J. Appl. Math. & Informatics Vol. 27(2009), No. 1 - 2, pp. 233 - 243
Website: http://www.kcam.biz

A LINE SEARCH TRUST REGION ALGORITHM AND ITS APPLICATION TO NONLINEAR PORTFOLIO PROBLEMS[1]

NENGZHU GU*, YAN ZHAO AND YAN GAO

ABSTRACT. This paper concerns an algorithm that combines line search and trust region step for nonlinear optimization problems. Unlike traditional trust region methods, we incorporate the Armijo line search technique into trust region method to solve the subproblem. In addition, the subproblem is solved accurately, but instead solved by inaccurate method. If a trial step is not accepted, our algorithm performs the Armijo line search from the failed point to find a suitable steplength. At each iteration, the subproblem is solved only one time. In contrast to interior methods, the optimal solution is derived by iterating from outside of the feasible region. In numerical experiment, we apply the algorithm to nonlinear portfolio optimization problems, primary numerical results are presented.

AMS Mathematics Subject Classification : 90C30, 65K05.
Key words and phrases : Portfolio optimization, convex programming, trust region method, line search, global convergence.

1. Introduction

Consider the following optimization problem

$$\begin{array}{ll} \min & f(x) \\ s.t. & x \in X, \end{array} \tag{1}$$

where $f(x)$ is a nonlinear function and X is a set that contains linear constraint conditions. In practice, many problems can be formulated as (1). For instance, portfolio problems, see section 4. In this paper, we assume that f is a convex twice continuously differentiable function and $X = \{x : Ax \leq b\}$, where A is an (m, n) matrix, b is an m-vector, x is an n-vector. For convenience of notation, we denote the derivative of $f(x)$ by g and $\nabla f(x)$ alternatively. We further assume that g is locally Lipschitzian and, for some $\bar{x} \in X$, $\{x \in X : f(x) \leq f(\bar{x})\}$ is bounded. Obviously, problem (1) has optimal solution.

Received November 7, 2007. Accepted April 6, 2008. *Corresponding author.
[1]Research was supported by the National Natural Science Foundation of China (No.10671126).
*School of Business, University of Shanghai for Science and Technology, Shanghai 200093, China.

It has been shown that trust region methods are desirable techniques for solving problem (1). For example, Byrd et al. [7], El-Alem [9], Gertz and Gill [11] and Byrd et al. [8]. In [13], Jiang and Qi proposed a Newton trust region algorithm for Convex SC^1 minimization problems. They gave priority to use Newton method to solve a relaxed problem (6), then let $x_{k+1} = x_k + d_k$ if line search condition $f(x_{k+1}) - f(x_k) \leq \sigma \nabla f(x_k)^T d_k$ is satisfied. Otherwise, they performed a trust region strategy to solve the subproblem (4). Our algorithm is inspired by their approach, the difference between our algorithm and Jiang-Qi algorithm is that we only need to solve the subproblem (4). If a trial step is not accepted, we incorporate the Armijo line search into the iteration, instead of resolving the subproblem by reducing the trust region radius.

This paper is organized as follows. In Section 2, we describe a algorithm for problem (1) that combines the Armijo line search technique and trust region step. In Section 3, we establish the global convergence for the algorithm. In Section 4, we apply the algorithm to nonlinear portfolio optimization problems and give a numerical example. Finally, our conclusions are given in Section 5.

2. Algorithm

Since problem (1) is a convex programming, solving (1) is equivalent to find $x^* \in X$ such that, for any $x \in X$,

$$g(x^*)^T (x - x^*) \geq 0, \tag{2}$$

or, equivalently, to find $x^* \in X$ such that

$$-g(x^*) \in N_x(x^*), \tag{3}$$

where $N_x(x^*)$ denotes the normal cone of the set X at x^*, that is,

$$N_x(x^*) = \{d : d^T(x - x^*) \leq 0\}, \forall x \in X.$$

Similar to [13], we use $\partial_B g(x)$ to denote the set of matrices G, where G is equal to the limit of some convergent sequence $\{\nabla g(\hat{x}_k)\}$ such that each $\hat{x}_k$ is a point in $\Re^n$ at which g is F-differentiable and the sequence $\{\hat{x}_k\}$ converges to x. Notice that g is locally Lipschitzian, thus $\partial_B g(x)$ is well defined. Furthermore, the convex hull of $\partial_B g(x)$ is the Clarke generalized Jacobian $\partial g(x)$, both $\partial g(x)$ and $\partial_B g(x)$ are nonempty and bounded.

Let $B = G + \beta I$, where $G \in \partial_B g(x)$, $\beta > 0$ and I is the $n \times n$ identity matrix. The subproblem of (1) is formulated as

$$\begin{aligned} \min_{d \in \Re^n} \quad & g_k^T d + \frac{1}{2} d^T B_k d = \phi_k(d) \\ s.t. \quad & \|d\| \leq \Delta_k, \quad d \in X \backslash x, \end{aligned} \tag{4}$$

where $\Delta_k > 0$ is a trust region radius. A relaxed problem of (4) is given as

$$\min_{d \in \Re^n} \quad g_k^T d + \frac{1}{2} d^T B_k d = \phi_k(d) \tag{5}$$

$$s.t. \qquad d \in X \backslash x.$$

Due to $f(x)$ is convex and g is locally Lipschitzian, it has been shown in [14] that every $G \in \partial_B g(x)$ is a symmetric and positive semidefinite matrix for all $x \in \Re^n$. This implies that B is positive definite matrix, thus (4) is strictly convex programming problem. There exists a unique solution for (4).

Traditional trust region methods solve an optimization problem iteratively. At each iteration, a trial step d_k is generated by solving the subproblem (4). If the ratio between the actual reduction and the predicted reduction is not less than a given constant, then the trial step is accepted. Otherwise, the subproblem (4) is resolved by reducing trust region radius until an acceptable step is found. Therefore, the subproblem may be solved several times at each iteration before an acceptable step is found, which can considerably increase the total cost of computation for large scale problems.

To improve efficiency of trust region methods, in the pioneering work [18], Necedal and Yuan presented a algorithm that combines trust region step with a backtrack line search technique for unconstrained optimization problem. When a trial step is not acceptable, their method performs a line search to find an iterative point instead of resolving the subproblem. Under reasonable conditions, they proved the convergence results. Numerical results showed that the combination of trust region step with line search is encouraging. More papers incorporated line searches into trust region methods can be seen in [17], [10-12], etc. Motivated by these proposed papers, we introduce an improved trust region method, our algorithm can be viewed as a combination of traditional trust region method and the Armijo line search for portfolio optimization problems. In contrast to interior methods, the optimal solution is derived by iterating from outside of the feasible region in our algorithm.

Since B_k is a positive definite matrix, we can perform an accurate solution for trust region subproblem. To ensure that d_k is not outside of a trust region, we compute d_k by

$$d_k = \begin{cases} -B_k^{-1} g_k, & if \ \ \|B_k^{-1} g_k\| < \Delta_k, \\ -\frac{\Delta_k}{\|B_k^{-1} g_k\|} B_k^{-1} g_k, & if \ \ \|B_k^{-1} g_k\| \geq \Delta_k. \end{cases} \tag{6}$$

This mechanism implies that in our algorithm the following condition

$$\phi_k(0) - \phi_k(d_k) \geq \tau \|g_k\| \min\{\Delta_k, \|g_k\|/\|B_k\|\},$$

where $\tau \in (0, 1)$ is a constant, is need not to be satisfied, which is an essential condition of inaccurate method for solving trust region subproblem.

Although several algorithms that combine trust region step with line searches have been presented in [10-12] and [17-18] respectively, these algorithms are proposed for unconstrained optimization problems. In addition, Approximate Hessian matrices B_k can not guaranteed to be positive definite in these algorithms, therefore inaccurate methods are used to solved subproblems in these algorithms. Compared with [10-12] and [17-18], the different points are that our algorithm is proposed for constrained optimization, and the subproblem is solved by accurate method.

Let the following notation to denote the distance between a set $S \subseteq \Re^n$ and a vector $x \in \Re^n$

$$dist\{x, S\} = inf\{\|x - y\| : y \in S\},$$

where $\|\cdot\|$ denotes a norm in $\Re^n$. We now state our algorithm as follows.

Algorithm 1

Step 1: Give $x_1 \in \Re^n$, $\Delta_1 > 0$. Choose constants β_0, c_1, c_2, μ, such that $\beta_0 > 0, 0 < c_1 < 1 < c_2$ and $\mu \in (0,1)$. Set $k := 1$. Compute $\zeta_1 = dist\{-g(x_1), N_x(x_1)\}$.

Step 2: Check termination, if x_k satisfies a prescribed stopping condition, then stop.

Step 3: Solve (4) accurately.

Step 4: Compute ρ_k by

$$\rho_k = \frac{f(x_k) - f(x_k + d_k)}{\phi_k(0) - \phi_k(d_k)}. \tag{7}$$

If $\rho_k \geq \mu$, go to Step 5. Otherwise, compute i_k, the minimum nonnegative integer i satisfies

$$f(x_k + \lambda^i d_k) \leq f_k + \delta \lambda^i g_k^T d_k. \tag{8}$$

Set $\alpha_k = \lambda^{i_k}$,

$$x_{k+1} = x_k + \alpha_k d_k, \tag{9}$$

and

$$\Delta_{k+1} \in [\|x_{k+1} - x_k\|, c_1 \Delta_k], \tag{10}$$

go to Step 6.

Step 5: Set

$$x_{k+1} = x_k + d_k, \tag{11}$$

and

$$\Delta_{k+1} \begin{cases} = \Delta_k, & if \;\; \|d_k\| < \Delta_k, \\ \in [\Delta_k, c_2 \Delta_k], & if \;\; \|d_k\| = \Delta_k. \end{cases} \tag{12}$$

Step 6: Compute $\zeta = dist\{-g(x_{k+1}), N_x(x_{k+1})\}$, let

$$\zeta_{k+1} = \begin{cases} \zeta, & if \;\; \zeta \leq \frac{1}{2}\zeta_k, \\ \zeta_k, & Otherwise. \end{cases} \tag{13}$$

Update the parameter β_k

$$\beta_{k+1} = \begin{cases} \frac{1}{2}\beta_k, & if \;\; \zeta_{k+1} \leq \frac{1}{2}\zeta_k, \\ \beta_k, & Otherwise. \end{cases} \tag{14}$$

Step 7: Update the symmetric matrix $B_{k+1} = G_{k+1} + \beta_{k+1} I$, set $k := k+1$, go to Step 2.

Remark 1. To update the matrix B_k, we introduce the parameters β_k and ζ_k, ζ_k denotes the distance between a point and a polyhedron. The aim of the algorithm is to have $\zeta_k \to 0$ as $k \to \infty$. Then, by (3), a solution of (1) is approached.

3. Global Convergence

We show that the accumulation points of sequence $\{x_k\}$ generated by Algorithm 1 are the optimal solutions, by this way the convergence property of Algorithm 1 can be established. We now turn to analyze the convergence behavior of Algorithm 1.

Remark 2. Recall that for $\bar{x} \in X$, $\{x \in X : f(x) \le f(\bar{x})\}$ is bounded and f is twice continuously differentiable, the two conditions imply that there exists a constant $M > 0$, such that

$$\|\nabla^2 f(x)\| \le M, \qquad \forall\, x \in X. \tag{15}$$

For simplicity, we define two index sets as follows:

$$I = \{k : \rho_k \ge \mu\} \text{ and } J = \{k : \rho_k < \mu\}.$$

We first prove that Algorithm 1 is well defined. To this end, we prove that there exists an integer i_k such that the line search (8) holds.

Lemma 1. *Assume that sequence $\{x_k\}$ is generated by Algorithm 1. Then line search (8) terminates in finite steps, i.e., there exists an integer i_k such that the line search (8) holds, for any $k \in J$.*

Proof. Suppose first, for the purpose of deriving a contradiction, that there exists $k \in J$ such that

$$f(x_k + \lambda^i d_k) > f_k + \delta\lambda^i g_k^T d_k, \qquad \forall\, i,$$

which implies

$$\frac{f(x_k + \lambda^i d_k) - f_k}{\lambda^i} > \delta g_k^T d_k.$$

Since f is differentiable, taking limit with $i \to \infty$, we have

$$g_k^T d_k \ge \delta g_k^T d_k. \tag{16}$$

Since $\delta \in (0, 1/2)$, it follows from (16) that $g_k^T d_k \ge 0$. However, we have from Step 3 of Algorithm 1 and the definition of d_k (6) that $g_k^T d_k < 0$. Therefore, for any $k \in J$, there exists $i_k > 0$ such that (8) holds. $\square$

In order to show that stepsize α_k is bounded, we now cite an important lemma from [13].

Lemma 2. *If there is an accumulation point x^* of $\{x_k\}$ such that x^* is not an optimal solution of (1), then there exists a positive number β such that*

$$\beta_k \geq \beta$$

for all k.

Proof. See Lemma 3.1 in [13]. □

Lemma 3. *Assume that sequence $\{x_k\}$ is generated by Algorithm 1. Then the stepsize α_k satisfies*

$$\alpha_k > \frac{(1-\delta)\lambda\beta}{M} \tag{17}$$

for all $k \in J$.

Proof. Due to Step 4 of Algorithm 1, we have

$$f(x_k + \lambda^{-1}\alpha_k d_k) > f_k + \delta\lambda^{-1}\alpha_k g_k^T d_k. \tag{18}$$

By Taylor's expansion, we obtain

$$f(x_k + \lambda^{-1}\alpha_k d_k) = f_k + \lambda^{-1}\alpha_k g_k^T d_k + \frac{1}{2}\lambda^{-2}\alpha_k^2 d_k^T \nabla^2 f(\xi_k) d_k, \tag{19}$$

where $\xi_k \in (x_k, x_k + \lambda^{-1}\alpha_k d_k)$. Relations (18), (19) and (15) ensure that

$$\delta\lambda^{-1}\alpha_k g_k^T d_k < \lambda^{-1}\alpha_k g_k^T d_k + \frac{1}{2}\lambda^{-2}\alpha_k^2 M \|d_k\|^2.$$

Consequently,

$$-(1-\delta) g_k^T d_k < \frac{1}{2}\lambda^{-1}\alpha_k M \|d_k\|^2. \tag{20}$$

Using the definition of d_k (6), we have

$$-g_k^T d_k - \frac{1}{2} d_k^T B_k d_k \geq 0. \tag{21}$$

Combining (21) and (20), we obtain

$$(1-\delta) d_k^T B_k d_k < \lambda^{-1}\alpha_k M \|d_k\|^2. \tag{22}$$

We can deduce from Lemma 2 that

$$\beta\|d_k\|^2 \leq d_k^T B_k d_k.$$

Thus, inequality (22) implies that (17) holds. □

Having proved that Algorithm 1 is well-defined, we now obtain an important proposition.

Proposition 1. *Suppose that sequence $\{x_k\}$ is generated by Algorithm 1. Then the following two inequalities hold.*

(1) $f(x_k + d_k) - f(x_k) \leq \frac{1}{2}\mu g_k^T d_k$ *for* $k \in I$.

(2) $f(x_k + s_k) - f(x_k) \leq \delta g_k^T s_k$ *for* $k \in J$, *where* $s_k = \alpha_k d_k$.

Proof. (1): If $k \in I$, we have from the definition ρ_k (7) that $f(x_k) - f(x_k + d_k) \geq \mu(\phi_k(0) - \phi_k(d_k))$. Since d_k is computed by Newton step, if $d_k = -B_k^{-1} g_k$, we obtain that $\phi_k(0) - \phi_k(d_k) = -\frac{1}{2} g_k^T d_k$. If $d_k = -\frac{\Delta_k}{\|B_k^{-1} g_k\|} B_k^{-1} g_k$, let $\frac{\Delta_k}{\|B_k^{-1} g_k\|} = \gamma_k$. Obviously, $\gamma_k \in (0, 1]$. Therefore, $\phi_k(0) - \phi_k(d_k) = -(1 - \frac{1}{2}\gamma_k) g_k^T d_k$. Since $(1 - \frac{1}{2}\gamma_k) \geq \frac{1}{2}$, this deduce that $f(x_k) - f(x_k + d_k) \geq -\frac{1}{2}\mu g_k^T d_k$.

(2): If $k \in J$, since Algorithm 1 is well-defined, we can deduce that the inequality holds from the Armijo line search (8). □

Based on Proposition 1 and Lemma 3, we establish the global convergence for Algorithm 1. To this end, we first recall two important propositions in [13].

Proposition 2. *Suppose that there are infinitely many Newton steps in the algorithm. Then any accumulation point of* $\{x_k\}$ *is an optimal solution of (1).*

Proof. See Proposition 3.2 in [13]. □

Proposition 3. *If only finitely many Newton steps are taken in the algorithm, then all the accumulation points of* $\{x_k\}$ *are optimal solutions of (1).*

Proof. See Proposition 3.3 in [13]. □

Obviously, applying Propositions 2 and 3, we can obtain the global convergence theorem for the algorithm.

Theorem 1. *Suppose sequence* $\{x_k\}$ *is generated by Algorithm 1. Then all the accumulation points are optimal solutions of (1).*

4. Applications and numerical examples

In this section, we apply Algorithm 1 to nonlinear portfolio optimization problems and present preliminary computational results for Algorithm 1. Portfolio optimization problems have received considerable attention since the seminal work introduced by Markowitz [16]. Recently, there are still many papers devoted to portfolio problems (e.g. Lobo et al. [15], Brogan and Stidham Jr. [6], Best and Hlouskova [2-5]). Most of these portfolio optimization problems can be formulated as a nonlinear optimization problem with nonlinear objective function and linear constraint conditions. In order to obtain a general formulation of portfolio optimization, we first recall several proposed portfolio selection models.

The Markowitz model. Portfolio optimization is a powerful tool for financial decision making under uncertainty. Markowitz first presented a model for portfolio selection in an uncertain environment with various simplifications. Let us recall the basic formulation: $x_i, i = 1, \cdots, n$ represent the proportion of fund invested in the n investments, $\sum_{i=1}^{n} x_i = e^T x = 1$, where $e = (1, 1, \cdots, 1)^T \in \Re^n$. The unit investment in the i^{th} asset provides the random return p_i over the considered fixed period. Assume that probability distribution of the vector p of returns of all assets is characterized by a known vector of expected returns $E(p) = r$ and by a covariance matrix $C = cov(p_i, p_j), i, j = 1, \cdots, n$

whose main diagonal consists of variances of individual returns. The return of the investment is $r(x) = \sum_{i=1}^{n} r_i x_i = r^T x$, and the risk of the investment is $\sigma^2(x) = \sum_{i,j} cov(p_i, p_j) x_i x_j = x^T C x$. If the financial object is to maximize the expected return of a portfolio of investments while minimizing the risk posed to the investors, then the decision making problem can be formulated as

$$\begin{array}{lll} \min & \tau x^T C x - r^T x & \\ s.t. & e^T x = 1, & \\ & x_i \geq 0, & i = 1, 2, \cdots, n, \end{array} \tag{23}$$

where $\tau \geq 0$ is a weighting parameter.

Portfolio models with transaction costs. Transaction costs have not been considered in classical Markowitz model. However, transaction costs are essential element of any realistic investment portfolio optimization. Recently, Lobo et al. [15] proposed a portfolio optimization model with linear and fixed transaction costs, which is given as

$$\begin{array}{ll} \max & \hat{a}^T (w + x) \\ s.t. & e^T x + \psi(x) \leq 0, \\ & w + x \in S, \end{array} \tag{24}$$

where $\hat{a} \in \Re^n$ is the vector of expected returns on each asset, $w \in \Re^n$ is the vector of current holdings in each asset, $x \in \Re^n$ is the vector of amounts transacted in each asset, $\psi(x) : \Re^n \to R$ is the transaction cost function, $S \subseteq \Re^n$ is the set of feasible portfolio. In their model, $\psi(x)$ can be chosen as a linear function such that the budget (or self-financing) constraint $e^T x + \psi(x) \leq 0$ is convex, risk measure is expressed as a second-order cone constraint $\| \sum^{1/2} (w + x) \|_2 \leq \sigma_{max}$ in S. Let x_i is denoted by $x_i = x_i^+ - x_i^-$, where x_i^+ and x_i^- denote the purchased positions and sold positions of asset i, respectively. If risks are measured by variance, the financial object is to minimize possible portfolio risks and transaction costs and, at the same time, maximize expected return. A nonlinear portfolio optimization is formulated as

$$\begin{array}{ll} \min & \tau (w + x_i^+ - x_i^-)^T \sum (w + x_i^+ - x_i^-) - \hat{a}^T (w + x^+ - x^-) \\ & + \sum_{i=1}^{n} (\kappa_i^+ x_i^+ + \kappa_i^- x_i^-) \\ s.t. & e^T (x^+ - x^-) + \sum_{i=1}^{n} (\kappa_i^+ x_i^+ + \kappa_i^- x_i^-) \leq 0, \\ & w + x^+ - x^- \geq 0, \\ & x^+ \geq 0, x^- \geq 0, \end{array} \tag{25}$$

where κ^+ and κ^- are the cost rates associated with buying and selling.

In [2], Best and Hlouskova considered the case when translation costs are denoted by piecewise nonlinear convex functions, they presented a portfolio model as follows

$$
\begin{aligned}
\min \quad & F(x) + \sum_{k=1}^{K} p^k(x^{+k}) + \sum_{k=1}^{K} q^k(x^{-k}) \\
s.t. \quad & x - \sum_{k=1}^{K}(x^{+k}) + \sum_{k=1}^{K}(x^{-k}) = \hat{x}, \\
& \bar{A}x \le b, \\
& 0 \le x^{+k} \le e^k, \qquad k = 1, \cdots, K, \\
& 0 \le x^{-k} \le d^k, \qquad k = 1, \cdots, K.
\end{aligned} \tag{26}
$$

where $-F(x)$ is an expected utility function, $\bar{A}$ is an (m, n) matrix, b is an m-vector, x is an n-vector. x^{+k} denote the amount purchased according to the kth transaction cost function $p^k(x^{+k})$. x^{-k} denote the amount sold according to the kth transaction cost function $q^k(x^{-k})$. $\hat{x}$ is the current holdings. K denotes the number of piecewise functions. e and d are upper bounds.

Formulations (23), (25), (26) can be reformulated as the form given by problem (1). Without loss of generality, model (25) is used in the experiment. We cite an numerical example from Bartholomew-Biggs [1, p.137]. Expected returns for each asset are

$$-0.028, 0.366, 0.231, -0.24, 0.535, -0.17, -0.881, 0.859, 0.128, 0.087,$$

and the elements of the variance-covariance matrix are

$$
\begin{pmatrix}
1.026 & -0.434 & 0.020 & -0.197 & -0.031 & -0.551 & 0.308 & -0.093 & -0.461 & -0.459 \\
-0.434 & 1.105 & -0.078 & 0.235 & -0.178 & -0.147 & -0.176 & -0.45 & 0.177 & -0.729 \\
0.020 & -0.078 & 0.433 & -0.124 & -0.189 & -0.586 & -0.020 & -0.611 & -0.209 & -0.127 \\
-0.197 & 0.235 & -0.124 & 8.076 & 1.009 & -1.879 & 4.558 & -0.011 & 0.197 & 0.54 \\
-0.031 & -0.178 & -0.189 & 1.009 & 2.901 & 0.027 & -1.113 & -0.529 & -0.176 & 0.125 \\
-0.551 & -0.147 & -0.586 & -1.879 & 0.027 & 5.163 & 0.107 & 2.484 & 0.394 & -0.142 \\
0.308 & -0.176 & -0.020 & 4.558 & -1.113 & 0.107 & 7.27 & 0.826 & -0.26 & -0.23 \\
-0.093 & -0.45 & -0.611 & -0.011 & -0.529 & 2.484 & 0.826 & 5.6 & 0.009 & 1.226 \\
-0.461 & 0.177 & -0.209 & 0.197 & -0.176 & 0.394 & -0.26 & 0.009 & 0.808 & 0.193 \\
-0.459 & -0.729 & -0.127 & 0.54 & 0.125 & -0.142 & -0.23 & 1.226 & 0.193 & 3.848
\end{pmatrix}.
$$

Our experiment is performed in MATLAB Version 6.5. Some fixed parameter values are given as follows

$$\omega = (0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.099, 0.1, 0.1, 0.1)^T, \kappa^+ = \kappa^- = 0.001.$$

$$\Delta_1 = 0.5, \ \lambda = 0.5, \ \delta = 0.25, \ \ c_1 = 0.5, \ \ c_2 = 1.5, \ \mu = 0.25, \ \beta_0 = 0.5.$$

The stopping condition is $\|d_k\| \le 10^{-6}$. We compare our algorithm with traditional trust region (TTR) method and the Algorithm in [13]. The numerical results are listed in Table 1.

Table 1. Comparisons of three trust region methods.

item	TTR	Algorithm in [13]	Algorithm 1
iterations	35	31	28
function evaluations	36	34	32
line searches	0	2	3

The numerical results show that line search is helpful to reduce iterations and function evaluations for the given problem.

5. Conclusions

This paper presented an algorithm for convex nonlinear optimization problems that combines line search and accurate trust region step. The Armijo line search is used in the algorithm when a trial step is not accepted, this scheme guarantees that the subproblem is only need to be solved one time at each iteration. The global convergence property has been established for the algorithm. We apply the algorithm to solve nonlinear portfolio optimization problems. Primary numerical results show that the algorithm is encouraging.

REFERENCES

1. M. Bartholomew-Biggs, *Nonlinear Optimization with Financial Application*, Kluwer Academic Publishers, Boston/Dordrecht/London, 2005.
2. M. J. Best and J. Hlouskova, *An algorithm for portfolio optimization with transaction costs*, Manage. Sci. **51** (2005), 1676-1688.
3. M. J. Best and J. Hlouskova, *Quadratic programming with transaction costs*, Comput. Oper. Res. **35** (2008), 18-33.
4. M. J. Best and J. Hlouskova, *An algorithm for portfolio optimization with variable transaction costs, part 1: theory*, J. Optim. Theory Appl. DOI:10.1007/s10957-007-9252-7.
5. M. J. Best and J. Hlouskova, *An algorithm for portfolio optimization with variable transaction costs, part 2: computational analysis*, J. Optim. Theory Appl. DOI:10.1007/s10957-007-9249-2.
6. A. J. Brogan and S. Stidham Jr., *Non-separation in the mean-lower-partial-moment portfolio optimization problem*, Euro. J. Oper. Res. **184** (2008), 701-710.
7. R. H. Byrd, R. B. Schnabel and G. A. Shultz, *A trust region algorithm for nonlinearly constrained optimization*, SIAM J. Numer. Anal. **24** (1987), 1152-1170.
8. R. H. Byrd, J. C. Gilbert and J. Nocedal, *A trust region method based on interior point Ttechniques for nonlinear programming*, Math. Prog. **89** (2000), 149–185.
9. M. El-Alem, *A robust trust-region algorithm with a non-monotonic penalty parameter scheme for constrained optimization*, SIAM J. Optim. **5** (1995), 348-378.
10. E. M. Gertz, *Combination Trust-region Line Search Methods for Unconstrained Optimization*, University of California, San Diego, 1999.
11. E. M. Gertz and P. E. Gill, *A primal-dual trust region algorithm for nonlinear optimization*, Math. Prog. **100** (2004), 49-94.
12. N. Gu and J. Mo, *Incorporating nonmonotone strategies into trust region method for unconstrained optimization*, Comput. Math. Appl. DOI:10.1016/j.camwa.2007.08.038.
13. H. Jiang and L. Qi, *Globally and superlinearly convergent trust-region algorithm for convex SC^1-minimization problems and its application to stochastic programs*, J. Optim. Theory Appl. **90** (1996), 649-669.
14. H. Jiang and L. Qi, *Local uniquences and iterative methods for nonsmooth variational inequalities*, J. Math. Analys. Appl. **196** (1995), 314-331.

15. M. S. Lobo, M. Fazel and S. Boyd, *Portfolio optimization with linear and fixed transaction costs*, Ann. Oper. Res. **152** (2007), 341-365.
16. H. M. Markowitz, *Portfolio selection*, J Finance, **6** (1952), 77-91.
17. J. Mo, K. Zhang and Z. Wei, *A nonmonotone trust region method for unconstrained optimization*, Appl. Math. Comput. **171** (2005), 371-384.
18. J. Nocedal and Y. Yuan, *Combining trust region and line search techniques*, in: Y. Yuan eds, Advances in Nonlinear Programming, Kluwer Academic Publishers, Dordrecht, (1996), 153-175.

Nengzhu Gu: Ph.D. of School of Business, University of Shanghai for Science and Technology, research interest is financial computation.
P. O. Box 2, 516 Jungong Road, School of Business, University of Shanghai for Science and Technology, Shanghai 200093, China
e-mail: `gnzemail@hotmail.com`

Yan Zhao: Ph.D. of School of Business, University of Shanghai for Science and Technology, research interest is Nonlinear science.

Yan Gao: Prof. of School of Business, University of Shanghai for Science and Technology, research interest is Nonlinear optimization and its application.