



A multivariate spectral projected gradient method for bound constrained optimization[☆]

Zhensheng Yu^{*}, Jing Sun, Yi Qin

College of Science, University of Shanghai for Science and Technology, Shanghai, 200093, PR China

ARTICLE INFO

Article history:

Received 3 November 2009

MSC:

90C30

65K05

Keywords:

Bounded constrained optimization

Multivariable spectral projected gradient

Nonmonotone line search

Global convergence

ABSTRACT

In this paper, we consider a multivariate spectral projected gradient (MSPG) method for bound constrained optimization. Combined with a quasi-Newton property, the multivariate spectral projected gradient method allows an individual adaptive step size along each coordinate direction. On the basis of nonmonotone line search, global convergence is established. A numerical comparison with the traditional SPG method shows that the method is promising.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

In this paper, we consider the bound constrained optimization problem

$$\begin{aligned} \min f(x) \\ \text{s.t. } x \in \Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}. \end{aligned} \quad (1)$$

where $l = (l^1, l^2, \dots, l^n)^T$, $u = (u^1, u^2, \dots, u^n)^T$ with $-\infty \leq l^i < u^i \leq +\infty$ for $i = 1, 2, \dots, n$. We denote by $g(x) = (g^1(x), g^2(x), \dots, g^n(x))^T$ the gradient of f at x .

We say that a vector $\bar{x} \in \Omega$ is a stationary point of problem (1) if it satisfies

$$\begin{cases} \bar{x}^i = l^i & \Rightarrow g^i > 0, \\ l^i < \bar{x}^i < u^i & \Rightarrow g^i = 0, \\ \bar{x}^i = u^i & \Rightarrow g^i < 0. \end{cases} \quad (2)$$

Problem (1) is very important in practical optimization, and many practical problems can be converted into form (1). In addition, problem (1) is often a subproblem of augmented Lagrangian or penalty schemes for general constrained optimization. Hence it has received much attention in recent decades, and many numerical algorithms have been developed [1–9]. The simplest of these methods is the projected gradient method which was originally proposed in [10,11] and extended in [12]. The advantage of the projected gradient method is that it is quite easy to implement and very effective for large scale problems. On the negative side, to keep the feasibility of the iterate, the projecting is in general expensive.

[☆] This work are supported by the Innovation Program of Shanghai Municipal Education Commission (No. 10YZ99) and Leading Academic Discipline Project of Shanghai Municipal Government, Project Number: S30501.

^{*} Corresponding author.

E-mail address: zhsh-yu@163.com (Z. Yu).

Moreover, even if the projecting is inexpensive, the method is considered to be very slow like its analogue, the optimal gradient method for unconstrained optimization.

To speed up the convergence of the classical projected gradient method, Birgin et al. [13] proposed a spectral projected gradient method for problem (1) which can be seen as an extension of the spectral gradient method for unconstrained optimization.

The spectral gradient method was originally proposed in [14] for quadratics and further analyzed in [15] for general functions, since it requires little computational work and greatly speeds up the convergence of gradient methods. Therefore, this technique has received successful applications in unconstrained and constrained optimizations [16–26].

Recently, by replacing the classical spectral step size with a vector, Han et al. [27] defined a new iterative scheme and proposed a multivariate spectral gradient method for unconstrained optimization. The new method is finitely convergent for positive definite quadratics and globally convergent for general functions based on the nonmonotone line search [28]. The numerical results show that the multivariate spectral gradient method works better than the classical spectral gradient method.

In this paper, we aim to extend the multivariate spectral gradient method to bound constrained optimization and propose a multivariate spectral projected gradient method (MSPG). We establish the global convergence on the basis of nonmonotone line search [28] and give a numerical comparison with the classical SPG method to show the efficiency of the proposed method.

The paper is organized as follows. In Section 2, we introduce the multivariate spectral projected gradient method. In Section 3, we discuss the global convergence of the proposed algorithm and numerical tests are given in Section 4. The conclusion is presented in Section 5.

2. The multivariate spectral projected gradient method

In what follows, we first introduce the spectral gradient step size and the multivariate spectral gradient method. Let g_k be the gradient of f at x_k ; the spectral gradient method for the unconstrained optimization $\min_{x \in \mathbb{R}^n} f(x)$ can be described as [14]

$$x_{k+1} = x_k - \frac{1}{\alpha_k} g_k$$

where α_k is given by

$$\alpha_k = \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}}, \quad (3)$$

with $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g(x_k) - g(x_{k-1})$. The choices of the α_k imposed some quasi-Newton property, which can be obtained by minimizing $\|\alpha I s_{k-1} - y_{k-1}\|$ with respect to α ; $\alpha_k I$ approximates to the Hessian matrix of f at x_k . On replacing α_k with a vector, the multivariate spectral gradient method is defined as follows [27]:

Consider $\text{diag}\{\lambda^1, \lambda^2, \dots, \lambda^n\}$ generated by minimizing

$$\|\text{diag}\{\lambda^1, \lambda^2, \dots, \lambda^n\} s_{k-1} - y_{k-1}\| \quad (4)$$

with respect to $\{\lambda^i\}_{i=1}^n$; then the multivariate spectral gradient iterate scheme is

$$x_{k+1} = x_k - \text{diag}\{1/\lambda_k^1, 1/\lambda_k^2, \dots, 1/\lambda_k^n\} g_k. \quad (5)$$

On the basis of (4) and (5), we now consider the multivariate spectral projected gradient method for problem (1). Given $z \in \mathbb{R}^n$, we define $P(z)$ as the orthogonal projection on Ω ; the well known result for $P(z)$ is

$$P(z) = \begin{cases} l^i & \text{if } z^i \leq l^i, \\ z^i & \text{if } l^i < z^i < u^i, \\ u^i & \text{if } z^i \geq u^i. \end{cases} \quad (6)$$

Now, we describe our algorithm, detailed as follows:

Algorithm 2.1.

Step 0. Given $x_0 \in \Omega$, $\alpha_0 \in \mathbb{R}^{n \times n}$, a positive integer $M \geq 1$, $\gamma \in (0, 1)$, $\delta > 0$, $0 < \sigma_1 < \sigma_2 < 1$, and $\varepsilon > 0$, set $k = 0$.

Step 1. If $\|P(x_k - g(x_k)) - x_k\| < \varepsilon$, stop.

Step 2.

(a) If $k=0$, set $x_1 = P(x_0 - \alpha_0 g_0)$; go to Step 5.

(b) If $y_{k-1}^i / s_{k-1}^i > 0$, then set $\lambda_k^i = y_{k-1}^i / s_{k-1}^i$; otherwise set $\lambda_k^i = \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}}$ for $i = 1, 2, \dots, n$.

(c) If $\lambda_k^i < \varepsilon$ or $\lambda_k^i > 1/\varepsilon$, then set $\lambda_k^i = \delta$ for $i = 1, 2, \dots, n$.

Step 3. Set $\alpha_k = \text{diag}\{1/\lambda_k^1, 1/\lambda_k^2, \dots, 1/\lambda_k^n\}$.

Step 4. (Nonmonotone line search.)

Step 4.1 Compute $d_k = P(x_k - \alpha_k g(x_k)) - x_k$; set $\tau = 1$.

Step 4.2. Set $x_+ = x_k + \tau d_k$.

Step 4.3. If

$$f(x_+) \leq \max_{j \in [0, \min\{k, M-1\}]} f(x_{k-j}) + \gamma \tau_k \langle d_k, g(x_k) \rangle, \quad (7)$$

then set $\tau_k = \tau$, $x_{k+1} = x_+$, $s_k = x_{k+1} - x_k$, $y_k = g(x_{k+1}) - g(x_k)$ and go to Step 5. If (7) does not hold, define $\tau_{\text{new}} \in [\sigma_1 \tau_k, \sigma_2 \tau_k]$, set $\tau_k = \tau_{\text{new}}$ and go to Step 4.2.

Step 5. Set $k = k + 1$; go to Step 1.

3. Global convergence

In this section, we analyze the global convergence properties of Algorithm 2.1. Throughout the paper, we assume that the following assumption holds.

Assumption 3.1. The level set $\mathcal{L} = \{x \in \Omega \mid f(x) \leq f(x_0)\}$ is compact.

The following lemma gives the descent property of d_k .

Lemma 3.1. Let $x_k \in \Omega$ and d_k be generated by Step 4.1. Then there exists a constant $c_1 > 0$ such that

$$g_k^T d_k \leq -c_1 \|d_k\|^2. \quad (8)$$

Proof. Define $g_k^i = g^i(x_k)$; by the definition of d_k , we have

$$d_k^i = P(x_k^i - g_k^i / \lambda_k^i) - x_k^i.$$

We consider three possible cases:

Case 1: $x_k^i - \frac{1}{\lambda_k^i} g_k^i \in (l^i, u^i)$. In this case, we have

$$d_k^i = P(x_k^i - g_k^i / \lambda_k^i) - x_k^i = -g_k^i / \lambda_k^i.$$

So we have

$$g_k^i d_k^i = -\lambda_k^i (d_k^i)^2.$$

Case 2: $x_k^i - \frac{1}{\lambda_k^i} g_k^i < l^i$. In this case, we have

$$d_k^i = l^i - x_k^i \leq 0 \quad \text{and} \quad g_k^i \geq -\lambda_k^i (l^i - x_k^i).$$

So we have

$$g_k^i d_k^i \leq -\lambda_k^i (l^i - x_k^i) d_k^i = -\lambda_k^i (d_k^i)^2.$$

Case 3: $x_k^i - \frac{1}{\lambda_k^i} g_k^i > u^i$. In this case, we have

$$d_k^i = u^i - x_k^i \geq 0 \quad \text{and} \quad g_k^i \leq -\lambda_k^i (l^i - x_k^i).$$

So we have

$$g_k^i d_k^i \leq -\lambda_k^i (l^i - x_k^i) d_k^i = -\lambda_k^i (d_k^i)^2.$$

Since $\lambda_k^i \geq \varepsilon$, let $c_1 = \varepsilon$; we have

$$g_k^T d_k \leq -c_1 \|d_k\|^2. \quad \square$$

Lemma 3.2. Let x_k and d_k be generated by Algorithm 2.1. Then $d_k = 0$ if and only if x_k is a stationary point of problem (1).

Proof. Define $L^k = \{i \mid x_k^i = l^i\}$, $F^k = \{i \mid l^i < x_k^i < u^i\}$ and $U^k = \{i \mid x_k^i = u^i\}$. We first show that $d_k = 0$ implies that x_k is a stationary point of problem (1).

For $i \in L^k$, from

$$0 = d_k^i = P\left(x_k^i - \frac{1}{\lambda_k^i} g_k^i\right) - x_k^i,$$

Table 1

Results for problem 1.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	$\ Pg\ $	I_t/I_f	Time	$\ Pg\ $
4(a)	23/24	0.156	1.4174E–07	1146/2310	0.652	9.6404E–06
4(b)	432/727	0.989	9.8231E–06	169/293	0.171	7.1694E–06
4(c)	23/24	0.165	1.4174E–07	208/368	0.089	9.0194E–06
4(d)	729/1296	2.282	3.6140E–07	1087/2290	0.353	9.9463E–06

Table 2

Results for problem 2.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	$\ Pg\ $	I_t/I_f	Time	$\ Pg\ $
10(a)	6/7	0.035	2.4252E–09	7/8	0.041	2.4252E–09
20(a)	20/23	0.176	6.9016E–09	20/24	0.313	8.6099E–09
50(a)	1/2	0.003	0	1/2	0.031	0

we have

$$P\left(x_k^i - \frac{1}{\lambda_k^i} g_k^i\right) = x_k^i = l^i,$$

by (6), we know that

$$x_k^i - \frac{1}{\lambda_k^i} g_k^i \leq l^i,$$

and since $\lambda_k^i > 0$, we get $g_k^i \geq 0$. Similarly, we get $g_k^i \leq 0$ for $i \in U^k$ and $g_k^i = 0$ for $i \in F^k$. Therefore x_k is a stationary point of problem (1).

On the other hand, suppose that x_k is a stationary point of problem (1); then for $i \in L^k$, $g_k^i \geq 0$, we have $x_k^i - \frac{1}{\lambda_k^i} g_k^i \leq l^i$, and therefore, $d_k^i = 0$ for $i \in L^k$. Similarly, we can get $d_k^i = 0$ for $i \in F^k$ and $i \in U^k$. This completes the proof. \square

On the basis of Lemmas 3.1 and 3.2 and following from Theorem 2.2 in [13], we deduce the following convergence theorem.

Theorem 3.1. Let $\{x_k\}$ be the sequence generated by Algorithm 2.1; then every limit point is a stationary point of (1).

4. Numerical results

In this section, we test our algorithm on some typical test problems which are taken from [3] with different dimensions and initial points. The program code was written in MATLAB and run in the MATLAB 7.1 environment. The parameters are chosen as: $M = 5$, $\gamma = 10^{-4}$, $\sigma_1 = 0.1$, $\sigma_2 = 0.9$, $\alpha_0 = \frac{1}{\|P(x_0 - g_0) - x_0\|}$,

$$\delta = \begin{cases} 1 & \text{if } \|P(x_k - g_k) - x_k\| > 1, \\ \|P(x_k - g_k) - x_k\| & \text{if } 10^{-5} \leq \|P(x_k - g_k) - x_k\| \leq 1, \\ 10^{-5} & \text{if } \|P(x_k - g_k) - x_k\| < 10^{-5}. \end{cases}$$

For deciding when to stop the execution of the algorithm, declaring convergence, we used the criterion $\|P(x_k - g_k) - x_k\| < 10^{-5}$, i.e., we choose $\varepsilon = 10^{-5}$. We also stopped the execution when 5000 iterations were completed without achieving convergence and denoted this by *.

To complete our numerical insight into the behavior of our MSPG, we compare it with Algorithm 2.2 (SPG2) of [13]; the test results are given in Tables 1–8. Here n denotes the dimension of the test problems, SP denotes the initial point, I_g and I_f denote the number of gradient estimations and function value estimations, “Time” denotes the CPU time (in seconds) used when the iteration is stopped, $\|Pg\|$ denotes the 2-norm of the final gradient projection. The numerical results show that our MSPG is competitive with the SPG2 method of [13], and that for problems 2, 6, 7, the performance of MSPG is better.

Test problems:

1. HS38U:

$$f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1).$$

Constraints:

$$-10 \leq x_i \leq 10, \quad i = 1, 2, 3, 4.$$

Initial point: $a = (-3, -1, -3, -1)^T$, $b = (3, 1, 3, 1)^T$, $c = (3, -1, 3, -1)^T$, $d = (-3, 1, -3, 1)^T$.

Table 3
Results for problem 3.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	$\ Pg\ $	I_t/I_f	Time	$\ Pg\ $
10(a)	54/56	0.452	3.3249E–06	78/100	0.406	6.6360E–06
10(b)	51/51	0.267	7.2507E–06	41/43	0.111	2.7661E–06
10(c)	79/86	0.450	6.6377E–06	65/85	0.047	6.2161E–06
100(a)	703/1227	6.233	9.8200E–06	580/1013	1.084	9.6949E–06
100(b)	579/1019	5.373	9.7380E–06	581/1036	0.990	9.5919E–06
100(c)	*	*	*	53/57	0.100	1.3422E–06
500(a)	1768/3263	87.349	9.9961E–06	2097/3181	67.549	9.9357E–06
500(b)	457/820	24.672	9.5160E–06	1212/2212	38.974	9.7672E–06
500(c)	1467/2720	73.296	9.9411E–06	56/65	1.639	9.7081E–06

Table 4
Results for problem 4.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	RE	I_t/I_f	Time	RE
10(a)	26/30	0.217	3.1316E–06	44/56	0.639	7.0182E–06
100(a)	73/87	4.811	5.8080E–06	59/77	3.710	5.0339E–06
500(a)	71/78	105.991	4.9478E–06	33/38	48.522	3.0030E–06
1000(a)	69/77	383.889	9.3035E–06	59/70	341.802	7.9566E–06

Table 5
Results for problem 5.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	RE	I_t/I_f	Time	RE
8(a)	116/139	0.847	7.4044E–06	112/147	1.061	7.7952E–06
8(b)	80/96	0.760	7.3456E–06	97/126	0.261	7.5414E–06
8(c)	81/99	1.022	2.8548E–06	76/96	0.121	5.9154E–06
20(a)	95/104	1.675	9.3397E–06	119/160	0.306	4.8117E–06
20(b)	140/200	2.148	8.2089E–06	112/149	0.289	8.3144E–06
20(c)	74/90	1.093	3.3328E–06	87/108	0.205	9.7532E–06
100(a)	121/159	4.115	9.6312E–06	117/145	1.630	9.1254E–06
100(b)	99/134	3.048	8.1354E–06	113/153	1.380	7.9284E–06
100(c)	109/150	3.872	4.2270E–06	109/151	1.240	4.2279E–06
1000(a)	99/113	28.487	7.6128E–06	105/135	21.222	4.5824E–06
1000(b)	100/126	28.306	9.3635E–06	96/130	19.302	6.7195E–06
1000(c)	72/84	22.132	9.6169E–06	99/124	19.562	9.5891E–06

Table 6
Results for problem 6.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	RE	I_t/I_f	Time	RE
15(a)	121/202	0.638	4.8813E–06	116/262	0.308	6.2949E–06
30(a)	34/35	0.685	8.4828E–06	37/37	0.106	7.9245E–06
100(a)	38/39	0.948	4.2437E–06	45/46	0.102	5.5055E–06
300(a)	22/23	0.709	3.7484E–06	60/67	0.262	7.2206E–06
900(a)	15/16	1.131	4.0861E–07	65/76	1.01	2.3793E–06
6000(a)	4/5	5.374	0	28/29	10.085	8.1934E–06

Table 7
Results for problem 7.

$n(S.P)$	MSPG			SPG2		
	I_t/I_f	Time	RE	I_t/I_f	Time	RE
10(a)	36/37	0.470	5.0784E–06	32/34	1.247	9.6356E–06
10(b)	58/68	0.809	7.9708E–06	41/41	0.183	9.8832E–06
10(c)	28/31	0.568	7.3295E–06	42/43	0.062	9.1032E–06
100(a)	41/42	3.798	9.0028E–06	*	*	*
100(b)	54/56	5.135	2.6930E–06	428/707	37.154	7.7178E–06
100(c)	43/43	4.099	7.2760E–06	408/672	33.655	8.8929E–06

Table 8

Results for problem 8.

$n(S,P)$	MSPG			SPG2		
	I_t/I_f	Time	RE	I_t/I_f	Time	RE
20(a)	38/67	0.257	8.8545E–07	22/31	0.401	8.6251E–07
20(b)	26/40	0.343	2.3554E–06	16/23	0.115	2.9573E–06
20(c)	31/43	0.305	7.1522E–06	23/32	0.039	7.6633E–06
100(a)	25/40	0.621	4.4196E–07	57/85	0.981	5.2790E–07
100(b)	51/135	1.194	1.1191E–06	26/30	0.5115	4.4929E–06
100(c)	66/96	1.369	4.5403E–06	38/59	0.066	5.2554E–06
200(a)	28/49	0.778	2.8305E–06	78/122	0.693	9.9050E–06
200(b)	30/34	0.838	2.1337E–06	26/27	0.221	7.7864E–07
200(c)	32/38	1.063	3.8684E–06	36/47	0.150	3.2144E–06

2. HS110U:

$$f(x) = \sum_{i=1}^n [(\ln(x_i - 2))^2 + (\ln(10 - x_i))^2] - \left(\prod_{i=1}^n x_i \right)^{0.2}.$$

Constraints:

$$2.001 \leq x_i \leq 9.999.$$

Initial point:

$$x^0 = (9.5, 9.5, \dots, 9.5)^T.$$

3. BRODYDENU:

$$f(x) = 1 + \sum_{i=1}^n ((3 - 2x_i)x_i - x_{i-1} - x_{i+1})^2.$$

Constraints:

$$-5 \leq x_i \leq 5.$$

Initial point: $a = (1, -1, 1, -1, \dots, 1, -1)^T$, $b = (1, 1, 1, 1, \dots, 1, 1)^T$, $c = (-1, -1, -1, -1, \dots, -1, -1)^T$.

4. TRIGU:

$$f(x) = \sum_{i=1}^n \left(n + i - \sum_{j=1}^n (a_{ij} \sin x_i + b_{ij} \cos x_j) \right),$$

where $a_{ij} = \delta_{ij}$, $b_{ij} = 1 + i\delta_{ij}$ and $\delta_{ij} = 1$ if $i = j$ and 0 otherwise.

Constraints:

$$0.01 \leq x_i \leq 10\,000.$$

Initial point: $a = (10, 10, \dots, 10)^T$.

5. CRAGGLEVYU:

$$f(x) = \sum_{i \in J} [(e^{x_i} - x_{i+1})^4 + 100(x_{i+1} - x_{i+2})^6 + \tan^4(x_{i+2} - x_{i+3}) + x_i^8 + (x_{i+3} - 1)^2],$$

where n is a multiple of 4 and $J = \{1, 5, 9, \dots, n-3\}$.

Constraints:

$$-5 \leq x_i \leq 5.$$

Initial point: $a = (1, 2, 1, 2, \dots, 1, 2)^T$, $x^0 = (1, 1, 1, 1, \dots, 1, 1)^T$, $c = (2, 2, 2, 2, \dots, 2, 2)^T$.

6. PENALTU:

$$f(x) = 1 + \sum_{i=1}^n x_i + 1000 \left(1 - \sum_{i=1}^n \frac{1}{x_i} \right)^2 + 1000 \left(1 - \sum_{i=1}^n \frac{i}{x_i} \right)^2.$$

Constraints:

$$0.01 \leq x_i \leq 10\,000.$$

Initial point: $a = (15, 15, 15, 15, \dots, 15)^T$.

7. TOINTRIGU:

$$f(x) = \sum_{(i,j) \in J} \alpha_{ij} \sin[\beta_i x_i + \beta_j x_j + \gamma_{ji}],$$

where $\alpha_{ij} = 5[1 + \text{mod}(i, 5) + \text{mod}(j, 5)]$, $\beta_i = 1 + i/n$, $\gamma_{ij} = (i + j)/n$, $J = \{(i, j) | \text{mod}(|i - j|, 4) = 0\}$.

Constraints:

$$-10 \leq x_i \leq 10.$$

Initial point: $a = (1, 1, \dots, 1)^T$, $b = (0, 0, \dots, 0)^T$.

8. BROWNU:

$$f(x) = \left[\sum_{i \in J} (x_i - 3) \right]^2 + \sum_{i \in J} [0.0001(x_i - 3)^2 - (x_i - x_{i-1})^2 + e^{20(x_i - x_{i+1})}].$$

Constraints:

$$-10 \leq x_i \leq 10,$$

where $J = \{1, 3, 5, 2n - 1\}$.

Initial point: $a = (0, 3, 0, 3, \dots, 0, 3)^T$, $b = (0, 0, 0, 0, \dots, 0, 0)^T$, $c = (3, 3, 3, 3, \dots, 3, 3)^T$.

5. Conclusion

In this paper, we presented the multivariate spectral gradient projected method for bound constrained optimization and compared it with the SPG2 method of [13]. This method allows an individual adaptive step size along each coordinate direction and global convergence is established on the basis of nonmonotone line search. The numerical tests show that our method is promising.

References

- [1] O.P. Burdakov, J.M. Martínez, E.A. Pilotta, A limited memory multipoint secant method for bound constrained optimization, *Ann. Oper. Res.* 117 (2002) 51–70.
- [2] R.H. Byrd, P.H. Lu, J. Nocedal, A limited memory algorithm for bound constrained optimization, *SIAM J. Sci. Stat. Comput.* 16 (1995) 1190–1208.
- [3] Z.W. Chen, J.Y. Han, D.C. Xu, A nonmonotone trust region method for nonlinear programming with simple bound constraints, *Appl. Math. Optim.* 43 (2001) 63–85.
- [4] A.R. Conn, N.I.M. Gould, Ph.L. Toint, Global convergence of a class of trust region algorithm for optimization with simple bounds, *SIAM J. Numer. Anal.* 25 (1988) 433–460.
- [5] R.S. Dembo, U. Tulowitzki, On the minimization of quadratic functions subject to box constraints, Working Paper, School of Organization and Management, Yale University, New Haven, CT, 1983.
- [6] F. Facchinei, J. Júdice, J. Soares, An active set Newton's algorithm for large-scale nonlinear programs with box constraints, *SIAM J. Optim.* 8 (1998) 158–186.
- [7] A. Friedlander, J.M. Martínez, S.A. Santos, A new trust region algorithm for bound constrained minimization, *Appl. Math. Optim.* 30 (1994) 235–266.
- [8] W.W. Hager, H. Zhang, A new active set algorithm for box constrained optimization, *SIAM J. Optim.* 17 (2006) 526–557.
- [9] J. Moré, G. Toraldo, On the solution of large scale quadratic programming problem with bound constraints, *SIAM J. Optim.* 1 (1991) 93–113.
- [10] A.A. Goldstein, Convex programming in Hilbert space, *Bull. Amer. Math. Soc.* 70 (1964) 709–710.
- [11] E.S. Leviton, B.T. Polyak, Constrained minimization problems, *USSR Comput. Math. Math. Phys.* 6 (1966) 1–50.
- [12] P.H. Calamai, J.J. Moré, Projected gradient methods for linearly constrained problems, *Math. Program.* 39 (1987) 93–116.
- [13] E.G. Birgin, J.M. Martinez, M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, *SIAM J. Optim.* 10 (2000) 1196–1211.
- [14] J. Barzilai, J.M. Borwein, Two point step size gradient method, *IMA J. Numer. Anal.* 8 (1988) 174–184.
- [15] M. Raydan, The Barzilai–Borwein gradient method for the large scale unconstrained optimization problem, *SIAM J. Optim.* 7 (1997) 26–33.
- [16] R. Andreani, E.G. Birgin, J.M. Martínez, J. Yuan, Spectral projected gradient and variable metric methods for optimization with linear inequalities, *IMA J. Numer. Anal.* 25 (2005) 221–252.
- [17] M. Andretta, E.G. Birgin, J.M. Martínez, Practical active-set Euclidian trust-region method with spectral projected gradients for bound-constrained minimization, *Optimization* 54 (2005) 305–325.
- [18] L. Bello, M. Raydan, Preconditioned spectral projected gradient method on convex sets, *J. Comput. Math.* 23 (2005) 225–232.
- [19] E.G. Birgin, J.M. Martinez, Large scale active-set box-constrained optimization method with spectral projected gradients, *Comput. Optim. Appl.* 23 (2002) 101–125.
- [20] E.G. Birgin, J.M. Martinez, M. Raydan, Inexact spectral projected gradient methods on convex sets, *IMA J. Numer. Anal.* 23 (2003) 539–559.
- [21] Y.H. Dai, R. Fletcher, Projected Barzilai–Borwein methods for large-scale box-constrained quadratic programming, *Numer. Math.* 100 (2005) 21–47.
- [22] Y.H. Dai, H.C. Zhang, Adaptive two-point stepsize gradient algorithm, *Numer. Algorithms* 27 (2001) 377–385.
- [23] M. Raydan, On the Barzilai and Borwein choice of steplength for the gradient method, *IMA J. Numer. Anal.* 13 (1993) 321–326.
- [24] C.Y. Wang, Q. Liu, X.M. Yang, Convergence properties of nonmonotone spectral projected gradient methods, *J. Comput. Appl. Math.* 182 (2006) 51–66.
- [25] Y.H. Xiao, Q.J. Hu, Subspace Barzilai–Borwein gradient method for large-scale bound constrained optimization, *Appl. Math. Optim.* 58 (2008) 275–290.
- [26] R. Fletcher, On the Barzilai–Borwein method, Department of Mathematics, University of Dundee NA/207, Dundee, Scotland, 2001.
- [27] L. Han, G.H. Yu, L.T. Guan, Multivariate spectral gradient method for unconstrained optimization, *Appl. Math. Comput.* 201 (2008) 621–630.
- [28] L. Grippo, F. Lampariello, S. Lucidi, A nonmonotone line search technique for Newton's methods, *SIAM J. Numer. Anal.* 23 (1986) 707–716.