Computational Economics **20:** 177–190, 2002.

Local Search Techniques for Constrained Portfolio Selection Problems

ANDREA SCHAERF
Dipartimento di Ingegneria Elettrica, Gestionale e Meccanica, Università di Udine, Via delle Scienze 208, I-33100, Udine, Italy; E-mail: schaerf@uniud.it

Abstract. We consider the problem of selecting a portfolio of assets that provides the investor a suitable balance of expected return and risk. With respect to the seminal *mean-variance* model of Markowitz, we consider additional constraints on the cardinality of the portfolio and on the quantity of individual shares. Such constraints better capture the real-world trading system, but make the problem more difficult to be solved with exact methods.

We explore the use of local search techniques, mainly tabu search, for the portfolio selection problem. We compare the combine previous work on portfolio selection that makes use of the local search approach and we propose new algorithms that combine different neighborhood relations. In addition, we show how the use of randomization and of a simple form of adaptiveness simplifies the setting of a large number of critical parameters. Finally, we show how our techniques perform on public benchmarks.

Key words: portfolio optimization, mean-variance portfolio selection, local search, tabu search

1. Introduction

The *portfolio selection* problem consists in selecting a portfolio of *assets* or *securities* that provides the investor a given expected return and minimizes the *risk*. One of the main contributions on this problem is the seminal work by Markowitz (1952), who introduced the so-called *mean-variance* model, which takes the variance of the portfolio as the measure of risk. According to Markowitz, the portfolio selection problem can be formulated as an optimization problem over real-valued variables with a quadratic objective function and linear constraints.

The basic Markowitz' model has been modified in the recent literature in various directions. First, Konno and Yamazaki (1991) propose a linear versions of the objective function, so as to make the problem easier to be solved using available software tools, such as the simplex method. On the other hand, with the aim of better capturing the intuitive notion of risk, Konno and Suzuki (1995) and Markowitz et al. (1993) studied more complex objective functions, based on the notions of *skewness* and *semi-variance*, respectively. Furthermore, several new constraints have been proposed, in order to make the basic formulation more adherent to the real world trading mechanisms.

Among others, there are constraints on the maximal cardinality of the portfolio (Chang et al., 2000; Bienstock, 1996) and on the minimum size of trading lots (Mansini and Speranza, 1999). Finally, Yoshimoto (1996) and Glover et al. (1996) consider multiperiod portfolio evolution with transaction costs.

In this paper we consider the basic objective function introduced by Markowitz, and we take into account two important additional constraints, namely the *cardinality* constraint and the *quantity* constraint, which limit the number of assets and the minimal and maximal shares of each individual asset in the portfolio, respectively.

The use of local search techniques for the portfolio selection problem has been proposed by Rolland (1997) and Chang et al. (2000). In this paper, we depart from the above two works, and we try to improve their techniques in various ways. First, we propose a broader set of possible neighborhood relations and search techniques. Second, we provide a deeper analysis on the effects of the parameter settings and employ adaptive evolution schemes for the parameters. Finally, we show how the interleaving of different neighborhood relations and different search techniques can improve the overall performances.

We test our techniques on the benchmarks proposed by Chang et al., which come from real stock markets.

2. Portfolio Selection Problems

We introduce the portfolio select problem in stages. First, we introduce the basic unconstrained version of Markowitz. Subsequently, we introduce the specific constraints of our formulation.

Given is a set of *n assets*, $A = \{a_1, \dots, a_n\}$. Each asset a_i has associated a real-valued *expected return* (per period) r_i, and each pair of assets $\langle a_i, a_j \rangle$ has a real-valued *covariance* σ_{ij}. The matrix $\sigma_{n \times n}$ is symmetric and each diagonal element σ_{ii} represents the *variance* of asset a_i. A positive value R represents the desired expected return.

A portfolio is a set of real values $X = \{x_1, \dots, x_n\}$ such that each x_i represents the fraction invested in the asset a_i. The value $\sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$ represents the variance of the portfolio, and it is considered as the measure of the risk associated with the portfolio. Consequently, the problem is to minimize the overall variance, still ensuring the expected return R. The formulation of the basic (unconstrained) problem is thus the following.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$$

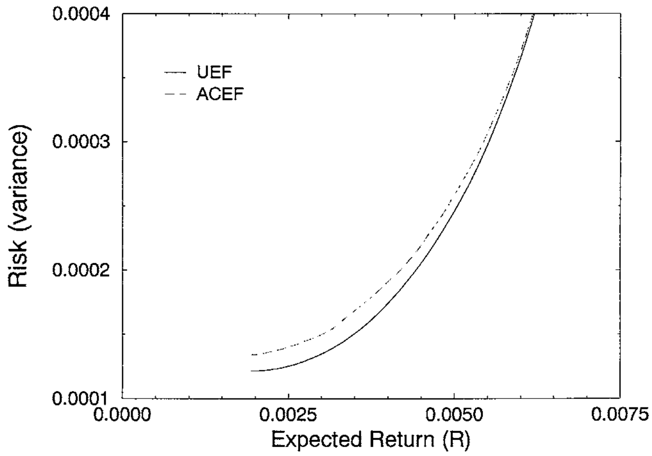

Figure 1. UEF and ACEF for instance no. 4.

$$\text{s.t.} \quad \sum_{i=1}^{n} r_i x_i \leq R \tag{1}$$

$$\sum_{i=1}^{n} x_i = 1 \tag{2}$$

$$0 \leq x_i \leq 1 \qquad (i = 1, \ldots, n) \tag{3}$$

This is a quadratic programming problem, and nowadays it can be solved optimally using available tools[1] despite the NP-completeness of the underlying decision problem.

By solving the problem as a function of R, we obtain the socalled *unconstrained efficiency frontier* (UEF), that gives for each expected return the minimum associated risk. The UEF for one of the benchmark problems of Chang et al. (2000) is provided in Figure 1 (solid line).

In our formulation, we consider the following two additional constraint types:

Cardinality constraint: The number of assets that compose the portfolio is limited. That is, a value $k \leq n$ is given such that the number of i's for which $x_i > 0$ is at most k.

Quantity constraints: The quantity of each asset i that is included in the portfolio is limited within a given interval. Specifically, a minimum ϵ_i and a maximum δ_i for each asset i are given, and we impose that either $x_i = 0$ or $\epsilon_i \leq x_i \leq \delta_i$.

These two constraint types can be modeled by adding n *binary* variables $z_1, \ldots, z_n$ and the following constraints.

$$\sum_{i=1}^{n} z_i \leq k \tag{4}$$

$$\epsilon_i z_i \leq x_i \leq \delta_i z_i \quad (i = 1, \ldots, n) \tag{5}$$

The variable z_i equals to 1 if asset a_i is included in the portfolio, $z_i = 0$ otherwise. The resulting problem is now a mixed integer programming problem, and iis it much harder to be solved using conventional techniques.

We call CEF the analogous of the UEF for the constrained problem. Given that we do not solve the problem with an exact method, we do not actually compute the CEF, but what we call the ACEF (*approximate constrained efficiency frontier*. Figure 1 shows the ACEF (dashed line) we computed for the same instance for the values $\epsilon_i = 0.01$, $\delta = 1$ (for $i = 1, \ldots, n$), and $k = 10$.

Notice that when the return is high, the distance to UEF is very small because typically large quantities of a few assets are used, and thus Constraints (4) and (5) don't come into play.

3. Local Search

Local search is a family of non-exhaustive general-purpose techniques for optimization problems. A local search algorithm starts from an initial state s_0, which can be obtained with some other technique or generated randomly, and enters a loop that *navigates* the search space, stepping from one state s_i to one of its neighbors s_{i+1}. The neighborhood is usually composed by the states that are obtained by some local change (called *move* from the current one.

The most common local search techniques are *hill climbing* (HC), *simulated annealing* (SA), and *tabu search* (TS). We describe here very briefly TS which is the technique that gave best results for one application (a full description of TS is out of the scope of this paper. see, e.g., Gover and Laguna, 1997)

At each state s_i, TS explores exhaustively the current neighborhood $N(s_i)$. Among the elements in $N(s_i)$, the one that gives the minimum value of the cost function becomes the new current state s_{i+1}, independently of the fact whether the cost of s_i is less or greater than the cost of s_{i+1}.

In order to escape from local minima, the so-called *tabu list* is used, which determines the forbidden moves. This list stores the most recently accepted moves. The *inverses* of the moves in the list are forbidden.

The stop criterion is based on the so-called *idle iterations*: The search terminates when it reaches a given number of iterations elapsed from the last improvement.

Different local search techniques can be combined and alternated to give rise to complex algorithms.

In particular, we explore what we call the *token-ring* strategy: Given an initial state s_0 and a set of basic local search techniques $t_1, \ldots, t_q$, that we call *runners*, the token-ring search makes circularly a run of each t_i, always starting from the best solution found by the previous runner t_{i-1} (or $_q$ if $i = 1$).

The full run stops when it performs one round without an improvement by any of the runners, whereas each component runner t_i stops according to its own criterion.

4. Portfolio Selection by Local Search

In order to apply local search techniques to portfolio selection we need to define the search space, the neighborhood structures, the cost function, and the selection rule for the initial state.

4.1. SEARCH SPACE AND NEIGHBORHOOD RELATIONS

For representing a state, we make use of two sequences $L = \{a_{l_1}, \ldots, a_{l_p}\}$ and $S = \{x_{l_1}, \ldots, x_{l_p}\}$ such that $a_{l_i} \in A$ and x_{l_i} is the fraction of a_{l_i} in the portfolio. All assets $a_j \notin L$ have the fraction x_j implicitly set to 0. With respect to the mathematical formulation, having $a_i \in L$ corresponds to setting z_i to 1.

We enforce that the length p of the sequence L is such that $p \leq k$, that the sum of x_{l_i} equals 1, and that $\epsilon_{l_i} \leq \delta_{l_i}$ for all elements in L. Therefore, all elements of the search space satisfy Constraints (2)–(5). Constraint (1) instead is not always satisfied and it is included in the cost function as explained below.

Given that the problem variables are continuous, the definition of the neighborhood relations refers to the notion of the *step* of a move m, which is a real-valued parameter q, with $0 < q < 1$, that determines the quantity of the move. Given a step q, we define the following three neighborhood relations:

idR ([i]ncrease, [d]ecrease, [R]eplace):

Description: The quantity of a chosen asset is increased or decreased. All other shares are changed accordingly so as to maintain the feasibility of the portfolio. If the share of the asset falls below the minimum it is replaced by a new one.

Attributes: $\langle a_i, s, a_j \rangle$ with $a_i \in A, s \in \{\uparrow, \downarrow\}, a_j \in A$

Preconditions: $a_i \in L$ and $a_j \notin L$

Effects: If $s = \uparrow$ then $x_i := x_i \cdot (1+q)$, otherwise $x_i := x_i \cdot (1-q)$. All values $x_k - \epsilon_k$ are renormalized so as to maintain the property that x_k's add up to 1. We renormalize $x_k - \epsilon_k$ and not x_k to ensure that no asset rather than a_i can fall below the minimum.

Special cases: If $s = \downarrow$ and $x_i(1-q) < \epsilon_i$, then a_i is deleted from L and a_j is inserted with $x_j = \epsilon_j$. If $s = \uparrow$ and $x_i(1+q) > \delta_i$, then x_i is set to δ_i.

Reference: Revised version of Chang et al. (2000).

idID ([i]ncrease, [d]ecrease, [I]nsert, [D]elete):

Description: Similar to idR, except that the deleted asset is not replaced and insertions of new assets are also considered.
Attributes: $\langle a_i, s\rangle$ with $a_i \in A$, $s \in \{\uparrow, \downarrow, \hookrightarrow\}$
Preconditions: If $s = \downarrow$ or $\uparrow$ then $a_i \in L$. If $s = \hookrightarrow$ then $a_i \notin L$.
Effects: If $s = \uparrow$ then $x_i := x_i \cdot (1 + q)$; if $s = \downarrow$ then $x_i \cdot (1 - q)$; if $s = \hookrightarrow$ then $a_i : \epsilon_i$ is inserted into L. The portfolio is repaired as explained above for idR.
Special cases: If $s = \downarrow$ and $x_i(1 - q) < \epsilon_i$, then a_i is delected from L, and it is not replaced. If $s = \uparrow$ and $x_i(1 + q) > \delta_i$, then x_i is set to δ_i.

TID ([T]ransfer, [I]nsert, [D]elete):

Description: A part of the share is transferred from one asset to another one. The transfer can go also toward an asset not in the portfolio, which is then inserted. If one asset falls below the minimum it is deleted.
Attributes: $\langle a_i, a_j\rangle$ with $a_i \in A$, $a_j \in A$
Preconditions: $a_i \in L$
Effects: The share x_i of asset a_i is decreased by $q \cdot x_i$ and x_j is increased by the same quantity. If $a_j \notin L$ than it is inserted in L with the quantity $q \cdot x_i$.
Special cases: The quantity transferred is larger than $q \cdot x_i$ in the following two cases: (i) If after the decrease of x_i we have that $x_i < \epsilon_i$ then also the remaining part of x_i is transferred to a_j. (ii) If $a_j \notin L$ and $q \cdot x_i < \epsilon_j$ then the quantity transferred is set to ϵ_j.
Reference: Extended version of Rolland (1997).

Notice that idR moves never change the number of assets in the portfolio, and thus the search space is not connected under idR. Therefore, the use of idR for the solution of the problem is limited. The relation idID in fact is a variant of idR that overcomes this drawback.

Notice also that under all three relations the size of the neighborhood is not fixed, w.r.t. the size of L, but it depends on the state. In particular, it depends on the number of assets that would fall below the minimum in case of a move that reduces the quantity of that asset. For example, for idR, the size is linear, $2 \cdot |L|$, if no asset a_i is such that $x_i(1 - q) < \epsilon_i$, but becomes quadratic, $|L| + |L| \cdot (n - |L|)$, if all assets are in such conditions.

We now define the inverse relations, which determines which moves are tabu. Our definitions are the following: For idR and idID, the inversse of m is any move with the same first asset and different arrow. For TID, it is the move with the two assets exchanged.

4.2. COST FUNCTION AND INITIAL STATE

Recalling that all constraints but Constraint (1) are automatically satisfied by all elements of the search space, the cost function $f(X)$ is composed by the objective function and the degree of violation of Constraint (1). Specifically, we define two components, $f_1(X) = \max(0, \sum_{i=1}^{n} r_i x_i - R)$ and $f_2(X) = \sum_{j=1}^{n} \sigma_{ij} x_i x_j$, which take into account the constraint and the objective function, respectively. The overall cost function is a linear combination of them: $f(X) = w_1 f_1(X) + w_2 f_2(X)$.

In order to ensure that a feasible solution is found, w_1 is (initially) set to a much larger value than w_2. However, during the search, w_1 is let to vary according to the so-called *shifting penalty* mechanism (see e.g., Gendreau et al., 1994): If for K consecutive iterations Constraint (1) is satisfied, w_1 is divided by a factor γ randomly chosen between 1.5 and 2. Conversely, if it is violated for H consecutive iterations, the corresponding weight is multiplied by a random factor in the same range (where H and K are parameters of the algorithm).

Notice that evaluation of the cost change associated to a move is computationally quite expensive for both idR and idID, due to the fact that a move changes the fraction of all assets in L. The computation of the cost is instead much cheaper for TID.

The initial state is selected as the best among $I = 100$ random portfolios with k assets. However, experiments show that the results are insensitive to I.

4.3. LOCAL SEARCH TECHNIQUES

We implemented all the three basic techniques, namely HC, SA, and TS, for all neighborhood relations. HC, which performs only improving and sideways moves, is implemented both using a random move selection and searching for the best move at each iteration (steepest descent). SA, which for the sake of brevity is not described in this paper, is implemented in the 'standard' way described in (Johnson et al., 1989). TS is implemented using a tabu list of variable size and the shifting penalty mechanism.

We also implemented several token-ring procedures. The main idea is to use one technique t_1, with a large step q, in conjunction with another t_2, with a smaller step. The technique t_1 guarantees diversification, whereas t_2 provides a 'finer-grain' intensification.

The step q is not kept fixed for the entire run, but it is allowed to vary according to a random distribution. Specifically, we introduce a further parameter d and for each iteration the step is selected with equal distribution in the interval $q - d$ and $q + d$.

Due to its limited exploration capabilities, idR is used only for t_2. Other combinations, of two or three techniques, have also been tested as described in the experimental results.

Table I. The benchmark instances.

No.	Origin	Assets	UEF	% Diff.
1	Hong Kong	31	1.55936	0.00344745
2	Germany	85	0.412213	2.53845
3	UK	89	0.454259	1.92711
4	USA	98	0.502038	4.69426
5	Japan	225	0.458285	0.204786

4.4. BENCHMARKS AND EXPERIMENTAL SETTING

We experiment our techniques on 5 instances taken from real stock markets.[2] We solve each instance for 100 equally distributed values for the expected return R.

We set the constraint parameters exactly as Chang et al. (2000): $\epsilon = 0.1$ and $\delta_i = 1$ for $i = 1, \ldots, n$, and $k = 10$ for all instances.

Given that the constraint problem has never been solved exactly, we cannot provide an absolute evaluation of our results. We measure the quality of our solutions in average percentage loss w.r.t. the UEF (available from the web site). We also refer to the ACEF, which we obtain by getting, for each point, the best solution found by the set of all runs using all techniques. The ACEF has been computed using a very large set of long runs, and reasonably provides a good approximation of the optimal solution of the constrained problem.

Table I illustrates, for all instances the original market, the average variance of UEF (multiplied by $\times 10^3$ for convenience), and the percentage average of the difference between ACEF and UEF.

Notice that the problem for which the discrepancy between UEF and ACEF is highest is no. 4 (with 4.69%). For this reason we illustrate our results for no. 4, in which the differences are more tangible.

Except for no. 1, all other instances give qualitatively similar results and they require almost the same parameter settings. Instance no. 1 instead, whose size is considerably smaller than the others, shows peculiar behaviors and requires completely different settings. Specifically, it requires shorter tabu list and much smaller steps.

5. Experimental results

In the following experiments, we run 4 trials for every point. For each parameter setting, we therefore run 2000 trials (4 trials $\times 100$ points $\times 5$ instances). Except for the first point of the UEF, in one of the four trials the initial state is not random, but it is the final state of the previous solved point of the UEF. The number of iterations is chosen in such a way that each single trial takes approximately 2 seconds (on a

Table II. Comparison of simple solvers.

Tech.	Nhb	Fixed step		Random step	
		Step	% Diff.	Base step	% Diff.
TS	idID	0.5	6.31568	0.4	5.60209
TS	TID	0.5	5.42842	0.3	4.85423
TS	idR	0.4	5.4743	0.4	5.4621
SA	TID	0.4	53.7006	0.4	56.5798
SA	idID	0.2	118.698	0.5	113.735
HC	TID	0.2	29.2577	0.2	29.039
HC	idID	0.2	41.4734	0.1	41.0438

300 MHz Pentium II, using the C++ compiler egcs-2.91.66), and therefore each test runs for approximately and hour.

We experimented with 20 different values of the step q. Regarding the step variablility d, preliminary experiments show that the best value is q, which means the step varies between 0 and $2q$. In all the following experiments, d is either set to 0 (fixed step) or is set to q (random step).

Regarding the parameters related to the shifting penalty mechanism, the experiments show that the performances are quite insensitive to their variations as far as they are in a given interval. Therefore, we set such parameters to fixed values throughout all our experiments ($h = 1$, $K = 20$).

5.1. SINGLE SOLVERS

The first set of experiments regards a comparison of algorithms using the three neighborhood relations idID, idR, and TID in isolation. Given that the search space is not connected under idR, the relation idR is run for initial states of all sizes from 2 to 10 (it is therefore granted a much longer running time). For the other two, idID and TID, we start always with an initial state of 10 assets.

Table II shows the best results for TS for both fixed and random steps, and the corresponding step valuse. For TS, the tabu list length is 10–25, and the maximum number of idle iterations is set to 1000.

The table shows also also the best performance of HC and SA for TID and idID. For the sake of fairness, we must say that the parameter setting of SA has not been investigated enough.

The results in Table II show that TS works much better than the others, and TID works better than idR and idID. They also show that the randomization of the step improves the results significantly.

Table III. Comparison of composite solvers.

Runner 1		Runner 2		Runner 3		% Diff
Nbh	Step	Nbh	Step	Nbh	Step	% Diff
TID	0.4	TID	0.05	–	–	4.70872
TID	0.4	TID	0.04	TID	0.004	4.70866
TID	0.4	idR	0.05	–	–	4.70804
TID	0.4	idR	0.05	TID	0.01	4.71221
idID	0.4	idID	0.04	–	–	5.06909
idID	0.3	idID	0.03	idID	0.003	4.99406
idID	0.4	idR	0.05	–	–	4.99206
idID	0.4	idR	0.04	idID	0.004	5.16368

5.2. COMPOSITE SOLVERS

Table III shows the best results for token-ring with various combinations of two or three neighborhoods all using TS and random steps. Notice that we consider as token-ring solver also the interleaving of the same technique with differnt steps.

The table shows that the best results are obtained using the combination of TID and idR, but TID with different steps performs almost as good. This results are very close to the ACEF (4.69426%), which is obtained using also much longer runs (24 hours each).

In conclusion, the best results (around 4.7%) are obtained by token ring solvers with random steps. Further experiments show that the most critical parameter is the size of the step of t_1, which must be in the range [0.3, 0.6]. They also show that using alternation of fixed steps the best result obtained is 4.84883.

5.3. EFFECTS OF CONSTRAINTS ON THE RESULTS

We conclude with a set of experiments that highlights the role played by constraints 4 and 5 on the problem. Figure 2 shows the best results for instance no. 4 for different values of the maximum number of assets k (ϵ_i and δ_i are fixed to the values 0.01 and 1).

The results show that the effect of the constraint decreases quite steeply when increasing k. The effect is negligible for $k > 30$.

Figure 3, instead, shows how the quality of the portfolio decreases while increasing the minimum quantity (ϵ_i). In order to focus on the minimum quantity constraint, we use a high value for the maximum cardinality ($k = 20$) so as to make the effect of the corresponding cardinality constraint less visible.

We don't show the results for different values of δ_i because the constraint on maximum quantity is less important from the practical point of view.

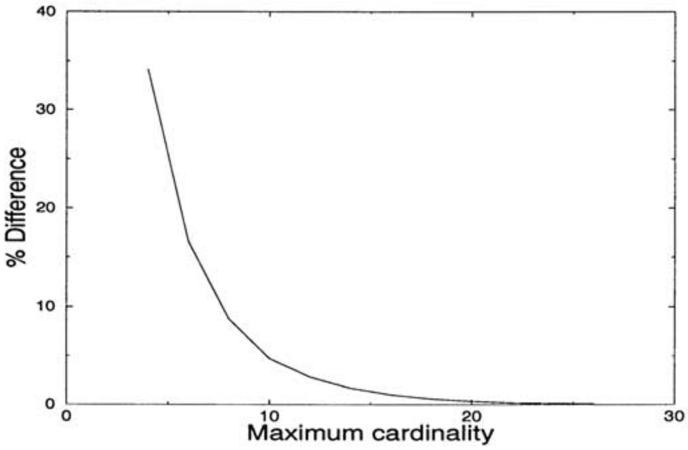

Figure 2. Results for different values of the maximum cardinality (k).

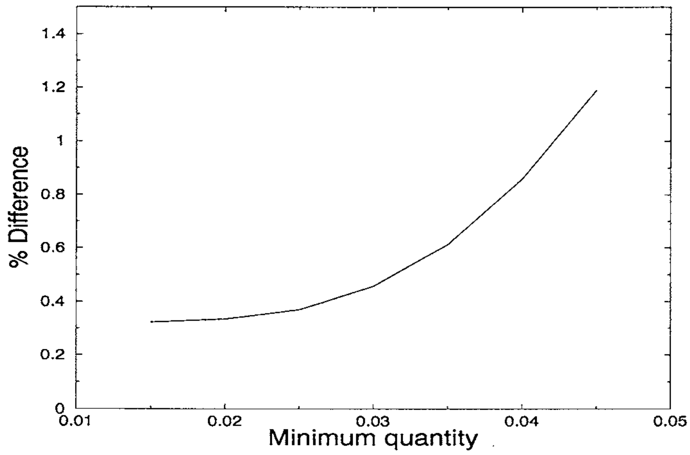

Figure 3. Results for different values of the minimum quantity (ϵ_i)

6. Related Work

This problem has been previously considered by Chang et al. (2000), who implemented three solvers based on TS, SA, and genetic algorithms (GA). Their experimental results that GA and SA work better than TS. Even though the TS procedure is not completely explained in the paper, we believe that this 'defeat' of TS in favor of SA and GA is due to the fact that their version of TS is not sufficiently optimized.

The neighborhood relation used by Chang et al. is a variant of idR. The difference stems from the fact that in their case a move m is represented by only the pair $\langle a_i, s\rangle$ and the replacing asset a_j is not considered part of m, but it is randomly generated whenever necessary. This definition makes incomplete the exploration of the full neighborhood because the quality of a move $\langle a_i, \downarrow\rangle$ may depends of the randomly generated a_j. In our work, instead, all possible replaces a_j are analized. In addition, the application of a move $m = \langle a_i, \downarrow \rangle$ is *non-deterministic*, and therefore it is not clear which is the definition of the inverse of m, and the definition of the tabu mechanism. Finally, with respect to our version, their TS misses the following important features: shifting penalty mechanism, random step (they use the fixed value 0.1), and variable-size tabu list.

Even though Chang et al. solve the same problem instances, a fair comparison between their and our results is not possible for two reasons:

First, they formulate Constraint (4) with the equality sign, i.e. $\sum_{i=1}^{n} z_i = k$, rather than as an inequality. As the authors themselves admit, constraining the solution to an exact number of assets in the portfolio is not meaningful by itself, but it is a tool to solve the inequality case. They claim that the solution of the problem with the inequality can be found solving their problem for all values from 1 to k. Unfortunately, though, they provide results only for the problem with equality.

Second, they do not solve a different instance for each value of R, but (following Perold, 1984), they reformulate the problem without Constraint (1) and with the following objective function: $f(X) = \lambda f_1(X) + (1 - \lambda) f_2(X)$. The problem is then solved for different values of λ. The quality of each solution is measured not based on the risk difference w.r.t. the UEF for the same return R, but using a metric that takes into account the distance to both axis. The disadvantage of this approach is that they obtain the solution for a set of values for R which are not an homogeneously distributed. Therefore their quality cannot be measured objectively, but it depends on how much they cluster toward the region in which the influence of Constraints (4) and (5) is less or more strong. In addition, these sets of points are not provided, and thus the results are not reproducible and not comparable.

Rolland (1997) considers the unconstrained problem therefore his results are not comparable. He introduces the TID neighborhood which turned out to be the most effective. Although, the definition of Rolland is different because he considers only transfers and no insertions and deletions. This is because, for the unconstrained problem, all assets can be present in the portfolio at any quantity, and therefore there is no need of inserting and deleting. The introduction of insert and delete moves is our way to adapt his (successful) idea to the constrained case.

Rolland makes use of a tabu list of fixed length equal to $0.4 \cdot n$, thus linearly related to the number of assets. He alternates the fixed step value 0.01 with the fized value 0.001, shifting every 100 moves. Our experiments confirm the need for two (and no more than two) step values, but they show that those values are too small for the constrained case. In addition, for the constrained problem, randomization works better than alternating two fixed values.

7. Conclusions and Future Work

We compared and combined different neighborhood relations and local search strategies to solve a version of the portfolio selection problem which involves a mixed-integer quadratic problem. Rather than exploring all techniques in the same depth, we focussed on TS that turned out to be the most promising from the beginning.

This work shows also how adaptive adjustments and randomization could help in reducing the burden of parameter setting. For example, the choice of the step parameter turned out to be particularly critical.

We solved public benchmark problems, but unfortunately no comparison with other results is possible at this stage.

In the future, we plan to adapt the current algorithms to different versions of the portfolio selection problem, both discrete and continuous, and to related problems. Possible hybridization of local search with other search paradigms, such as genetic algorithms, will also be investigated.

Acknowledgements

I thank T.-J. Chang and E. Rolland for answering all questions about their work, and K.R. Apt and M. Cadoli for comments on earlier drafts of this paper.

Notes

[1] For example, an online portfolio selection solver is available at http://www-fp.mcs.anl.gov/otc/Guide/CaseStudies/port/

[2] Available at the URL http://mscmga.ms.ac.uk/jeb/orlib/portfolio.html

References

Bienstock, D. (1996). Computational study of a family of mixed-integer quadratic programming problems. *Mathematical Programming*, **74**, 121–140.

Chang, T.-J., Meade, N., Beasley, J.E. and Sharaiha, Y.M. (2000). Heuristics for cardinality constrained portfolio optimisation. *Computers and Operational Research*, **27**, 1271–1302.

Gendreau, M., Hertz, A. and Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management Science*, **40** (10), 1276–1290.

Glover, F. and Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers.

Glover, F., Mulvey, J.M. and Hoyland, K. (1996). Solving dynamic stochastic control problems in finance using tabu search with variable scaling. In I.H. Osman and J.P. Kelly (eds.): *Meta-Heuristics: Theory & Applications*. Kluwer Academic Publishers, pp. 429–448.

Johnson, D.S., Aragon, C.R., McGeoch, L.A. and Schevon, C. (1989). Optimization by simulated annealing: An experimental evaluation; part I, graph partitioning. *Operations Research*, **37** (6), 865–892.

Konno, H. and Suzuki, K.-I. (1995). A mean-variance-skewness portfolio optimization model. *Journal of the Operations Research Society of Japan*, **38** (2), 173–187.

Konno, H. and Yamazaki, H. (1991). Mean-absolute deviation portfolio optimization model and its applications to Tokio stock market. *Management Science*, **37** (5), 519–531.

Mansini, R. and Speranza, M.G. (1999). Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research* **114**, 219–233.

Markowitz, H. (1952). Portfolio selection. *Journal of Finance*, **7** (1), 77–91.

Markowitz, H. Todd, P., Xu, G. and Yamane, Y. (1993). Computation of mean-semivariance efficient sets by the critical line algorithm. *Annals of Operations Research*, **45**, 307–317.

Perold, A.F. (1984). Large-scale portfolio optimization. *Management Science*, **30** (10), 1143–1160.

Rolland, E. (1997). A tabu search method for constrained real-number search: Applications to portfolio selection. Technical Report, Dept. of accounting & management information systems. Ohio State University, Columbus.

Yoshimoto, A. (1996). The mean-variance approach to portfolio optimization subject to transaction costs. *Journal of the Operations Research Society of Japan*, **39** (1), 99–117.