Steplength Selection in Interior-Point Methods for Quadratic Programming*

Frank Curtis† Jorge Nocedal‡

December 19, 2005

Abstract

We present a new strategy for choosing primal and dual steplengths in a primal-dual interior-point algorithm for convex quadratic programming. Current implementations often scale steps equally to avoid increases in dual infeasibility between iterations. We propose that this method can be too conservative, while safeguarding an unequally-scaled steplength approach will often require fewer steps toward to a solution. Computational results are given.

1 Introduction

In this paper we consider the convex quadratic program in standard form:

$$\min_{x \in \mathbb{R}^n} q(x) = \tfrac{1}{2} x^T Q x + c^T x \qquad \text{s.t. } Ax = b,\ x \geq 0, \tag{1.1}$$

where $Q \in \mathbb{R}^{n \times n}$ is symmetric positive semidefinite, $A \in \mathbb{R}^{m \times n}$ has full row rank, $c \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$. The corresponding dual program is given by

$$\max_{x,z \in \mathbb{R}^n,\ y \in \mathbb{R}^m} -\tfrac{1}{2} x^T Q x + b^T y \qquad \text{s.t. } -Qx + A^T y + z = c,\ z \geq 0. \tag{1.2}$$

An interior-point method applied to (1.1)-(1.2) will determine, from an initial point (x, y, z) with $(x, z) > 0$, an appropriate step $(\Delta x, \Delta y, \Delta z)$ in the primal-dual space. If $(\Delta x, \Delta y, \Delta z)$ does not maintain positivity of the bounded variables (x, z), then the algorithm must backtrack from this infeasible step.

A simple technique is to update

$$(x^+, y^+, z^+) \leftarrow (x, y, z) + \alpha(\Delta x, \Delta y, \Delta z),$$

where the common steplength multiplier α is chosen as large as possible subject to the fraction to the boundary rule. Scaling steps equally in this manner has been recommended in the literature; see for example [1, 4, 11] and the references therein. The main advantage of this approach is the guaranteed reduction in infeasibility that will be attained along this scaled Newton step.

*This work was supported in part by a grant from the Intel Corporation.

†Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 60208-3118, USA.

‡Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 60208-3118, USA.

Empirical evidence has shown, however, that this approach is unnecessarily conservative in numerous cases. Convergence can still be attained with an unequally-scaled steplength approach that allows for longer steps. We propose the use of three separate steplength multipliers, one for each of Δx, Δy, and Δz, to allow more freedom in the primal and dual steps once we find that taking the full Newton step is prohibited.

Throughout the paper, e denotes the vector of 1s of appropriate dimension. All norms referenced are considered Euclidean norms.

2 The Primal-Dual Algorithm

The first-order optimality conditions, also known as the KKT conditions, for (x, y, z) to be a solution to (1.1)-(1.2) are

$$Ax - b = 0 \tag{2.3a}$$

$$A^T y + z - Qx - c = 0 \tag{2.3b}$$

$$Xz = 0 \tag{2.3c}$$

$$(x, z) \geq 0 \tag{2.3d}$$

where we denote

$$X = \text{diag}(x) \quad \text{and} \quad Z = \text{diag}(z).$$

Newton's method for solving (2.3a)-(2.3c) obtains a search direction as the solution to the linear system

$$\begin{bmatrix} -Q & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} Qx + c - A^T y - z \\ b - Ax \\ -Xz \end{bmatrix} \equiv \begin{bmatrix} r_d \\ r_p \\ r_c \end{bmatrix}. \tag{2.4}$$

We refer to the norms of the residual vectors r_p and r_d as *primal* and *dual infeasibility*, respectively, and together as the *infeasibility* of (x, y, z). The expression $-r_c^T e/n = x^T z/n$ is known as the *complementarity* of (x, z). We are satisfied with an estimate of the solution if (2.3d) holds and $\|(r_p, r_d, r_c)\|$ is below a given tolerance.

Primal-dual interior-point methods for quadratic programming have received much attention; see e.g. [1, 4, 8, 9, 13]. Rather than try to solve the system (2.3) directly, these methods include a centering term to promote steps toward the interior of the feasible set. They aim to find points on the *central path*, which is defined as the set of points solving (2.3) with (2.3c) replaced by

$$Xz = \gamma e \tag{2.5}$$

for some $\gamma > 0$. The search direction is obtained by solving

$$\begin{bmatrix} -Q & A^T & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} r_d \\ r_p \\ r_c + \gamma e \end{bmatrix}, \tag{2.6}$$

where $\gamma = \sigma\mu$ such that $\sigma \in (0,1)$ and $\mu = x^T z/n$ to reflect the complementarity of the current iterate. Steplength multipliers $(\alpha_x, \alpha_y, \alpha_z)$ are chosen to ensure that the update

$$(x^+, y^+, z^+) \leftarrow (x, y, z) + (\alpha_x \Delta x, \alpha_y \Delta y, \alpha_z \Delta z) \tag{2.7}$$

maintains $(x^+, z^+) > 0$.

Let us consider the effect of the steplength multipliers on primal-dual infeasibility and complementarity of the following iterate. We find from (2.6) and (2.7)

$$\begin{aligned} r_p^+(\alpha_x) &\equiv b - Ax^+ \\ &= b - A(x + \alpha_x \Delta x) \\ &= (1 - \alpha_x) r_p, \end{aligned}$$

$$\begin{aligned} r_d^+(\alpha_x, \alpha_y, \alpha_z) &\equiv Qx^+ + c - A^T y^+ - z^+ \\ &= Q(x + \alpha_x \Delta x) + c - A^T(y + \alpha_y \Delta y) - (z + \alpha_z \Delta z) \\ &= r_d + \alpha_x Q \Delta x - \alpha_y A^T \Delta y - \alpha_z \Delta z \\ &= (1 - \alpha_y) r_d + (\alpha_x - \alpha_y) Q \Delta x + (\alpha_y - \alpha_z) \Delta z, \end{aligned} \tag{2.8}$$

and

$$\begin{aligned} \mu^+(\alpha_x, \alpha_z) &\equiv {x^+}^T z^+ / n \\ &= (x + \alpha_x \Delta x)^T (z + \alpha_z \Delta z)/n \\ &= \mu + \alpha_x \Delta x^T z/n + \alpha_z x^T \Delta z/n + \alpha_x \alpha_z \Delta x^T \Delta z/n. \end{aligned}$$

Notice that if a common steplength multiplier α is used, then primal and dual infeasibility will decrease by the common factor $(1 - \alpha)$. Moreover, for $\alpha > 0$ sufficiently small, we find

$$\mu^+(\alpha) = \mu(1 - \alpha(1 - \sigma)) + \alpha^2 \Delta x^T \Delta z/n \ < \ \mu$$

so complementarity will also decrease.

When $Q = 0$, as in linear programming, steps are chosen as large as possible as long as the bounded variables (x, z) remain positive. As can be seen above, primal and dual infeasibility will decrease the most when choosing the largest allowable α_x and $\alpha_y = \alpha_z$. The presence of a nonzero Q in (2.8), however, couples the primal and dual steplength multipliers and dual infeasibility may increase if large unequal steplength multipliers are used; this may cause the algorithm to diverge. Any unequally-scaled steplength strategy must take these considerations into account.

In the algorithm proposed in this paper we require that the steplengths provide a decrease in the merit function

$$\phi(x, y, z) = \|Ax - b\|^2 + \|A^T y + z - Qx - c\|^2 + x^T z. \tag{2.9}$$

This choice of ϕ is made for three main reasons. First, it is closely related to the norm of the KKT residual vector, which must be small in the vicinity of the optimal solution. Second, slightly more weight is applied to large (greater than 1) primal and dual infeasibility terms. This is reasonable since *feasibility*, or the state of having $\|r_p\|$ and $\|r_d\|$ below a given tolerance, should generally be achieved at least as fast as reduced complementarity on the solution path. Finally, in Section 4, we will show that the unique minimum of ϕ can easily be found in a particular set of steplength multiplier vectors that guarantees a decrease in the function.

3 Steplength Strategies

In light of the difficulties imposed by a nonzero quadratic term in (1.1), a conservative strategy is generally employed. In this section we outline this approach and remark on an alternative method proposed in [8].

Current practice is to set primal and dual steplength multipliers to a common value while preserving the positivity of (x, z). That is, to set $(\alpha_x, \alpha_y, \alpha_z) = \bar{\alpha}e$ where

$$\bar{\alpha}_x = \beta \left[\max_{k=1,\ldots,n} \{1, -\Delta x_k / x_k\} \right]^{-1}, \quad \bar{\alpha}_z = \beta \left[\max_{k=1,\ldots,n} \{1, -\Delta z_k / z_k\} \right]^{-1}$$

for a given $0 < \beta < 1$ and

$$\bar{\alpha} = \min\{\bar{\alpha}_x, \bar{\alpha}_z\}.$$

This process of setting $(\alpha_x, \alpha_z) < (\bar{\alpha}_x, \bar{\alpha}_z)$ is known as the applying the *fraction to the boundary rule*. As seen in Section 2, this strategy leads to a guaranteed reduction in primal and dual infeasibility. Once feasibility is obtained, one can choose $\alpha \in (0, \bar{\alpha}]$ for a guaranteed reduction in complementarity. More precisely, at a feasible point we can choose α to solve

$$\min_{\alpha \in I\!R} \ (x + \alpha \Delta x)^T (z + \alpha \Delta z) \qquad \text{s.t.} \ \ 0 < \alpha < \bar{\alpha}. \tag{3.10}$$

An alternative approach has been proposed by Mészáros in [8]. He observes that feasibility can be obtained more rapidly by choosing unequal steplength multipliers based on the efficient set of points for the quadratic multiobjective problem

$$\left.\begin{array}{ll} \min & \|r_p^+(\alpha_x)\| \\ \min & \|r_d^+(\alpha_x, \alpha_z)\| \end{array}\right\} \ \text{s.t.} \quad \begin{array}{l} 0 \le \alpha_x \le \bar{\alpha}_x, \\ 0 \le \alpha_z \le \bar{\alpha}_z. \end{array} \tag{3.11}$$

Note that he maintains $\alpha_y = \alpha_z$. A point (α_x^*, α_z^*) is in the efficient set of (3.11) if, for any (α_x, α_z) satisfying the bounds in (3.11), we have

$$\begin{array}{rcl} \|r_p^+(\alpha_x)\| < \|r_p^+(\alpha_x^*)\| & \Rightarrow & \|r_d^+(\alpha_x^*, \alpha_z^*)\| < \|r_d^+(\alpha_x, \alpha_z)\|; \\ \|r_d^+(\alpha_x, \alpha_z)\| < \|r_d^+(\alpha_x^*, \alpha_z^*)\| & \Rightarrow & \|r_p^+(\alpha_x^*)\| < \|r_p^+(\alpha_x)\|. \end{array} \tag{3.12}$$

In other words, the pair (α_x^*, α_z^*) is not dominated by any other feasible pair. Assuming $r_p \neq 0$, $r_d \neq 0$, and $Q\Delta x \neq 0$, conditions which are normally achieved in practice, the efficient set can be shown to be equal to

$$\{(\alpha_x, \alpha_z^*) \mid \alpha_x^* \le \alpha_x \le \bar{\alpha}_x\}, \tag{3.13}$$

where (α_x^*, α_z^*) is chosen as the pair that minimizes dual infeasibility over the set

$$\{(\alpha_x, \alpha_z) \ge 0 \mid \alpha_x = \bar{\alpha}_x \text{ or } \alpha_z = \bar{\alpha}_z\}. \tag{3.14}$$

In particular, Mészáros chooses $(\alpha_x, \alpha_z) = (\max\{\bar{\alpha}, \alpha_x^*\}, \alpha_z^*)$. Once feasibility is attained, he also chooses the common steplength multiplier found by solving (3.10).

4 A New Steplength Strategy

When an adequate globalization technique is not used, setting the bounded steplength multipliers to their largest permissible values $(\alpha_x = \bar{\alpha}_x, \alpha_z = \bar{\alpha}_z)$ may cause divergence. A strategy

such as those described in Section 3 must be employed. However, these particular methods have some disadvantages. Equal steplengths are guaranteed to reduce primal and dual infeasibility, but, as observed by Mészáros, other choices of $(\alpha_x, \alpha_y, \alpha_z)$ may reduce primal and dual infeasibility even more. Strictly observing the step's effect on infeasibility of the next iterate may not be the most efficient method either. We stress that the third measure, complementarity, should not be ignored.

Large unequally-scaled steps are often accepted in algorithms for linear and nonlinear programming; see [7] and [2, 3, 10, 12], respectively. We conjecture that such a choice can be beneficial in convex quadratic programming as well, as long as necessary globalization techniques are implemented; e.g. the observation of a merit function ϕ. We propose a method for selecting steplengths with motivation behind both the specific merit function observed as well as the particular subset of feasible multipliers considered. Rather than perform the costly minimization of ϕ (see (2.9)) over the entire feasible set, we choose a set over which we can guarantee a reduction in the function via a subproblem whose unique solution requires minimal extra computational effort compared to the equally-scaled steplength approach. By creating a separate multiplier for the step in the dual variables Δy, which can be freely scaled, we allow more freedom of movement in the dual space.

A risk in using Mészáros' method is the possible selection of the multipliers $(\bar{\alpha}_x, 0, 0)$ over a number of consecutive iterations, if this choice minimizes primal and dual infeasibility of the following iterate. This myopic choice may stagnate the algorithm as no step is taken in the dual space. Therefore, we would generally prefer to promote long steps over the search space. If short steplengths are to be selected, say for a large decrease in complementarity, then we wish to take advantage of the guaranteed simultaneous reduction in primal and dual infeasibility characteristic of equally-scaled steplengths. We also aim to provide a decrease in the merit function at least as that obtained by equally-scaled steplengths. We now present our method formally.

Let α_y be an unrestricted steplength multiplier. The multipliers (α_x, α_z) will be chosen based on the two sets

$$\mathcal{A}_1 = \{(\alpha, \alpha) \mid 0 \le \alpha \le \bar{\alpha}\}$$

and

$$\mathcal{A}_2 = \begin{cases} \{(\bar{\alpha}, \alpha_z) \mid \bar{\alpha} \le \alpha_z \le \bar{\alpha}_z\} & \text{if } \bar{\alpha}_x \le \bar{\alpha}_z; \\ \{(\alpha_x, \bar{\alpha}) \mid \bar{\alpha} \le \alpha_x \le \bar{\alpha}_x\} & \text{if } \bar{\alpha}_x \ge \bar{\alpha}_z. \end{cases}$$

Their union, $\mathcal{A}_1 \cup \mathcal{A}_2$, can be visualized as the bold line segments in Figures 1 and 2.

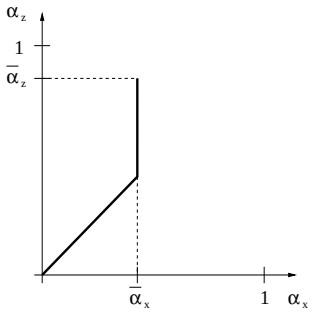

Figure 1: $\mathcal{A}_1 \cup \mathcal{A}_2$ if $\bar{\alpha}_x \le \bar{\alpha}_z$

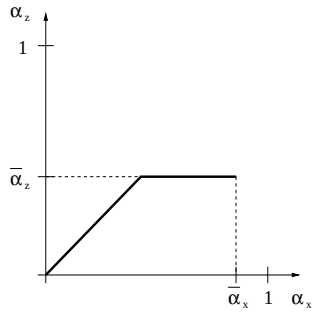

Figure 2: $\mathcal{A}_1 \cup \mathcal{A}_2$ if $\bar{\alpha}_x \ge \bar{\alpha}_z$

We will choose steplength multipliers as the unique solution to

$$\min_{\alpha_x,\alpha_y,\alpha_z\in\mathbb{R}} \phi(x+\alpha_x\Delta x, y+\alpha_y\Delta y, z+\alpha_z\Delta z) \qquad \text{s.t. } (\alpha_x,\alpha_z)\in\mathcal{A}_1\cup\mathcal{A}_2. \tag{4.15}$$

Notice that every feasible truncated Newton step is considered in this approach, so this choice of steplength multipliers will decrease the merit function as least as much as equally-scaled steplengths. Moreover, we have the following remark.

Remark 4.1. *The global optimum of* (4.15) *can be determined by solving two 2-dimensional quadratic programs, each with simple bound constraints. (Restricting $\alpha_y=\alpha_z$, the subproblems are 1-dimensional quadratics with simple bounds.)*

We describe these quadratic problems, so that the effort of forming and solving them can easily be quantified. Define

$$r=\begin{bmatrix} r_p \\ r_d\end{bmatrix}, \quad s=\begin{bmatrix}-A\\ Q\end{bmatrix}\Delta x, \quad t=\begin{bmatrix}0\\ -A^T\end{bmatrix}\Delta y, \quad \text{and} \quad u=\begin{bmatrix}0\\ -I\end{bmatrix}\Delta z,$$

so the objective function of (4.15) reduces to

$$\phi(x+\alpha_x\Delta x, y+\alpha_y\Delta y, z+\alpha_z\Delta z) = \|r+\alpha_x s+\alpha_y t+\alpha_z u\|^2 + (x+\alpha_x\Delta x)^T(z+\alpha_z\Delta z).$$

The first trial point, which we define as the minimizer of ϕ over the set $\mathcal{A}_1$, can be computed as the solution to

$$\min_{\alpha_x,\alpha_y\in\mathbb{R}} q_1(\alpha_x,\alpha_y) \qquad \text{s.t. } 0\le\alpha_x\le\bar{\alpha},$$

with $\alpha_z=\alpha_x$, and where q_1 is a quadratic function of the form (1.1) with

$$Q=\begin{bmatrix}(s+u)^T(s+u)+\Delta x^T\Delta z & (s+u)^Tt\\ (s+u)^Tt & t^Tt\end{bmatrix}, \quad c=\begin{bmatrix} r^T(s+u)+\frac{1}{2}\left(\Delta x^Tz+x^T\Delta z\right)\\ r^Tt\end{bmatrix}.$$

If $\bar{\alpha}_x\le\bar{\alpha}_z$, then the trial point in $\mathcal{A}_2$ is given by $\alpha_x=\bar{\alpha}_x$ and the solution to

$$\min_{\alpha_y,\alpha_z\in\mathbb{R}} q_2(\alpha_y,\alpha_z) \qquad \text{s.t. } \bar{\alpha}\le\alpha_z\le\bar{\alpha}_z,$$

where q_2 is given by (1.1) with

$$Q=\begin{bmatrix}t^Tt & t^Tu\\ t^Tu & u^Tu\end{bmatrix}, \qquad c=\begin{bmatrix} r^Tt+\bar{\alpha}_x s^Tt\\ r^Tu+\frac{1}{2}x^T\Delta z+\bar{\alpha}_x\left(s^Tu+\frac{1}{2}\Delta x^T\Delta z\right)\end{bmatrix}.$$

Otherwise, the trial point is given by $\alpha_z=\bar{\alpha}_z$ and the solution to

$$\min_{\alpha_x,\alpha_y\in\mathbb{R}} q_3(\alpha_x,\alpha_y) \qquad \text{s.t. } \bar{\alpha}\le\alpha_x\le\bar{\alpha}_x.$$

where q_3 is defined by (1.1) with

$$Q=\begin{bmatrix}s^Ts & s^Tt\\ s^Tt & t^Tt\end{bmatrix}, \qquad c=\begin{bmatrix} r^Ts+\frac{1}{2}\Delta x^Tz+\bar{\alpha}_z\left(s^Tu+\frac{1}{2}\Delta x^T\Delta z\right)\\ r^Tt+\bar{\alpha}_z t^Tu\end{bmatrix}.$$

We choose the one of these two trial points with a smaller merit function value. Recalling the analysis in Section 2, we note that a reduction in the merit function is guaranteed over this set.

We claim that the extra work required to compute and compare the trial steplengths above is small, and reduces to a handful of inner products. These products are used to construct the components of the subproblems and evaluate the merit function at the resulting trial points. In fact, what appear to be the most costly operations, namely the required matrix-vector products $A\Delta x$, $Q\Delta x$, $A^T\Delta y$ that define s and t, are necessary in other parts of the algorithm. For example, they appear in the calculations of iterative refinement for the solution of the linear system (2.6). Therefore, we can compare our method to the methods described in Section 3 simply by observing iteration counts. The required run-time per iteration will not increase significantly. We also note that the matrices in the subproblems for $\mathcal{A}_2$ are positive semidefinite, so the trial point in this set is obtained by solving a 2-dimensional *convex* quadratic program.

5 Computational Results

A Matlab code was written to solve a variety of problems from the Maros and Mészáros test set; see [6]. Problems from this set, for which AMPL models were available, were selected based on size — fewer than 10,000 variables. Steplength multipliers were chosen according to an equally-scaled steplength strategy ("eq.") where ($\alpha_x = \alpha_y = \alpha_z = \bar{\alpha}$), Mészáros' method where we set $(\alpha_x, \alpha_z) = (\max\{\bar{\alpha}, \alpha_x^*\}, \alpha_z^*)$ ("Més."), and our new steplength strategy described in Section 4 ("new"). We employed Matlab's `quadprog` function to solve the subproblems for determining $(\alpha_x, \alpha_y, \alpha_z)$ during each iteration.

Table 1 contains iteration counts for the problems solved. The relative differences between equally-scaled steplengths and Mészáros' method and between equally-scaled steplengths and our method are given, rounded to the nearest integer. Negative numbers refer to an improvement over employing equally-scaled steplengths. Problems were considered solved when the norm of the KKT residual vector was less than 10^{-6}.

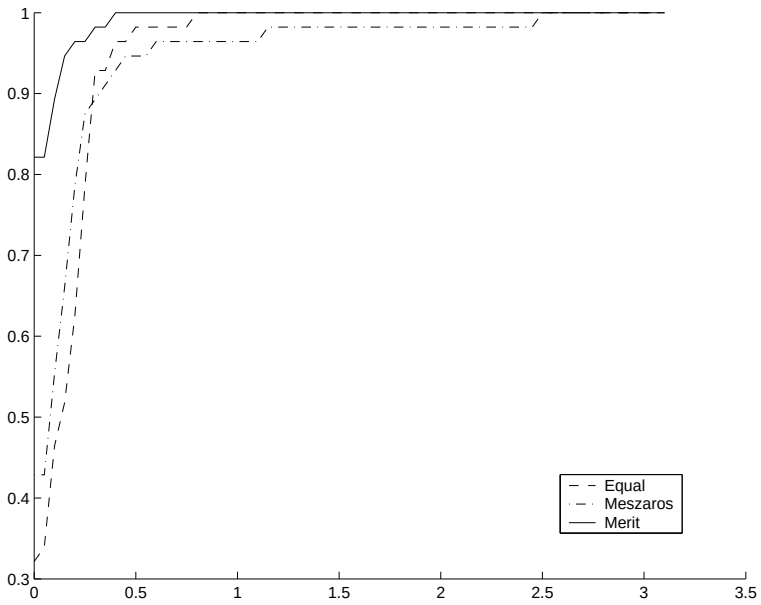

Figure 3: Performance profile for iteration counts

	Iter.			Rel. Diff.			Iter.			Rel. Diff.	
Problem	eq.	Més.	new	Més.	new	Problem	eq.	Més.	new	Més.	new
aug3dcqp	19	19	20	0%	5%	qfffff80	63	60	53	-5%	-16%
aug3dqp	20	20	19	0%	-5%	qisrael	51	49	45	-4%	-12%
cvxqp1	19	18	19	-5%	0%	qpcboei2	40	39	50	-3%	25%
dual1	13	13	14	0%	8%	qsc205	21	21	20	0%	-5%
dual2	9	9	10	0%	11%	qscagr25	30	28	25	-7%	-17%
dual3	9	9	11	0%	22%	qscagr7	29	48	22	66%	-24%
dual4	9	9	9	0%	0%	qscfxm1	45	45	39	0%	-13%
dualc1	17	20	17	18%	0%	qscfxm2	56	57	48	2%	-14%
dualc2	15	15	15	0%	0%	qscfxm3	59	57	51	-3%	-14%
dualc5	9	9	9	0%	0%	qscrs8	43	39	33	-9%	-23%
dualc8	11	9	9	-18%	-18%	qscsd1	14	13	13	-7%	-7%
hs021	9	9	8	0%	-11%	qscsd6	20	20	19	0%	-5%
hs035	10	10	10	0%	0%	qscsd8	19	18	17	-5%	-11%
hs053	4	4	4	0%	0%	qsctap1	27	26	24	-4%	-11%
hs076	12	12	12	0%	0%	qsctap2	24	24	21	0%	-13%
hs35mod	9	9	9	0%	0%	qsctap3	26	25	22	-4%	-15%
hues-mod	23	34	23	48%	0%	qseba	55	54	52	-2%	-5%
huestis	24	23	17	-4%	-29%	qshare1b	40	38	34	-5%	-15%
lotschd	12	9	7	-25%	-42%	qshare2b	30	31	25	3%	-17%
mosarqp1	17	17	17	0%	0%	qshell	61	61	51	0%	-16%
mosarqp2	19	19	19	0%	0%	qship04l	23	23	20	0%	-13%
q25fv47	50	48	44	-4%	-12%	qship04s	22	22	20	0%	-9%
qadlittl	22	20	19	-9%	-14%	qship08l	26	26	26	0%	0%
qafiro	16	13	13	-19%	-19%	qship08s	26	25	26	-4%	0%
qbandm	32	32	29	0%	-9%	qship12s	28	27	29	-4%	4%
qbeaconf	24	22	20	-8%	-17%	qsierra	43	–	36	–	-16%
qbrandy	30	26	27	-13%	-10%	qstandat	23	20	19	-13%	-17%
qetamacr	38	38	42	0%	11%	tame	4	4	4	0%	0%

Table 1: Iteration counts for three steplength strategies

Notice that by employing Mészáros' method we fail to solve the problem `qsierra` before the maximum number (200) of iterations is reached. The algorithm continually chooses steplength multipliers of the form $(\bar{\alpha}_x, 0, 0)$ which, as described in Section 4, causes no step to be taken in the dual space. Consequently, progress towards the optimal solution is slowed considerably.

The results are summarized in figure 3 in terms of a logarithmic performance profile described in [5].

We observe that the new strategy produces a fairly consistent reduction in required iterations when compared to the common technique of choosing equally-scaled steplengths. Furthermore, the gains are more pronounced on more difficult problems. We also tested a variant of the new algorithm in which the dual steplengths are equal, i.e., $\alpha_y = \alpha_z$, and observed that it is slightly less efficient.

References

[1] K. M. Anstreicher and J. Ph. Vial. On the convergence of an infeasible primal-dual interior-point method for convex programming. *Optimization Methods and Software*, 3(4):273–283, 1994.

[2] J. Betts, S. K. Eldersveld, P. D. Frank, and J. G. Lewis. An interior-point nonlinear programming algorithm for large scale optimization. Technical report MCT TECH-003, Mathematics and Computing Technology, The Boeing Company, P.O. Box 3707, Seattle WA 98124-2207, 2000.

[3] R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, 1999.

[4] T. J. Carpenter, I. J. Lustig, J. M. Mulvey, and D. F. Shanno. Higher-order predictor-corrector interior point methods with application to quadratic objectives. *SIAM Journal on Optimization*, 3(4):696–725, 1993.

[5] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming, Series A*, 91:201–213, 2002.

[6] I. Maros and C. Meszaros. A repository of convex quadratic programming problems. *Optimization Methods and Software*, 11 & 12:671–681, 1999. Special Issue on Interior Point Methods.

[7] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, 2(4):575–601, 1992.

[8] Cs. Meszaros. Steplengths in infeasible primal-dual interior point methods for quadratic programming. *Operations Research Letters*, 213:39–45, 1999.

[9] R. J. Vanderbei. LOQO: an interior point code for quadratic programming. Technical Report SOR 94-15, Program in Statistics and Operations,Research, Princeton University, New Jersey, USA, 1994.

[10] R. J. Vanderbei and D. F. Shanno. An interior point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13:231–252, 1999.

[11] R.J. Vanderbei. *Linear Programming: Foundations and Extensions*. Springer Verlag, New York, second edition, 2001.

[12] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. Technical Report RC 23149, IBM T. J. Watson Research Center, Yorktown Heights, NY, USA, March 2004.

[13] S. Wright. *Primal-dual interior-point methods*. Society for Industrial and Applied Mathematics, Philadelphia, 1997.