asset i is in the *down* state. It is pinned to its lower bound. We need to determine the range over which the Kuhn-Tucker value in (31) is ≤ 0. There are four cases:

$$[p_i, q_i] = \begin{cases} [0, \infty] & \text{if } \tilde{f}_i = 0 \text{ and } \tilde{e}_i \leq 0 \\ [0, \tilde{e}_i/\tilde{f}_i] & \text{if } \tilde{f}_i < 0 \\ [\tilde{e}_i/\tilde{f}_i, \infty] & \text{if } \tilde{f}_i > 0 \\ [0, -1] & \text{otherwise} \end{cases} \tag{32}$$

Note the last case above, which we have recorded as $[0, -1]$. This notation indicates that there are no values of A for which the Kuhn-Tucker conditions are satisfied for this asset. If this case holds, then the state vector s we are considering is not valid.

Now suppose that asset i is in the *up* state. It is pinned to its upper bound. We need to determine the range over which the Kuhn-Tucker value in (31) is ≥ 0. Again there are four cases:

$$[p_i, q_i] = \begin{cases} [0, \infty] & \text{if } \tilde{f}_i = 0 \text{ and } \tilde{e}_i \geq 0 \\ [0, \tilde{e}_i/\tilde{f}_i] & \text{if } \tilde{f}_i > 0 \\ [\tilde{e}_i/\tilde{f}_i, \infty] & \text{if } \tilde{f}_i < 0 \\ [0, -1] & \text{otherwise} \end{cases} \tag{33}$$

Finally, suppose that asset i is in the *in* state. It is between its upper and lower bounds. We need to determine the range over which its value as given by equation (30) remains within its bounds. That is, we need to determine the range of A values over which the following inequality is true:

$$L_i \leq \frac{1}{A}\tilde{c}_i + \tilde{d}_i \leq H_i$$

There are two inequalities here which we will deal with separately:

$$A(L_i - \tilde{d}_i) \leq \tilde{c}_i \tag{34}$$

$$A(H_i - \tilde{d}_i) \geq \tilde{c}_i \tag{35}$$

For the first inequality (34) we have:

$$[p_i^L, q_i^L] = \begin{cases} [0, \infty] & \text{if } L_i = \tilde{d}_i \text{ and } \tilde{c}_i \geq 0 \\ [\tilde{c}_i/(L_i - \tilde{d}_i), \infty] & \text{if } L_i < \tilde{d}_i \\ [0, \tilde{c}_i/(L_i - \tilde{d}_i)] & \text{if } L_i > \tilde{d}_i \\ [0, -1] & \text{otherwise} \end{cases} \tag{36}$$

For the second inequality (35) we have:

$$[p_i^H, q_i^H] = \begin{cases} [0, \infty] & \text{if } H_i = \tilde{d}_i \text{ and } \tilde{c}_i \leq 0 \\ [\tilde{c}_i/(H_i - \tilde{d}_i), \infty] & \text{if } H_i > \tilde{d}_i \\ [0, \tilde{c}_i/(H_i - \tilde{d}_i)] & \text{if } H_i < \tilde{d}_i \\ [0, -1] & \text{otherwise} \end{cases} \tag{37}$$

We then set:

$$p_i = \max(p_i^L, p_i^H) \tag{38}$$

$$q_i = \min(q_i^L, q_i^H) \tag{39}$$

$$[p_i, q_i] = [p_i^L, q_i^L] \ \cap \ [p_i^H, q_i^H] \tag{40}$$

The range of values of A for which this vector of asset states s is valid is given by:

$$A^s_{min} = \max(p_i)$$

$$A^s_{max} = \min(q_i)$$

$$[A^s_{min}, A^s_{max}] = \textstyle\bigcap_{i=1}^{n} [p_i, q_i]$$

The state vector s is valid iff $A^s_{min} \leq A^s_{max}$.

We can visualize these computations graphically, which sometimes helps. The intervals $[p_i, q_i]$ and $[A^s_{min}, A^s_{max}]$ are computed based on the two equations (30) and (31):

$$w_i = \frac{1}{A}\tilde{c}_i + \tilde{d}_i \tag{41}$$

$$\frac{\partial f}{\partial w_i} - \lambda = \tilde{e}_i - A\tilde{f}_i \tag{42}$$

For a given state vector s, the solution vectors $\tilde{c}$, $\tilde{d}$, $\tilde{e}$ and $\tilde{f}$ are constants that are independent of A. So the right hand sides of these two equations define functions of A that are in a simple form.

Think of these equations graphically, where A is graphed on the x axis. Each equation plots as a curve on the graph. For *down* and *up* assets, we care about the second equation for the partial derivative minus λ, which graphs as a straight line, and we are concerned about where that line crosses the x axis (where it changes sign from ≤ 0 to ≥ 0 or vice-versa). For *in* assets, we care about the

first equation for the asset weights, which graphs as a hyperbola, and we are concerned about where that curve crosses the horizontal constraint lines $w_i = L_i$ and $w_i = H_i$.

Consider a particular point A which is valid for s. Some of the curves intersect their horizontal constraint lines to the left of A, some of them intersect to the right of A, and some may not intersect at all. The intersection points to the left of A are our values p_i, p_i^L, and p_i^H. The intersection points to the right of A are our values q_i, q_i^L, and q_i^H. As we move left or right from A, the first intersection point we encounter is the earliest point where one of our constraint equations becomes invalid. These points are our values A^s_{min} and A^s_{max}.

Theorem 4.2 *A state vector s is valid for A iff $A^s_{min} \leq A \leq A^s_{max}$.*

Proof:

By the way we constructed (defined) the interval $[A^s_{min}, A^s_{max}]$, the Kuhn-Tucker conditions have a solution for A iff $A^s_{min} \leq A \leq A^s_{max}$.

4.4 Equality Constraints and Overlapping Segments

The Kuhn-Tucker conditions in Definition 3.2 impose two sets of conditions on solutions to the problem of maximizing f over F. One set of conditions imposes equality constraints on the *down* and *up* assets. The other set of conditions imposes constraints on the first partial derivatives of f at the solution.

What happens if we only consider the first set of equality constraints without the first partial derivative constraints? The problem certainly becomes simpler, almost trivial, and it turns out that the answer is important.[3]

Definition 4.1 *For a state vector s, the* equality constraints *for s are:*

$$\sum_{i=1}^{n} w_i = 1$$
$$w_i = L_i \quad \textit{for all } i \in D$$
$$w_i = H_i \quad \textit{for all } i \in U$$

$E(s)$ is the set of all asset weight vectors that satisfy these constraints:

$$E(s) = \left\{ w \mid \sum_{i=1}^{n} w_i = 1 \textit{ and } w_i = L_i \textit{ for all } i \in D \textit{ and } w_i = H_i \textit{ for all } i \in U \right\}$$

[3]Lemma 4.4 in this section is a critical element of the proof we will give in section 5.3 that the critical-line algorithm terminates.

Lemma 4.1 *For a state vector s, $w \in E(s)$ maximizes f over $E(s)$ for A iff there exists a λ such that:*

$$\frac{\partial f}{\partial w_i}(w) = \lambda \quad \textit{for all } i \in I$$

Proof:

$w \in E(s)$ maximizes f over $E(s)$ for A iff it minimizes $-f$ subject to:

$$h(w) = \sum_{i=1}^{n} w_i - 1 = 0$$
$$h_i(w) = w_i - L_i = 0 \quad \text{for } i \in D$$
$$h_i(w) = w_i - H_i = 0 \quad \text{for } i \in U$$

By the Kuhn-Tucker Theorem 3.1,[4] $w \in E(s)$ maximizes f over $E(s)$ for A iff there exist Lagrange multipliers λ and λ_i for $i \in D \cup U$ which satisfy:

$$-\frac{\partial f}{\partial w_i}(w) + \lambda \frac{\partial h}{\partial w_i}(w) + \sum_{j \in D \cup U} \lambda_j \frac{\partial h_j}{\partial w_i}(w) = 0 \quad \text{for all } i = 1 \ldots n$$

Evaluate the partial derivatives of h and h_j and rearrange and simplify to get:

$$\frac{\partial f}{\partial w_i}(w) = \begin{cases} \lambda & \text{if } i \in I \\ \lambda + \lambda_i & \text{if } i \in D \cup U \end{cases} \tag{43}$$

The only if direction of the Lemma is now immediate. For the if direction, suppose there exists a λ such that:

$$\frac{\partial f}{\partial w_i}(w) = \lambda \quad \text{for all } i \in I$$

For $i \in D \cup U$ define:

$$\lambda_i = \frac{\partial f}{\partial w_i}(w) - \lambda$$

Then equation (43) is satisfied, so w maximizes f over $E(s)$ for A.

[4] This is a rather trivial application of the Kuhn-Tucker theorem, because we have no inequality constraints. This is really just classical Lagrange multiplier theory.

Lemma 4.2 *w maximizes f over $E(s)$ for A iff there is a λ such that (w, λ) satisfies the simultaneous linear equations (15) for s and A.*

Proof:

Suppose w maximizes f over $E(s)$ for A. By Definition 4.1 and Lemma 4.1, there is a λ such that all of the following equations are satisfied:

$$\begin{aligned}
&\text{For } i \in D: \quad w_i = L_i \\
&\text{For } i \in I: \quad \frac{\partial f}{\partial w_i}(w) = \alpha_i - A\sum_{j=1}^{n} \rho_{i,j} w_j = \lambda \\
&\text{For } i \in U: \quad w_i = H_i \\
&\qquad\qquad \sum_{i=1}^{n} w_i = 1
\end{aligned}$$

This is the same as the set of equations (15).

Conversely, if (w, λ) satisfies the equations (15), then we have $w \in E(s)$, and Lemma 4.1 says that w maximizes f over $E(s)$ for A.

Lemma 4.3 *If s is a valid state vector for A with solution w, then w uniquely maximizes f over $E(s)$ for A.*

Proof:

Suppose s is valid for A with solution w. Then for some λ, (w, λ) is a solution to the simultaneous linear equations (15). By Lemma 4.2, w maximizes f over $E(s)$. If any other asset weight vector w' maximizes f over $E(s)$, Lemma 4.2 tells us that (w', λ') must also be solution to equations (15) for some λ'. In section 4.2, we showed that these equations always have a unique solution, so $w = w'$ and w uniquely maximizes f over $E(s)$ for A.

What happens when solution segments overlap? That is, suppose the solution segments for two state vectors s and s' are both valid for some coefficient of relative risk aversion A. Each state vector provides a solution to the problem. Are the solutions the same? In other words, are solutions unique?

Our suspicion, of course, is that the answer must be "yes." This is not immediately obvious, however. We can prove that the solutions are indeed the same, provided s and s' satisfy some conditions that make them in a certain sense "close" to each other.

We will need this result as a crucial element of our argument in section 5.3 that the critical-line algorithm terminates.

Lemma 4.4 *Suppose s is a valid state vector for A with solution (w, λ), and s' is another valid state vector for A with solution (w', λ'), with:*

$$\begin{aligned} D &= \{i \mid s_i = \ down\} \\ I &= \{i \mid s_i = \ in\} \\ U &= \{i \mid s_i = \ up\} \end{aligned}$$

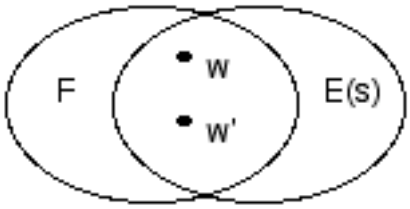
$$\begin{aligned} D' &= \{i \mid s'_i = \ down\} \\ I' &= \{i \mid s'_i = \ in\} \\ U' &= \{i \mid s'_i = \ up\} \end{aligned}$$

If $D \cup U \subset D' \cup U'$ and $I \cap I' \neq \{\}$ then $w = w'$ and $\lambda = \lambda'$.

Proof:

Let E(s) be the set of equality constraints for s in Definition 4.1.

The following Venn diagram helps illustrate the proof:



Because w and w' are both solutions, $w \in F$ and $w' \in F$.

$w \in E(s)$ by the definition of $E(s)$.

Because $D \cup U \subset D' \cup U'$, $w' \in E(s)$.

So both w and $w' \in F \cap E(s)$.

Because w' is a solution for A, it maximizes f over F for A. w' therefore must maximize f over $F \cap E(s)$ for A.

By Lemma 4.3, w uniquely maximizes f over $E(s)$ for A. w therefore must uniquely maximize f over $F \cap E(s)$ for A.

So we must have $w = w'$.

Because $I \cap I' \neq \{\}$, w and w' share some asset k with state *in*. By the Kuhn-Tucker conditions for s and s' for asset k, we have:

$$\lambda = \frac{\partial f}{\partial w_k}(w) = \frac{\partial f}{\partial w_k}(w') = \lambda'$$

4.5 A Brute Force Solution

We have now accumulated enough understanding of the problem to formulate a complete solution. The solution uses brute force.

The number of possible state vectors s is finite, albeit potentially large for large values of n. For n assets, the number of possible state vectors is at most 3^n—exactly 3^n for a fully constrained problem, and less than 3^n for a problem with some of the low and high bounds unconstrained.

Iterate over all of the possible state vectors s with at least one *in* state. Use the computations of sections 4.1 and 4.3 to determine the intervals $[A^s_{min}, A^s_{max}]$ and the solution vectors $\tilde{c}$ and $\tilde{d}$ for each s. Discard the invalid state vectors with $A^s_{min} > A^s_{max}$. Sort the remaining valid state vectors in increasing order by A^s_{min}. Theorem 3.3 tells us that the resulting set of solutions for each valid s covers the entire domain $A \in (0, \infty)$.[5] So we have a complete solution to the problem. Given a value of A, we can quickly locate a valid state vector s for A and use its solution vectors $\tilde{c}$ and $\tilde{d}$ to compute the efficient portfolio for A.

The problem with this approach, of course, is its inefficiency for all but a very small number of assets. For most problems with larger numbers of assets, the vast majority of the possible state vectors are invalid. We would much prefer an approach that determined the valid state vectors directly. We develop such an approach in the sections that follow.

[5] Provided there are any solutions at all, i.e., $F \neq \{\}$.

5 The Critical-Line Algorithm

5.1 Introduction

Suppose we have a valid vector of asset states s. In the previous section we saw how to compute the range of values of A for which the state vector is valid, and how to compute the efficient portfolios for all the values of A in that range.

The range of A values over which s is valid is $[A^s_{min}, A^s_{max}]$. The solution we computed in the previous section is valid over this range, but not outside the range.

What happens as A decreases down to and across its endpoint A^s_{min} or up to and across its endpoint A^s_{max}? We must enter a new solution segment with a new state vector s'.

Given s, and the fact that s is valid, our intuition provides a reasonable conjecture as to what the new state vector s' must be. At A^s_{min} or A^s_{max}, there are only three possibilities:

1. Some *in* asset k hits its low bound. In this case the asset changes state to *down*.
2. Some *in* asset k hits its high bound. In this case the asset changes state to *up*.
3. The Kuhn-Tucker value for some *down* or *up* asset k becomes zero. In this case the asset changes state to *in*.

We can make this conjecture formal via the following definitions. For these definitions, the vectors p, q, $\tilde{c}$ and $\tilde{d}$ are the ones we computed in the previous section 4.

Definition 5.1 *Let s be a valid state vector.*

A minimum critical asset *for s is a value of k for which $A^s_{min} = p_k > 0$.*

A maximum critical asset *for s is a value of k for which $A^s_{max} = q_k < \infty$.*

Note that a valid state vector s with $A^s_{min} = 0$ has no minimum critical asset. If $A^s_{min} > 0$ there is at least one minimum critical asset, and maybe more than one. If $A^s_{max} = \infty$, s has no maximum critical asset. If $A^s_{max} < \infty$ there is at least one maximum critical asset, and maybe more than one.

Definition 5.2 *The* previous state vector *for a valid state vector s and a minimum critical asset k for s is the state vector $s' = prev(s,k)$ defined as follows:*

$$s'_i = \begin{cases} s_i & \text{if } i \neq k \\ down & \text{if } i = k,\ s_k = in,\ \text{and } p_k(L_k - \tilde{d}_k) = \tilde{c}_k \\ up & \text{if } i = k,\ s_k = in,\ \text{and } p_k(L_k - \tilde{d}_k) \neq \tilde{c}_k \\ in & \text{if } i = k \text{ and } s_k = down \text{ or } up \end{cases}$$

Definition 5.3 *The* next state vector *for a valid state vector s and a maximum critical asset k for s is the state vector $s' = next(s,k)$ defined as follows:*

$$s'_i = \begin{cases} s_i & \text{if } i \neq k \\ down & \text{if } i = k,\ s_k = in,\ \text{and } q_k(L_k - \tilde{d}_k) = \tilde{c}_k \\ up & \text{if } i = k,\ s_k = in,\ \text{and } q_k(L_k - \tilde{d}_k) \neq \tilde{c}_k \\ in & \text{if } i = k \text{ and } s_k = down \text{ or } up \end{cases}$$

The *critical-line algorithm* starts with a valid state vector s and repeatedly finds previous and next state vectors until the resulting set of solution segments covers the entire domain from $A = 0$ to $A = \infty$.

```
Initialize the result set S to the single element s
t = s
while (A^t_min > 0) {
    k = first minimum critical asset for t
    t = prev(t,k)
    add t to S
}
t = s
while (A^t_max < ∞) {
    k = first maximum critical asset for t
    t = next(t,k)
    add t to S
}
```

Note that we could have said "any" instead of "first" in the first lines of each of the two while loops, in which case the algorithm would be non-deterministic. At each iteration, we could in theory select *any* minimum or maximum critical asset for t. There may be more than one choice in the case of "ties." Thus there could be many paths through the algorithm. To make this deterministic, we choose the first one in each case. That is, in the first while loop, we choose the smallest value of k which is a minimum critical asset for t, and in the second while loop we choose the smallest value of k which is a maximum critical asset for t.

In order to convince ourselves that this algorithm actually works, we need to prove the following:

- The *prev* and *next* functions lead from valid state vectors to valid state vectors.
- The *prev* and *next* functions go in the proper directions with no gaps. That is, if $s' = prev(s, k)$, then the solution segment for s' is to the left of the solution segment for s, and if $s' = next(s, k)$, then the solution segment for s' is to the right of the solution segment for s, for some suitable definition of "to the left of," "to the right of" and "no gaps."
- The algorithm terminates—neither of the two *while* loops goes on forever.

5.2 Validity

In this section we prove that the *prev* and *next* functions lead from valid state vectors to valid state vectors, and that they go in the proper directions (to the left and right respectively) with no gaps.

One of the requirements for a state vector to be valid is that it must contain at least one *in* asset. So we must show that the *prev* and *next* functions lead from valid state vectors to state vectors that always have at least one *in* asset. We prove this using a sequence of lemmas.

Lemma 5.1 *If k is the only* in *asset in a valid state vector s, then $\tilde{c}_k = 0$.*

Proof:

Equation (19) computes $\tilde{c}$ as:

$$\tilde{c} = \tilde{V}^{-1}\tilde{x}$$

So we have:

$$\tilde{V}\tilde{c} = \tilde{x}$$

Because k is the only *in* asset, this equation has a particularly simple form:

$$\begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ \rho_{k,1} & \cdots & \rho_{k,k} & \cdots & \rho_{k,n} & 1 & \\ & & & \ddots & & & \\ & & & & 1 & & \\ 1 & \cdots & 1 & \cdots & 1 & & \end{pmatrix} \begin{pmatrix} \tilde{c}_1 \\ \vdots \\ \tilde{c}_k \\ \vdots \\ \tilde{c}_n \\ \tilde{c}_{n+1} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ \alpha_k \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

where the elements we have left blank in the matrix $\tilde{V}$ are all 0.

For $i \neq k$ and $i \leq n$, multiplying out row i shows that $\tilde{c}_i = 0$.

Multiplying out the last row gives $\sum_{i=1}^{n} \tilde{c}_i = 0$. Because $\tilde{c}_i = 0$ for all $i \neq k$ and $i \leq n$, we must have $\tilde{c}_k = 0$.

Lemma 5.2 *If k is the only* in *asset in a valid state vector s, then $[p_k, q_k] = [0, \infty]$.*

Proof:

s is valid, so it is valid for some A with a solution $w \in F$.

By Lemma 5.1, $\tilde{c}_k = 0$.

By equation (17) $w_k = \frac{1}{A}\tilde{c}_k + \tilde{d}_k$. So we must have $w_k = \tilde{d}_k$.

$w \in F$, so $L_k \leq w_k \leq H_k$. Because $w_k = \tilde{d}_k$, $L_k \leq \tilde{d}_k \leq H_k$.

Because $\tilde{c}_k = 0$, $L_k \leq \tilde{d}_k \leq H_k$, and because s is valid, inspecting equations (36), (37), and (40) shows that we must have:

$$\begin{aligned} [p_k^L, q_k^L] &= [0, \infty] \\ [p_k^H, q_k^H] &= [0, \infty] \\ [p_k, q_k] &= [0, \infty] \end{aligned}$$

Lemma 5.3 *If k is the only* in *asset in a valid state vector s, then k cannot be a minimum or maximum critical asset for s.*

Proof: By Definition 5.1, if k is a minimum critical asset for s, we must have $A^s_{min} = p_k > 0$. But by Lemma 5.2, we know that $p_k = 0$. So this is impossible.

Similarly, if k is a maximum critical asset for s, we must have $A^s_{max} = q_k < \infty$. But we know that $q_k = \infty$, so this is impossible.

Lemma 5.4 *The* prev *and* next *functions always lead from valid state vectors to state vectors that have at least one* in *asset.*

Proof:

Let s be a valid state vector, k a minimum critical asset for s, and $s' = prev(s, k)$. Suppose s' contains only *down* and *up* assets. The only way this can happen is if s contains only one *in* asset, and that asset is the minimum critical asset k that changes state to *down* or *up*. But Lemma 5.3 says this is impossible.

The proof for the *next* function is similar.

Lemma 5.5 *If s is valid and $s' = prev(s, k)$ or $s' = next(s, k)$, then s and s' share at least one* in *asset in common.*

Proof:

The only way s and s' could fail to share an *in* asset is if s contains only one *in* asset and that asset is the one that changes state. But we showed above that this is impossible.

Theorem 5.1 *If s is a valid state vector, k is a minimum critical asset for s, and $s' = prev(s,k)$, then s' is also valid. In particular, s' is valid for $A = A^s_{min}$.*

Similarly, if s is a valid state vector, k is a maximum critical asset for s, and $s' = next(s,k)$, then s' is also valid. In particular, s' is valid for $A = A^s_{max}$.

Proof:

We will do the case where k is a minimum critical asset for s. The maximum critical asset case is similar, and we leave the proof for that case as an exercise for the reader.

Lemma 5.4 establishes that s' has at least one *in* asset. So the only concern remaining is the Kuhn-Tucker conditions for s'.

Let $A = A^s_{min}$. We have $0 < p_k = A$. Let w be the efficient portfolio for A.

We know that s is valid for A, and that (s, w, λ, A) satisfies the Kuhn-Tucker conditions for some λ. We will show that (s', w, λ, A) also satisfies the Kuhn-Tucker conditions. This shows that s' is valid for A, and hence s' is valid.

The Kuhn-Tucker conditions for (s', w, λ, A) are:

$$w \in F$$

$$w_i = \begin{cases} L_i & \text{if } s'_i = down \\ H_i & \text{is } s'_i = up \end{cases}$$

$$\frac{\partial f}{\partial w_i}(w) - \lambda \begin{cases} \leq 0 & \text{if } s'_i = down \\ = 0 & \text{if } s'_i = in \\ \geq 0 & \text{if } s'_i = up \end{cases}$$

We know that $w \in F$, and we have $s_i = s'_i$ for all $i \neq k$. So we only need to demonstrate that the second two conditions hold for $i = k$.

First suppose that $s'_k = down$. Then

$$\begin{aligned} p_k(L_k - \tilde{d}_k) &= \tilde{c}_k \quad \text{(by Definition 5.2)} \\ L_k &= \frac{1}{p_k}\tilde{c}_k + \tilde{d}_k \\ L_k &= \frac{1}{A}\tilde{c}_k + \tilde{d}_k \quad \text{(because } p_k = A\text{)} \\ &= w_k \quad \text{(by equation (17))} \end{aligned}$$

Now suppose that $s'_k = up$. Then $s_k = in$, and by Definition 5.2, $p_k(L_k - \tilde{d}_k) \neq \tilde{c}_k$. We also know that $0 < p_k$. By inspection of equations (36) and (37), we must have:

$$\begin{aligned} p_k &= \tilde{c}_k/(H_k - \tilde{d}_k) \\ H_k &= \frac{1}{p_k}\tilde{c}_k + \tilde{d}_k \end{aligned}$$

$$\begin{aligned} H_k &= \frac{1}{A}\tilde{c}_k + \tilde{d}_k \quad \text{(because } p_k = A\text{)} \\ &= w_k \quad \text{(by equation (17))} \end{aligned}$$

Our final task is to verify the Kuhn-Tucker condition on the first partial derivative. We will in fact show that no matter what kind of transition has taken place, we have:

$$\frac{\partial f}{\partial w_k} - \lambda = 0$$

First suppose that the transition is from $s_k = in$ to $s'_k = down$ or up. Because $s_k = in$, the Kuhn-Tucker value is 0, and we are done. So suppose that the transition is from $s_k = down$ or up to $s'_k = in$. Because $0 < p_k$, inspecting equations (32) and (33) shows that we must have:

$$\begin{aligned} p_k &= \tilde{e}_k/\tilde{f}_k \\ A &= \tilde{e}_k/\tilde{f}_k \quad \text{(because } p_k = A\text{)} \\ \frac{\partial f}{\partial w_k} - \lambda &= \tilde{e}_k - A\tilde{f}_k \quad \text{(by equation(31))} \\ &= 0 \end{aligned}$$

Theorem 5.2 *The* prev *and* next *functions go in the proper directions with no gaps. That is, if s is valid, k is a minimum critical asset for s, and $s' = prev(s,k)$, then*

$$A^{s'}_{min} \le A^s_{min} \le A^{s'}_{max}$$

Similarly, if s is valid, k is a maximum critical asset for s, and $s' = next(s,k)$, then

$$A^{s'}_{min} \le A^s_{max} \le A^{s'}_{max}$$

Proof:

In the minimum critical asset case, Theorem 5.1 shows that s' is valid for A^s_{min}. So we must have:

$$A^{s'}_{min} \le A^s_{min} \le A^{s'}_{max}$$

Similarly, in the maximum critical asset case, s' is valid for A^s_{max}, so we must have:

$$A^{s'}_{min} \le A^s_{max} \le A^{s'}_{max}$$

Note that Theorem 5.2 does not prove that the new segment for s' is strictly to the left or right of the one for s. That is, it does not prove that $A^{s'}_{max} = A^s_{min}$ for the *prev* direction or that $A^{s'}_{min} = A^s_{max}$ for the *next* direction. We will not prove those properties until Theorem 5.3 in the next section, and then only under certain assumptions. The properties we have proven here, however, are enough to establish the absence of any "gaps" in the sequence of generated segments ($A^{s'}_{max} \ge A^s_{min}$ for *prev* and $A^{s'}_{min} \le A^s_{max}$ for *next*).

5.3 Termination

We now know that the critical-line algorithm always leads from valid state vectors to valid state vectors, and it moves in the proper directions with no gaps. It remains to show that the algorithm terminates.

We present the algorithm again here for convenience:

```
Initialize the result set S to the single element s
t = s
while (A^t_min > 0) {
    k = the first minimum critical asset for t
    t = prev(t,k)
    add t to S
}
t = s
while (A^t_max < ∞) {
    k = the first maximum critical asset for t
    t = next(t,k)
    add t to S
}
```

The first important point to note is that the number of valid state vectors is finite, as we discussed in section 4.5. Our main theorem 3.3 tells us that this finite set of all possible valid state vectors does indeed provide a full solution—every possible value of A from 0 to ∞ is part of the solution segment for some valid state vector.[6]

Given these observations, it is tempting to try to resolve the termination problem by showing that the values of A^t_{min} are strictly decreasing at each iteration of the first while loop, and that the values of A^t_{max} are strictly increasing at each iteration of the second while loop. If we could in fact demonstrate this, we would be done. This is not immediate, however. Note that theorem 5.2 only guarantees that these endpoints are nonincreasing and nondecreasing (respectively). It may be possible for an iteration of the algorithm to produce a new solution segment that is valid for only a single point ($A^t_{min} = A^t_{max}$), without causing any decrease in the value of A^t_{min} in the first loop, or increase in the value of A^t_{max} in the second loop.

Because the number of valid state vectors is finite, another approach would be to try to show that the two loops never revisit a previously examined state vector. This is called a *cycle*. It we could prove that cycles never happen, we would be finished.

These turn out to be rather difficult problems, and it will take some effort to work our way through the solutions.

[6] Provided we have any solutions at all, i.e., $F \neq \{\}$.

We begin with some useful definitions and lemmas.

Definition 5.4 *An asset k is* strict *for a state vector s if all the inequalities in the Kuhn-Tucker conditions for asset k are strict in the interior of the solution segment for s. That is, if $A^s_{min} < A < A^s_{max}$, and w is the efficient portfolio for A, then:*

$$\text{if } k \in D \text{ then } \frac{\partial f}{\partial w_k}(w) < \lambda$$

$$\text{if } k \in U \text{ then } \frac{\partial f}{\partial w_k}(w) > \lambda$$

$$\text{if } k \in I \text{ then } L_k < w_k < H_k$$

Definition 5.5 *A state vector s is* strict *if $A^s_{min} < A^s_{max}$ (the solution segment contains more than a single point), and all assets k are strict for s.*

Lemma 5.6 *Suppose k is a minimum or maximum critical asset for a valid state vector s. Then k is strict for s.*

Proof:

We will do the case where k is a minimum critical asset for s. The maximum critical asset case is similar.

k is a minimum critical asset for s, so $0 < p_k = A^s_{min}$. Suppose that $A^s_{min} < A < A^s_{max}$. Let w be the efficient portfolio for A.

Suppose $k \in D$. By inspection of equation (32), we must have $A^s_{min} = p_k = \tilde{e}_k/\tilde{f}_k$ and $\tilde{f}_k > 0$. So we have:

$$\begin{aligned} \frac{\partial f}{\partial w_k}(w) - \lambda &= \tilde{e}_k - A\tilde{f}_k \\ &< \tilde{e}_k - A^s_{min}\tilde{f}_k \\ &= 0 \end{aligned}$$

Suppose $k \in U$. By inspection of equation (33), we must have $A^s_{min} = p_k = \tilde{e}_k/\tilde{f}_k$ and $\tilde{f}_k < 0$. So we have:

$$\begin{aligned} \frac{\partial f}{\partial w_k}(w) - \lambda &= \tilde{e}_k - A\tilde{f}_k \\ &> \tilde{e}_k - A^s_{min}\tilde{f}_k \\ &= 0 \end{aligned}$$

Suppose $k \in I$. Let v be the efficient portfolio for A^s_{max}. v must satisfy all the constraints, so we have $L_k \leq v_k \leq H_k$.

Suppose $k \in I$ and $p_k(L_k - \tilde{d}_k) = \tilde{c}_k$. Then $A^s_{min} = p_k = \tilde{c}_k/(L_k - \tilde{d}_k)$. By inspection of equation (36), we must have $L_k < d_k$, which together with $0 < p_k$ implies that $\tilde{c}_k < 0$. So we have:

$$\begin{aligned} L_k &= \frac{1}{A^s_{min}}\tilde{c}_k + \tilde{d}_k \\ &< \frac{1}{A}\tilde{c}_k + \tilde{d}_k \\ &= w_k \\ &< \frac{1}{A^s_{max}}\tilde{c}_k + \tilde{d}_k \\ &= v_k \\ &\leq H_k \end{aligned}$$

Suppose $k \in I$ and $p_k(L_k - \tilde{d}_k) \neq \tilde{c}_k$. Then $A^s_{min} = p_k = \tilde{c}_k/(H_k - \tilde{d}_k)$. By inspection of equation (37), we must have $H_k > d_k$, which together with $0 < p_k$ implies that $\tilde{c}_k > 0$. So we have:

$$\begin{aligned} H_k &= \frac{1}{A^s_{min}}\tilde{c}_k + \tilde{d}_k \\ &> \frac{1}{A}\tilde{c}_k + \tilde{d}_k \\ &= w_k \\ &> \frac{1}{A^s_{max}}\tilde{c}_k + \tilde{d}_k \\ &= v_k \\ &\geq L_k \end{aligned}$$

Lemma 5.7 *Suppose s is a valid state vector. If $s' = prev(s, k)$ for a minimum critical asset k for s, or if $s' = next(s, k)$ for a maximum critical asset k for s, then the solution segments for s and s' intersect in at most a single point.*

Proof:

Again we do only the minimum critical asset case. The proof is by contradiction.

If the solution segments for s and s' intersect in more than a single point, then they mut share an internal point A with:

$$A^s_{min} < A < A^s_{max}$$

By Lemma 5.5, s and s' share at least one *in* asset in common. In addition, the *down* and *up* assets on of one of the state vectors is always a subset of the

down and *up* assets of the other state vector. By Lemma 4.4, s and s' have the same solution (w, λ) for A.

Suppose $s_k = down$ and $s'_k = in$. Then:

$$\begin{aligned} \frac{\partial f}{\partial w_k}(w) &= \lambda \quad \text{(because } s'_k = in\text{)} \\ \frac{\partial f}{\partial w_k}(w) &< \lambda \quad \text{(by Lemma 5.6 applied to } s, k\text{)} \end{aligned}$$

Suppose $s_k = up$ and $s'_k = in$. Then:

$$\begin{aligned} \frac{\partial f}{\partial w_k}(w) &= \lambda \quad \text{(because } s'_k = in\text{)} \\ \frac{\partial f}{\partial w_k}(w) &> \lambda \quad \text{(by Lemma 5.6 applied to } s, k\text{)} \end{aligned}$$

Suppose $s_k = in$ and $s'_k = down$. Then:

$$\begin{aligned} w_k &= L_k \quad \text{(because } s'_k = down\text{)} \\ w_k &> L_k \quad \text{(by Lemma 5.6 applied to } s, k\text{)} \end{aligned}$$

Suppose $s_k = in$ and $s'_k = up$. Then:

$$\begin{aligned} w_k &= H_k \quad \text{(because } s'_k = up\text{)} \\ w_k &< H_k \quad \text{(by Lemma 5.6 applied to } s, k\text{)} \end{aligned}$$

So all four possible cases lead to a contradiction.

Lemma 5.8 *Suppose s is strict.*

If (w, λ) is the solution for A^s_{min}, then:

$$\begin{aligned} &\textit{If } i \in D \cup U: && \frac{\partial f}{\partial w_i}(w) = \lambda \textit{ iff } i \textit{ is minimum critical for } s \\ &\textit{If } i \in I: && w_i = L_i \textit{ or } w_i = H_i \textit{ iff } i \textit{ is minimum critical for } s \end{aligned}$$

Similarly, if (w, λ) is the solution for A^s_{max}, then:

$$\begin{aligned} &\textit{If } i \in D \cup U: && \frac{\partial f}{\partial w_i}(w) = \lambda \textit{ iff } i \textit{ is maximum critical for } s \\ &\textit{If } i \in I: && w_i = L_i \textit{ or } w_i = H_i \textit{ iff } i \textit{ is maximum critical for } s \end{aligned}$$

Proof:

As usual, we only do the minimum critical case. The maximum critical case is similar.

For $i \in D$, suppose $\frac{\partial f}{\partial w_i}$ is equal to λ at the left endpoint of the solution segment. Because s is strict, it is less than λ at interior points of the segment. So we must have $\tilde{f}_i > 0$, and equation (32) shows that k is a minimum critical asset for s. The other direction is equally trivial.

For $i \in U$, suppose $\frac{\partial f}{\partial w_i}$ is equal to λ at the left endpoint of the solution segment. Because s is strict, it is greater than λ at interior points of the segment. So we must have $\tilde{f}_i < 0$, and equation (33) shows that k is a minimum critical asset for s. The other direction is equally trivial.

For $i \in I$, suppose $w_i = L_i$ at the left endpoint. Because s is strict, it is greater that L_i in the interior, so we must have $\tilde{c}_i < 0$ with $L_i < d_i$, and equation (36) shows that k is a minimum critical asset. The case where $w_i = H_i$ and the other direction are equally trivial.

We can now prove that under certain assumptions which eliminate the need to deal with special cases and other issues, the critical-line algorithm does indeed terminate. Our proof uses that same line of reasoning as Markowitz' proof in Appendix A of [7].

Theorem 5.3 *Suppose s is strict and $L_i < H_i$ for all i.*

If k is the only minimum critical asset for s (there are no ties), then $s' = prev(s, k)$ is also strict.

Similarly, if k is the only maximum critical asset for s (there are no ties), then $s' = next(s, k)$ is also strict.

Proof:

Again we do only the minimum critical asset case.

Let $A = A^s_{min}$. In Theorem 5.1 we showed that both s and s' are valid for A, and they share the same solution (w, λ) at A.

Let $(\tilde{c}', \tilde{d}', \tilde{e}', \tilde{f}')$ be the solution vectors we computed for s' in section 4.1.

We first prove that $A^{s'}_{min} < A^{s'}_{max}$, that is, that the solution segment for s' contains more than a single point.

Because there aren't any ties, i is not minimum critical for s for all $i \neq k$. For $i \neq k$ we also have $s_i = s'_i$ (these assets keep the same state). By Lemma 5.8, we have:

$$\text{If } i \in D, i \neq k: \quad \frac{\partial f}{\partial w_i}(w) - \lambda = \tilde{e}'_i - A\tilde{f}'_i < 0$$

$$\text{If } i \in I, i \neq k: \quad L_i < w_i = \frac{1}{A}\tilde{c}'_i + \tilde{d}'_i < H_i$$

$$\text{If } i \in U, i \neq k: \quad \frac{\partial f}{\partial w_i}(w) - \lambda = \tilde{e}'_i - A\tilde{f}'_i > 0$$

Because all of the inequalities are strict, they remain true if we replace A by

$A \pm \epsilon$ for all sufficiently small $\epsilon > 0$.

Asset k is the one that changes state in the transition from s to s'. There are four cases.

Suppose k changes from *down* to *in*. Then at the endpoint A:

$$w_k = \frac{1}{A}\tilde{c}'_k + \tilde{d}'_k = L_k$$

Suppose for the purpose of contradiction that $\tilde{c}'_k \leq 0$. Then for all sufficiently small $\epsilon > 0$ we have:

$$\frac{1}{A+\epsilon}\tilde{c}'_k + \tilde{d}'_k \geq L_k$$

In this case (using our assumption that $L_k < H_k$), for all sufficiently small $\epsilon > 0$, $A+\epsilon$ satisfies all of the conditions of Theorem 4.1 for s', and therefore s' is valid for $A+\epsilon$ for all sufficiently small $\epsilon > 0$. But then s and s' overlap in more than a single point, which contradicts Lemma 5.7.

So we must have $\tilde{c}'_k > 0$. Then for all sufficiently small $\epsilon > 0$ we have:

$$\frac{1}{A-\epsilon}\tilde{c}'_k + \tilde{d}'_k > L_k$$

Thus, for all sufficiently small $\epsilon > 0$, $A - \epsilon$ satisfies all of the conditions of Theorem 4.1 for s', and therefore s' is valid for $A - \epsilon$ for all sufficiently small $\epsilon > 0$. This proves that s' contains more than a single point.

The other three kinds of transitions for k are similar, and we leave them as exercises (*up* to *in*, *in* to *down*, and *in* to *up*).

Proving that all the assets are strict for s' is easy but tedious. There are seven cases, for $i \neq k \in I$, $i \neq k \in D$, $i \neq k \in U$, and the four kinds of transitions for $i = k$. We will do just the first of the seven cases, the others being similar.

Suppose $i \neq k \in I$. Over the solution segment for s', the asset weight w_i is given by the equation $\frac{1}{B}\tilde{c}'_i + \tilde{d}'_i$. At the right endpoint A the value of this equation is strictly between the bounds L_i and H_i. At the left endpoint the value might be at one of these bounds, but clearly it cannot be at either bound in the interior of the segment. So i is strict for s'.

The key conclusion of theorem 5.3 is that under its assumptions, the *prev* and *next* functions produce sequences of strict solution segments. Thus under the assumptions, the sequences of A^t_{min} and A^t_{max} values produced by the while loop iterations are strictly decreasing/increasing respectively. As we observed at the beginning of this section, this is enough to prove that the algorithm terminates.

We aren't finished yet, but we are getting closer. At this point, if we make a few important assumptions, we know that the critical-line algorithm works, in the sense that it terminates in a finite amount of time with a full and correct solution.

The assumptions are:

1. We can find a strict starting state vector for the algorithm.
2. $L_i < H_i$ for all i.
3. We never encounter any ties, where at an iteration there is more than one candidate for the minimum critical asset or maximum critical asset.

We deal with these problems in the following sections.

5.4 Finding the First Segment

In this section we deal with the problem of finding a strict starting state vector for the algorithm (the first solution segment).

Markowitz and Sharpe both start at the end of the efficient frontier for $A = 0$, and solve a linear programming problem to find the efficient portfolio held by an infinitely risk-tolerant investor—the portfolio that maximizes the expected return. Unfortunately, this technique, while trivial in many common cases, only works in the general case if there is an upper bound on the expected return. This is true if all the assets are fully constrained, but it is not always true in the general case we are considering, where some assets may be only partially constrained or, for that matter, unconstrained.

Markowitz mentions this problem and suggests starting at the other end for an infinitely risk-averse investor with $A = \infty$, by solving the quadratic programming problem to find the minimum variance portfolio. This is the approach we take.

For any value of A, maximizing our objective function:

$$f(w) = w'x - \frac{1}{2}Aw'Vw$$

is equivalent to minimizing:

$$f^*(w) = -\frac{2}{A}w'x + w'Vw$$

In the limit, as $A \to \infty$, we get the variance of w:

$$\mathrm{Var}(w) = \lim_{A\to\infty} f^*(w) = w'Vw$$

Thus as risk aversion increases towards $A = \infty$, we suspect that the optimal portfolio approaches the minimum variance portfolio. If we can find a valid state vector s that corresponds to the minimum variance portfolio, that state vector should be valid for all sufficiently large values of A, with $A^s_{min} < A^s_{max} = \infty$.

That is our goal and plan of attack. The first step towards the goal is to solve the minimization problem to find the minimum variance portfolio v.

Minimizing the variance $w'Vw$ over the constraint set F is a convex quadratic programming problem with linear constraints. These problems are well understood and not difficult to solve numerically. One popular technique is to use the Kuhn-Tucker conditions to reduce the problem to a linear complementarity programming problem, then use the Lemke-Howson algorithm to solve that problem. A good description of this algorithm may be found in Friedman [4].

Definition 5.6 *A* linear complementarity programming problem (LCPP) *has the following form:*

Given a vector q and a matrix M, find vectors w and z so that the following conditions are all satisfied:

$$w = q + Mz$$

$$w \geq 0,\ z \geq 0 \text{ and } wz = 0$$

Before we can derive the LCPP that we are going to solve, we need to develop some notation.

We face the problem that our lower and upper bound constraints are optional. We have been using the convention that $L_i = -\infty$ means that asset i has no lower bound, and $H_i = +\infty$ means that asset i has no upper bound. We need to specify these constraints in our problem in a different form. We must include only the constraints that are present, and leave out the ones that are missing. To accomplish this, we define two helper matrices Y and Z that "gather together" the lower bound constraints and upper bound constraints that we need to include respectively.

An example is the best way to illustrate how these helper matrices work. Suppose we have a three asset problem. Assets 1 and 3 have lower bounds, but asset 2 does not. The helper matrix Y is defined as the identity matrix with row 2 removed:

$$Y = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Let L be the column vector of all the lower bounds:

$$L = \begin{pmatrix} L_1 \\ -\infty \\ L_3 \end{pmatrix}$$

Then multiplying Y times L gathers together just the bounds that are present (the ones that are not $-\infty$):

$$YL = \begin{pmatrix} L_1 \\ L_3 \end{pmatrix}$$

The helper matrix Z works the same way for the upper bound constraints. We define H to be the full vector of upper bound constraints, and ZH gathers together just the ones that are not $+\infty$.

If w is a vector of asset weights, note that:

$$YL \leq Yw \text{ iff } L \leq w$$
$$Zw \leq ZH \text{ iff } w \leq H$$

One of the results we will get by solving the LCPP is a pair of Lagrange multiplier vectors μ^L and μ^H for the low and high bound constraints. The solution will only find these multipliers for assets which are actually constrained. We establish in advance that we will define $\mu_i^L = 0$ for assets i without low bounds, and $\mu_i^H = 0$ for assets i without high bounds. We use the notation μ^L and μ^H for the full vectors. $Y\mu^L$ and $Z\mu^H$ gather together just the multiplier variables for which we actually need to find values.

Note that:

$$Y'Y\mu^L = \mu^L \text{ and } Z'Z\mu^H = \mu^H$$

We use our new notation to express the problem in the following form:

$$\text{Minimize } \text{Var}(v) = v'Vv \text{ subject to:}$$

$$\begin{gathered} 1 - v\mathbf{1} = 0 \\ YL - Yv \leq 0 \\ -ZH + Zv \leq 0 \end{gathered}$$

By the Kuhn-Tucker theorem, v minimizes $\text{Var}(v)$ over F iff there exist Lagrange multipliers λ, $\mu^L \geq 0$, and $\mu^H \geq 0$ with:

$$\begin{gathered} v \in F \\ \nabla\text{Var}(v) - \lambda\mathbf{1} - \mu^L + \mu^H = 0 \\ L_i < v_i \Rightarrow \mu_i^L = 0 \\ v_i < H_i \Rightarrow \mu_i^H = 0 \end{gathered} \tag{44}$$

$\nabla\text{Var}(v) = 2Vv$, so we have:

$$2Vv - \lambda\mathbf{1} - \mu^L + \mu^H = 0 \tag{45}$$

LCP problems require that all variables must be non-negative. To this end, we use the standard technique of splitting our variables v and λ into the difference of two variables each of which is non-negative:

$$\begin{aligned} v &= v_1 - v_2 \quad \text{where } v_1 \geq 0 \text{ and } v_2 \geq 0 \\ \lambda &= \lambda_1 - \lambda_2 \quad \text{where } \lambda_1 \geq 0 \text{ and } \lambda_2 \geq 0 \end{aligned}$$

After making these substitutions, we get:

$$\begin{gathered} 2Vv_1 - 2Vv_2 - \lambda_1\mathbf{1} + \lambda_2\mathbf{1} - \mu^L + \mu^H = 0 \\ YL - Yv_1 + Yv_2 \leq 0 \\ -ZH + Zv_1 - Zv_2 \leq 0 \\ 1 - v_1\mathbf{1} + v_2\mathbf{1} = 0 \end{gathered}$$

LCP problems also require that we use $\geq$ inequalities. So we express each of the two equalities above as a pair of $\geq$ inequalities, and we negate the two $\leq$ inequalities:

$$\begin{gathered}
2Vv_1 - 2Vv_2 - \lambda_1\mathbf{1} + \lambda_2\mathbf{1} - \mu^L + \mu^H \geq 0 \\
-2Vv_1 + 2Vv_2 + \lambda_1\mathbf{1} - \lambda_2\mathbf{1} + \mu^L - \mu^H \geq 0 \\
-YL + Yv_1 - Yv_2 \geq 0 \\
ZH - Zv_1 + Zv_2 \geq 0 \\
1 - v_1\mathbf{1} + v_2\mathbf{1} \geq 0 \\
-1 + v_1\mathbf{1} - v_2\mathbf{1} \geq 0
\end{gathered}$$

We are now ready to state our LCPP. It is simply the equations above expressed using vectors and a matrix:

$$w = q + Mz =$$

$$\begin{pmatrix} 0 \\ 0 \\ -YL \\ ZH \\ 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 2V & -2V & -Y' & Z' & \mathbf{1}_c & -\mathbf{1}_c \\ -2V & 2V & Y' & -Z' & -\mathbf{1}_c & \mathbf{1}_c \\ Y & -Y & 0 & 0 & 0 & 0 \\ -Z & Z & 0 & 0 & 0 & 0 \\ -\mathbf{1}_r & \mathbf{1}_r & 0 & 0 & 0 & 0 \\ \mathbf{1}_r & -\mathbf{1}_r & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ Y\mu^L \\ Z\mu^H \\ \lambda_2 \\ \lambda_1 \end{pmatrix} \tag{46}$$

$$w \geq 0,\ z \geq 0 \text{ and } wz = 0$$

Note that the matrix M has the following form:

$$M = \begin{pmatrix} A & B' \\ -B & 0 \end{pmatrix} \text{ where } A = \begin{pmatrix} 2V & -2V \\ -2V & 2V \end{pmatrix} \text{ and } B = \begin{pmatrix} -Y & Y \\ Z & -Z \\ \mathbf{1}_r & -\mathbf{1}_r \\ -\mathbf{1}_r & \mathbf{1}_r \end{pmatrix}$$

Because V is positive semidefinite, it is easy to show that both A and M are also positive semidefinite. Friedman shows in [4] that when M is positive semidefinite, the Lemke-Howson algorithm can be used to solve the LCPP.

We supply the constant vector q and the constant matrix M. We use the Lemke-Howson algorithm to solve the problem to find the vectors w and z.[7] The vector z is the one that contains the solution we need for v_1, v_2, $Y\mu^L$, $Z\mu^H$, λ_1 and λ_2.[8] We recombine that vectors v_1 and v_2 into $v = v_1 - v_2$ and the multipliers λ_1 and λ_2 into $\lambda = \lambda_1 - \lambda_2$. The resulting vector of asset weights v and multipliers (λ, μ^L, μ^H) satisfy all the Kuhn-Tucker conditions in (44), so v minimizes Var over F and we have our minimum variance portfolio.

[7] It is possible that there is no solution, if the feasible set is empty ($F = \{\}$). In this case the Lemke-Howson algorithm tells us that the LCPP is infeasible and has no solution.

[8] We don't care about the contents of the solution vector w, although it is easy enough to see what its contents must be by multiplying all the rows of equation (46). We only care that it is non-negative, so our inequalities are all true, which is guaranteed by the Lemke-Howson algorithm.

Define the state vector s as follows:

$$s_i = \begin{cases} down & \text{if } v_i = L_i \\ in & \text{if } L_i < v_i < H_i \\ up & \text{if } v_i = H_i > L_i \end{cases}$$

In order to serve as a starting state vector for the critical-line algorithm, s must be strict with $A^s_{max} = \infty$. We can prove that this is true if we make a few important assumptions.

Theorem 5.4 *The starting state vector s as defined above is strict with $A^s_{max} = \infty$ if all the following conditions are satisfied:*

- *s contains at least one* in *asset.*
- *$L_i < H_i$ for all i.*
- *$v_i = L_i \Rightarrow \mu_i^L > 0$ and $v_i = H_i \Rightarrow \mu_i^H > 0$.*

Proof:

Let v be the minimum variance portfolio computed above along with its Lagrange multipliers λ, μ^L and μ^H. Let $\tilde{V}$, $\tilde{x}$, $\tilde{y}$, $\tilde{c}$, $\tilde{d}$, $\tilde{e}$ and $\tilde{f}$ be the various solution vectors for s computed by the equations in section 4.1 (we know these solutions exist because s contains at least one *in* asset).

Define:

$$\tilde{v} = \begin{pmatrix} v \\ -\lambda/2 \end{pmatrix}$$

We first show that:

$$\tilde{V}\tilde{v} = \tilde{y}$$

To see this, multiply out each row i. For $i \in D$, row i of $\tilde{V}$ is all 0 except for a 1 in column i, and element i of $\tilde{y}$ is $v_i = L_i$. Similarly, for $i \in U$, row i of $\tilde{V}$ is the same, and element i of $\tilde{y}$ is $v_i = H_i$. For $i \in I$, first note that $L_i < v_i < H_i$, so $\mu_i^L = \mu_i^H = 0$. Then note that row i of $\tilde{V}\tilde{v}$ is one half of row i of equation (45), which is 0, which is element i of $\tilde{y}$. The last row $n+1$ is $v\mathbf{1} = 1$.

We also have by the definition of $\tilde{d}$:

$$\tilde{V}\tilde{d} = \tilde{y}$$

$\tilde{V}$ is non-singular, so $\tilde{V}\tilde{v} = \tilde{y}$ and $\tilde{V}\tilde{d} = \tilde{y}$ together imply that:

$$\tilde{d} = \tilde{v}$$

We compute the solution vector $\tilde{f}$ as follows:

$$\begin{aligned}
\tilde{f} &= \hat{V}\tilde{d} \\
&= \hat{V}\tilde{v} \\
&= \begin{pmatrix} V & \mathbf{1}_c \\ \mathbf{1}_r & 0 \end{pmatrix} \begin{pmatrix} v \\ -\lambda/2 \end{pmatrix} \\
&= \begin{pmatrix} Vv - \dfrac{\lambda}{2}\mathbf{1} \\ v\mathbf{1} \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{2}(\mu^L - \mu^H) \\ 1 \end{pmatrix}
\end{aligned}$$

We now examine the three cases $i \in D$, $i \in I$, and $i \in U$.

For $i \in D$, because $v_i = L_i < H_i$, $\mu_i^H = 0$. By assumption, $\mu_i^L > 0$. So $\tilde{f}_i = \frac{1}{2}(\mu_i^L - \mu_i^H) > 0$. Then:

$$\tilde{e}_i - A\tilde{f}_i < 0 \quad \text{for all sufficiently large } A$$

$$[p_i, q_i] = [\tilde{e}_i/\tilde{f}_i, \infty]$$

For $i \in I$, we have $L_i < \tilde{d}_i = v_i < H_i$. Then:

$$L_i < \frac{1}{A}\tilde{c}_i + \tilde{d}_i < H_i \quad \text{for all sufficiently large } A$$

$$[p_i, q_i] = [max(\tilde{c}_i/(H_i - \tilde{d}_i), \tilde{c}_i/(L_i - \tilde{d}_i)), \infty]$$

For $i \in U$, because $L_i < H_i = v_i$, $\mu_i^L = 0$. By assumption, $\mu_i^H > 0$. So $\tilde{f}_i = \frac{1}{2}(\mu_i^L - \mu_i^H) < 0$. Then:

$$\tilde{e}_i - A\tilde{f}_i > 0 \quad \text{for all sufficiently large } A$$

$$[p_i, q_i] = [\tilde{e}_i/\tilde{f}_i, \infty]$$

So we have $A^s_{min} = \max(p_i) < A^s_{max} = \infty$, and each asset i is clearly strict for s, so the proof is complete.

We can now verify the conjecture we made at the beginning of this section that as risk aversion increases towards $A = \infty$, the optimal portfolio approaches the minimum variance portfolio. Let $w(A)$ be the optimal portfolio for A. Within the solution segment for our starting state vector s we have:

$$\tilde{w}(A) = \frac{1}{A}\tilde{c} + \tilde{d} = \frac{1}{A}\tilde{c} + \tilde{v}$$

Thus:

$$\lim_{A \to \infty} w(A) = v$$

At the end of section 5.3, we concluded that if we made the following assumptions, we could solve the full problem:

1. We can find a strict starting state vector for the algorithm.
2. $L_i < H_i$ for all i.
3. We never encounter any ties, where at an iteration there is more than one candidate for the minimum critical asset or maximum critical asset.

The first assumption has been transformed into two new assumptions:

1. The solution to the LCPP used to find the minimum variance portfolio v satisfies $v_i = L_i \Rightarrow \mu_i^L > 0$ and $v_i = H_i \Rightarrow \mu_i^H > 0$.
2. The solution v to the LCPP has at least one asset with $L_i < v_i < H_i$.
3. $L_i < H_i$ for all i.
4. We never encounter any ties, where at an iteration there is more than one candidate for the minimum critical asset or maximum critical asset.

We deal with all of these remaining issues in the next section.

5.5 Degeneracy and Cycles

The assumptions listed at the end of the previous section are all instances of a general problem called *degeneracy*. Is there some kind of pattern that is common to these cases of degeneracy that we can exploit to modify the algorithm to deal properly with all of them?

There is indeed a pattern, and it is a simple one to see. In all of these cases, the problem is "bumping up against a wall" or "lack of wiggle room." In each of the degeneracy situations, we could get around the problem if we were permitted to "move the wall just a little bit" or "make just a little bit more wiggle room." To make this vague insight more explicit, we'll examine the different kinds of degeneracy in more detail to see how we could make them go away if we had this kind of power.

Let $\epsilon > 0$ be a small number. For example, ϵ might be 10^{-6}.

We start with the easiest problem, which is $L_i = H_i$ for some asset i. If we replace L_i by $L_i - \epsilon$ or replace H_i by $H_i + \epsilon$, the problems go away both in Theorem 5.4 where we need $L_i < H_i$ to prove that the starting state vector is strict, and in Theorem 5.3 where we need $L_i < H_i$ to prove our main termination theorem. In both theorems, the problem is the *exact* equality of L_i and H_i. If they are not exactly equal, even if the difference between them is only our tiny

amount ϵ, we no longer have any problem. The problem is lack of wiggle room, and any amount of wiggle room, no matter how small, solves the problem.

Next consider the assumption that the solution to the LCPP satisfies $v_i = L_i \Rightarrow \mu_i^L > 0$ and $v_i = H_i \Rightarrow \mu_i^H > 0$.

Suppose we have a degenerate situation where $v_i = L_i$ with $\mu_i^L = 0$ for some asset i. If we "move the wall" and replace L_i by $L_i' = L_i - \epsilon$, asset i becomes an *in* asset which is strictly between its lower and upper bounds. Because $\mu_i^L = 0$, it is also easy to see that asset i still satisfies the Kuhn-Tucker conditions (44) after this change to the problem, so v still minimizes Var over the slightly enlarged feasible set F'.

Similarly, if we have a degeneracy where $v_i = H_i$ and $\mu_i^H = 0$ for some asset i, we can simply "move the wall" and replace H_i by $H_i' = H_i + \epsilon$.

Assume that we have gone ahead and made the adjustments described so far. We now have $L_i < H_i$ for all i, and we have $v_i = L_i \Rightarrow \mu_i^L > 0$ and $v_i = H_i \Rightarrow \mu_i^H > 0$ for all i. Suppose we still do not have any *in* assets with $L_i < v_i < H_i$. We need to move a wall to make at least one asset *in*. We proceed along much the same lines as the proof of Theorem 3.3. There are two cases.

If there is at least one *down* asset, let a be the smallest value of μ_i^L for all the *down* assets. For each *down* asset i with $\mu_i^L = a$, replace L_i by $L_i' = L_i - \epsilon$ and change the state of asset i to *in*.

If there aren't any *down* assets, we use the *up* assets instead. Let a be the smallest value of μ_i^H for all the *up* assets. For each *up* asset i with $\mu_i^H = a$, replace H_i by $H_i' = H_i + \epsilon$ and change the state of asset i to *in*.

We need to show that the Kuhn-Tucker conditions are still satisfied, so that v still minimizes Var over our enlarged feasible set F'. We verify this for the case where there is at least one *down* asset and leave the other case as an exercise. Define a new set of Lagrange multipliers as follows:

$$\begin{aligned}
\lambda' &= \lambda + a \\
\mu_i^{L'} &= \mu_i^L - a \quad \text{if } v_i = L_i \\
\mu_i^{L'} &= 0 \quad \text{if } v_i \neq L_i \\
\mu_i^{H'} &= \mu_i^H + a \quad \text{if } v_i = H_i \\
\mu_i^{H'} &= 0 \quad \text{if } v_i \neq H_i
\end{aligned}$$

The portfolio v together with the new feasible set F' and the new set of multipliers $(\lambda', \mu^{L'}, \mu^{H'})$ satisfies the Kuhn-Tucker conditions (44), so v minimizes Var over F'.

At this point, we have used our powers to modify the problem slightly in such a way that all the conditions of Theorem 5.4 are satisfied for the modified problem. Thus the starting state vector s for the modified problem is strict with $A_{max}^s = \infty$, and we have a good starting state vector for the critical-line

algorithm.

The only remaining problem is ties. We have seen that ties are a theoretical problem in the critical-line algorithm. A *tie* happens at an iteration when there is more than one minimum or maximum critical asset, and hence more than one choice for the asset that will change state. Theorem 5.3 is only valid if there are no ties, and there is no way to rescue the proof and repair it if there are ties. The key conclusion of Theorem 5.3 is that the previous and next segments contain more than a single point. It is this conclusion that permits us to claim that the algorithm must terminate. In the case of ties, we cannot use this idea to prove termination.

If there is a tie at an iteration, it is possible for the previous or next segment to contain only a single point. If this happens, it opens up the possibility that as we continue with the iterations in the algorithm, we may eventually come back to revisit the same state vector and its solution segment. This is called a *cycle*. Cycles are obviously bad—if we encounter one, the algorithm goes into an infinite loop.

We can use our wall-moving powers to deal with ties too. As an example, suppose that we are at some iteration in the critical-line algorithm moving in the *prev* direction and assets 1, 2 and 3 are all minimum critical, with $p_1 = p_2 = p_3 = A^t_{min}$. The trick in this case is to pick asset 1 as the one that will change state, and move walls to push away assets 2 and 3 just a little bit. For example, suppose asset 2 is an *in* asset that has hit its lower bound and wants to change state to *down*. Replace L_2 by $L_2 - \epsilon$, so that asset 2 is no longer critical—it's almost critical, but not exactly critical, and that's all we need. Similarly, suppose asset 3 is an *up* asset that wants to change to *in*. If we replace H_3 by $H_3 + \epsilon$, we make asset 3 already *in*, and no longer critical. It is easy to verify that the Kuhn-Tucker conditions are still satisfied after these small changes. After we have "pushed away" assets 2 and 3 in this way, asset 1 becomes the only critical asset, and we can go ahead and change its state and move on to the next segment, which is guaranteed to be strict, although it might be very small.

The key insight that we have now developed in detail is that all of the degeneracy problems are caused by *exact* equality conditions. We can make all of them disappear by moving one of the things that's equal away in an appropriate direction by our tiny amount ϵ.

There are two ways to implement this basic idea, one practical and one theoretical.

The practical approach simply makes these adjustments directly whenever they are needed. At the start of the problem, for any assets with $L_i = H_i$, we add our small number ϵ to H_i. This transforms the problem into one which has none of this kind of degeneracy. After we have found the minimum variance portfolio, we fix any degeneracies involving the Lagrange multipliers μ^L and μ^H, and we

make sure we have an *in* asset, as described above. During an iteration, if we encounter any ties, we pick the first critical asset to change state, and we push away the other ones by ϵ as described above.

This direct implementation works well in practice. The answers produced by the modified algorithm are indistinguishable from the "true" answers for the unmodified problem when displayed with the typical few decimal places of accuracy.

A variant of this practical approach is to use the first three techniques at the beginning of the algorithm to guarantee the strictness and correctness of the starting segment, but ignore the problem of ties altogether in the middle of the algorithm during the iterations. While ties are common, and cycles are possible in theory, in practice cycles never seem to happen even with ties.

This practical approach is quite unsatisfying in theory, of course, because we are changing the original problem as stated, even if only by a little bit.

There is a theoretical solution which satisfies the purists. It turns out that the problems we have been discussing are another manifestation of similar issues that appear in the theory of mathematical programming, where the simplex algorithm used to solve linear programming problems and the variations of the simplex algorithm used to solve non-linear programming problems also suffer from degeneracies, ties, and cycling problems.

In mathematical programming, the standard theoretical technique for solving these problems is based on the same basic idea we presented above, but in a formal way. Instead of adding and subtracting actual small values ϵ whenever there is a problem, we effectively add a whole new set of variables ϵ, ϵ^2, ϵ^3, and so on to the problem. It is possible to prove using arguments involving polynomials and their roots that with this technique ties and cycles are impossible, and the computed result for the modified problem is also exactly the same as the solution to the original problem. It is possible to implement this theoretical solution algorithmically, at the cost of increased space and time requirements to perform all the computations for the additional variables.

In [6], Markowitz adapts this technique to his critical-line algorithm and works through the proof that it does indeed resolve the degeneracy, tie and cycling issues. He says that these "techniques are available if needed." We do not pursue the details of this formal theoretical solution further here.

5.6 Corner Portfolios, Critical Lines and C-Fund Separation

For a valid state vector s, the two efficient portfolios at the segment endpoints A^s_{min} and A^s_{max} are called *corner portfolios*. The interior efficient portfolios in the segment are linear combinations of the two corner portfolios. Markowitz [7] calls this a *critical line*.

Suppose there is a set of $C-1$ valid state vectors with C corner portfolios whose solution segments completely cover all possible values of A over its domain $(0, \infty)$. Every possible efficient portfolio is a linear combination of two of the C corner portfolios. Sharpe [9] calls this the *C-fund separation theorem*. This is the constrained version of the two-fund separation theorem for unconstrained problems.

5.7 A Performance Optimization

Markowitz [7] mentions the following useful optimization which avoids having to do an expensive matrix inversion operation at each iteration of the critical-line algorithm.

When moving from a segment s to the next or previous segment s', we can compute the new $\tilde{V}_{s'}^{-1}$ matrix from the old $\tilde{V}_s^{-1}$ matrix directly, using one vector/matrix multiplication operation and one matrix/matrix multiplication operation.

Define the row vector r to be row k of $\tilde{V}_{s'}$. This is the only row where $\tilde{V}_{s'}$ and $\tilde{V}_s$ differ.

If asset k is switching to state *down* or *up*:

$$r_j = \begin{cases} 0 & \text{if } j \neq k \\ 1 & \text{if } j = k \end{cases}$$

If asset k is switching to state *in*:

$$r_j = \begin{cases} \rho_{k,j} & \text{if } j \leq n \\ 1 & \text{if } j = n+1 \end{cases}$$

Compute:

$$z = r\tilde{V}_s^{-1}$$

Define E to be the identity matrix with row k replaced by z:

$$E = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ z_1 & \cdots & z_k & \cdots & z_{n+1} & \\ & & & \ddots & & \\ & & & & 1 & \end{pmatrix}$$

Then:

$$\tilde{V}_{s'} = E\tilde{V}_s$$
$$\tilde{V}_{s'}^{-1} = \tilde{V}_s^{-1}E^{-1}$$

where it is easily verified that E^{-1} is:

$$E^{-1} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ -z_1/z_k & \cdots & 1/z_k & \cdots & -z_{n+1}/z_k \\ & & & \ddots & \\ & & & & 1 \end{pmatrix}$$

6 Generalizations and Other Approaches

The version of the critical-line algorithm we have presented here is a special case of the more general solution developed by Markowitz.

Markowitz' general version of the problem permits an arbitrary number of extra equality constraints in addition to the budget constraint, and he permits arbitrary linear inequality constraints, not just low and high bound constraints.

The theorems and proofs are much the same for the general problem, although the intuitions behind them become a bit more opaque, at least for beginners, which is why we chose to develop the solution for the simpler problem in this paper.

In some ways, however, the mathematics becomes more elegant and somewhat easier, which is often the case with generalizations. For example, we no longer have to deal with three *in*, *up* and *down* states. We instead have only two states, which Markowitz calls *in* and *out*. This reduces the number of separate cases which need to be considered in the proofs. There are now only two kinds of state transitions, from *in* to *out* and from *out* to *in*, rather than the four kinds we used in this paper.

Markowitz' treatment, which centers around the computational details of matrix manipulations, also reveals how similar the critical-line algorithm is to standard linear and non-linear programming algorithms. Indeed, these kinds of "tableau" algorithms were all being actively discovered and developed by researchers during the same time period, after World War II and during the 1950's. In [7] he addresses this similarity in more detail, and discusses in an appendix how his algorithm can be implemented as yet another variation of the simplex method, quite similar to the Lemke-Howson variation of the simplex method that is often used to solve non-parametric convex quadratic programming problems.

Our presentation is more theoretical and formal and less computational than Markowitz'. It uses more calculus, and the Kuhn-Tucker conditions play a more central role. In this sense our approach is closer to that outlined briefly by Sharpe in [10]. We feel that Sharpe's approach is more intuitive and better reveals the underlying economic meaning of the equations, but this is almost certainly a matter of personal taste.

7 The Limitations of Portfolio Optimization

Markowitz' theory of portfolio optimization, especially the unconstrained theory, is at the heart of modern finance.[9] Together with Sharpe's Capital Asset Pricing Model, which is a direct extension of the theory, the models provide a flawed but not unreasonable first-order approximation to how financial markets work. They are the departure point for further extensions and modifications such as modern intertemporal and multi-factor models and asset pricing theories. Their theories teach us many important practical lessons, especially the importance of diversification and the central role that is played by the capitalization-weighted total market portfolio, two lessons which also remain central in the modern theories.

There are, however, severe problems with trying to use portfolio optimization theory to actively manage real-life portfolios.

One problem is the difficulty of estimating the input parameters and the sensitivity of the algorithm to relatively small variations in those parameters, especially the important expected return parameters.

Estimating expected returns is notoriously difficult and controversial. Given the large amount of noise in the prices of risky assets like stocks, it is statistically impossible to accurately estimate expected returns from even long historical time series data with more accuracy than a percent or two. Trying to estimate expected returns based on fundamental considerations is even more difficult, and it is impossible to trust any such estimate with an accuracy of more than a percent or two.[10] The optimization algorithm is simply too sensitive to this kind of inaccuracy to be useful.

The algorithm is also sensitive to the other parameters. As an example, consider a simple two asset problem where the assets are domestic and foreign stocks. If we make the not totally unreasonable simplifying assumption that the expected returns and standard deviations of the two assets are the same, say 10% and 20% respectively, with a correlation of 0.7, we get the solution where a 50/50 asset allocation is optimal for all levels of risk aversion. If we take currency risk into account, however, which increases the volatility of foreign stocks when denominated in domestic currency, and we assume a correlation of 0 between currency and stocks, the optimal asset allocation changes dramatically. For example, a seemingly small increase in the standard deviation of foreign stocks from 20% to 22% changes the optimal asset allocation to roughly 1/3 foreign

[9] The constrained theory and the critical-line algorithm do not appear to be anywhere near as central as is the unconstrained theory to the subsequent development of the discipline of financial economics. Thus, in the unlikely event that the patient reader has actually read this far, he has probably wasted his time, unless he finds the mathematics interesting in its own right, as does the author.

[10] Consider the wide range of such return forecasts provided by academic and other "experts" on an almost daily basis. Which experts do you trust? Which numbers do you use? How accurate are they?

and 2/3 domestic![11]

Another problem is the complex interplay between all the different assets which makes it problematic to attempt to consider asset subsets in isolation.[12] For example, suppose there are only three assets A, B and C. An optimization analysis of A and B in isolation may reveal some optimal asset allocation, say 50/50. When C is added to the analysis, however, the result can change dramatically. For the same level of risk aversion, the new analysis may have a completely different suballocation to the portion of the portfolio that contains A and B. The allocation might change to 10/90 for A/B, or 90/10, when we add C to our considerations. This makes the portfolio optimization problem something of an "all or nothing" proposition, a difficult task indeed.

A third problem is that the theory rests on the assumption of a single factor risk model. This makes the theory inappropriate for use with multi-factor risk models. For example, the Fama-French model has three risk factors, with a multi-dimensional efficient frontier. Most of the efficient portfolios in that model, including the market portfolio, are not mean-variance efficient in the Markowitz model. See Cochrane [2, 3] for interesting discussions of these issues.[13]

While these problems should not be understated, there is an even more fundamental problem involving market equilibrium and competition.

Modern financial institutions match buyers with sellers almost instantaneously. Thus market prices for securities adjust quickly to reflect wealth and information-weighted aggregated beliefs as to the underlying value of the securities. Markets are nearly always in supply/demand equilibrium.

Markets are also extremely competitive. Analysts use all the tools available to them to investigate opportunities to buy and sell securities and asset classes, including the theories and tools described in this paper.

Thus, the output of an optimizer can have an asset or asset class weighted with non-market proportions for the representative investor only if the inputs used to the optimization problem do not reflect the wealth and information-weighted aggregated beliefs of other analysts.

In other words, an analyst who uses an optimizer to make asset allocation

[11] A good deal of "home bias" is almost certainly irrational. But at least over short horizons where the theory of purchasing power parity is not a consideration, and for investors for whom it is not possible to hedge currency risk (or it is costly to hedge), it appears that some significant degree of home bias is justified.

[12] This is admittedly a sin we just committed in the previous example.

[13] It is interesting to speculate, however, about what might happen if we add human capital as another major asset that must be considered. It might be possible to reconcile the Fama-French model with CAPM if, as some current research suggests, value stocks have a higher correlation with labor income for the representative investor than do growth stocks. A complete reconciliation along these lines is probably too much to hope for, however. In any case, this conjecture about value stocks and labor income is only a conjecture, not a proven fact, and it might just be wishful thinking.

decisions can succeed in beating the market only if his estimates of the inputs to the optimizer are significantly more accurate than those used by competing analysts. Such an analyst has to be consistently smarter than his competitors to make an abnormal profit by a large enough margin to overcome his expenses.

The high level of talent in the financial analysis world combined with the extraordinary high level of competition in that world must cause us to be suspicious of the possibility of earning any consistent abnormal profits by using optimizers. Indeed, the long and detailed historical record does not reveal any serious instances of such success. If it were possible, one would think that over the last 50 years since Markowitz made his discoveries, we would be able to identify at least one person who has been a consistent winner based on his use of Markowitz' theories. If that person exists, it is a well-kept secret.

The proper way to think of these theories is that the models they build are a good way to at least start to think about how markets work, but they do not give us any kind of edge in trying to beat those markets.

Put yet another way, portfolio optimizer programs by themselves do not help one become smarter than other investors. It is impossible for such a program by itself to give an investor an advantage, because everyone has access to these programs.

Eugene Fama has been reported to have said that if one feels like wasting some time, it's OK to play around with optimization programs, but it is indeed a waste of time. We largely concur with this sentiment, except that playing around with such programs can often help students visualize the mathematics and underlying economic ideas when they are first becoming exposed to the theories. For practical applications, however, they are useless.

For all of these reasons, we strongly advise individual investors to resist the temptation to try to use optimization analysis to make asset allocation decisions about their personal portfolios. Doing so is extremely dangerous, and usually leads to disastrous results. Individual investors are much better advised to simply hold inexpensive total market index funds, and forget about trying to beat the market.

8 Examples and an Implementation

For examples of constrained optimization problems and the critical-line algorithm used to solve them, see the "PortOpt" program. PortOpt is a free cross-platform Java program which is a companion to this paper. The program is available at the author's web site at the address given on the title page of this paper.

PortOpt has an "Examples" menu which contains commands to generate a number of sample problems, included both unconstrained and constrained versions of all five examples from [8], various end-cases, and randomly generated problems. Users may also enter their own problems using a parameter entry panel.

While PortOpt is a full implementation of the critical-line algorithm as described in this paper, with no limit on the number of assets, its purpose is educational, not practical. It should be useful for students learning the theory (constrained or unconstrained), and it is occasionally useful for doing experiments as part of some kinds of research projects. It is not, however, useful for trying to build market-beating monstrosities in real life, for all of the reasons given in the previous section.

PortOpt displays solutions both numerically and graphically. Both displays are interactive.

For the numeric solution, a calculator is provided that can be used to enter any of the relevant parameters and calculate all of the other ones (e.g., find the efficient portfolio and its expected return given the standard deviation.)

The graph shows the efficient frontier curve with markers at the corners, along with a slider control at the bottom that can be used to move up and down the curve. As the slider is moved, the efficient portfolios at the corresponding points on the curve are displayed in bar chart and numeric format. This visualization aid makes it easy to see how the solution changes and how the various assets change state at the corners.

We also display the coefficient of relative risk aversion A, the parabolic iso-elastic indifference curve that is tangent to the efficient frontier curve at the solution, the certainty equivalent, and other information that may be relevant and educational depending on the context.

For unconstrained problems which contain a risk-free asset and have at least three assets, we display the efficient frontier for the risky asset subset and the optimal risky portfolio M which is at the tangency point of the two efficient frontiers. This helps illustrate the two-fund separation theorem.

For constrained problems we also compute and graph the unconstrained efficient frontier. This helps see how much effect the constraints have on the solution.

Problem sets can be saved to disk in XML format and read back in later.

We do careful error checking of parameters entered by the user. We check for inconsistent constraints (empty feasible sets), inconsistent correlations (non-positive semidefinite correlation matrices), linearly dependent assets, and arbitrage opportunities. In each of these cases we issue a detailed error message explaining the problem.

For dealing with degeneracies, PortOpt uses the variant of the practical approach described in section 5.5. The techniques described in that section are used to guarantee the strictness and correctness of the starting state vector, but the problem of ties and cycles is ignored. Cycles are detected, however, and in the extremely unlikely event that one occurs, an "internal error" message is displayed.

While PortOpt's implementation of the critical-line algorithm does not impose any limit on the number of assets other than enough available memory to hold the vectors and matrices used by the algorithm, with a large number of assets the human interface of the program begins to perform poorly due to the large number of Java Swing human interface components. For example, with 20 assets, the correlation matrix display and parameter entry area requires $20^2 = 400$ text components. 20 assets seems to be about the upper limit on the size of the problems that the human interface can accommodate gracefully.

While we have made no attempt to optimize performance other than using the optimization described in section 5.7, the program is reasonably fast. The following tests were done an a 1 GHz Apple Macintosh PowerBook G4 computer:

20 assets or fewer: Under 1 second.
100 random assets: 8 seconds.
200 random assets: 1 minute 26 seconds.

For the linear algebra computations, we use CERN's Colt Distribution [1]. For generating random asset problems, we use the algorithm described in Lin and Bendel [5].

This paper is the user manual for the program. There is no other documentation.

References

[1] CERN. The colt distribution. http://hoschek.home.cern.ch/hoschek/colt/.

[2] John Cochrane. New facts in finance. *Economic Perspectives (Federal Reserve Bank of Chicago)*, 1999.

[3] John Cochrane. Portfolio advice for a multifactor world. *Economic Perspectives (Federal Reserve Bank of Chicago)*, 1999.

[4] Joel Friedman. Linear complementarity and mathematical (non-linear) programming. http://www.math.ubc.ca/~jf/courses/340/pap.pdf.

[5] Shang P. Lin and Robert B. Bendel. Generation of population correlation matrices with specified eigenvalues (algorithm as 213). *Applied Statistics (Royal Statistical Society)*, 1985.

[6] Harry M. Markowitz. The optimization of a quadratic function subject to linear constraints. *Naval Research Logistics Quarterly*, 1956.

[7] Harry M. Markowitz. *Portfolio Selection: Efficient Diversification of Investments*. Blackwell, second edition, 1991. (Originally published in 1959.).

[8] John Norstad. Portfolio optimization: Part 1 – unconstrained portfolios. http://www.norstad.org/finance, Sep 2002.

[9] William F. Sharpe. Macro-investment analysis (an electronic work-in-progress). http://www.stanford.edu/~wfsharpe/mia/mia.htm.

[10] William F. Sharpe. *Portfolio Theory and Capital Markets*. McGraw-Hill, 2000. (Originally published in 1970.).