Portfolio Optimization under Uncertainty
Part l

J-F. Pusztaszeri
CS522-Computational Tools and Methods in Finance

1.0 Asset and Liability Models

1.1 General stochastic programming model

This is an extension of linear/nonlinear programming using probabilistic coefficients. The following is a *multi-stage (linear) stochastic programming problem with recourse and with discretely distributed random elements* (Wets 1996):

Given

$$t \in (1, \ldots, T) \qquad \xi^s_t = (A^s_{t1}, \ldots, A^s_{tt}, b^s_t, c^s_t),\ \ \forall s \in S$$

$$\xi = (\xi_1, \ldots, \xi_T) \qquad \mathrm{Prob}[\vec{\xi} = (\xi^s_1, \ldots, \xi^s_T)] = p_s$$

$$\underline{\xi}_t \equiv (\xi_1, \ldots, \xi_T)$$

minimize

$$\sum_S p_s \sum_{t=1}^{T} c^s_t x^s_t$$

such that

$$\sum_{r=1}^{t} A^s_{tr} x^s_r = b^s_t, \quad t = 1, \ldots, T, \quad s \in S \qquad \text{deterministic constraints}$$

$$x^s_r = x^r_r, \quad \text{if } \underline{\xi}^s_t = \underline{\xi}^r_t, \quad t = 1, \ldots, T, \quad s \in S \qquad \text{non-anticipativity constraints}$$

$$v^s_t \le x^s_r \le u^s_t \quad t = 1, \ldots, T, \quad s \in S$$

In this formulation, every time step represents a *stage* (i.e., a node in some decision tree where branching takes place). S is generally a set of *scenarios*, each scenario corresponding to a specific realization of the uncertain parameters of the model. Note that scenarios span all the stages, so they can be interpreted as specific paths from root to end nodes in the decision tree. The non-anticipativity constraints restrict the variable from anticipating a future state: a decision at time t can only be based on the observations accumulated up to t. These constraints are stochastic.

By convention, there is no randomness in the first stage ($t = 1$), i.e., $\xi_1^s = \xi_1^r$, and $x_1^s = x_1^s, \forall r, s \in S$.

1.2 Solution methods

There are three classes of numerical methods which are employed to solve the general problem:

- quasi-gradient methods
- stochastic decomposition
- Aggregation methods

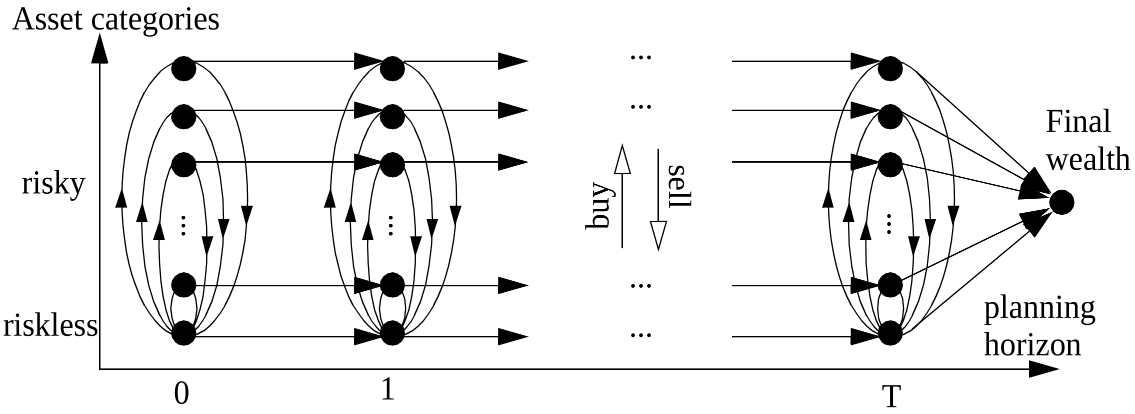
It should be noted that these methods are not as yet up to the standards of implementation of techniques currently used in commercial integer and linear programming solvers. For detail, see (Mulvey and Ziemba 1995).

For specific types of stochastic programming problems, the following are used:

- logarithm penalty for chance constraints
- extended linear quadratic techniques for discrete-time systems (Jensen and King 1992).

1.3 Generalized Network Model (GNM) for portfolio management

This is a special case of the general model described above, taken from (Mulvey 1994; Mulvey and Ziemba 1995). This model returns an optimal position of funds which must be allocated to different asset categories over several stages. This is best explained using a network diagram, where the arcs represent cash flows between assets:



The horizontal axis represents a discrete set of stages, up to some arbitrary planning horizon T. The vertical axis represents assets which enter the portfolio. A node on the diagram represent a given stage-asset pair. The first element of the asset category is a riskless asset (to be seen as a cash account) to/from which a cash flow can take place to any other asset.in the portfolio. The portfolio is rebalanced when a cash flow between assets actually takes place. A time period represents a stage only if rebalancing occurs at that time. From this point on, it is assumed that each time step represents a stage.

Notes: the direction of the arc can be inverted to represent a short-sell.

A stochastic optimization model is built on this network by making use of the flow conservation constraints implied by it. It is important to note that, while this model is general enough, many asset and liability management problems require side constraints which are not captured by the network approach. A good example of a more general instance is given in (Carino, Kent et al. 1994). The so-called Russell-Yasuka-Kasai model described in it is useful to look at for a comparative study of a stochastic programming approach and a Markowitz model applied to the same problem. The superiority of the former is clearly demonstrated.

Definitions:

The planning horizon is divided into two parts:

$t = \{0, 1, \ldots, \tau\}$, during which investment decisions are made, and

$t = \{\tau + 1, \ldots, T\}$, during which the present value of the liability cash flow (i.e., risk) at the end of the planning period is computed.

The investment categories in the set I are represented by the index $i = \{1, \ldots, I\}$. One defines likewise a set S of scenarios, indexed by $s = \{1, \ldots, S\}$. The following stochastic decision variables must be defined:

- $x_{i,t}^{s}$: investment in asset category i at the start of period t under scenario s. This variable defines the state of the system after rebalancing has taken place.
- $r_{i,t}^{s}$: single period return for asset i at t and under s.
- $y_{i,t}^{s}$: decision variable to purchase asset i at t and under s.
- $z_{i,t}^{s}$: decision variable to sell asset i at t and under s.

This last variable is decomposed into two parts which model the source of the cash flow: a dividend part is denoted by $d_{i,t}^{s}$ and the cash flow generated by actually selling the asset is represented by $u_{i,t}^{s}$, so that $z_{i,t}^{s} = d_{i,t}^{s} + u_{i,t}^{s}$.

- e_{t-1}^{s}: cash outflow at period $t-1$ under s.
- g_{t-1}^{s}: cash inflow at period $t-1$ under s (i.e., a cash flow representing savings).
- l_{t-1}^{s}: payment on loan principal at period $t-1$ and scenario s. This represents the liability of the investment process. In this model, all the borrowing is done over a single period.

Symmetric transaction costs are defined for $y_{i,t}^{s}$ and $z_{i,t}^{s}$, and are denoted by ζ_i. With reference to the network model, these determine the costs of the arcs.

One need the following state variable as well:

- b_{t}^{s}: amount of borrowing at t under s, with a borrowing rate given by δ_{t}^{s}.

To simplify notation, the following is defined:

- $\sum_{i} x_{i,t}^{s} = c_{t}^{s}$, $s \in S, t \in T$: total asset for a given scenario and a given period
- $x_{i,t}^{s}(1 + r_{i,t}^{s}) = v_{i,t}^{s}$, $i \in I, s \in S, t \in T$: accumulated cash flow at the end of period t and before rebalancing in asset i takes place.

Constraints:

The constraints are established to ensure flow conservation in the underlying network. For each asset category i,

$$x_{i,t}^{s} = v_{i,t-1}^{s} + y_{i,t-1}^{s}(1 - \zeta_i) - z_{i,t-1}^{s} \quad , \quad i \in I, s \in S, t \in T \tag{1.1}$$

and for the nodes at each period t,

$$c_t^s = g_{t-1}^s + \sum_i (u_{i,t-1}^s(1-\zeta_i) + d_{i,t-1}^s) - \sum_i y_{i,t}^t - b_{i,t-1}^s(1-\delta_t^s) + c_{t-1}^s - e_{t-1}^s - l_{t-1}^s + b_t^s \;\; , s \in S, t \in T \tag{1.2}$$

All stochastic decision variables are subject to the non-anticipativity constraints. For given i, t, two variables based on scenarios with the same history up to t must be equal, i.e.,

$$x_{i,t}^s = x_{i,t}^r \text{ , for } r, s \text{ equal up to } t \tag{1.3}$$

for $x_{i,t}^s$, and likewise for all other decision variables.

Constraints may also apply to restrict the range of investment in specific asset categories, without changing the structure of the model.

Objective functions:

The elementary building block of any objective function is a measure of wealth at τ, the last time step at which investment decision are made. This is expressed as

$$w_\tau^s = \sum_i v_{i,t}^s - PV(l_{\tau,T}^s) - b_\tau^s . \tag{1.4}$$

The terms of the right hand side represent the sum of the total wealth to time τ, the present value of the liability (or risk) from τ to T, and the loan portion yet to be repaid at τ.

A generic objective function to be maximized has the form

$$\sum_s p_s U(w_\tau^s) \tag{1.5}$$

for $U(w_\tau^s)$ an arbitrary function of w_τ^s.A special case of this expression is given by

$$E(w_\tau^s) - \beta Var(w_t^s) . \tag{1.6}$$

1.4 A special case: the Markowitz mean-variance model

Recall that the first step of a stochastic programming problem is deterministic. By setting $\tau = 1$ and by using Equation 1.6 as an objective function, the original Markowitz model is recovered.

2.0 Portfolio Optimization with the RiskWatch Software

2.1 Software functionality

RiskWatch is a risk management software tool. Its prime use in the industry is for calculating the risk position of a corporation over its investments, and to facilitate reporting of this position. The software consists of several “modules”, as shown in Figure 2-1.

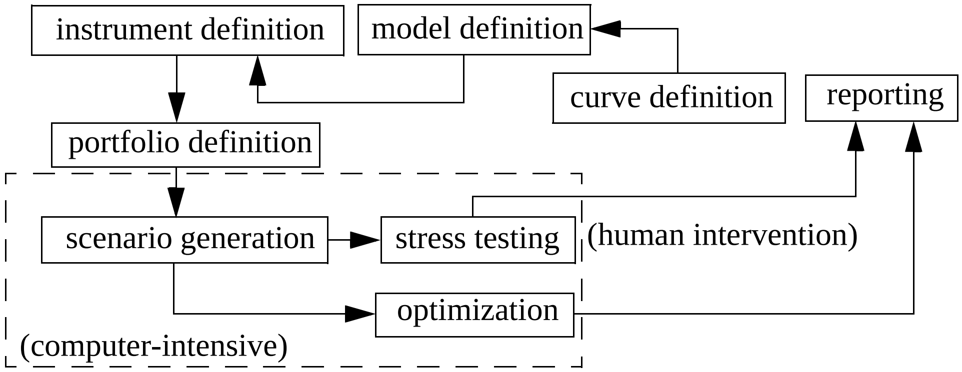


Figure 2-1. RiskWatch operating diagram

The instrument, curve and portfolio modules essentially provide a data structure (objects and attributes) allowing users to define securities, yield curves and (possibly nested) portfolios. Instruments and curves are defined first, and are then employed to construct portfolios. The model definition module includes a vast library of pricing functions, and specific choices must be matched to instruments as they are being created. The present value of instruments is calculated in the portfolio module

The scenario generation, stress testing and optimization modules are computer-intensive. The scenario module relies on the perturbation of a curve (or pricing model) to generate a set of scenarios by Monte Carlo simulation. The stress testing module propagates the portfolio forward in time under specific scenarios and user-defined guidelines. This is a sensitivity analysis step, and the price vectors of portfolios are calculated for specified horizons. The optimization module is redundant to the stress testing module, and its aim is to automate the latter and, to some extent, the user intervention that it requires. Mathematical optimization methods, and in particular linear programming, are used here. Calls to the CPLEX linear programming solver are made from this module.

2.2 The RiskWatch optimization module

Three portfolio optimization techniques are used in this module, and their usage is summarized in Table 2-1. (Algorithmics 1997) can be consulted for a full description of these methods. The simplest one is the *scenario optimization* approach, which is used for hedging, index or cash flow tracking, and to replicate a large portfolio with a much smaller one (portfolio compression). Optimization is only performed over a measure or risk, taking the form of a tracking error. A generalization of this approach is provided by two variants of the same method, so-called risk/reward analysis, which extend the risk measure to one

Usefulness/technique	scenario optimization	risk/reward (K/MR(K))	risk/reward (K-MR(K))
hedging positions	****************	****************	****************
creating synthetic securities		****************	****************
tracking a market index	****************		
portfolio compression	****************		
cash flow matching	****************		
efficient frontier (Markowitz)		****************	****************

Table 2-1. Use of optimization tools in RiskWatch

capturing the trade-off between risk and excess return. The analog of a Markowitz efficient frontier can be created using this approach.

What follows is a detailed description of these three techniques. It should be noted that in the stochastic programming framework discussed earlier, all are *single-stage* methods. Given I instruments and S scenarios, all three share the common set of definitions summarized in Figure 2-2, where $\vec{q}$ is the $I \times 1$ vector of present values of the portfolio, b is the present value of the target, $\vec{p}$ is the $S \times 1$ vector of scenario probabilities, D is the $I \times S$ matrix of portfolio values at target date under all scenarios, and $\vec{b}$ is the $S \times 1$ vector of benchmark value at the target date.

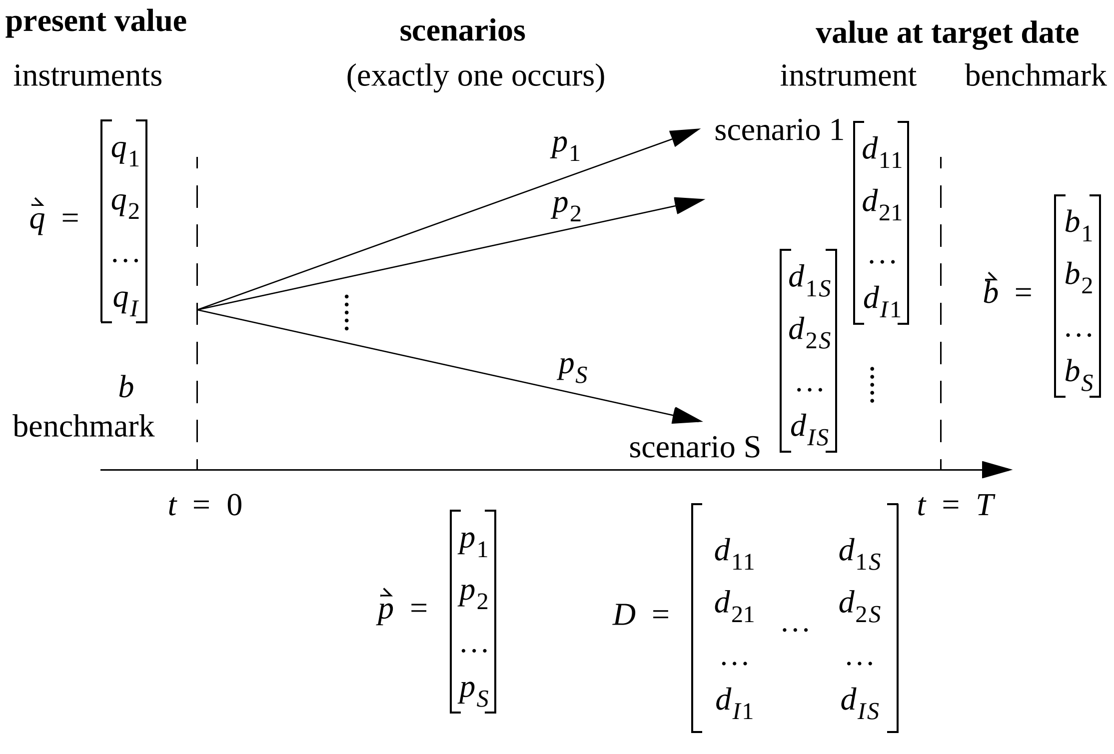


Figure 2-2. RiskWatch scenario-based optimization framework

2.3 Scenario optimization

This approach is described in (Dembo 1991). Its goal is to find the portfolio which best replicates a benchmark over a given horizon. A first step consists in defining a measure of “closeness” of target and replicating portfolio. This is achieved by defining $\vec{x}$ as the $I \times 1$ vector of positions in each instrument entering the replicating portfolio, and constructing the $S \times 1$ vector $\vec{y}$ representing the *tracking error*, as

$$\vec{y} = D^T\vec{x} - \vec{b}. \tag{2.3}$$

For a perfect match between replication and target, the tracking error is zero, i.e., $D^T\vec{x} = \vec{b}$. Note that the elements of the position vector may be positive (we buy the corresponding instrument), negative (sell) or zero (instrument is not selected).

Based on this definition of tracking error, a “regret” function (which we ultimately want to minimize) over all scenarios is defined as

$$R = \|\vec{y}\|. \tag{2.4}$$

The choice of norm is arbitrary. See (Dembo and King 1992) for a thorough description of regret functions. Only two norms are currently implemented in RiskWatch. With the l_1 (absolute value) norm, and with $\vec{y}$ a random vector, Equation 2.4 becomes

$$R = \sum_{s=1}^{S} p_s|y_s| \tag{2.5}$$

For the l_∞ (maximum error) norm, this expression is

$$R = max_S|y_s| \tag{2.6}$$

The definition of the tracking error can be refined to describe either an upside and a downside error, defined respectively as

$\vec{y}^+$, with $y_s^+ = max(0, y_s)$, $\forall s \in S$ and $\vec{y}^-$, with $y_s^- = |min(0, y_s)|$, $\forall s \in S$. A regret function utilizing these errors will be referred to as an upside and a downside regret respectively. Note that $\vec{y} = \vec{y}^+ - \vec{y}^-$.

The scenario optimization problem consists of minimizing the regret function (Equation 2.4, and its special cases Equation 2.5 and Equation 2.6, with either expression equipped with a choice of tracking error), subject to the S tracking constraints (Equation 2.3). Optionally, one may add a set of linear constraints describing the maximum amount to be spent on a portfolio, the trading limits, and bounds on the tracking errors. These expressions involve the vector $\vec{q}$ and the scalar c and can take many forms. Note that with the choice of the l_1 norm, the problem transforms into a linear programming problem by decomposing the vectors of variables $\bar{x}$ and $\vec{y}$ into sums of vectors with non negative elements, as done earlier, to ensure that the nonnegativity conditions of the linear program are satisfied, that is

$$\vec{x} = \vec{x}^+ - \vec{x}_S^-. \tag{2.7}$$

2.4 Risk/reward optimization

This is a generalization of the previous method, in which the regret function is used in conjunction with an additional constraint to define a relationship between risk (regret) and return on investment. The new constraint is called excess profit and is constructed as follows:

For any scenario s, the change in value of the replicating portfolio from 0 to T is given by $\sum_i (d_{is} - q_i)x_i$. Likewise, the change of value of the benchmark in the same interval is given by $b_s - b$. so the gain in the replicating portfolio in this interval and under s is $\sum_i (d_{is} - q_i)x_i - (b_s - b)$. The expected gain for the same is therefore

$$(b - \vec{q}^T\vec{x}) + \left\|D^T\vec{x} - \vec{b}\right\| \tag{2.8}$$

In component form, using the l_1 norm, and after rearranging the terms, this expression can be analyzed in terms of cash flows, as shown in Figure 2-9. This indicates that only two rebalancing operations are taking place, one at time $t = 0$ when we the benchmark portfolio is sold and the replicating portfolio is bought (as determined by the position vector $\vec{x}$), and the other at the horizon date when the trade is exactly inverted.

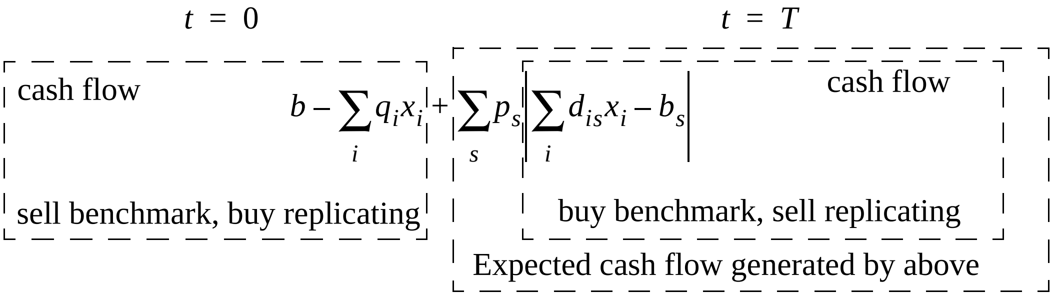


Figure 2-9. Cash flows in expected gain

The next step consists of generating an efficient frontier. Do do so, one starts with the original scenario optimization problem, removes the initial cash flow constraints alluded to in the in preceding section (keeping them would prove too restrictive), and adds the following parametric constraint based on the expected gain

$$(b - \vec{q}^T\vec{x}) + \left\|D^T\vec{x} - \vec{b}\right\| \geq K, \tag{2.10}$$

where K is a non-zero scalar, which indicates a minimum expected excess profit specified by an investor. If the l_1 norm is used, the problem is, as before, a linear programming problem, but which is now *parametric in the right-hand side* with parameter K. For illustration purpose, one can choose the downside regret and write the problem as

$$\min \vec{p}^T\vec{y}^- \tag{2.11}$$

s.t.

$$-\vec{y}^{+} + \vec{y}^{-} + D(\vec{x}^{+} - \vec{x}^{-}) = \vec{b}, \quad \forall s \in S \tag{2.12}$$

$$\vec{p}^{T}(\vec{y}^{+} - \vec{y}^{-}) - \vec{q}^{T}(\vec{x}^{+} - \vec{x}^{-}) \geq K - b \tag{2.13}$$

$$\vec{x}^{+}, \vec{x}^{-}, \vec{y}^{+}, \vec{y}^{-} \geq 0 \tag{2.14}$$

Solving this problem, one gets a a minimum regret as a function of K. It can be shown that minimizing a linear programming problem parametric in the right-hand side produces a piecewise-linear and convex function, as shown in Figure 2-15. See Section 6.8 of (Bazaraa, Jarvis et al. 1977) for a complete analysis of parametric linear programs. The diagram represents the optimal risk/reward frontier $MR_{\vec{b}}(K)$. A point in the plane represents a replicating portfolio, infeasible with respect to the constraints of the problem if lying above the curve, feasible but not optimal if lying below it. The target portfolio is at the origin. The break points on the curve correspond to values K for which alternative optimal dual solutions exist, and different optimal bases span the time intervals delimited by the break points. The diagram also indicates a region where no feasible solution exists, which is determined by the constraints used in the problem. The example (Equation 2.11 to Equation 2.14) here always has a solution.

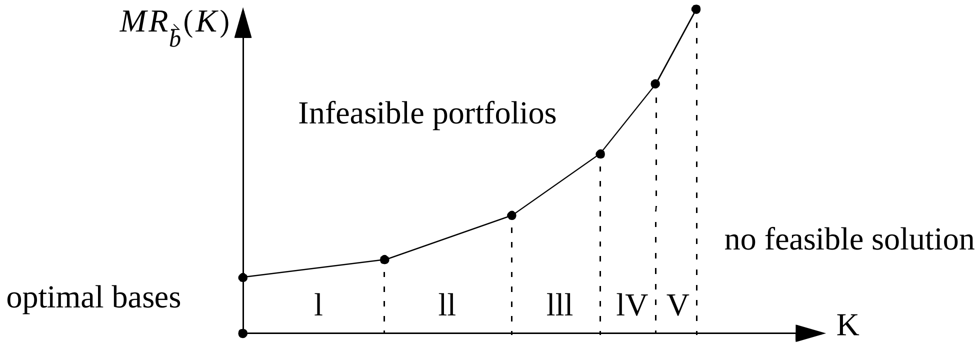


Figure 2-15. The risk/reward frontier

Choosing a portfolio on the risk/reward curve

This is achieved by constructing an empirical measure of reward to risk which uses the risk/reward curve $MR_{\vec{b}}(K)$. With analogy to the *Sharpe ratio,* the reward to risk ratio is defined as

$$W(K) = \frac{K}{MR_{\vec{b}}(K)}, \qquad MR_{\vec{b}}(K) > 0, \tag{2.16}$$

Maximizing this reward to risk ratio is equivalent to finding a point (points) on the frontier such that the zero-intercept line tangent to it (them) has minimal slope. This is one variant of the risk/reward optimization technique used in RiskWatch. The other variant is obtained from defining

$$W(K) = K - \alpha MR_{\vec{b}}(K)\,. \tag{2.17}$$

There is a clear analogy between this expression and the Markowitz mean variance formulation, seen earlier as a special case of the generalized network model for asset and liability analysis (Equation 1.6). In this case, the expected wealth is replaced by the expected excess return, and the variance of the wealth function by the minimum regret describing risk. Both formulations make use of a damping factor (α or β). The two RiskWatch variants are illustrated in Figure 2-18. The choice of replicating portfolios is determined uniquely by K (the optimal solution may not be unique however).

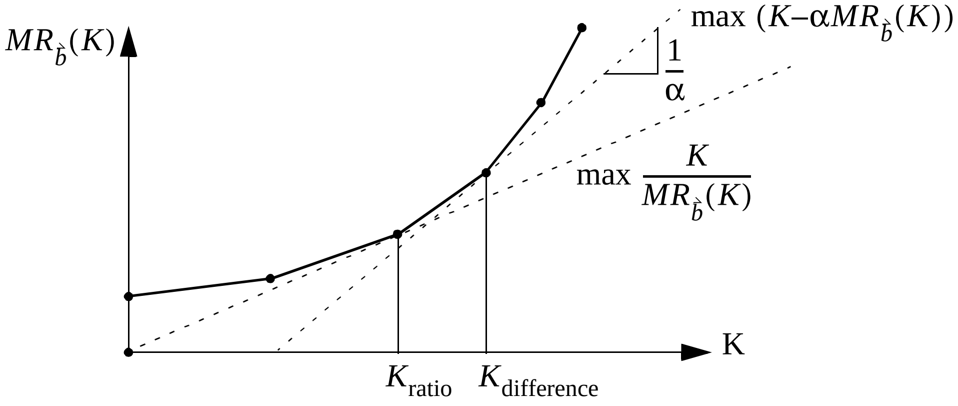


Figure 2-18. Portfolio selection on the risk/reward frontier

2.5 Change of target

This section addresses the computational advantage created by using a linear programming formulation (and implicit use of the $l1$-norm) to generate a risk/reward curve when a change of target is made. A risk/reward curve is assumed to have been calculated for the example described by Equation 2.11 to Equation 2.14. A change of target is then made as follows:

$$b \to b^* \text{ and hence } \vec{b} \to \vec{b}^* \tag{2.19}$$

The question is whether one need to solve this new problem from scratch. To answer this, one takes the dual[1] of the original problem to obtain

$$\max\ \vec{b}^T \vec{\pi} + (K - b)\lambda \tag{2.20}$$

s.t.

$$D\vec{\pi} - \lambda\vec{q} = \vec{0}, (\vec{x}^{+}, \vec{x}^{-}) \tag{2.21}$$

$$0 \le \vec{\pi} - \lambda\vec{p} \le \vec{p}, (\vec{y}^{+}, \vec{y}^{-}) \tag{2.22}$$

$$\lambda \ge 0 . \tag{2.23}$$

Here, $\vec{\pi}$ is an $S \times 1$ vector of variables associated with the S constraints described by Equation 2.12, and λ is the variable associated with Equation 2.13. These are the dual variables of the problem. Observing that the target $(b, \vec{b})$ only appears in the objective function, the weak duality theorem of linear programming ensures that the dual objective function evaluated at the optimal solution of the original problem will be a lower bound for the solution to the new problem created using the target map of Equation 2.19. Knowing this lower bound can speed up calculation significantly.

2.6 Model extensions

The discussion so far has involved only single period (and single stage) problems. The extension of the methodology used in RiskWatch to multi-stage stochastic programming problems is particularly difficult, as it requires the construction of a risk/reward framework which must handle, among others, the non-anticipativity constraints discussed earlier. Also, the simple scenario approach is unwieldy, since the growth of scenarios is exponential in the number of stages.

The extension of the model to a multi-period system is, however, trivial. For a fixed set of scenarios, the target and price vectors can now be assumed to be time-dependent. The regret function now takes the form

$$R = \sum_{t=0}^{T} \left\| D_t^T \vec{x} - \vec{b} \right\| \tag{2.24}$$

as shown in Figure 2-25. In the diagram, $\vec{d}_{i,j}$ represents column i of the D matrix at time t.

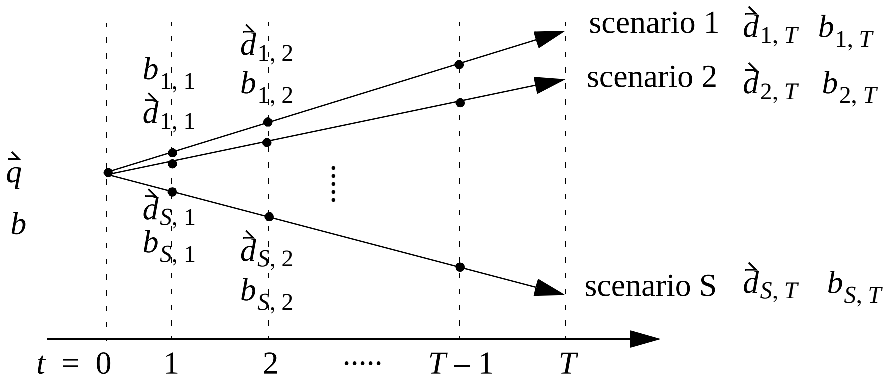


Figure 2-25. Multi-period extension of the optimization module

---

[1] i.e., if $\min\{\vec{c}^{T}\vec{x} : A\vec{x} \le \vec{b}, \vec{x} \in \Re^{n}{}_{+}\}$ is the original linear programming problem, its *dual* is expressed as $\max\{\vec{u}^{T}\vec{b} : \vec{u}^{T}A \ge \vec{c}^{T}, \vec{u} \in \Re^{m}{}_{+}\}$.

References

Algorithmics (1997). Scenario Optimization and Risk/Reward Analysis. RiskWatch User's Manual, Toronto, Algorithmics, Inc.

Bazaraa M. S., Jarvis J. J., et al. (1977). Linear Programming and Network Flows. New York, John Wiley & Sons.

Carino D. R., Kent T., et al. (1994). "The Russel-Yasuda Kasai model: An asset/liability model for a Japanese insurance company using multistage stochastic programming." Interfaces **24**(1): 29-49.

Dembo R. S. (1991). "Scenario Optimization." Annals of Operations Research **30**: 63-80.

Dembo R. S. and King A. J. (1992). "Tracking Models and the Optimal Regret Distribution in Asset Allocation." Applied Stochastic Models and Data Analysis **8**: 151-157.

Jensen D. L. and King A. J. (1992). "Frontier: A Graphical Interface for Portfolio Optimization in a Piecewise Linear-Quadratic Risk Framework." IBM Systems Journal **31**(1): 62-70.

Mulvey J. M. (1994). "An asset-liability investment system." Interfaces **24**(3): 22-33.

Mulvey J. M. and Ziemba W. T. (1995). Asset and Liability Allocation in a Global Environment. Finance. R. A. Jarrow, V. Maksimovic and W. T. Ziemba. Amsterdam, Elsevier. **9:** 435-463.

Wets R. J.-B. (1996). "Challenges in Stochastic Programming." Mathematical Programming **75**: 115-135.