Part I

OPTIMIZATION ALGORITHMS

Chapter 2

The spherical quadratic steepest descent algorithm

2.1 Introduction

In this chapter an extremely simple gradient only algorithm is proposed that, in terms of storage requirement (only 3 n-vectors need be stored) and computational efficiency, may be considered as an alternative to the conjugate gradient methods. The method effectively applies the steepest descent (SD) method to successive simple spherical quadratic approximations of the objective function in such a way that no explicit line searches are performed in solving the minimization problem. It is shown that the method is convergent when applied to general positive-definite quadratic functions. The method is tested by its application to some standard and other test problems. On the evidence presented the new method, called the SQSD algorithm, appears to be reliable and stable, and very competitive compared to the well established conjugate gradient methods. In particular, it does very well when applied to extremely ill-conditioned problems.

2.2 The classical steepest descent method

Consider the following unconstrained optimization problem:

$$\min f(\mathbf{x}), \ \ \mathbf{x} \in \Re^n \tag{2.1}$$

where f is a scalar objective function defined on $\Re^n$, the n-dimensional real Euclidean space, and $\mathbf{x}$ is a vector of n real components $x_1, x_2, \ldots, x_n$. It is assumed that f is differentiable so that the gradient vector $\nabla f(\mathbf{x})$ exists everywhere in $\Re^n$. The solution is denoted by $\mathbf{x}^*$.

The steepest descent (SD) algorithm for solving problem (2.1) may then be stated as follows:

Algorithm 2.1 SD algorithm

Initialization: Specify convergence tolerances ε_g and ε_x, select starting point $\mathbf{x}^0$. Set $k := 1$ and go to main procedure.
Main procedure:

1. If $\|\nabla f(\mathbf{x}^{k-1})\| < \varepsilon_g$, then set $\mathbf{x}^* \cong \mathbf{x}^c = \mathbf{x}^{k-1}$ and stop; otherwise set $\mathbf{u}^k := -\nabla f(\mathbf{x}^{k-1})$.

2. Let λ_k be such that $f(\mathbf{x}^{k-1} + \lambda_k \mathbf{u}^k) = \min_\lambda f(\mathbf{x}^{k-1} + \lambda \mathbf{u}^k)$ subject to $\lambda \geq 0$ {line search step}.

3. Set $\mathbf{x}^k := \mathbf{x}^{k-1} + \lambda_k \mathbf{u}^k$; if $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| < \varepsilon_x$, then $\mathbf{x}^* \cong \mathbf{x}^c = \mathbf{x}^k$ and stop; otherwise set $k := k + 1$ and go to Step 1.

It can be shown that if the steepest descent method is applied to a general positive-definite quadratic function of the form[1] $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{A}\mathbf{x} + \mathbf{b}^{\mathsf{T}}\mathbf{x} + c$, then the sequence $\{f(\mathbf{x}^k)\} \rightarrow f(\mathbf{x}^*)$. Depending, however, on the starting point $\mathbf{x}^0$ and the condition number of $\mathbf{A}$ associated with the quadratic form, the rate of convergence may become extremely slow.

[1]A superscript T means transpose.

It is proposed here that for general functions $f(\mathbf{x})$, better overall performance of the steepest descent method may be obtained by applying it successively to a sequence of very simple quadratic approximations of $f(\mathbf{x})$. The proposed modification, named here the spherical quadratic steepest descent (SQSD) method, remains a first order method since only gradient information is used with no attempt being made to construct the Hessian of the function. The storage requirements therefore remain minimal, making it ideally suitable for problems with a large number of variables. Another significant characteristic is that the method requires no explicit line searches.

2.3 The spherical quadratic steepest descent method

In the SQSD approach, given an initial approximate solution $\mathbf{x}^0$, a sequence of spherically quadratic optimization subproblems $P[k], k = 0, 1, 2, \ldots$ is solved, generating a sequence of approximate solutions $\mathbf{x}^{k+1}$. More specifically, at each point $\mathbf{x}^k$ the constructed approximate subproblem is $P[k]$:

$$\min_{\mathbf{x}} \tilde{f}_k(\mathbf{x}) \tag{2.2}$$

where the approximate objective function $\tilde{f}_k(\mathbf{x})$ is given by

$$\tilde{f}_k(\mathbf{x}) = f(\mathbf{x}^k) + \boldsymbol{\nabla}^{\mathrm{T}} f(\mathbf{x}^k)(\mathbf{x} - \mathbf{x}^k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^k)\mathbf{C}_k^{\mathrm{T}}(\mathbf{x} - \mathbf{x}^k) \tag{2.3}$$

and $\mathbf{C}_k = \mathrm{diag}(c_k, c_k, \ldots, c_k) = c_k\mathbf{I}$. The solution to this problem will be denoted by $\mathbf{x}^{*k}$, and for the construction of the next subproblem $P[k+1]$, $\mathbf{x}^{k+1} := \mathbf{x}^{*k}$.

For the first subproblem the curvature c_0 is set to $c_0 := \|\boldsymbol{\nabla} f(\mathbf{x}^0)\| / \rho$, where $\rho > 0$ is some arbitrarily specified step limit. Thereafter, for $k \geq 1$, c_k is chosen such that $\tilde{f}(\mathbf{x}^k)$ interpolates $f(\mathbf{x})$ at both $\mathbf{x}^k$ and $\mathbf{x}^{k-1}$. The latter conditions imply that for $k = 1, 2, \ldots$

$$c_k := \frac{2\left[f(\mathbf{x}^{k-1}) - f(\mathbf{x}^k) - \boldsymbol{\nabla}^{\mathrm{T}} f(\mathbf{x}^k)(\mathbf{x}^{k-1} - \mathbf{x}^k)\right]}{\|\mathbf{x}^{k-1} - \mathbf{x}^k\|^2} \tag{2.4}$$

Clearly the identical curvature entries along the diagonal of the Hessian, mean that the level surfaces of the quadratic approximation $\tilde{f}_k(\mathbf{x})$, are indeed concentric hyper-spheres. The approximate subproblems $P[k]$ are therefore aptly referred to as spherical quadratic approximations.

It is now proposed that for a large class of problems the sequence $\mathbf{x}^0, \mathbf{x}^1, \ldots$ will tend to the solution of the original problem (2.1), i.e.

$$\lim_{k\to\infty} \mathbf{x} = \mathbf{x}^* \tag{2.5}$$

For subproblems $P[k]$ that are convex, i.e. $c_k > 0$, the solution occurs where $\boldsymbol{\nabla}\tilde{f}_k(\mathbf{x}) = \mathbf{0}$, that is where

$$\boldsymbol{\nabla} f(\mathbf{x}^k) + c_k\mathbf{I}(\mathbf{x} - \mathbf{x}^k) = \mathbf{0} \tag{2.6}$$

The solution to the subproblem, $\mathbf{x}^{*k}$ is therefore given by

$$\mathbf{x}^{*k} = \mathbf{x}^k - \frac{\boldsymbol{\nabla} f(\mathbf{x}^k)}{c_k} \tag{2.7}$$

Clearly the solution to the spherical quadratic subproblem lies along a line through $\mathbf{x}^k$ in the direction of steepest descent. The SQSD method may formally be stated in the form given in Algorithm 2.2.

Step size control is introduced in Algorithm 2.2 through the specification of a step limit ρ and the test for $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| > \rho$ in Step 2 of the main procedure. Note that the choice of c_0 ensures that for $P[0]$ the solution $\mathbf{x}^1$ lies at a distance ρ from $\mathbf{x}^0$ in the direction of steepest descent. Also the test in Step 3 that $c_k < 0$, and setting $c_k := 10^{-60}$ where this condition is true ensures that the approximate objective function is always positive-definite.

Algorithm 2.2 SQSD algorithm

Initialization: Specify convergence tolerances ε_g and ε_x, step limit $\rho > 0$ and select starting point $\mathbf{x}^0$. Set $c_0 := \|\nabla f(\mathbf{x}^0)\| / \rho$. Set $k := 1$ and go to main procedure.

Main procedure:

1. If $\left\|\nabla f(\mathbf{x}^{k-1})\right\| < \varepsilon_g$, then $\mathbf{x}^* \cong \mathbf{x}^c = \mathbf{x}^{k-1}$ and stop; otherwise set

$$\mathbf{x}^k := \mathbf{x}^{k-1} - \frac{\nabla f(\mathbf{x}^{k-1})}{c_{k-1}}.$$

2. If $\left\|\mathbf{x}^k - \mathbf{x}^{k-1}\right\| > \rho$, then set

$$\mathbf{x}^k := \mathbf{x}^k - \rho\frac{\nabla f(\mathbf{x}^{k-1})}{\|\nabla f(\mathbf{x}^{k-1})\|};$$

   if $\left\|\mathbf{x}^k - \mathbf{x}^{k-1}\right\| < \varepsilon_x$, then $\mathbf{x}^* \cong \mathbf{x}^c = \mathbf{x}^k$ and stop.

3. Set

$$c_k := \frac{2\left[f(\mathbf{x}^{k-1}) - f(\mathbf{x}^k) - \nabla^T f(\mathbf{x}^k)(\mathbf{x}^{k-1} - \mathbf{x}^k)\right]}{\left\|\mathbf{x}^{k-1} - \mathbf{x}^k\right\|^2};$$

   if $c_k < 0$ set $c_k := 10^{-60}$.

4. Set $k := k + 1$ and go to Step 1 for next iteration.

2.4 Convergence of the SQSD method

An analysis of the convergence rate of the SQSD method, when applied to a general positive-definite quadratic function, affords insight into the convergence behavior of the method when applied to more general functions. This is so because for a large class of continuously differentiable functions, the behavior close to local minima is quadratic. For quadratic functions the following theorem may be proved.

THEOREM. *The SQSD algorithm (without step size control) is convergent when applied to the general quadratic function of the form* $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}$, *where* $\mathbf{A}$ *is a* $n \times n$ *positive-definite matrix and* $\mathbf{b} \in \Re^n$.

PROOF. Begin by considering the bivariate quadratic function, $f(\mathbf{x}) = x_1^2 + \gamma x_2^2$, $\gamma \geq 1$ and with $\mathbf{x}^0 = [\alpha, \beta]^T$. Assume $c_0 > 0$ given, and for convenience in what follows set $c_0 = 1/\delta, \delta > 0$. Also employ the notation $f_k = f(\mathbf{x}^k)$.

Application of the first step of the SQSD algorithm yields

$$\mathbf{x}^1 = \mathbf{x}^0 - \frac{\nabla f_0}{c_0} = [\alpha(1-2\delta), \beta(1-2\gamma\delta)]^T \tag{2.8}$$

and it follows that

$$\left\|\mathbf{x}^1 - \mathbf{x}^0\right\|^2 = 4\delta^2(\alpha^2 + \gamma^2\beta^2) \tag{2.9}$$

and

$$\nabla f_1 = [2\alpha(1-2\delta), 2\gamma\beta(1-2\gamma\delta)]^T \tag{2.10}$$

For the next iteration the curvature is given by

$$c_1 = \frac{2[f_0 - f_1 - \nabla^T f_1(\mathbf{x}^0 - \mathbf{x}^1)]}{\|\mathbf{x}^0 - \mathbf{x}^1\|^2} \tag{2.11}$$

Utilizing the information contained in (2.8)-(2.10), the various entries in expression (2.11) are known, and after substitution c_1 simplifies to

$$c_1 = \frac{2(\alpha^2 + \gamma^3\beta^2)}{\alpha^2 + \gamma^2\beta^2} \tag{2.12}$$

In the next iteration, Step 1 gives

$$\mathbf{x}^2 = \mathbf{x}^1 - \frac{\nabla f_1}{c_1} \tag{2.13}$$

And after the necessary substitutions for $\mathbf{x}^1$, ∇f_1 and c_1, given by (2.8), (2.10) and (2.12) respectively, (2.13) reduces to

$$\mathbf{x}^2 = [\alpha(1-2\delta)\mu_1, \beta(1-2\gamma\delta)\omega_1]^T \tag{2.14}$$

where

$$\mu_1 = 1 - \frac{1+\gamma^2\beta^2/\alpha^2}{1+\gamma^3\beta^2/\alpha^2} \tag{2.15}$$

and

$$\omega_1 = 1 - \frac{\gamma+\gamma^3\beta^2/\alpha^2}{1+\gamma^3\beta^2/\alpha^2} \tag{2.16}$$

Clearly if $\gamma = 1$, then $\mu_1 = 0$ and $\omega_1 = 0$. Thus by (2.14) $\mathbf{x}^2 = \mathbf{0}$ and convergence to the solution is achieved within the second iteration.

Now for $\gamma > 1$, and for any choice of α and β, it follows from (2.15) that

$$0 \leq \mu_1 \leq 1 \tag{2.17}$$

which implies from (2.14) that for the first component of $\mathbf{x}^2$:

$$\left|x_1^{(2)}\right| = |\alpha(1-2\delta)\mu_1| < |\alpha(1-2\delta)| = \left|x_1^{(1)}\right| \tag{2.18}$$

or introducing α notation (with $\alpha_0 = \alpha$), that

$$|\alpha_2| = |\mu_1\alpha_1| < |\alpha_1| \tag{2.19}$$

{Note: because $c_0 = 1/\delta > 0$ is chosen arbitrarily, it cannot be said that $|\alpha_1| < |\alpha_0|$. However α_1 is finite.}

The above argument, culminating in result (2.19), is for the two iterations $\mathbf{x}^0 \rightarrow \mathbf{x}^1 \rightarrow \mathbf{x}^2$. Repeating the argument for the sequence of overlapping pairs of iterations $\mathbf{x}^1 \rightarrow \mathbf{x}^2 \rightarrow \mathbf{x}^3$; $\mathbf{x}^2 \rightarrow \mathbf{x}^3 \rightarrow \mathbf{x}^4; \ldots$, it follows similarly that $|\alpha_3| = |\mu_2\alpha_2| < |\alpha_2|$; $|\alpha_4| = |\mu_3\alpha_3| < |\alpha_3|; \ldots$, since $0 \leq \mu_2 \leq 1$; $0 \leq$

$\mu_3 \leq 1; \ldots$, where the μs are given by (corresponding to equation (2.15) for μ_1):

$$\mu_1 = 1 - \frac{1 + \gamma^2 \beta_{j-1}^2 / \alpha_{j-1}^2}{1 + \gamma^3 \beta_{j-1}^2 / \alpha_{j-1}^2} \tag{2.20}$$

Thus in general

$$0 \leq \mu_j \leq 1 \tag{2.21}$$

and

$$|\alpha_{j+1}| = |\mu_j \alpha_j| < |\alpha_j| \tag{2.22}$$

For large positive integer m it follows that

$$|\alpha_m| = |\mu_{m-1}\alpha_{m-1}| = |\mu_{m-1}\mu_{m-2}\alpha_{m-2}| = |\mu_{m-1}\mu_{m-2}\cdots\mu_1\alpha_1| \tag{2.23}$$

and clearly for $\gamma > 0$, because of (2.21)

$$\lim_{m\to\infty} |\alpha_m| = 0 \tag{2.24}$$

Now for the second component of $\mathbf{x}^2$ in (2.14), the expression for ω_1, given by (2.16), may be simplified to

$$\omega_1 = \frac{1 - \gamma}{1 + \gamma^3 \beta^2 / \alpha^2} \tag{2.25}$$

Also for the second component:

$$x_2^{(2)} = \beta(1 - 2\gamma\delta)\omega_1 = \omega_1 x_2^{(1)} \tag{2.26}$$

or introducing β notation

$$\beta_2 = \omega_1 \beta_1 \tag{2.27}$$

The above argument is for $\mathbf{x}^0 \to \mathbf{x}^1 \to \mathbf{x}^2$ and again, repeating it for the sequence of overlapping pairs of iterations, it follows more generally for $j = 1, 2, \ldots$, that

$$\beta_{j+1} = \omega_j \beta_j \tag{2.28}$$

where ω_j is given by

$$\omega_j = \frac{1-\gamma}{1+\gamma^3\beta_{j-1}^2/\alpha_{j-1}^2} \tag{2.29}$$

Since by (2.24), $|\alpha_m| \to 0$, it follows that if $|\beta_m| \to 0$ as $m \to \infty$, the theorem is proved for the bivariate case. Make the assumption that $|\beta_m|$ does not tend to zero, then there exists a finite positive number ε such that

$$|\beta_j| \geq \varepsilon \tag{2.30}$$

for all j. This allows the following argument:

$$|\omega_j| = \left|\frac{1-\gamma}{1+\gamma^3\beta_{j-1}^2/\alpha_{j-1}^2}\right| \leq \left|\frac{1-\gamma}{1+\gamma^3\varepsilon^2/\alpha_{j-1}^2}\right| = \left|\frac{(1-\gamma)\alpha_{j-1}^2}{\alpha_{j-1}^2+\gamma^3\varepsilon^2}\right| \tag{2.31}$$

Clearly since by (2.24) $|\alpha_m| \to 0$ as $m \to \infty$, (2.31) implies that also $|\omega_m| \to 0$. This result taken together with (2.28) means that $|\beta_m| \to 0$ which contradicts the assumption above. With this result the theorem is proved for the bivariate case.

Although the algebra becomes more complicated, the above argument can clearly be extended to prove convergence for the multivariate case, where

$$f(\mathbf{x}) = \sum_{i=1}^{n} \gamma_i x_i^2, \;\; \gamma_1 = 1 < \gamma_2 < \gamma_3 < \ldots < \gamma_n \tag{2.32}$$

Finally since the general quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}, \;\; \mathbf{A} \text{ positive} - \text{definite} \tag{2.33}$$

may be transformed to the form (2.32), convergence of the SQSD method is also ensured in the general case.

2.5 Numerical results and conclusion

The SQSD method is now demonstrated by its application to some test problems. For comparison purposes the results are also given for the standard SD method and both the Fletcher-Reeves (FR) and Polak-Ribiere (PR) conjugate gradient methods. The latter two methods are implemented using the CG+ FORTRAN conjugate gradient program of Gilbert and Nocedal [111]. The CG+ implementation uses the line search routine of Moré and Thuente [112]. The function and gradient values are evaluated together in a single subroutine. The SD method is applied using CG+ with the search direction modified to the steepest descent direction. The FORTRAN programs were run on a 266 MHz Pentium 2 computer using double precision computations.

The standard (references [113, 114, 115, 116]) and other test problems used are listed in Appendix A and the results are given in Tables 2.1 and 2.2. The convergence tolerances applied throughout are $\varepsilon_g = 10^{-5}$ and $\varepsilon_x = 10^{-8}$, except for the extended homogenous quadratic function with $n = 50000$ (Problem 12) and the extremely ill-conditioned Manevich functions (Problems 14). For these problems the extreme tolerances $\varepsilon_g \cong 0 (= 10^{-75})$ and $\varepsilon_x = 10^{-12}$, are prescribed in an effort to ensure very high accuracy in the approximation $\mathbf{x}^c$ to $\mathbf{x}^*$. For each method the number of function-cum-gradient-vector evaluations (N^{fg}) are given. For the SQSD method the number of iterations is the same as N^{fg}. For the other methods the number of iterations (N^{it}) required for convergence, and which corresponds to the number of line searches executed, are also listed separately. In addition the relative error (E^r) in optimum function value, defined by

$$E^r = \left| \frac{f(\mathbf{x}^*) - f(\mathbf{x}^c)}{1 + |f(\mathbf{x}^*)|} \right| \tag{2.34}$$

where $\mathbf{x}^c$ is the approximation to $\mathbf{x}^*$ at convergence, is also listed. For the Manevich problems, with $n \geq 40$, for which the other (SD, FR and PR) algorithms fail to converge after the indicated number of steps, the infinite norm of the error in the solution vector (I^∞), defined by $\|\mathbf{x}^* - \mathbf{x}^c\|_\infty$ is also tabulated. These entries, given instead of the relative error in function value (E^r), are made in italics.

Inspection of the results shows that the SQSD algorithm is consistently com-

Prob. #	n	SQSD			Steepest Descent		
		ρ	N^{fg}	E^r	N^{fg}	N^{it}	E^r/I^∞
1	3	1	12	3.E-14	41	20	6.E-12
2	2	1	31	1.E-14	266	131	9.E-11
3	2	1	33	3.E-08	2316	1157	4.E-08
4	2	0.3	97	1.E-15	> 20000		3.E-09
5(a)	3	1	11	1.E-12	60	29	6.E-08
5(b)	3	1	17	1.E-12	49	23	6.E-08
6	4	1	119	9.E-09	> 20000		2.E-06
7	3	1	37	1.E-12	156	77	3.E-11
8	2	10	39	1.E-22	12050*	6023*	26*
9	2	0.3	113	5.E-14	6065	3027	2.E-10
10	2	1	43	1.E-12	1309	652	1.E-10
11	4	2	267	2.E-11	16701	8348	4.E-11
12	20	1.E+04	58	1.E-11	276	137	1.E-11
	200	1.E+04	146	4.E-12	2717	1357	1.E-11
	2000	1.E+04	456	2.E-10	> 20000		2.E-08
	20000	1.E+04	1318	6.E-09	> 10000		8.E+01
	50000	1.E+10	4073	3.E-16	> 10000		5.E+02
13	10	0.3	788	2.E-10	> 20000		4.E-07
	100	1	2580	1.E-12	> 20000		3.E+01
	300	1.73	6618	1.E-10	> 20000		2.E+02
	600	2.45	13347	1.E-11	> 20000		5.E+02
	1000	3.16	20717	2.E-10	> 30000		9.E+02
14	20	1	3651	2.E-27	> 20000		9.E-01
		10	3301	9.E-30			
	40	1	13302	5.E-27	> 30000		*1.E+00*
		10	15109	2.E-33			
	60	1	19016	7.E-39	> 30000		*1.E+00*
		10	16023	6.E-39			
	100	1	39690	1.E-49	> 50000		*1.E+00*
		10	38929	3.E-53			
	200	1	73517	5.E-81	> 100000		*1.E+00*
		10	76621	4.E-81			

* Convergence to a local minimum with $f(\mathbf{x}^c) = 48.9$.

Table 2.1: Performance of the SQSD and SD optimization algorithms when applied to the test problems listed in Appendix A

Prob. #	n	Fletcher-Reeves			Polak-Ribiere		
		N^{fg}	N^{it}	E^r/I^∞	N^{fg}	N^{it}	E^r/I^∞
1	3	7	3	0$	7	3	0$
2	2	30	11	2.E-11	22	8	2.E-12
3	2	45	18	2.E-08	36	14	6.E-11
4	2	180	78	1.E-11	66	18	1.E-14
5(a)	3	18	7	6.E-08	18	8	6.E-08
5(b)	3	65	31	6.E-08	26	11	6.E-08
6	4	1573	783	8.E-10	166	68	3.E-09
7	3	132	62	4.E-12	57	26	1.E-12
8	2	72*	27*	26*	24*	11*	26*
9	2	56	18	5.E-11	50	17	1.E-15
10	2	127	60	6.E-12	30	11	1.E-11
11	4	193	91	1.E-12	99	39	9.E-14
12	20	42	20	9.E-32	42	20	4.E-31
	200	163	80	5.E-13	163	80	5.E-13
	2000	530	263	2.E-13	530	263	2.E-13
	20000	1652	825	4.E-13	1652	825	4.E-13
	50000	3225	1161	1.E-20	3225	1611	1.E-20
13	10	> 20000		2.E-02	548	263	4.E-12
	100	> 20000		8.E+01	1571	776	2.E-12
	300	> 20000		3.E+02	3253	1605	2.E-12
	600	> 20000		6.E+02	5550	2765	2.E-12
	1000	> 30000		1.E+03	8735	4358	2.E-12
14	20	187	75	8.E-24	1088	507	2.E-22
	40	> 30000		*1.E+00*	> 30000		*1.E+00*
	60	> 30000		*1.E+00*	> 30000		*1.E+00*
	100	> 50000		*1.E+00*	> 50000		*1.E+00*
	200	> 100000		*1.E+00*	> 100000		*1.E+00*

* Convergence to a local minimum with $f(\mathbf{x}^c) = 48.9$; $ Solution to machine accuracy.

Table 2.2: Performance of the FR and PR algorithms when applied to the test problems listed in Appendix A

petitive with the other three methods and performs notably well for large problems. Of all the methods the SQSD method appears to be the most reliable one in solving each of the posed problems. As expected, because line searches are eliminated and consecutive search directions are no longer forced to be orthogonal, the new method completely overshadows the standard SD method. What is much more gratifying, however, is the performance of the SQSD method relative to the well-established and well-researched conjugate gradient algorithms. Overall the new method appears to be very competitive with respect to computational efficiency and, on the evidence presented, remarkably stable.

In the implementation of the SQSD method to highly non-quadratic and non-convex functions, some care must however be taken in ensuring that the chosen step limit parameter ρ, is not too large. A too large value may result in excessive oscillations occurring before convergence. Therefore a relatively small value, $\rho = 0.3$, was used for the Rosenbrock problem with $n = 2$ (Problem 4). For the extended Rosenbrock functions of larger dimensionality (Problems 13), correspondingly larger step limit values ($\rho = \sqrt{n}/10$) were used with success.

For quadratic functions, as is evident from the convergence analysis of Section 2.4, no step limit is required for convergence. This is borne out in practice by the results for the extended homogenous quadratic functions (Problems 12), where the very large value $\rho = 10^4$ was used throughout, with the even more extreme value of $\rho = 10^{10}$ for $n = 50000$. The specification of a step limit in the quadratic case also appears to have little effect on the convergence rate, as can be seen from the results for the ill-conditioned Manevich functions (Problems 14), that are given for both $\rho = 1$ and $\rho = 10$. Here convergence is obtained to at least 11 significant figures accuracy ($\|\mathbf{x}^* - \mathbf{x}^c\|_\infty < 10^{-11}$) for each of the variables, despite the occurrence of extreme condition numbers, such as 10^{60} for the Manevich problem with $n = 200$.

The successful application of the new method to the ill-conditioned Manevich problems, and the analysis of the convergence behavior for quadratic functions, indicate that the SQSD algorithm represents a powerful approach to solving quadratic problems with large numbers of variables. In particular, the SQSD method can be seen as an *unconditionally convergent*, *stable* and *economic* alternative iterative method for solving large systems of linear equations, ill-conditioned or not, through the minimization of the sum of the squares of the residuals of the equations.