Math. Prog. Comp. (2015) 7:237–267
DOI 10.1007/s12532-015-0084-4

FULL LENGTH PAPER

# On fast trust region methods for quadratic models with linear constraints

**M. J. D. Powell**[1]

Received: 25 August 2014 / Accepted: 22 April 2015 / Published online: 12 June 2015

**Abstract** Quadratic models $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, of the objective function $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, are used by many successful iterative algorithms for minimization, where $k$ is the iteration number. Given the vector of variables $\underline{x}_k \in \mathcal{R}^n$, a new vector $\underline{x}_{k+1}$ may be calculated that satisfies $Q_k(\underline{x}_{k+1}) < Q_k(\underline{x}_k)$, in the hope that it provides the reduction $F(\underline{x}_{k+1}) < F(\underline{x}_k)$. Trust region methods include a bound of the form $\|\underline{x}_{k+1} - \underline{x}_k\| \leq \Delta_k$. Also we allow general linear constraints on the variables that have to hold at $\underline{x}_k$ and at $\underline{x}_{k+1}$. We consider the construction of $\underline{x}_{k+1}$, using only of magnitude $n^2$ operations on a typical iteration when $n$ is large. The linear constraints are treated by active sets, which may be updated during an iteration, and which decrease the number of degrees of freedom in the variables temporarily, by restricting $\underline{x}$ to an affine subset of $\mathcal{R}^n$. Conjugate gradient and Krylov subspace methods are addressed for adjusting the reduced variables, but the resultant steps are expressed in terms of the original variables. Termination conditions are given that are intended to combine suitable reductions in $Q_k(\cdot)$ with a sufficiently small number of steps. The reason for our work is that $\underline{x}_{k+1}$ is required in the LINCOA software of the author, which is designed for linearly constrained optimization without derivatives when there are hundreds of variables. Our studies go beyond the final version of LINCOA, however, which employs conjugate gradients with termination at the trust region boundary. In particular, we find that, if an active set is updated at a point that is not the trust region centre, then the Krylov method may terminate too early due to a degeneracy. An extension to the conjugate gradient method for searching round the trust region

Sadly, M. J. D. Powell died on April 19th 2015. Article related queries can be directed to Nick Gould (nick.gould@stfc.ac.uk).

✉ M. J. D. Powell

[1] Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, Wilberforce Road, Cambridge CB3 0WA, England

boundary receives attention too, but it was also rejected from LINCOA, because of poor numerical results. The given techniques of LINCOA seem to be adequate in practice.

**Keywords** Conjugate gradients · Krylov subspaces · Linear constraints · Quadratic models · Trust region methods

**Mathematics Subject Classification** 65K05 · 90C20 · 90C26 · 90C56 · 49M37

## 1 Introduction

Let an iterative algorithm be applied to seek the least value of a general objective function $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, subject to the linear constraints

$$\underline{a}_j^T \underline{x} \le b_j, \quad j = 1, 2, \ldots, m, \tag{1.1}$$

the value $m = 0$ being reserved for the unconstrained case. We assume that, for any vector of variables $\underline{x} \in \mathcal{R}^n$, the function value $F(\underline{x})$ can be calculated. The vectors $\underline{x}$ of these calculations are generated automatically by the algorithm after some preliminary work. A feasible point $\underline{x}_1$ with $F(\underline{x}_1)$ are required for the first iteration, where feasible means that the constraints (1.1) are satisfied. For every iteration number $k$, we define $\underline{x}_k$ to be the point that, at the start of the $k$-th iteration, has supplied the least calculated value of the objective function so far, subject to $\underline{x}_k$ being feasible. If this point is not unique, we choose the candidate that occurs first, so $\underline{x}_{k+1} \neq \underline{x}_k$ implies the strict reduction $F(\underline{x}_{k+1}) < F(\underline{x}_k)$.

The main task of the $k$-th iteration is to pick a new vector of variables, $\underline{x}_k^+$ say, or to decide that the sequence of iterations is complete. On some iterations, $\underline{x}_k^+$ may be infeasible, in order to investigate changes to the objective function when moving away from a constraint boundary, an extreme case being when the boundary is the set of points that satisfy a linear equality constraint that is expressed as two linear inequalities. We restrict attention, however, to an iteration that makes $\underline{x}_k^+$ feasible, and that tries to achieve the reduction $F(\underline{x}_k^+) < F(\underline{x}_k)$. We also restrict attention to algorithms that employ a quadratic model $Q_k(\underline{x}) \approx F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, and a trust region radius $\Delta_k > 0$.

We let the model function of the $k$-th iteration be the quadratic

$$Q_k(\underline{x}) \;=\; F(\underline{x}_k) + (\underline{x} - \underline{x}_k)^T \underline{g}_k + \frac{1}{2}\,(\underline{x} - \underline{x}_k)^T H_k\,(\underline{x} - \underline{x}_k), \quad \underline{x} \in \mathcal{R}^n, \tag{1.2}$$

the vector $\underline{g}_k \in \mathcal{R}^n$ and the $n \times n$ symmetric matrix $H_k$ being chosen before the start of the iteration. The trust region radius $\Delta_k$ is also chosen in advance. The ideal vector $\underline{x}_k^+$ would be the vector $\underline{x}$ that minimizes $Q_k(\underline{x})$ subject to the linear constraints (1.1) and the trust region bound

$$\|\underline{x} - \underline{x}_k\| \;\le\; \Delta_k. \tag{1.3}$$

The purpose of our work, however, is to generate a vector $\underline{x}_k^+$ that is a useful approximation to this ideal vector, and whose construction requires only $\mathcal{O}(n^2)$ operations on most iterations. This is done by the LINCOA Fortran software [12], developed by the author for linearly constrained optimization when derivatives of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, are not available. LINCOA has been applied successfully to several test problems that have hundreds of variables without taking advantage of any sparsity, which would not be practicable if the average amount of work on each iteration were $\mathcal{O}(n^3)$, this amount of computation being typical if an accurate approximation to the ideal vector is required.

When first derivatives of $F$ are calculated, the choice $\underline{g}_k = \underline{\nabla} F(\underline{x}_k)$ is usual for the model function (1.2). Furthermore, after choosing the second derivative matrix $H_1$ for the first iteration, the $k$-th iteration may construct $H_{k+1}$ from $H_k$ by the symmetric Broyden formula [see equation (3.6.5) of [3], for instance]. There is also a version of that formula for derivative-free optimization ([9]). It generates both $\underline{g}_{k+1}$ and $H_{k+1}$ from $\underline{g}_k$ and $H_k$ by minimizing $\| H_{k+1} - H_k \|_F$ subject to some interpolation conditions, where the subscript "$F$" denotes the Frobenius matrix norm. The techniques that provide $\underline{g}_k$, $H_k$ and $\Delta_k$ are separate from our work, however, except for one feature. It is that, instead of requiring $H_k$ to be available explicitly, we assume that, for any $\underline{v} \in \mathcal{R}^n$, the form of $H_k$ allows the matrix vector product $H_k \underline{v}$ to be calculated in $\mathcal{O}(n^2)$ operations. Thus our work is relevant to the LINCOA Fortran software, where the expression for $H_k$ includes a linear combination of about $2n+1$ outer products of the form $\underline{y}\,\underline{y}^T$, $\underline{y} \in \mathcal{R}^n$.

Table 4 of [11] gives some numerical results, on the efficiency of the symmetric Broyden formula in derivative-free unconstrained optimization when $F$ is a strictly convex quadratic function. In all of these tests, the optimal vector of variables is calculated to high accuracy, using only about $\mathcal{O}(n)$ values of $F$ for large $n$, which shows that the updating formula is successful at capturing enough second derivative information to provide fast convergence. There is no need for $\| H_k - \nabla^2 F(\underline{x}_k) \|$ to become small, as explained by [1] when $\underline{\nabla} F$ is available. In the tests of [11] with a quadratic $F$, every initial matrix $H_1$ satisfies $\| H_1 - \nabla^2 F \|_F > \frac{1}{2} \| \nabla^2 F \|_F$. Further, although the sequence $\| H_k - \nabla^2 F \|_F$, $k = 1, 2, \ldots, K$, decreases monotonically, where $K$ is the final value of $k$, the property

$$\left\| H_K - \nabla^2 F \right\|_F > 0.9 \left\| H_1 - \nabla^2 F \right\|_F \tag{1.4}$$

is not unusual when $n$ is large.

Therefore the setting for our choice of $\underline{x}_k^+$ is that we seek a vector $\underline{x}$ that provides a relatively small value of $Q_k(\underline{x})$ subject to the constraints (1.1) and (1.3), although we expect the accuracy of the approximation $Q_k(\underline{x}) \approx F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, to be poor. After choosing $\underline{x}_k^+$, the value $F(\underline{x}_k^+)$ is calculated usually, and then, in derivative-free optimization, the next quadratic model satisfies the equation

$$Q_{k+1}\left(\underline{x}_k^+\right) = F\left(\underline{x}_k^+\right), \tag{1.5}$$

even if $\underline{x}_k^+$ is not the best vector of variables so far. Thus, if $|Q_k(\underline{x}_k^+) - F(\underline{x}_k^+)|$ is large, then $Q_{k+1}$ is substantially different from $Q_k$. Fortunately, the symmetric Broyden formula has the property that, if $F$ is quadratic, then $\| H_{k+1} - H_k \|$ tends to zero as $k$

becomes large, so it is usual on the later iterations for the error $|Q_k(\underline{x}_k^+) - F(\underline{x}_k^+)|$ to be much less than a typical error $|Q_k(\underline{x}) - F(\underline{x})|$, $\|\underline{x} - \underline{x}_k\| \le \Delta_k$. Thus the reduction $F(\underline{x}_k^+) < F(\underline{x}_k)$ is inherited from $Q_k(\underline{x}_k^+) < Q_k(\underline{x}_k)$ much more often than would be predicted by theoretical analysis, if the theory employed a bound on $|Q_k(\underline{x}_k^+) - F(\underline{x}_k^+)|$ that is derived only from the errors $\|\underline{g}_k - \nabla F(\underline{x}_k)\|$ and $\|H_k - \nabla^2 F(\underline{x}_k)\|$, assuming that $F$ is twice differentiable.

Suitable ways of choosing $\underline{x}_k^+$ in the unconstrained case ($m = 0$) are described by [2] and by [10], for instance. They are addressed in Sect. 2, where both truncated conjugate gradient and Krylov subspace methods receive attention. They are pertinent to our techniques for linear constraints ($m > 0$), because, as explained in Sect. 3, active set methods are employed. Those methods generate sequences of unconstrained problems, some of the constraints (1.1) being satisfied as equations, which can reduce the number of variables, while the other constraints (1.1) are ignored temporarily. The idea of combining active set methods with Krylov subspaces is considered in Sect. 4. It was attractive to the author during the early development of the LINCOA software, but two severe disadvantages of this approach are exposed in Sect. 4. The present version of LINCOA combines active set methods with truncated conjugate gradients, which is the subject of Sect. 5. The conjugate gradient calculations are complete if they generate a vector $\underline{x}$ on the boundary of the trust region constraint (1.3), but moves round this boundary that preserve feasibility may provide a useful reduction in the value of $Q_k(\underline{x})$. This possibility is studied in Sect. 6. Some further remarks, including a technique for preserving feasibility when applying the Krylov method, are given in Sect. 7.

## 2 The unconstrained case

There are no linear constraints on the variables throughout this section, $m$ being zero in expression (1.1). We consider truncated conjugate gradient and Krylov subspace methods for constructing a point $\underline{x}_k^+$ at which the quadratic model $Q_k(\underline{x})$, $\|\underline{x} - \underline{x}_k\| \le \Delta_k$, is relatively small. These methods are iterative, a sequence of points $\underline{p}_\ell$, $\ell = 1, 2, \ldots, L$, being generated, with $\underline{p}_1 = \underline{x}_k$, with $Q_k(\underline{p}_{\ell+1}) < Q_k(\underline{p}_\ell)$, $\ell = 1, 2, \ldots, L-1$, and with $\underline{x}_k^+ = \underline{p}_L$. The main difference between them is that the conjugate gradient iterations are terminated if the $\ell$-th iteration generates a point $\underline{p}_{\ell+1}$ that satisfies $\|\underline{p}_{\ell+1} - \underline{x}_k\| = \Delta_k$, but both $\underline{p}_\ell$ and $\underline{p}_{\ell+1}$ may be on the trust region boundary in a Krylov subspace iteration. The second derivative matrix $H_k$ of the quadratic model (1.2) is allowed to be indefinite. We keep in mind that we would like the total amount of computation for each $k$ to be $\mathcal{O}(n^2)$.

If the gradient $\underline{g}_k$ of the model (1.2) were zero, then, instead of investigating whether $H_k$ has any negative eigenvalues, we would abandon the search for a vector $\underline{x}_k^+$ that provides $Q_k(\underline{x}_k^+) < Q_k(\underline{x}_k)$. In this case LINCOA would try to improve the model by changing one of its interpolation conditions. We assume throughout this section, however, that $\underline{g}_k$ is nonzero.

The first iteration ($\ell = 1$) of both the conjugate gradient and Krylov subspace methods picks the vector

$$\underline{p}_2 = \underline{p}_1 - \alpha_1 \underline{g}_k = \underline{x}_k - \alpha_1 \underline{g}_k, \tag{2.1}$$

where $\alpha_1$ is the value of $\alpha$ that minimizes the quadratic function of one variable $Q_k(\underline{p}_1 - \alpha\, \underline{g}_k), \alpha \in \mathcal{R}$, subject to $\|\underline{p}_1 - \alpha\, \underline{g}_k - \underline{x}_k\| \le \Delta_k$. Further, whenever a conjugate gradient iteration generates $\underline{p}_{\ell+1}$ from $\underline{p}_\ell$, a line search from $\underline{p}_\ell$ along the direction

$$\underline{d}_\ell = \begin{cases} -\underline{g}_k, & \ell = 1, \\ -\underline{\nabla} Q_k(\underline{p}_\ell) + \beta_\ell\, \underline{d}_{\ell-1}, & \ell \ge 2, \end{cases} \tag{2.2}$$

is made, the value of $\beta_\ell$ being defined by the conjugacy condition $\underline{d}_\ell^T H_k\, \underline{d}_{\ell-1} = 0$. Thus $\underline{p}_{\ell+1}$ is the vector

$$\underline{p}_{\ell+1} = \underline{p}_\ell + \alpha_\ell\, \underline{d}_\ell, \tag{2.3}$$

where $\alpha_\ell$ is the value of $\alpha$ that minimizes $Q_k(\underline{p}_\ell + \alpha\, \underline{d}_\ell), \alpha \in \mathcal{R}$, subject to the bound $\|\underline{p}_\ell + \alpha\, \underline{d}_\ell - \underline{x}_k\| \le \Delta_k$. This procedure is well-defined, provided it is terminated if $\|\underline{p}_{\ell+1} - \underline{x}_k\| = \Delta_k$ or $\underline{\nabla} Q_k(\underline{p}_{\ell+1}) = 0$ occurs. In theory, these conditions provide termination after at most $n$ iterations (see [3], for instance).

The only task of a conjugate gradient iteration that requires $\mathcal{O}(n^2)$ operations is the calculation of the product $H_k\, \underline{d}_\ell$. All of the other tasks of the $\ell$-th iteration can be done in only $\mathcal{O}(n)$ operations, by taking advantage of the availability of $H_k\, \underline{d}_{\ell-1}$ when $\underline{\nabla} Q_k(\underline{p}_\ell)$, $\beta_\ell$ and $\underline{d}_\ell$ are formed. Therefore the target of $\mathcal{O}(n^2)$ work for each $k$ is maintained if $L$, the final value of $\ell$, is bounded above by a number that is independent of both $k$ and $n$. Further, because the total work of the optimization algorithm depends on the average value of $L$ over $k$, a few large values of $L$ may be tolerable.

The two termination conditions that have been mentioned are not suitable for keeping $L$ small. Indeed, if $H_k$ is positive definite, then, for every $\ell \ge 1$, the property $Q_k(\underline{p}_{\ell+1}) < Q_k(\underline{x}_k)$ implies $\|\underline{p}_{\ell+1} - \underline{x}_k\| < \Delta_k$ for sufficiently large $\Delta_k$. Furthermore, if $\underline{\nabla} Q_k(\underline{p}_{\ell+1}) = 0$ is achieved in exact arithmetic, then $\ell$ may have to be close to $n$, but it is usual for computer rounding errors to prevent $\underline{\nabla} Q_k(\underline{p}_{\ell+1})$ from becoming zero. Instead, the termination conditions of LINCOA, which are described below, are recommended for use in practice. They require a positive constant $\eta_1 < 1$ to be prescribed, the LINCOA value being $\eta_1 = 0.01$.

Termination occurs immediately with $\underline{x}_k^+ = \underline{p}_\ell$ if the calculated $\underline{d}_\ell$ is not a descent direction, which is the condition

$$\underline{d}_\ell^T\, \underline{\nabla} Q_k(\underline{p}_\ell) \;\ge\; 0. \tag{2.4}$$

Otherwise, the positive steplength $\hat{\alpha}_\ell$, say, that puts $\underline{p}_\ell + \hat{\alpha}_\ell\, \underline{d}_\ell$ on the trust region boundary is found, and termination occurs also with $\underline{x}_k^+ = \underline{p}_\ell$ if this move is expected to give a relatively small reduction in $Q_k(\cdot)$, which is the test

$$\hat{\alpha}_\ell\, \left| \underline{d}_\ell^T\, \underline{\nabla} Q_k(\underline{p}_\ell) \right| \;\le\; \eta_1\, \left\{ Q_k(\underline{x}_k) - Q_k(\underline{p}_\ell) \right\}. \tag{2.5}$$

In the usual situation, when both inequalities (2.4) and (2.5) fail, the product $H_k\, \underline{d}_\ell$ is generated for the construction of the new point (2.3). The calculation ends with $\underline{x}_k^+ = \underline{p}_{\ell+1}$ if $\underline{p}_{\ell+1}$ is on the trust region boundary, or if the reduction in $Q_k$ by the $\ell$-th iteration has the property

$$Q_k(\underline{p}_\ell) - Q_k(\underline{p}_{\ell+1}) \;\le\; \eta_1 \left\{ Q_k(\underline{x}_k) - Q_k(\underline{p}_{\ell+1}) \right\}. \tag{2.6}$$

It also ends in the case $\ell = n$, which is unlikely unless $n$ is small. In all other cases, $\ell$ is increased by one for the next conjugate gradient iteration.

Condition (2.4) is equivalent to $\underline{\nabla} Q_k(\underline{p}_\ell) = 0$ in exact arithmetic, because, if $\underline{\nabla} Q_k(\underline{p}_\ell)$ is nonzero, then the given choices of $\underline{p}_\ell$ and $\underline{d}_\ell$ give the property

$$\underline{d}_\ell^T \underline{\nabla} Q_k(\underline{p}_\ell) \;=\; -\left\| \underline{\nabla} Q_k(\underline{p}_\ell) \right\|^2 \;<\; 0, \tag{2.7}$$

the vector norm being Euclidean. The requirement $\underline{d}_\ell^T \underline{\nabla} Q_k(\underline{p}_\ell) < 0$ before $\alpha_\ell$ is calculated provides $\alpha_\ell > 0$. For $\ell \ge 2$, the test (2.5) causes termination when $\|\underline{\nabla} Q_k(\underline{p}_\ell)\|$ is small. Specifically, as $\|\underline{p}_\ell - \underline{x}_k\| < \Delta_k$ and $\|\underline{p}_\ell + \hat{\alpha}_\ell \underline{d}_\ell - \underline{x}_k\| = \Delta_k$ imply $\|\hat{\alpha}_\ell \underline{d}_\ell\| < 2\Delta_k$, we deduce from the Cauchy–Schwarz inequality that condition (2.5) holds in the case

$$\left\| \underline{\nabla} Q_k(\underline{p}_\ell) \right\| \;\le\; \eta_1 \left\{ Q_k(\underline{x}_k) - Q_k(\underline{p}_\ell) \right\} / (2\Delta_k). \tag{2.8}$$

The test (2.6) gives $L = \ell + 1$ and $\underline{x}_k^+ = \underline{p}_{\ell+1}$ when $Q_k(\underline{p}_\ell) - Q_k(\underline{p}_{\ell+1})$ is relatively small. We find in Sect. 5 that the given conditions for termination are also useful for search directions $\underline{d}_\ell$ that satisfy linear constraints on the variables.

In the remainder of this section, we consider a Krylov subspace method for calculating $\underline{x}_k^+$ from $\underline{x}_k$, where $k$ is any positive integer such that the gradient $\underline{g}_k$ of the model (1.2) is nonzero. For $\ell \ge 1$, the Krylov subspace $\mathcal{K}_\ell$ is defined to be the span of the vectors $H_k^{j-1} \underline{g}_k \in \mathcal{R}^n$, $j = 1, 2, \ldots, \ell$, the matrix $H_k^{j-1}$ being the $(j-1)$-th power of $H_k = \nabla^2 Q_k$, and $H_k^0$ being the unit matrix even if $H_k$ is zero. Let $\ell^*$ be the greatest integer $\ell$ such that $\mathcal{K}_\ell$ has dimension $\ell$, which implies $\mathcal{K}_\ell = \mathcal{K}_{\ell^*}$, $\ell \ge \ell^*$. We retain $\underline{p}_1 = \underline{x}_k$. The $\ell$-th iteration of our method constructs $\underline{p}_{\ell+1}$, and the number of iterations is at most $\ell^*$. Our choice of $\underline{p}_{\ell+1}$ is a highly accurate estimate of the vector $\underline{x}$ that minimizes $Q_k(\underline{x})$, subject to $\|\underline{x} - \underline{x}_k\| \le \Delta_k$ and subject to the condition that $\underline{x} - \underline{x}_k$ is in $\mathcal{K}_\ell$. We find later in this section that the calculation of $\underline{p}_{\ell+1}$ for each $\ell$ requires only $\mathcal{O}(n^2)$ operations. An introduction to Krylov subspace methods is given by [2], for instance.

It is well known that each $\underline{p}_{\ell+1}$ of the conjugate gradient method has the property that $\underline{p}_{\ell+1} - \underline{x}_k$ is in the Krylov subspace $\mathcal{K}_\ell$, which can be deduced by induction from Eqs. (2.2) and (2.3). It is also well known that the search directions of the conjugate gradient method satisfy $\underline{d}_\ell^T H_k \underline{d}_j = 0$, $1 \le j < \ell$. It follows that the points $\underline{p}_\ell$, $\ell = 1, 2, 3, \ldots$, of the conjugate gradient and Krylov subspace methods are the same while they are strictly inside the trust region $\{\underline{x} : \|\underline{x} - \underline{x}_k\| \le \Delta_k\}$. We recall that the conjugate gradient method is terminated when its $\underline{p}_{\ell+1}$ is on the trust region boundary, but the iterations of the Krylov subspace method may continue, in order to provide a smaller value of $Q_k(\underline{x}_k^+)$.

Our Krylov subspace method has a standard form with termination conditions for use in practice. For $\ell = 1, 2, \ldots, \ell^*$, we let $\underline{v}_\ell$ be a vector of unit length in $\mathcal{K}_\ell$ that, for $\ell \ge 2$, satisfies the orthogonality conditions

$$\underline{v}_\ell^T \underline{d} \;=\; 0, \quad \underline{d} \in \mathcal{K}_{\ell-1}, \tag{2.9}$$

which defines every $\underline{v}_\ell$ except its sign. Thus $\{\underline{v}_j : j = 1, 2, \ldots, \ell\}$ is an orthonormal basis of $\mathcal{K}_\ell$. Further, because $H_k \underline{v}_j$ is in the subspace $\mathcal{K}_{j+1}$, the conditions (2.9) provide the highly useful property

$$\underline{v}_\ell^T H_k \, \underline{v}_j \;=\; 0, \quad 1 \leq j \leq \ell - 2. \tag{2.10}$$

The usual way of constructing $\underline{v}_\ell$, $\ell \geq 2$, is described below. It is extended in Sect. 4 to calculations with linear constraints on the variables.

After setting $\underline{p}_1 = \underline{x}_k$, the vector $\underline{p}_2$ of our Krylov subspace method is given by Eq. (2.1) as mentioned already. For $\ell \geq 2$, we write $\underline{p}_{\ell+1}$ as the sum

$$\underline{p}_{\ell+1} \;=\; \underline{x}_k + \sum_{i=1}^{\ell} \theta_i \, \underline{v}_i . \tag{2.11}$$

Therefore, assuming that every vector norm is Euclidean, we seek the vector of coefficients $\underline{\theta} \in \mathcal{R}^\ell$ that minimizes the quadratic function

$$\Phi_{k\ell}(\underline{\theta}) \;=\; Q_k\left( \underline{x}_k + \sum_{i=1}^{\ell} \theta_i \, \underline{v}_i \right), \quad \underline{\theta} \in \mathcal{R}^\ell, \tag{2.12}$$

subject to $\|\underline{\theta}\| \leq \Delta_k$. This function has the second derivatives

$$d^{\,2}\Phi_{k\ell}(\underline{\theta})/d\theta_i \, d\theta_j \;=\; \underline{v}_i^T H_k \, \underline{v}_j, \quad 1 \leq i, j \leq \ell, \tag{2.13}$$

and Eq. (2.10) shows that the matrix $\nabla^2 \Phi_{k\ell}(\cdot)$ is tridiagonal. Moreover, the gradient $\underline{\nabla} \Phi_{k\ell}(0)$ has the components $\underline{v}_i^T \underline{g}_k$, $i = 1, 2, \ldots, \ell$, so only the first component is nonzero, its value being $\pm \|\underline{g}_k\|$. Hence, after forming the tridiagonal part of $\nabla^2 \Phi_{k\ell}(\cdot)$, the required $\underline{\theta}$ can be calculated accurately, conveniently and rapidly by the algorithm of [6].

At the beginning of the $\ell$-th iteration of our Krylov subspace method, where $\ell \geq 2$, the vectors $\underline{v}_i$, $i = 1, 2, \ldots, \ell - 1$, are available, with the tridiagonal part of the second derivative matrix $\nabla^2 \Phi_{k\ell-1}(\cdot)$, its last diagonal element being $\underline{v}_{\ell-1}^T H_k \underline{v}_{\ell-1}$, so the product $H_k \underline{v}_{\ell-1}$ has been calculated already. A condition for termination at this stage is suggested in the next paragraph. If it fails, then the Arnoldi formula

$$\underline{v}_\ell \;=\; \frac{H_k \, \underline{v}_{\ell-1} - \sum_{j=1}^{\ell-1} \left( \underline{v}_j^T H_k \, \underline{v}_{\ell-1} \right) \underline{v}_j}{\left\| H_k \, \underline{v}_{\ell-1} - \sum_{j=1}^{\ell-1} \left( \underline{v}_j^T H_k \, \underline{v}_{\ell-1} \right) \underline{v}_j \right\|} \tag{2.14}$$

is applied, the amount of work being only $\mathcal{O}(n)$, because Eq. (2.10) provides $\underline{v}_j^T H_k \underline{v}_{\ell-1} = 0$, $j \leq \ell - 3$. The calculation of $H_k \underline{v}_\ell$ requires $\mathcal{O}(n^2)$ operations, however, which is needed for the second derivatives $\underline{v}_{\ell-1}^T H_k \underline{v}_\ell$ and $\underline{v}_\ell^T H_k \underline{v}_\ell$. Then $\underline{p}_{\ell+1}$ is generated by the method of the previous paragraph, followed by termination with $\underline{x}_k^+ = \underline{p}_{\ell+1}$ if inequality (2.6) is satisfied. Otherwise, $\ell$ is increased by one for the next iteration.

The left hand side of the test (2.5) is the reduction that occurs in the linear function

$$\Lambda_{k\ell}(\underline{x}) \;=\; Q_k(\underline{p}_\ell) + (\underline{x} - \underline{p}_\ell)^T \underline{\nabla} Q_k(\underline{p}_\ell), \quad \underline{x} \in \mathcal{R}^n, \tag{2.15}$$

if a step is taken from $\underline{p}_\ell$ along the direction $\underline{d}_\ell$ to the trust region boundary. For Krylov subspace methods, we replace that left hand side by the greatest reduction that can be achieved in $\Lambda_{k\ell}(\cdot)$ by any step from $\underline{p}_\ell$ to the trust region boundary. We see that this step is to the point

$$\widehat{\underline{p}}_{\ell+1} \;=\; \underline{x}_k - \Delta_k \,\underline{\nabla} Q_k(\underline{p}_\ell) / \left\| \underline{\nabla} Q_k(\underline{p}_\ell) \right\|, \tag{2.16}$$

assuming $\underline{\nabla} Q_k(\underline{p}_\ell) \neq 0$, so inequality (2.5) becomes the termination condition

$$\left(\underline{p}_\ell - \widehat{\underline{p}}_{\ell+1}\right)^T \underline{\nabla} Q_k(\underline{p}_\ell) \;\leq\; \eta_1 \left\{ Q_k(\underline{x}_k) - Q_k(\underline{p}_\ell) \right\}, \tag{2.17}$$

which is the test

$$\Delta_k \left\| \underline{\nabla} Q_k(\underline{p}_\ell) \right\| - (\underline{x}_k - \underline{p}_\ell)^T \underline{\nabla} Q_k(\underline{p}_\ell) \;\leq\; \eta_1 \left\{ Q_k(\underline{x}_k) - Q_k(\underline{p}_\ell) \right\}. \tag{2.18}$$

If it is satisfied, then, instead of forming $\underline{v}_\ell$ and $\nabla^2 \Phi_{k\ell}(\cdot)$ for the calculation of $\underline{p}_{\ell+1}$, the choice $\underline{x}_k^+ = \underline{p}_\ell$ is made, which completes the Krylov subspace iterations.

The vectors $\underline{p}_\ell$ and $\underline{\nabla} Q_k(\underline{p}_\ell)$ are required for the termination condition (2.18) when $\ell \geq 2$. The construction of $\underline{p}_\ell$ provides the parameters $\gamma_i$, say, of the sum

$$\underline{p}_\ell \;=\; \underline{x}_k + \sum_{i=1}^{\ell-1} \gamma_i \,\underline{v}_i, \tag{2.19}$$

which corresponds to Eq. (2.11), and we retain the vectors $\underline{v}_i$, $i = 1, 2, \ldots, \ell - 1$. Further, the quadratic function (1.2) has the gradient

$$\underline{\nabla} Q_k(\underline{p}_\ell) \;=\; \underline{g}_k + \sum_{i=1}^{\ell-1} \gamma_i \, H_k \,\underline{v}_i, \tag{2.20}$$

but we prefer not to store $H_k \underline{v}_i$, $i = 1, 2, \ldots, \ell - 2$. Instead, because $H_k \underline{v}_i$ is in the space $\mathcal{K}_{i+1}$, and because any vector in $\mathcal{K}_{\ell-1}$ is unchanged if it is multiplied by $\sum_{j=1}^{\ell-1} \underline{v}_j \,\underline{v}_j^T$, we can write Eq. (2.20) in the form

$$\underline{\nabla} Q_k(\underline{p}_\ell) \;=\; \underline{g}_k + \gamma_{\ell-1} H_k \,\underline{v}_{\ell-1} + \sum_{j=1}^{\ell-1} \left\{ \sum_{i=1}^{\ell-2} \gamma_i \,\underline{v}_j^T H_k \,\underline{v}_i \right\} \underline{v}_j. \tag{2.21}$$

Thus it is straightforward to calculate $\underline{\nabla} Q_k(\underline{p}_\ell)$, and again we take advantage of the property (2.10).

## 3 Active sets

We now turn to linear constraints on the variables, $m$ being positive in expression (1.1). We apply an active set method, this technique being more than 40 years old (see [4], for instance). The active set $\mathcal{A}$, say, is a subset of the indices $\{1, 2, \ldots, m\}$ such that the constraint gradients $\underline{a}_j$, $j \in \mathcal{A}$, are linearly independent. Usually, until $\mathcal{A}$ is updated, the variables $\underline{x} \in \mathcal{R}^n$ are restricted by the equations

$$\underline{a}_j^T \underline{x} \;=\; b_j, \quad j \in \mathcal{A}, \tag{3.1}$$

but, if $j$ is not in $\mathcal{A}$, the constraint $\underline{a}_j^T \underline{x} \le b_j$ is ignored, until updating of the active set is needed to prevent a constraint violation. This updating may occur several times during the search on the $k$-th iteration for a vector $\underline{x}_k^+ = \underline{x}$ that provides a relatively small value of $Q_k(\underline{x})$, subject to the inequality constraints (1.1) and the bound $\|\underline{x} - \underline{x}_k\| \le \Delta_k$. As in Sect. 2, the search generates a sequence of points $\underline{p}_\ell$, $\ell = 1, 2, \ldots, L$, in the trust region with $\underline{p}_1 = \underline{x}_k$, $\underline{x}_k^+ = \underline{p}_L$, and $Q_k(\underline{p}_{\ell+1}) < Q_k(\underline{p}_\ell)$, $\ell = 1, 2, \ldots, L-1$. Also, every $\underline{p}_\ell$ is feasible. Let $\mathcal{A}$ be the current active set when $\underline{p}_{\ell+1}$ is constructed. We replace the Eq. (3.1) by the conditions

$$\underline{a}_j^T \left(\underline{p}_{\ell+1} - \underline{p}_\ell\right) \;=\; 0, \quad j \in \mathcal{A}, \tag{3.2}$$

throughout this section.

This replacement brings a strong advantage if one (or more) of the residuals $b_j - \underline{a}_j^T \underline{x}$, $j = 1, 2, \ldots, m$, is very small and positive, because it allows the indices of those constraints to be in $\mathcal{A}$. The Eq. (3.1), however, require the choice of $\mathcal{A}$ at $\underline{x} = \underline{x}_k$ to be a subset of $\{j : b_j - \underline{a}_j^T \underline{x}_k = 0\}$. Let $b_i - \underline{a}_i^T \underline{x}_k$ be tiny and positive, where $i$ is one of the constraint indices. If $i$ is not in the set $\mathcal{A}$, then the condition $b_i - \underline{a}_i^T \underline{p}_2 \ge 0$ is ignored in the first attempt to construct $\underline{p}_2$, so it is likely that the $\underline{p}_2$ of this attempt violates the $i$-th constraint. Then $\underline{p}_2$ would be shifted somehow to a feasible position in $\mathcal{R}^n$, and usually the new $\underline{p}_2$ would satisfy $\underline{a}_i^T \underline{p}_2 = b_i$, with $i$ being included in a new active set for the construction of $\underline{p}_3$. Thus the active set of the $k$-th iteration may be updated after a tiny step from $\underline{p}_1 = \underline{x}_k$ to $\underline{p}_2$, which tends to be expensive if there are many tiny steps, the work of a typical updating being $\mathcal{O}(n^2)$. Therefore the conditions (3.2) are employed instead of the Eq. (3.1) in the LINCOA software, the actual choice of $\mathcal{A}$ at $\underline{x}_k$ being as follows.

For every feasible $\underline{x} \in \mathcal{R}^n$, and for the current $\Delta_k$, we let $\mathcal{J}(\underline{x})$ be the set

$$\mathcal{J}(\underline{x}) \;=\; \left\{ j : b_j - \underline{a}_j^T \underline{x} \le \eta_2 \, \Delta_k \, \|\underline{a}_j\| \right\} \;\subset\; \{1, 2, \ldots, m\}, \tag{3.3}$$

where $\eta_2$ is a positive constant, the value $\eta_2 = 0.2$ being set in the LINCOA software. In other words, the constraint index $j$ is in $\mathcal{J}(\underline{x})$ if and only if the distance from $\underline{x}$ to the boundary of the $j$-th constraint is at most $\eta_2 \, \Delta_k$. The choice of $\mathcal{A}$ at $\underline{x}_k$ is a subset of $\mathcal{J}(\underline{x}_k)$. As in Sect. 2, the step from $\underline{p}_1 = \underline{x}_k$ to $\underline{p}_2$ is along a search direction $\underline{d}_1$, which is defined below. This direction has the property $\underline{a}_j^T \underline{d}_1 \le 0$, $j \in \mathcal{J}(\underline{x}_k)$, in order that every positive step along $\underline{d}_1$ goes no closer to the boundary of the $j$-th constraint for every $j \in \mathcal{J}(\underline{x}_k)$. It follows that, if the length of the step from $\underline{p}_1$ to $\underline{p}_2$ is governed by the need for $\underline{p}_2$ to be feasible, then the length $\|\underline{p}_2 - \underline{p}_1\|$ is greater than $\eta_2\,\Delta_k$, which prevents the tiny step that is addressed in the previous paragraph. This device is taken from the TOLMIN algorithm of [7].

The direction $\underline{d}_1$ is the unique vector $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to the conditions

$$\underline{a}_j^T \underline{d} \;\le\; 0, \quad j \in \mathcal{J}(\underline{x}_k), \tag{3.4}$$

and the choice of $\mathcal{A}$ at $\underline{x}_k$ is derived from properties of $\underline{d}_1$. Let $\mathcal{I}(\underline{x}_k)$ be the set

$$\mathcal{I}(\underline{x}_k) \;=\; \left\{ j \in \mathcal{J}(\underline{x}_k) : \underline{a}_j^T \underline{d}_1 = 0 \right\}. \tag{3.5}$$

If the index $j$ is in $\mathcal{J}(\underline{x}_k)$ but not in $\mathcal{I}(\underline{x}_k)$, then a sufficiently small perturbation to the $j$-th constraint does not alter $\underline{d}_1$. It follows that $\underline{d}_1$ is also the vector $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to $\underline{a}_j^T \underline{d} \le 0$, $j \in \mathcal{I}(\underline{x}_k)$. Furthermore, the definition of $\mathcal{I}(\underline{x}_k)$ implies that $\underline{d}_1$ is the $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to the equations

$$\underline{a}_j^T \underline{d} \;=\; 0, \quad j \in \mathcal{I}(\underline{x}_k). \tag{3.6}$$

We pick $\mathcal{A} = \mathcal{I}(\underline{x}_k)$ if the vectors $\underline{a}_j$, $j \in \mathcal{I}(\underline{x}_k)$, are linearly independent. Otherwise, $\mathcal{A}$ is any subset of $\mathcal{I}(\underline{x}_k)$ such that $\underline{a}_j$, $j \in \mathcal{A}$, is a basis of the linear subspace spanned by $\underline{a}_j$, $j \in \mathcal{I}(\underline{x}_k)$. The actual choice of basis does not matter, because the conditions (3.2) are always equivalent to $\underline{a}_j^T(\underline{p}_{\ell+1} - \underline{p}_\ell) = 0$, $j \in \mathcal{I}(\underline{x}_k)$.

In the LINCOA software, $\underline{d}_1$ is calculated by the Goldfarb-Idnani algorithm [5] for quadratic programming. A subset $\mathcal{A}$ of $\mathcal{J}(\underline{x}_k)$ is updated until it becomes the required active set. Let $\underline{d}(\mathcal{A})$ be the vector $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to $\underline{a}_j^T \underline{d} = 0$, $j \in \mathcal{A}$. The vectors $\underline{a}_j$, $j \in \mathcal{A}$, are linearly independent for every $\mathcal{A}$ that occurs. Also, by employing the signs of some Lagrange multipliers, every $\mathcal{A}$ is given the property that $\underline{d}(\mathcal{A})$ is the vector $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to $\underline{a}_j^T \underline{d} \le 0$, $j \in \mathcal{A}$. It follows that the Goldfarb–Idnani calculation is complete when $\underline{d} = \underline{d}(\mathcal{A})$ satisfies all the inequalities (3.4). Otherwise, a strict increase in $\|\underline{g}_k + \underline{d}(\mathcal{A})\|_2$ is obtained by picking an index $j \in \mathcal{J}(\underline{x}_k)$ with $\underline{a}_j^T \underline{d}(\mathcal{A}) > 0$ and adding it to $\mathcal{A}$, combined with deletions from $\mathcal{A}$ if necessary to achieve the stated conditions on $\mathcal{A}$. For $k \ge 2$, the initial $\mathcal{A}$ of this procedure is derived from the previous active set, which is called a "warm start". Usually, when the final $\mathcal{A}$ is different from the initial $\mathcal{A}$, the amount of work of this part of LINCOA is within our target, namely $\mathcal{O}(n^2)$ operations for each $k$.

Let $A$ be the $n \times |\mathcal{A}|$ matrix that has the columns $\underline{a}_j$, $j \in \mathcal{A}$, where $\mathcal{A}$ is any of the sets that occur in the previous paragraph. When $\mathcal{A}$ is updated in the Goldfarb–Idnani algorithm, the QR factorization of $A$ is updated too. We let $\widehat{Q}R$ be this factorization, where $\widehat{Q}$ is $n \times |\mathcal{A}|$ with orthonormal columns and where $R$ is square, upper triangular and nonsingular. Furthermore, an $n \times (n - |\mathcal{A}|)$ matrix $\check{Q}$ is calculated and updated such that the $n \times n$ matrix $(\widehat{Q} \mid \check{Q})$ is orthogonal. We employ the matrices $\widehat{Q}$, $R$ and $\check{Q}$ that are available when the choice of the active set at $\underline{x}_k$ is complete.

Indeed, the direction $\underline{d}$ that minimizes $\|\underline{g}_k + \underline{d}\|_2$ subject to the Eq. (3.6) is given by the formula

$$\underline{d}_1 \;=\; -\check{Q}\,\check{Q}^T \underline{g}_k. \tag{3.7}$$

Further, the step $\underline{p}_{\ell+1} - \underline{p}_\ell$ satisfies the conditions (3.2) if and only if $\underline{p}_{\ell+1} - \underline{p}_\ell$ is in the column space of $\check{Q}$. Thus, until $\mathcal{A}$ is changed from its choice at $\underline{x}_k$ to avoid a constraint violation, our search for a small value of $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, subject to the active constraints and the bound $\|\underline{x} - \underline{x}_k\| \le \Delta_k$, is equivalent to seeking a small value of the quadratic function $Q_k(\underline{x}_k + \check{Q}\underline{\sigma})$, $\underline{\sigma} \in \mathcal{R}^{n-|\mathcal{A}|}$, subject to $\|\underline{\sigma}\| \le \Delta_k$. The calculation is now without linear constraints, so we can employ some of the techniques of Sect. 2. We address the construction of $\check{Q}\underline{\sigma}$ by Krylov subspace and truncated conjugate gradient methods in Sects. 4 and 5, respectively. Much cancellation is usual in the product $\check{Q}^T \underline{g}_k$ for large $k$, due to the first order conditions at the solution of a smooth optimization problem, and this cancellation occurs in the second term of the product $\underline{d}_1 = -\check{Q}(\check{Q}^T \underline{g}_k)$. Fortunately, because the definition of $\check{Q}$ provides $\underline{a}_j^T \check{Q} = 0$, $j \in \mathcal{A}$, the vector (3.7) satisfies the constraints $\underline{a}_j^T \underline{d}_1 = 0$, $j \in \mathcal{A}$, for every product $\check{Q}^T \underline{g}_k$.

Here is a simple example in two dimensions, illustrated in Fig. 1, that makes an important point. Letting $x_1$ and $x_2$ be the components of $\underline{x} \in \mathcal{R}^2$, and letting $\underline{x}_k$ be at the origin, we seek a relatively small value of the linear function

$$Q_k(\underline{x}) \;=\; -2\,x_1 - x_2, \quad \underline{x} \in \mathcal{R}^2, \tag{3.8}$$

subject to the constraints

$$\underline{a}_1^T \underline{x} = x_2 \le 0, \quad \underline{a}_2^T \underline{x} = x_1 + x_2 \le 2 \quad \text{and} \quad \|\underline{x} - \underline{x}_k\|_2 \le \sqrt{10}. \tag{3.9}$$

The feasible region of Fig. 1 contains the points that satisfy the constraints (3.9), and the steepest descent direction $-\underline{\nabla} Q_k(\cdot)$ is shown too. We pick the LINCOA value $\eta_2 = 0.2$ for expression (3.3), which gives $\mathcal{J}(\underline{x}_k) = \{1\}$ with $m = 2$. A positive step from $\underline{p}_1 = \underline{x}_k$ along $-\underline{\nabla} Q_k(\cdot)$ would violate $x_2 \le 0$, so the first active set $\mathcal{A}$ is

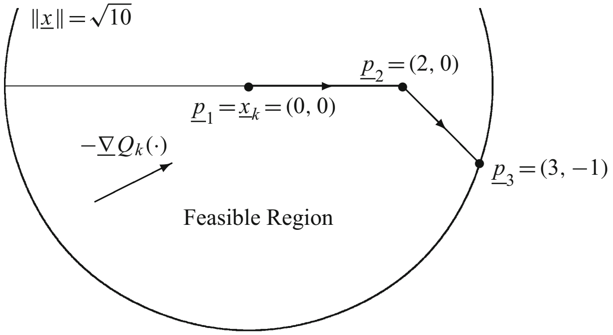
$\|\underline{x}\| = \sqrt{10}$
$\underline{p}_2 = (2, 0)$
$\underline{p}_1 = \underline{x}_k = (0, 0)$
$-\underline{\nabla} Q_k(\cdot)$
$\underline{p}_3 = (3, -1)$
Feasible Region

**Fig. 1** A change to the active set in two dimensions

also {1}. Thus $\underline{p}_2$ is at (2, 0) as shown in Fig. 1, the length of the step to $\underline{p}_2$ being restricted by the constraints because $Q_k(\cdot)$ is linear. Further progress can be made from $\underline{p}_2$ only by deleting the index 1 from $\mathcal{A}$, but an empty set is still excluded by the direction of $-\underline{\nabla} Q_k(\cdot)$. Therefore $\mathcal{A}$ is updated from {1} to {2}, which causes $\underline{p}_3$ to be the feasible point on the trust region boundary that satisfies $\underline{a}_2^T(\underline{p}_3 - \underline{p}_2) = 0$. We find that $\underline{x} = \underline{p}_3 = (3, -1)$ is the $\underline{x}$ that minimizes the function (3.8) subject to the constraints (3.9), and that the algorithm supplies the strictly decreasing sequence $Q_k(\underline{p}_1) = 0$, $Q_k(\underline{p}_2) = -4$ and $Q_k(\underline{p}_3) = -5$.

The important point of this example is that, when $\mathcal{A}$ is updated at $\underline{p}_2$, it is necessary not only to add a constraint index to $\mathcal{A}$ but also to make a deletion from $\mathcal{A}$. Furthermore, in all cases when $\mathcal{A}$ is updated at $\underline{x} = \underline{p}_\ell$, say, we want the length of the step from $\underline{p}_\ell$ to $\underline{p}_{\ell+1}$ to be at least $\eta_2\,\Delta_k$ if it is restricted by the linear constraints. Therefore, if a new $\mathcal{A}$ is required at the feasible point $\underline{p}_\ell$, it is generated by the procedure that is described earlier in this section, after replacing $\underline{g}_k$ by $\underline{\nabla} Q_k(\underline{p}_\ell)$ and $\underline{x}_k$ by $\underline{p}_\ell$. We are reluctant to update the active set at $\underline{p}_\ell$, however, when $\underline{p}_\ell$ is close to the boundary of the trust region. Indeed, if a new active set at $\underline{p}_\ell$ is under consideration in the LINCOA software, then the change to $\mathcal{A}$ is made if and only if the distance from $\underline{p}_\ell$ to the trust region boundary is at least $\eta_2\Delta_k$, which is the condition $\|\underline{p}_\ell - \underline{x}_k\| \leq (1-\eta_2)\,\Delta_k$. Otherwise, the calculation of $\underline{x}_k^+$ is terminated with $L = \ell$ and $\underline{x}_k^+ = \underline{p}_\ell$.

## 4 Krylov subspace methods

Let the active set $\mathcal{A}$ be chosen at the trust region centre $\underline{p}_1 = \underline{x}_k$ as in Sect. 3. We recall from the paragraph that includes Eq. (3.7) that the minimization of $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, subject to the active and trust region constraints

$$\underline{a}_j^T(\underline{x} - \underline{x}_k) \;=\; 0, \quad j \in \mathcal{A}, \quad \text{and} \quad \|\underline{x} - \underline{x}_k\| \;\leq\; \Delta_k \tag{4.1}$$

is equivalent to the minimization of the quadratic function

$$Q_k^{\text{red}}(\underline{\sigma}) \;=\; Q_k\left(\underline{x}_k + \check{Q}\,\underline{\sigma}\right), \quad \underline{\sigma} \in \mathcal{R}^{n-|\mathcal{A}|}, \tag{4.2}$$

subject only to the trust region bound $\|\underline{\sigma}\| \leq \Delta_k$, where $\check{Q}$ is an $n \times (n - |\mathcal{A}|)$ matrix with orthonormal columns that are orthogonal to $\underline{a}_j$, $j \in \mathcal{A}$. Because there is a trust region bound but no linear constraints on $\underline{\sigma}$, either the Krylov subspace method or the conjugate direction method can be taken from Sect. 2 to construct a relatively small value of $Q_k^{\text{red}}(\underline{\sigma})$, $\|\underline{\sigma}\| \leq \Delta_k$. The Krylov subspace alternative receives attention in this section. First we express it in terms of the original variables $\underline{x} \in \mathcal{R}^n$.

Recalling that the Krylov subspace $\mathcal{K}_\ell$ in Sect. 2 is the span of the vectors $H_k^{j-1}\underline{g}_k \in \mathcal{R}^n$, $j = 1, 2, \ldots, \ell$, we now let $\mathcal{K}_\ell$ be the subspace of $\mathcal{R}^{n-|\mathcal{A}|}$ that is spanned by $(\nabla^2 Q_k^{\text{red}})^{j-1}\,\underline{\nabla} Q_k^{\text{red}}(0)$, $j = 1, 2, \ldots, \ell$. Further, corresponding to the sequence $\underline{p}_\ell \in \mathcal{R}^n$, $\ell = 1, 2, 3, \ldots$, in Sect. 2, we consider the sequence $\underline{s}_\ell \in \mathcal{R}^{n-|\mathcal{A}|}$, $\ell = 1, 2, 3, \ldots$, where $\underline{s}_1$ is zero, and where the $\ell$-th iteration of the Krylov subspace method sets $\underline{s}_{\ell+1}$

to the vector $\underline{\sigma}$ in $\mathcal{K}_\ell$ that minimizes $Q_k^{\text{red}}(\underline{\sigma})$ subject to $\|\underline{\sigma}\| \leq \Delta_k$. The analogue of Eq. (2.11) is that we write $\underline{s}_{\ell+1}$ as the sum

$$\underline{s}_{\ell+1} = \sum_{i=1}^{\ell} \theta_i \, \underline{w}_i \in \mathcal{R}^{n-|\mathcal{A}|}, \tag{4.3}$$

each $\underline{w}_i$ being a vector of unit length in $\mathcal{K}_i$ with the property $\underline{w}_i^T \underline{w}_j = 0, i \neq j$. We pick $\underline{w}_1 = -\underline{\nabla} Q_k^{\text{red}}(0)/\|\underline{\nabla} Q_k^{\text{red}}(0)\|$, and, for $\ell \geq 2$, the Arnoldi formula (2.14) supplies the vector

$$\underline{w}_\ell = \frac{\nabla^2 Q_k^{\text{red}} \, \underline{w}_{\ell-1} - \sum_{j=1}^{\ell-1} \left( \underline{w}_j^T \, \nabla^2 Q_k^{\text{red}} \, \underline{w}_{\ell-1} \right) \underline{w}_j}{\left\| \nabla^2 Q_k^{\text{red}} \, \underline{w}_{\ell-1} - \sum_{j=1}^{\ell-1} \left( \underline{w}_j^T \, \nabla^2 Q_k^{\text{red}} \, \underline{w}_{\ell-1} \right) \underline{w}_j \right\|}. \tag{4.4}$$

Equation (2.10), which is very useful for trust region calculations without linear constraints, takes the form

$$\underline{w}_\ell^T \, \nabla^2 Q_k^{\text{red}} \, \underline{w}_j = 0, \quad 1 \leq j \leq \ell - 2, \tag{4.5}$$

so again there are at most two nonzero terms in the sum of the Arnoldi formula.

Instead of generating each $\underline{w}_i$ explicitly, however, we work with the vectors $\underline{v}_i = \check{Q} \underline{w}_i \in \mathcal{R}^n$, $i \geq 1$, which are analogous to the vectors $\underline{v}_i$ in the second half of Sect. 2. In particular, because $\check{Q}^T \check{Q}$ is the $(n - |\mathcal{A}|) \times (n - |\mathcal{A}|)$ unit matrix, they enjoy the orthonormality property

$$\underline{v}_i^T \underline{v}_j = \left( \check{Q} \underline{w}_i \right)^T \left( \check{Q} \underline{w}_j \right) = \underline{w}_i^T \underline{w}_j = \delta_{ij}, \tag{4.6}$$

for all relevant positive integers $i$ and $j$. Furthermore, Eqs. (4.2) and (4.3) show that $Q_k^{\text{red}}(\underline{s}_{\ell+1})$ is the same as $Q_k(\underline{p}_{\ell+1})$, where $\underline{p}_{\ell+1}$ is the point

$$\underline{p}_{\ell+1} = \underline{x}_k + \sum_{i=1}^{\ell} \theta_i \, \check{Q} \, \underline{w}_i = \underline{x}_k + \sum_{i=1}^{\ell} \theta_i \, \underline{v}_i. \tag{4.7}$$

It follows that the required values of the parameters $\theta_i$, $i = 1, 2, \ldots, \ell$, are given by the minimization of the function

$$\Phi_{k\ell}(\underline{\theta}) = Q_k \left( \underline{x}_k + \sum_{i=1}^{\ell} \theta_i \, \underline{v}_i \right), \quad \underline{\theta} \in \mathcal{R}^\ell, \tag{4.8}$$

subject to the trust region bound $\|\underline{\theta}\| \leq \Delta_k$, which is like the calculation in the paragraph that includes expressions (2.11)–(2.13).

In order to construct $\underline{v}_i$, $i \geq 1$, we replace $\underline{w}_i$ in the definition $\underline{v}_i = \check{Q}\underline{w}_i$ by a vector that is available. The quadratic (4.2) has the first and second derivatives

$$\left.\begin{array}{rcl} \underline{\nabla} Q_k^{\rm red}(\underline{\sigma}) &=& \check{Q}^T \underline{\nabla} Q_k \left(\underline{x}_k + \check{Q}\,\underline{\sigma}\right), \quad \underline{\sigma} \in \mathcal{R}^{n-|\mathcal{A}|}, \\ \text{and} \qquad \nabla^2 Q_k^{\rm red} &=& \check{Q}^T \nabla^2 Q_k\, \check{Q} \;=\; \check{Q}^T H_k\, \check{Q}. \end{array}\right\} \tag{4.9}$$

Hence $\underline{\nabla} Q_k(\underline{x}_k) = \underline{g}_k$ yields the vector

$$\underline{v}_1 \;=\; \check{Q}\,\underline{w}_1 \;=\; -\check{Q}\,\underline{\nabla} Q_k^{\rm red}(0) / \left\| \underline{\nabla} Q_k^{\rm red}(0) \right\| \;=\; -\check{Q}\,\check{Q}^T \underline{g}_k / \left\| \check{Q}\,\check{Q}^T \underline{g}_k \right\|. \tag{4.10}$$

For $\ell \geq 2$, $\underline{v}_\ell = \check{Q}\underline{w}_\ell$ is obtained from Eq. (4.4) multiplied by $\check{Q}$. Specifically, the identities

$$\nabla^2 Q_k^{\rm red}\, \underline{w}_{\ell-1} \;=\; \check{Q}^T H_k\, \check{Q}\, \underline{w}_{\ell-1} \;=\; \check{Q}^T H_k\, \underline{v}_{\ell-1} \tag{4.11}$$

and

$$\underline{w}_\ell^T\, \nabla^2 Q_k^{\rm red}\, \underline{w}_j \;=\; \underline{w}_\ell^T \left(\check{Q}^T H_k\, \check{Q}\right) \underline{w}_j \;=\; \underline{v}_\ell^T H_k\, \underline{v}_j, \quad 1 \leq j \leq \ell, \tag{4.12}$$

give the Arnoldi formula

$$\underline{v}_\ell \;=\; \check{Q}\,\underline{w}_\ell \;=\; \frac{\check{Q}\,\check{Q}^T H_k\, \underline{v}_{\ell-1} - \sum_{j=1}^{\ell-1} \left(\underline{v}_j^T H_k\, \underline{v}_{\ell-1}\right) \underline{v}_j}{\left\| \check{Q}\,\check{Q}^T H_k\, \underline{v}_{\ell-1} - \sum_{j=1}^{\ell-1} \left(\underline{v}_j^T H_k\, \underline{v}_{\ell-1}\right) \underline{v}_j \right\|}, \quad \ell \geq 2. \tag{4.13}$$

Only the last two terms of its sum can be nonzero as before due to Eqs. (4.5) and (4.12).

The Krylov subspace method with the active set $\mathcal{A}$ is now very close to the method described in the two paragraphs that include Eqs. (2.11)–(2.14). The first change to the description in Sect. 2 is that, instead of the form (2.1), $\underline{p}_2$ is now the point $\underline{p}_1 - \alpha_1 \check{Q}\check{Q}^T \underline{g}_k$, where $\alpha_1$ is the value of $\alpha$ that minimizes $Q_k(\underline{p}_1 - \alpha\check{Q}\check{Q}^T \underline{g}_k)$, $\alpha \in \mathcal{R}$, subject to $\|\underline{p}_2 - \underline{x}_k\| \leq \Delta_k$. Equation (2.10) is a consequence of the properties (4.5) and (4.12), so $\nabla^2 \Phi_{k\ell}(\cdot)$ is still tridiagonal. The last $\ell - 1$ components of the gradient $\underline{\nabla}\Phi_{k\ell}(0) \in \mathcal{R}^\ell$ are still zero, but its first component is now $\pm\|\check{Q}\check{Q}^T \underline{g}_k\| = \pm\|\check{Q}^T \underline{g}_k\|$. Finally, the Arnoldi formula (2.14) is replaced by expression (4.13). We retain termination if inequality (2.6) holds.

The Krylov method was employed in an early version of the LINCOA software. If the calculated point (4.7) violates an inactive linear constraint, then $\underline{p}_{\ell+1}$ has to be replaced by a feasible point, and often the active set is changed. Some difficulties may occur, however, which are addressed in the remainder of this section. Because of them, the current version of LINCOA applies the version of conjugate gradients given in Sect. 5, instead of the Krylov method.

Here is an instructive example in only two variables with an empty active set, where both $\underline{p}_1 = \underline{x}_k$ and $\underline{p}_2$ are feasible, but the $\underline{p}_3$ of the Krylov method violates a constraint. We seek a small value of the quadratic model

$$Q_k(\underline{x}) \;=\; -50\,x_1 - 8\,x_1\,x_2 - 44\,x_2^2, \quad \underline{x} \in \mathcal{R}^2, \tag{4.14}$$

subject to $\|\underline{x}\| \le 1$ and $x_2 \le 0.2$, the trust region centre being at the origin, which is well away from the boundary of the linear constraint. The point $\underline{p}_2$ has the form (2.1), where $\underline{p}_1 = \underline{x}_k = 0$ and $\underline{g}_k = \underline{\nabla} Q_k(\underline{x}_k) = (-50, 0)^T$. The function $Q_k(\underline{p}_1 - \alpha \underline{g}_k), \alpha \in \mathcal{R}$, is linear, so $\underline{p}_2$ is on the trust region boundary at $(1, 0)^T$, which is also well away from the boundary of the linear constraint. Therefore $\underline{p}_3$ has the form (2.11) with $\ell = 2$, the coefficients $\theta_1$ and $\theta_2$ being calculated to minimize expression (2.12) subject to $\|\underline{\theta}\| \le 1$. In other words, because $n = 2$, the point $\underline{p}_3$ is the vector $\underline{x}$ that minimizes the model (4.14) subject to $\|\underline{x}\| \le 1$, and one can verify that this point is the unconstrained minimizer of the strictly convex quadratic function $Q_k(\underline{x}) + 47\,\|\underline{x}\|^2$, $\underline{x} \in \mathcal{R}^2$. Thus we find the sequence

$$\underline{p}_1 \;=\; \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \underline{p}_2 \;=\; \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \underline{p}_3 \;=\; \begin{pmatrix} 0.6 \\ 0.8 \end{pmatrix}, \tag{4.15}$$

with $Q_k(\underline{p}_1) = 0$, $Q_k(\underline{p}_2) = -50$ and $Q_k(\underline{p}_3) = -62$.

The point $\underline{p}_3$ is unacceptable, however, as it violates the constraint $x_2 \le 0.2$. The condition $Q_k(\underline{p}_3) < Q_k(\underline{p}_2)$ suggests that it may be helpful to consider the function of one variable

$$Q_k\left(\underline{p}_2 + \alpha\left\{\underline{p}_3 - \underline{p}_2\right\}\right) \;=\; -50 + 13.6\,\alpha - 25.6\,\alpha^2, \quad \alpha \in \mathcal{R}, \tag{4.16}$$

in order to move to the feasible point on the straight line through $\underline{p}_2$ and $\underline{p}_3$ that provides the least value of $Q_k(\cdot)$ subject to the trust region bound. Feasibility demands $\alpha \le 0.25$, but the function (4.16) increases strictly monotonically for $0 \le \alpha \le 17/64$, so all positive values of $\alpha$ are rejected. Furthermore, all negative values of $\alpha$ are excluded by the trust region bound. In this case, therefore, the point $\underline{p}_3$ of the Krylov method fails to provide an improvement over $\underline{p}_2$, even when $\underline{p}_3 - \underline{p}_2$ is used as a search direction from $\underline{p}_2$. On the other hand, $\underline{p}_2$ can be generated easily by the conjugate gradient method, and then $\underline{x}_k^+ = \underline{x}_k + \underline{p}_2$ is chosen, because $\underline{p}_2$ is on the trust region boundary.

The minimization of the function (4.14) subject to the constraints

$$\|\underline{x}\| \le 1, \quad x_2 \le 0.2 \quad \text{and} \quad x_1 + 0.1\,x_2 \le 0.6 \tag{4.17}$$

is also instructive. The data $\underline{p}_1 = \underline{x}_k = 0$ and $\underline{g}_k = (-50, 0)^T$ imply again that the step from $\underline{p}_1$ to $\underline{p}_2$ is along the first coordinate direction, and now $\underline{p}_2$ is at $(0.6, 0)^T$, because of the last of the conditions (4.17). The move from $\underline{p}_2$ is along the boundary of $x_1 + 0.1 x_2 \le 0.6$, the index of this constraint being put into the active set $\mathcal{A}$, so $\underline{p}_3$ and $Q_k(\underline{p}_3)$ take the values

$$\underline{p}_3 \;=\; \begin{pmatrix} 0.6 + 0.1\alpha \\ -\alpha \end{pmatrix} \quad \text{and} \quad Q_k(\underline{p}_3) \;=\; -30 - 0.2\,\alpha - 43.2\,\alpha^2, \tag{4.18}$$

for some steplength $\alpha$ that has to be chosen. The Krylov method would calculate the $\alpha$ that minimizes $Q_k(\underline{p}_3)$ subject to $\|\underline{p}_3\| \le 1$, which is $\alpha = -0.8576$, and then $\underline{p}_3$ is on the trust region boundary at $(0.5142, 0.8576)^T$ with $Q_k(\underline{p}_3) = -61.0647$, the exact figures being rounded to four decimal places. This $\underline{p}_3$ violates the constraint $x_2 \le 0.2$, however, so, if the sign of $\alpha$ is accepted, and if $|\alpha|$ is reduced by a line search to provide feasibility, then we find $\alpha = -0.2$, $\underline{p}_3 = (0.58, 0.2)^T$ and $Q_k(\underline{p}_3) = -31.688$. On the other hand, a conjugate gradient search would go downhill from $\underline{p}_2$ to $\underline{p}_3$, and expression (4.18) for $Q_k(\underline{p}_3)$ shows that $\alpha$ would be positive. This search would reach the trust region boundary at the feasible point $\underline{p}_3 = (0.6739, -0.7388)^T$, with $\alpha = 0.7388$ and $Q_k(\underline{p}_3) = -53.7298$. Thus it can happen that the Krylov method is highly disadvantageous if its steplengths are reduced without a change of sign for the sake of feasibility.

Another infelicity of the Krylov method when there are linear constraints is shown by the final example of this section. We consider the minimization of the function

$$Q_k(\underline{x}) = x_1^2 - 0.1\,x_2^2 - 20\,x_3^2 - 10\,x_1x_3 + 4.8\,x_2x_3 - 4\,x_2 - x_3 + 21, \quad \underline{x} \in \mathcal{R}^3, \tag{4.19}$$

subject to the constraints

$$x_3 \le 1 \quad \text{and} \quad \|\underline{x}\| \le \sqrt{26}, \tag{4.20}$$

so again the trust region centre $\underline{x}_k$ is at the origin. The step to $\underline{p}_2$ from $\underline{p}_1 = \underline{x}_k$ is along the direction $-\underline{\nabla} Q_k(\underline{x}_k) = (0, 4, 1)^T$, and, because $Q_k(\cdot)$ has negative curvature along this direction, the step is the longest one allowed by the constraints, giving $\underline{p}_2 = (0, 4, 1)^T$. Further, the active set becomes nonempty, in order to follow the boundary of $x_3 \le 1$, which, after the step from $\underline{p}_1$ to $\underline{p}_2$, reduces the calculation to the minimization of the function of two variables

$$\widetilde{Q}_k(\underline{\widetilde{x}}) = Q_k\begin{pmatrix} \widetilde{x}_1 \\ \widetilde{x}_2 \\ 1 \end{pmatrix} = \widetilde{x}_1^2 - 0.1\,\widetilde{x}_2^2 - 10\,\widetilde{x}_1 + 0.8\,\widetilde{x}_2, \quad \underline{\widetilde{x}} \in \mathcal{R}^2, \tag{4.21}$$

subject to $\|\underline{\widetilde{x}}\| \le 5$, starting at the point $\underline{\widetilde{p}}_1 = (0, 4)^T$, although the centre of the trust region is still at the origin.

The Krylov subspace $\mathcal{K}_\ell$, $\ell \ge 1$, for the minimization of $\widetilde{Q}_k(\underline{\widetilde{x}})$, $\underline{\widetilde{x}} \in \mathcal{R}^2$, is the space spanned by the vectors $(\nabla^2 \widetilde{Q}_k)^{j-1} \underline{\nabla} \widetilde{Q}_k(\underline{\widetilde{p}}_1)$, $j = 1, 2, \ldots, \ell$. The example has been chosen so that the matrix $\nabla^2 \widetilde{Q}_k$ is diagonal and so that $\underline{\nabla} \widetilde{Q}_k(\underline{\widetilde{p}}_1)$ is a multiple of the first coordinate vector. It follows that all searches in Krylov subspaces, starting at $\underline{\widetilde{p}}_1 = (0, 4)^T$, cannot alter the value $\widetilde{x}_2 = 4$ of the second variable, assuming that computer arithmetic is without rounding errors. The search from $\underline{\widetilde{p}}_1$ to $\underline{\widetilde{p}}_2$ is along the direction $-\underline{\nabla} \widetilde{Q}_k(\underline{\widetilde{p}}_1) = (10, 0)^T$. It goes to the point $\underline{\widetilde{p}}_2 = (3, 4)^T$ on the trust region boundary, which corresponds to $\underline{p}_3 = (3, 4, 1)^T$. Here $\widetilde{Q}_k(\cdot)$ is least subject to $\|\underline{\widetilde{x}}\| \le 5$ and $\widetilde{x}_2 = 4$, so the iterations of the Krylov method are complete. They generate the monotonically decreasing sequence

$$Q_k(\underline{p}_1) = 21, \quad Q_k(\underline{p}_2) = \widetilde{Q}_k(\underline{\widetilde{p}}_1) = 1.6 \quad \text{and} \quad Q_k(\underline{p}_3) = \widetilde{Q}_k(\underline{\widetilde{p}}_2) = -19.4. \tag{4.22}$$

In this case, the Krylov method brings the remarkable disadvantage that $\widetilde{Q}_k(\underline{\widetilde{p}}_2)$ is not the least value of $\widetilde{Q}_k(\underline{\widetilde{x}})$ subject to $\|\underline{\widetilde{x}}\| \leq 5$. Indeed, $\widetilde{Q}_k(\underline{\widetilde{x}}) = -25$ is achieved by $\underline{\widetilde{x}} = (5, 0)^T$, for example. The conjugate gradient method would also generate the sequence $\underline{p}_\ell$, $\ell = 1, 2, 3$, of the Krylov method, with termination at $\underline{p}_3$ because it is on the boundary of the trust region. A way of extending the conjugate gradient alternative by searching round the trust region boundary is considered in Sect. 6. It can calculate the vector $\underline{\widetilde{x}}$ that minimizes the quadratic (4.21) subject to $\|\underline{\widetilde{x}}\| \leq 5$, and it is included in the NEWUOA software for unconstrained optimization without derivatives ([10]).

## 5 Conjugate gradient methods

Taking the decision to employ conjugate gradients instead of Krylov subspace methods in the LINCOA software provided much relief, because the difficulties in the second half of Sect. 4 are avoided. In particular, if a step of the conjugate gradient method goes from $\underline{p}_\ell \in \mathcal{R}^n$ to $\underline{p}_{\ell+1} \in \mathcal{R}^n$, then the quadratic model along this step, which is the function $Q_k(\underline{p}_\ell + \alpha\{\underline{p}_{\ell+1} - \underline{p}_\ell\})$, $0 \leq \alpha \leq 1$, decreases strictly monotonically. The initial point of every step is feasible. It follows that, if one or more of the linear constraints (1.1) are violated at $\underline{x} = \underline{p}_{\ell+1}$, then it is suitable to replace $\underline{p}_{\ell+1}$ by the point

$$\underline{p}_{\text{new}} \;=\; \underline{p}_\ell + \alpha^* \left(\underline{p}_{\ell+1} - \underline{p}_\ell\right), \tag{5.1}$$

where $\alpha^*$ is the greatest $\alpha \in \mathcal{R}$ such that $\underline{p}_{\text{new}}$ is feasible. Thus $0 \leq \alpha^* < 1$ holds, and $\underline{p}_{\text{new}}$ is on the boundary of a constraint whose index is not in the current $\mathcal{A}$.

The conjugate gradient method of Sect. 2 without linear constraints can be applied to the reduced function

$$Q_k^{\text{red}}(\underline{\sigma}) \;=\; Q_k\left(\underline{p}_1 + \check{Q}\,\underline{\sigma}\right), \quad \underline{\sigma} \in \mathcal{R}^{n-|\mathcal{A}|}, \tag{5.2}$$

after generating the active set $\mathcal{A}$ at $\underline{p}_1$, as suggested in the paragraph that includes Eq. (3.7). Thus, starting at $\underline{\sigma} = \underline{\sigma}_1 = 0$, and until a termination condition is satisfied, we find the conjugate gradient search directions of the function (5.2) in $\mathcal{R}^{n-|\mathcal{A}|}$. Letting these directions be $\underline{d}_\ell^{\text{red}}$, $\ell = 1, 2, 3, \ldots$, which have the downhill property $(\underline{d}_\ell^{\text{red}})^T \underline{\nabla} Q_k^{\text{red}}(\underline{\sigma}_\ell) < 0$, each new point $\underline{\sigma}_{\ell+1} \in \mathcal{R}^{n-|\mathcal{A}|}$ has the form

$$\underline{\sigma}_{\ell+1} \;=\; \underline{\sigma}_\ell + \alpha_\ell\, \underline{d}_\ell^{\text{red}}, \tag{5.3}$$

where $\alpha_\ell$ is the value of $\alpha \geq 0$ that minimizes $Q_k^{\text{red}}(\underline{\sigma}_\ell + \alpha\, \underline{d}_\ell^{\text{red}})$ subject to a trust region bound. Equation (2.2) for the reduced problem provides the directions

$$\underline{d}_\ell^{\text{red}} \;=\; \begin{cases} -\underline{\nabla} Q_k^{\text{red}}(0), & \ell = 1, \\ -\underline{\nabla} Q_k^{\text{red}}(\underline{\sigma}_\ell) + \beta_\ell\, \underline{d}_{\ell-1}^{\text{red}}, & \ell \geq 2, \end{cases} \tag{5.4}$$

where $\beta_\ell$ is defined by the conjugacy condition

$$(\underline{d}_\ell^{\text{red}})^T \nabla^2 Q_k^{\text{red}} \underline{d}_{\ell-1}^{\text{red}} = 0. \tag{5.5}$$

As in Sect. 4, we prefer to work with the original variables $\underline{x} \in \mathcal{R}^n$, instead of with the reduced variables $\underline{\sigma} \in \mathcal{R}^{n-|\mathcal{A}|}$, so we express the techniques of the previous paragraph in terms of the original variables. In particular, the line search of the $\ell$-th conjugate gradient iteration sets $\alpha_\ell$ to the value of $\alpha \geq 0$ that minimizes $Q_k(\underline{p}_\ell + \alpha\, \underline{d}_\ell)$ subject to $\|\underline{p}_\ell + \alpha\, \underline{d}_\ell - \underline{x}_k\| \leq \Delta_k$, where $\underline{p}_\ell$ and $\underline{d}_\ell$ are the vectors

$$\underline{p}_\ell = \underline{p}_1 + \check{Q}\, \underline{\sigma}_\ell \quad \text{and} \quad \underline{d}_\ell = \check{Q}\, \underline{d}_\ell^{\text{red}}, \quad \ell \geq 1, \tag{5.6}$$

which follows from the definition (5.2). Thus Eq. (5.3) supplies the sequence of points $\underline{p}_{\ell+1} = \underline{p}_\ell + \alpha_\ell\, \underline{d}_\ell$, $\ell = 1, 2, 3, \ldots$, and there is no change to each steplength $\alpha_\ell$. Furthermore, formula (5.4) and the definition (5.2) supply the directions

$$\underline{d}_\ell = \check{Q}\, \underline{d}_\ell^{\text{red}} = \begin{cases} -\check{Q}\, \underline{\nabla} Q_k^{\text{red}}(0) = -\check{Q}\, \check{Q}^T\, \underline{\nabla} Q_k(\underline{p}_1), & \ell = 1, \\ -\check{Q}\, \check{Q}^T\, \underline{\nabla} Q_k(\underline{p}_\ell) + \beta_\ell\, \underline{d}_{\ell-1}, & \ell \geq 2, \end{cases} \tag{5.7}$$

and there is no change to the value of $\beta_\ell$. Because Eqs. (5.6) and (4.9) give the identities

$$\underline{d}_\ell^T H_k\, \underline{d}_j = (\underline{d}_\ell^{\text{red}})^T\, \check{Q}^T H_k\, \check{Q}\, \underline{d}_j^{\text{red}} = (\underline{d}_\ell^{\text{red}})^T\, \nabla^2 Q_k^{\text{red}}\, \underline{d}_j^{\text{red}}, \quad j = 1, 2, \ldots, \ell, \tag{5.8}$$

the condition (5.5) that defines $\beta_\ell$ is just $\underline{d}_\ell^T H_k\, \underline{d}_{\ell-1} = 0$, which agrees with the choice of $\beta_\ell$ in formula (2.2). Therefore, until a termination condition holds, the conjugate direction method for the active set $\mathcal{A}$ is the same as the conjugate direction method in Sect. 2, except that the gradients $\underline{\nabla} Q_k(\underline{p}_\ell)$, $\ell \geq 1$, are multiplied by the projection operator $\check{Q}\check{Q}^T$. Thus the directions (5.7) have the property $\underline{a}_j^T \underline{d}_\ell = 0$, $j \in \mathcal{A}$, in order to satisfy the constraints (3.2).

The conditions of LINCOA for terminating the conjugate gradient steps for the current active set $\mathcal{A}$ are close to the conditions in the paragraph that includes expressions (2.4)–(2.6). Again there is termination with $\underline{x}_k^+ = \underline{p}_\ell$ if inequality (2.4) or (2.5) holds, where $\hat{\alpha}_\ell$ is still the nonnegative value of $\alpha$ such that $\underline{p}_\ell + \alpha\, \underline{d}_\ell$ is on the trust region boundary. Alternatively, the new point $\underline{p}_{\ell+1} = \underline{p}_\ell + \alpha_\ell\, \underline{d}_\ell$ is calculated. If the test (2.6) is satisfied, or if $\underline{p}_{\ell+1}$ is a feasible point on the trust region boundary, or if $\underline{p}_{\ell+1}$ is any feasible point with $\ell = n - |\mathcal{A}|$, then the conjugate gradient steps for the current iteration number $k$ are complete, the value $\underline{x}_k^+ = \underline{p}_{\ell+1}$ being chosen, except that $\underline{x}_k^+ = \underline{p}_{\text{new}}$ is preferred if $\underline{p}_{\ell+1}$ is infeasible, as suggested in the first paragraph of this section. Another possibility is that $\underline{p}_{\ell+1}$ is a feasible point that is strictly inside the trust region with $\ell < n - |\mathcal{A}|$. Then $\ell$ is increased by one in order to continue the conjugate gradient steps for the current $\mathcal{A}$. In all other cases, $\underline{p}_{\ell+1}$ is infeasible, and we let $\underline{p}_{\text{new}}$ be the point (5.1).

In these other cases, a choice is made between ending the conjugate gradient steps with $\underline{x}_k^+ = \underline{p}_{\text{new}}$, or generating a new active set at $\underline{p}_{\text{new}}$. We recall from the last paragraph of Sect. 3 that the LINCOA choice is to update $\mathcal{A}$ if and only if the distance from $\underline{p}_{\text{new}}$ to the trust region boundary is at least $\eta_2 \Delta_k$. Furthermore, after using the notation $\underline{p}_1 = \underline{x}_k$ at the beginning of the $k$-th iteration as in Sect. 2, we now revise the meaning of $\underline{p}_1$ to $\underline{p}_1 = \underline{p}_{\text{new}}$ whenever a new active set is constructed at $\underline{p}_{\text{new}}$. Thus the description in this section of the truncated conjugate gradient method is valid for every $\mathcal{A}$.

The number of conjugate gradient steps is finite for each $\mathcal{A}$, even if we allow $\ell$ to exceed $n - |\mathcal{A}|$. Indeed, failure of the termination test (2.6) gives the condition

$$Q_k(\underline{x}_k) - Q_k\left(\underline{p}_{\ell+1}\right) > \left\{Q_k(\underline{x}_k) - Q_k(\underline{p}_\ell)\right\}/(1-\eta_1), \tag{5.9}$$

which cannot happen an infinite number of times as $Q_k(\underline{x})$, $\|\underline{x} - \underline{x}_k\| \le \Delta_k$, is bounded below. Also, the number of new active sets is finite for each iteration number $k$, which can be proved in the following way.

Let $\underline{p}_1$ be different from $\underline{x}_k$ due to a previous change to the active set, and let the work with the current $\mathcal{A}$ be followed by the construction of another active set at the point (5.1) with $\ell = \ell^*$, say. Then condition (5.9) and $Q_k(\underline{p}_{\text{new}}) \le Q_k(\underline{p}_{\ell^*})$ provide the bound

$$Q_k(\underline{x}_k) - Q_k\left(\underline{p}_{\text{new}}\right) \ge \left\{Q_k(\underline{x}_k) - Q_k(\underline{p}_1)\right\}/(1-\eta_1)^{\ell^*-1}, \tag{5.10}$$

which is helpful to our proof for $\ell^* \ge 2$, because $0 < \eta_1 < 1$, $0 < \eta_2 < 1$ and $\ell^* \ge 2$ imply $1/(1-\eta_1)^{\ell^*-1} > (1 + \frac{1}{4}\eta_1\eta_2)$. In the alternative case $\ell^* = 1$, we recall that the choice of $\mathcal{A}$ gives the search direction $\underline{d}_1$ the property $\underline{a}_j^T \underline{d}_1 \le 0$, $j \in \mathcal{J}(\underline{p}_1)$, where $\mathcal{J}(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, is the set (3.3). Thus the infeasibility of $\underline{p}_2$ and Eq. (5.1) yield $\|\underline{p}_{\text{new}} - \underline{p}_1\| > \eta_2 \Delta_k$. Moreover, the failure of the termination condition (2.5) for $\ell = 1$ with $\|\hat{\alpha}_1 \underline{d}_1\| < 2\Delta_k$ supply the inequality

$$\begin{aligned} \left|\left(\underline{p}_{\text{new}} - \underline{p}_1\right)^T \underline{\nabla} Q_k(\underline{p}_1)\right| &= \left|\hat{\alpha}_1\, \underline{d}_1^T \underline{\nabla} Q_k(\underline{p}_1)\right| \left\|\underline{p}_{\text{new}} - \underline{p}_1\right\| / \|\hat{\alpha}\, \underline{d}_1\| \\ &> \eta_1 \left\{Q_k(\underline{x}_k) - Q_k(\underline{p}_1)\right\} \left\|\underline{p}_{\text{new}} - \underline{p}_1\right\| /(2\,\Delta_k), \end{aligned} \tag{5.11}$$

the first line being true because $\underline{p}_{\text{new}} - \underline{p}_1$ is a multiple of $\underline{d}_1$. Now the function $\phi(\alpha) = Q_k(\underline{p}_1 + \alpha\{\underline{p}_{\text{new}} - \underline{p}_1\})$, $0 \le \alpha \le 1$, is a quadratic that decreases monotonically, which implies $\phi(0) - \phi(1) \ge \frac{1}{2}|\phi'(0)|$, and $|\phi'(0)|$ is the left hand side of expression (5.11). Thus, remembering $\|\underline{p}_{\text{new}} - \underline{p}_1\| > \eta_2 \Delta_k$, we deduce the property

$$\phi(0) - \phi(1) = Q_k(\underline{p}_1) - Q_k(\underline{p}_{\text{new}}) > \frac{1}{4}\eta_1\eta_2 \left\{Q_k(\underline{x}_k) - Q_k(\underline{p}_1)\right\}, \tag{5.12}$$

which gives the bound

$$Q_k(\underline{x}_k) - Q_k(\underline{p}_{\text{new}}) > \left(1 + \frac{1}{4}\eta_1\eta_2\right)\left\{Q_k(\underline{x}_k) - Q_k(\underline{p}_1)\right\} \tag{5.13}$$

in the case $\ell^* = 1$. It follows from expressions (5.10) and (5.13) that, whenever consecutive changes to the active set occur, the new value of $Q_k(\underline{x}_k) - Q_k(\underline{p}_1)$ is greater than the old one multiplied by $(1 + \frac{1}{4}\eta_1\eta_2)$. Again, due to the boundedness of $Q_k(\underline{x})$, $\|\underline{x} - \underline{x}_k\| \le \Delta_k$, this cannot happen an infinte number of times, so the number of changes to the active set is finite for each $k$.

An extension of the given techniques for truncated conjugate gradients subject to linear constraints is included in LINCOA, which introduces a tendency for $\underline{x}_k^+$ to be on the boundaries of the active constraints. Without the extension, more changes than necessary are made often to the active set in the following way. Let the $j$-th constraint be in the current $\mathcal{A}$, which demands the condition

$$b_j - \underline{a}_j^T \underline{p}_1 \;\le\; \eta_2 \, \Delta_k \, \|\underline{a}_j\| \tag{5.14}$$

for the current $\Delta_k$, let $\underline{p}_{\text{curr}}$ be the current $\underline{p}_1$, and let $(b_j - \underline{a}_j^T \underline{p}_{\text{curr}})/(\eta_2\|\underline{a}_j\|)$ be greater than the final trust region radius. Further, let the directions of $\underline{a}_j$ and $\underline{\nabla} F(\underline{x})$ for every relevant $\underline{x}$ be such that it would be helpful to keep the $j$-th constraint active for the remainder of the calculation. The $j$-th constraint cannot be in the final active set, however, unless changes to $\underline{p}_1$ cause $(b_j - \underline{a}_j^T \underline{p}_1)/(\eta_2\|\underline{a}_j\|)$ to be at most the final value of $\Delta_k$. On the other hand, while $j$ is in $\mathcal{A}$, condition (3.2) shows that all the conjugate gradient steps give $b_j - \underline{a}_j^T \underline{p}_{\ell+1} = b_j - \underline{a}_j^T \underline{p}_\ell$. Thus the index $j$ may remain in $\mathcal{A}$ until $\Delta_k$ becomes less than $(b_j - \underline{a}_j^T \underline{p}_{\text{curr}})/(\eta_2\|\underline{a}_j\|)$. Then $j$ has to be removed from $\mathcal{A}$, which allows the conjugate gradient method to generate a new $\underline{p}_1$ that supplies the reduction $b_j - \underline{a}_j^T \underline{p}_1 < b_j - \underline{a}_j^T \underline{p}_{\text{curr}}$. If $j$ is the index of a linear constraint that is important to the final vector of variables, however, then $j$ will be reinstated in $\mathcal{A}$ by yet another change to the active set.

The projected steepest descent direction $-\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_1)$ is calculated as before when the new active set $\mathcal{A}$ is constructed at $\underline{p}_1$, but now we call this direction $\underline{d}_{\text{old}}$, because the main feature of the extension is that it picks a new direction $\underline{d}_1$ for the formula $\underline{p}_2 = \underline{p}_1 + \alpha_1 \underline{d}_1$, in order that the residuals of the active constraints can be reduced by the move from $\underline{p}_1$ to $\underline{p}_2$. We still require $\underline{d}_{\text{old}}$ to be a direction of descent, so the conjugate gradient steps are terminated by setting $\underline{x}_k^+ = \underline{p}_1$ if $\underline{d}_{\text{old}}^T \underline{\nabla} Q_k(\underline{p}_1) \ge 0$ holds in practice, which is the test (2.4). Also, if all the residuals $b_j - \underline{a}_j^T \underline{p}_1$, $j \in \mathcal{A}$, are sufficiently small, which means in LINCOA that they satisfy the bounds

$$b_j - \underline{a}_j^T \underline{p}_1 \;\le\; 10^{-4} \, \Delta_k \, \|\underline{a}_j\|, \quad j \in \mathcal{A}, \tag{5.15}$$

then the conjugate gradient steps from $\underline{p}_1$ are as before, with $\underline{d}_1 = \underline{d}_{\text{old}}$.

In all other cases, the extension begins by calculating the shortest step from $\underline{p}_1$ to the boundaries of the active constraints. This step is the vector $\underline{d}_{\text{perp}}$, say, in the column space of $A$ that satisfies $A^T \underline{d}_{\text{perp}} = \underline{r}$, where $A$ has the columns $\underline{a}_j$, $j \in \mathcal{A}$, and where $\underline{r}$ has the components $b_j - \underline{a}_j^T \underline{p}_1$, $j \in \mathcal{A}$. We recall from near the middle of Sect. 3 that the QR factorization $A = \widehat{Q}R$ is available, which assists the construction of $\underline{d}_{\text{perp}}$. Indeed, it has the form $\underline{d}_{\text{perp}} = \widehat{Q}\underline{s}$, and $\underline{s}$ is defined by the equations

$$A^T \underline{d}_{\text{perp}} = (\widehat{Q}R)^T \widehat{Q}\, \underline{s} = R^T \underline{s} = \underline{r}, \tag{5.16}$$

so $\underline{s}$ can be calculated easily, using the triangularity and nonsingularity of $R$. The new search direction $\underline{d}_1$ of the extension is a nonnegative linear combination of $\underline{d}_{\text{old}}$ and $\underline{d}_{\text{perp}}$, as explained below.

The choice $\underline{p}_2 = \underline{p}_1 + \underline{d}_1$, where $\underline{d}_1$ is the direction

$$\underline{d}_1 = \eta_2\, \Delta_k\, \underline{d}_{\text{old}} / \|\underline{d}_{\text{old}}\| + \theta\, \underline{d}_{\text{perp}} \tag{5.17}$$

with $\theta = 1$, would bring the advantages that the length of the step from $\underline{p}_1$ to $\underline{p}_2$ is greater than $\eta_2\, \Delta_k$, and $\underline{p}_2$ would be on the boundaries of all the active constraints. This point $\underline{p}_2$, however, can be infeasible and can violate the trust region bound, although $\underline{p}_1 + \underline{d}_1$ would satisfy all the constraints in the case $\theta = 0$. Therefore the extension picks the direction (5.17), after setting $\theta$ to the greatest number in $[0, 1]$ such that all the constraints hold at $\underline{p}_1 + \underline{d}_1$. The value $\theta = 0$ may occur, and then $\underline{d}_1$ is a multiple of $\underline{d}_{\text{old}}$, so we would generate the sequence of steps from $\underline{p}_1$ by the conjugate gradient method without the extension.

When $\theta$ is positive in formula (5.17), we pick $\underline{p}_2 = \underline{p}_1 + \alpha_1\, \underline{d}_1$, where $\alpha_1$ is the value of $\alpha$ that minimizes $Q_k(\underline{p}_1 + \alpha\, \underline{d}_1)$, $0 \le \alpha \le 1$. The point $\underline{p}_2$ is feasible due to the choice of $\theta$, but, for every $\alpha > 1$, at least one constraint is violated at $\underline{p}_1 + \alpha\, \underline{d}_1$, in the usual case when $\underline{p}_1 + \underline{d}_1$ is on the boundary of a constraint that is satisfied as a strict inequality at $\underline{p}_1$. Our step from $\underline{p}_1$ to $\underline{p}_2$ achieves the reductions

$$b_j - \underline{a}_j^T \underline{p}_2 = (1 - \alpha_1\, \theta) \left( b_j - \underline{a}_j^T \underline{p}_1 \right), \quad j \in \mathcal{A}, \tag{5.18}$$

in the residuals of the active constraints.

The following argument shows that $\alpha_1$ is positive in Eq. (5.18). We have to rule out $\alpha_1 = 0$, so it is sufficient to establish $\underline{d}_1^T\, \underline{\nabla} Q_k(\underline{p}_1) < 0$, and we recall that termination with $\underline{x}_k^+ = \underline{p}_1$ would have happened earlier in the case $\underline{d}_{\text{old}}^T \underline{\nabla} Q_k(\underline{p}_1) \ge 0$. Hence, because of the choice (5.17), it is sufficient to prove $\underline{d}_{\text{perp}}^T \underline{\nabla} Q_k(\underline{p}_1) \le 0$. We find in the paragraph that includes expressions (3.4)–(3.6) that $\underline{d}_{\text{old}}$ is the vector $\underline{d}$ that minimizes $\|\underline{\nabla} Q_k(\underline{p}_1) + \underline{d}\|$ subject to $\underline{a}_j^T \underline{d} \le 0$, $j \in \mathcal{A}$, and the first order conditions at the solution of this quadratic programming problem provide the relation

$$\underline{d}_{\text{old}} = -\underline{\nabla} Q_k(\underline{p}_1) + \sum_{j \in \mathcal{A}} \lambda_j\, \underline{a}_j, \tag{5.19}$$

for some multipliers $\lambda_j$ that are all nonpositive. Moreover, $\underline{d}_{\text{perp}}$ is derived from the equations $\underline{a}_j^T \underline{d}_{\text{perp}} = r_j$, $j \in \mathcal{A}$, each $r_j = b_j - \underline{a}_j^T \underline{p}_1$ being nonnegative because $\underline{p}_1$ is feasible. Thus, remembering that $\underline{d}_{\text{perp}}$ is orthogonal to $\underline{d}_{\text{old}}$, we deduce the required inequality

$$\underline{d}_{\text{perp}}^T\, \underline{\nabla} Q_k(\underline{p}_1) = \underline{d}_{\text{perp}}^T \left\{ -\underline{d}_{\text{old}} + \sum_{j \in \mathcal{A}} \lambda_j\, \underline{a}_j \right\} = \sum_{j \in \mathcal{A}} \lambda_j\, r_j \le 0. \tag{5.20}$$

We now know from the property (5.18) that, with the extension, the step from $\underline{p}_1$ to $\underline{p}_2$ achieves a strict reduction in every positive residual of an active constraint. Termination with $\underline{x}_k^+ = \underline{p}_2$ is suitable if $Q_k(\underline{p}_1 + \alpha_1 \underline{d}_1)$ is the least value of $Q_k(\underline{p}_1 + \alpha \underline{d}_1)$, $\alpha \geq 0$, and if condition (2.6) holds for $\ell = 1$. Otherwise, a search direction $\underline{d}_2$ is chosen for a step from $\underline{p}_2$ to $\underline{p}_3$. The constraints (3.2) with $\ell = 2$ require $\underline{d}_2$ to be in the column space of $\check{Q}$, but the direction (5.17) is not in this space for $\theta > 0$. Therefore $\beta_2$ has to be zero in formula (5.7), giving $\underline{d}_2 = -\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_2)$. It is convenient to call this direction $\underline{d}_1 = -\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_1)$, so we replace $\underline{p}_2$ by $\underline{p}_1$ without making a further change to $\mathcal{A}$. Thus the given description of the truncated conjugate gradient method for the current $\mathcal{A}$ becomes valid for the continuation of the method. Formula (5.7) still provides search directions that are mutually conjugate, because the new $\underline{d}_1$ is a projected steepest descent direction.

## 6 Moves round the trust region boundary

The conjugate gradient method of Sect. 5 terminates at $\underline{x}_k^+ = \underline{p}_{\ell+1}$ if $\underline{p}_{\ell+1}$ is a feasible point on the boundary of the trust region, but usually a move round the boundary can generate another feasible point, $\underline{x}_k^{++}$ say, that provides the strict reduction $Q_k(\underline{x}_k^{++}) < Q_k(\underline{x}_k^+)$. In the case (4.19)–(4.20), for example, the conjugate gradient method yields $\underline{x}_k^+ = (3, 4, 1)^T$ with $Q_k(\underline{x}_k^+) = -19.4$, although the point $\underline{x}_k^{++} = (5, 0, 1)^T$ is feasible with $Q_k(\underline{x}_k^{++}) = -25$, as mentioned at the end of Sect. 4. The early versions of the LINCOA software included an extension to the conjugate gradient method that seeks reductions in $Q_k(\cdot)$ by searching round the trust region boundary, which is described briefly below. Now, however, LINCOA has been made simpler by the removal of the extension, because of some numerical results that are given too in this section.

A move from $\underline{x}_k^+$ to $\underline{x}_k^{++}$ round the trust region boundary is made by the early versions of LINCOA only if the Taylor series linear function

$$\Lambda_k^+(\underline{x}) \;=\; Q_k\left(\underline{x}_k^+\right) + \left(\underline{x} - \underline{x}_k^+\right)^T \, \underline{\nabla} Q_k\left(\underline{x}_k^+\right), \quad \underline{x} \in \mathcal{R}^n, \tag{6.1}$$

suggests that $Q_k(\underline{x}_k^{++})$ is going to be substantially less than $Q_k(\underline{x}_k^+)$. Indeed, letting $\widehat{\underline{x}}_k^{++}$ be the vector $\underline{x}$ that minimizes $\Lambda_k^+(\underline{x})$ subject to $\underline{a}_j^T(\underline{x} - \underline{x}_k^+) = 0$, $j \in \mathcal{A}$, and to $\|\underline{x} - \underline{x}_k\| \leq \Delta_k$, the search for a relatively small value of $Q_k(\cdot)$ is terminated at $\underline{x}_k^+$ if the inequality

$$\Lambda_k^+\left(\underline{x}_k^+\right) - \Lambda_k^+\left(\widehat{\underline{x}}_k^{++}\right) \;=\; \left(\underline{x}_k^+ - \widehat{\underline{x}}_k^{++}\right)^T \, \underline{\nabla} Q_k\left(\underline{x}_k^+\right) \;\leq\; \eta_1 \left\{Q_k(\underline{x}_k) - Q_k\left(\underline{x}_k^+\right)\right\} \tag{6.2}$$

holds, which corresponds to the test (2.17). The vector $\widehat{\underline{x}}_k^{++}$ is not calculated explicitly, however, because, using $\|\underline{x}_k^+ - \underline{x}_k\| = \Delta_k$, it can be shown that the reduction $\Lambda_k^+(\underline{x}_k^+) - \Lambda_k^+(\widehat{\underline{x}}_k^{++})$ is the sum

$$\left\| \check{Q}^T \left(\underline{x}_k^+ - \underline{x}_k\right)\right\| \, \left\| \check{Q}^T \underline{\nabla} Q_k \left(\underline{x}_k^+\right)\right\| + \left\{\check{Q}^T \left(\underline{x}_k^+ - \underline{x}_k\right)\right\}^T \check{Q}^T \underline{\nabla} Q_k \left(\underline{x}_k^+\right). \quad (6.3)$$

Thus condition (6.2) causes termination if the vectors $\check{Q}^T(\underline{x}_k^+ - \underline{x}_k)$ and $\check{Q}^T \underline{\nabla} Q_k(\underline{x}_k^+)$ are parallel with opposite signs. Also, a difficulty is avoided by termination in the highly unusual case when these two vectors are parallel (or nearly parallel) with the same sign. Specifically, there is a search round the trust region boundary from $\underline{x}_k^+$ only if $\underline{x}_k^+$ has the property

$$\begin{aligned} &\left\| \check{Q}^T \left(\underline{x}_k^+ - \underline{x}_k\right)\right\| \, \left\| \check{Q}^T \underline{\nabla} Q_k \left(\underline{x}_k^+\right)\right\| - \left| \left\{\check{Q}^T \left(\underline{x}_k^+ - \underline{x}_k\right)\right\}^T \check{Q}^T \underline{\nabla} Q_k \left(\underline{x}_k^+\right)\right| \\ &\quad > \eta_1 \left\{Q_k(\underline{x}_k) - Q_k\left(\underline{x}_k^+\right)\right\}. \end{aligned} \quad (6.4)$$

When condition (6.4) holds, the vectors $\check{Q}\check{Q}^T(\underline{x}_k^+ - \underline{x}_k)$ and $\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{x}_k^+)$ are linearly independent, and we let $\mathcal{S} \subset \mathcal{R}^n$ be the two dimensional linear space that is spanned by them. Each move $\underline{x}_k^{++} - \underline{x}_k^+$ is chosen to be in $\mathcal{S}$, which is reasonable because $\widehat{\underline{x}}_k^{++} - \underline{x}_k^+$ is in $\mathcal{S}$. The constraint $\|\underline{x}_k^{++} - \underline{x}_k\| = \Delta_k$ is achieved by giving $\underline{x}_k^{++} - \underline{x}_k$ the form

$$\underline{s}(\theta) \;=\; (\cos\theta - 1)\, \check{Q}\, \check{Q}^T \left(\underline{x}_k^+ - \underline{x}_k\right) + \sin\theta \, \underline{v}, \quad \theta \in \mathcal{R}, \quad (6.5)$$

where $\underline{v}$ is the vector in $\mathcal{S}$ that is orthogonal to $\check{Q}\check{Q}^T(\underline{x}_k^+ - \underline{x}_k)$, and that satisfies $\|\underline{v}\| = \|\check{Q}\check{Q}^T(\underline{x}_k^+ - \underline{x}_k\|$ and $\underline{v}^T \underline{\nabla} Q_k(\underline{x}_k^+) < 0$. The value of $\theta$ is calculated by seeking the first local minimum of the function

$$\phi(\theta) \;=\; Q_k\left(\underline{x}_k^+ + \underline{s}(\theta)\right), \quad \theta \geq 0, \quad (6.6)$$

subject to the feasibility of $\underline{x}_k^+ + \underline{s}(\theta)$. The inequality $\underline{v}^T \underline{\nabla} Q_k(\underline{x}_k^+) < 0$ supplies $\phi'(0) < 0$.

Equations (1.2) and (6.5) show that the function (6.6) is a trigonometric polynomial of degree two. The coefficients of this polynomial are generated, the amount of work when $n$ is large being dominated by the need for the product $H_k \underline{v}$, the product $H_k \check{Q}\check{Q}^T(\underline{x}_k^+ - \underline{x}_k)$ being available. We calculate an estimate, $\theta^*$ say, of the least positive value of $\theta$ that satisfies $\phi'(\theta) = 0$, the relative error of the estimate being at most 0.01. By considering every inactive constraint whose boundary is within distance $\Delta_k$ of $\underline{x}_k$, the value of $\theta^*$ is reduced if necessary so that all the points $\underline{x}_k^+ + \underline{s}(\theta)$, $0 \leq \theta \leq \theta^*$, are feasible. Then we make the choice $\underline{x}_k^{++} = \underline{x}_k^+ + \underline{s}(\theta^*)$.

We take the view that $\underline{x}_k^{++}$ is a new vector $\underline{x}_k^+$. There are no more searches round the trust region boundary for the current $k$ if an inactive constraint causes a decrease in $\theta^*$ in the previous paragraph, or if the change above to $\underline{x}_k^+$ reduces $Q_k(\underline{x}_k^+)$ by an amount that is at most the new value of $\eta_1\{Q_k(\underline{x}_k) - Q_k(\underline{x}_k^+)\}$, which corresponds to the test (2.6), or if condition (6.4) fails for the new $\underline{x}_k^+$. Otherwise, another move is made from $\underline{x}_k^+$ to a new $\underline{x}_k^{++}$ in the way that has been described already. This procedure is continued until termination occurs.

The success of searches round the trust region boundary was investigated by numerical experiments, including the application of versions of LINCOA to the following problem. We let $n$ be even, and, for each $\underline{x} \in \mathcal{R}^n$, we set the points

$$\underline{p}_i = \begin{pmatrix} x_{2i-1} \\ x_{2i} \end{pmatrix} \in \mathcal{R}^2, \quad i = 1, 2, \ldots, n/2. \tag{6.7}$$

We seek the least value of the function

$$F(\underline{x}) = n^{-2} \sum_{i=2}^{n/2} \sum_{j=1}^{i-1} \min \left[ \left\| \underline{p}_i - \underline{p}_j \right\|^{-1}, \, 10^3 \right], \quad \underline{x} \in \mathcal{R}^n, \tag{6.8}$$

subject to $3n/2$ linear constraints, namely that every $\underline{p}_i$ is in the triangle with the vertices (0, 0), (2, 0) and (0, 2). The initial positions of the points are chosen randomly within the triangle. For example, the left hand and middle parts of Fig. 2 show the initial random positions and the final calculated positions of the points in a case with $n = 80$, while the right hand part of Fig. 2 shows calculated positions for a different random start. Both sets of final points satisfy to high accuracy the first order conditions for the solution of the test problem, but the numbers of final points that are strictly inside the triangle are different, the two final values of the objective function being $F = 0.15626737$ and $F = 0.15603890$. This test problem has several local minima, and LINCOA tries to find only one of them.

LINCOA requires not only an initial vector of variables but also the initial and final values of $\Delta_k$, which are set to 0.1 and $10^{-6}$ in these calculations. The value of NPT is also required, which is the number of interpolation conditions satisfied by each approximation $Q_k(\underline{x}) \approx F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$. The amount of routine work for each $k$ is of magnitude NPT squared, due to a linear system of equations that supplies each new quadratic model, so it is helpful for NPT to be of magnitude $n$ when $n$ is large. The values NPT $= n + 6$ and NPT $= 2n + 1$ are compared in the numerical results of this section.

Tables 1 and 2 give these results for $n = 10 \times 2^\ell$, $\ell = 0, 1, 2, 3, 4, 5$. There are five cases for each $n$, the cases being distinguished by different random choices of the

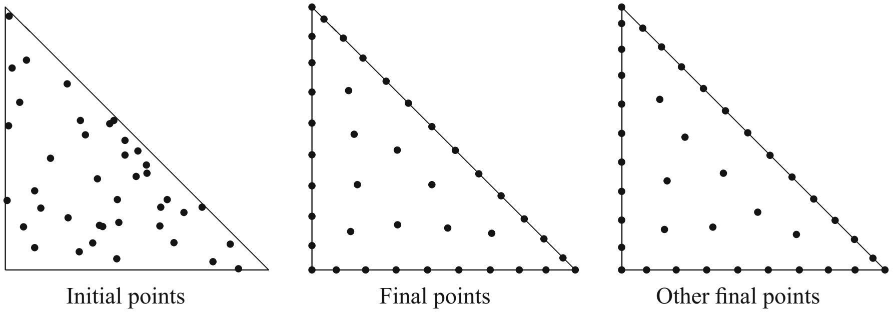

**Fig. 2** The points in triangle problem with $n = 80$

**Table 1** The points in triangle problem with NPT $= n + 6$

| $n$ | #$F$ | #TRI | [1, 3] | [4, 10] | [11, ∞) |
|---|---|---|---|---|---|
| 10 | 144/127 | 100/91 | 96/68 | 5/19 | 0/3 |
| 20 | 483/451 | 324/318 | 243/101 | 80/198 | 0/19 |
| 40 | 1472/2270 | 983/1554 | 822/430 | 161/881 | 0/243 |
| 80 | 7324/5873 | 4861/3927 | 4555/2183 | 304/1433 | 2/311 |
| 160 | 34815/33396 | 22641/22192 | 22327/17702 | 313/3790 | 2/700 |
| 320 | 226065/287538 | 146717/189130 | 146042/173010 | 667/13758 | 9/2362 |

**Table 2** The points in triangle problem with NPT $= 2n + 1$

| $n$ | #$F$ | #TRI | [1, 3] | [4, 10] | [11, ∞) |
|---|---|---|---|---|---|
| 10 | 179/193 | 121/129 | 117/82 | 4/43 | 0/4 |
| 20 | 584/457 | 354/292 | 250/54 | 104/207 | 0/31 |
| 40 | 1472/1733 | 898/1078 | 608/115 | 287/667 | 3/296 |
| 80 | 5462/5838 | 3459/3741 | 2583/292 | 837/1716 | 39/1733 |
| 160 | 19645/26569 | 12588/18010 | 9766/2145 | 2726/6801 | 96/9064 |
| 320 | 73560/73061 | 48483/49428 | 39116/7878 | 9092/16559 | 275/24991 |

initial vector of variables. For each application of LINCOA, we let #$F$ be the number of calculations of the objective function, and we let #TRI (Trust Region Iterations) be the number of iterations that construct $\underline{x}_k^+$ by the truncated conjugate gradient method of Sect. 5, with or without searches round the trust region boundary. The second and third columns of the tables show the averages to the nearest integer of #$F$ and #TRI over the five cases for each $n$. We recall that every step in the construction of $\underline{x}_k^+$ requires a vector to be multiplied by the matrix $H_k = \nabla^2 Q_k$, which is the most expensive part of every step. For each $k$, the number of multiplications is in the range $[1, 3]$, $[4, 10]$ or $[11, \infty)$. The number of occurrences of each range is counted for each test problem, the sum of these numbers being #TRI. These counts are also averaged to the nearest integer over the five cases for each $n$, the results being shown in the last three columns of the tables. Two versions of LINCOA are employed, the first one being the current version that is without searches round the trust region boundary, and the second one being the extension of the current version that includes the boundary searches and termination conditions of this section. The main entries in the table have the form $p/q$, where $p$ and $q$ are the averages of the version of LINCOA that is without and with boundary searches, respectively. Good accuracy is achieved throughout the experiments of Tables 1 and 2, the greatest residual of a KKT first order condition at a final vector of variables being about $3 \times 10^{-5}$.

The results in the last rows of both tables are highly unfavourable to searches round the trust region boundary. We find in the $n = 320$ row of Table 1 that the extra work of the searches causes #$F$ to become worse by 27 %, while the 0.7 % improvement in #$F$ in the $n = 320$ row of Table 2 is very expensive. Indeed, although the conjugate gradient method is truncated after at most 3 steps in 39,116 out of 48,483 applications, the method with boundary searches requires more than 10 multiplications of a vector by $H_k = \nabla^2 Q_k$ in about half of its 49,428 applications. Furthermore, the boundary searches in the $n = 80$ and $n = 160$ rows of Table 2 also take much more effort, and they cause #$F$ to increase by about 7 and 35%, respectively. Table 2 is more relevant than Table 1 for $n \geq 80$ because its values of #$F$ are smaller. Moreover, the #$F$ entries in the $n = 40$ rows of both tables suggest strongly that boundary searches are unhelpful. We give less attention to smaller values of $n$, because LINCOA is designed to be particularly useful for minimization without derivatives when there are hundreds of variables, by taking advantage of the discovery that the symmetric Broyden updating method makes such calculations possible. Thus Tables 1 and 2 provide excellent support for the decision, taken in 2013, to terminate the calculation of $\underline{x}_k^+$ by LINCOA when the steps of the conjugate gradient method reach the trust region boundary.

## 7 Further remarks and discussion

We begin our conclusions by noting that, when there are no linear constraints on the variables, the Krylov subspace method provides searches round the boundary of the trust region that compare favourably with the searches of Sect. 6. Let $\widehat{\underline{p}}_{\ell+1} \in \mathcal{R}^n$ and $\check{\underline{p}}_{\ell+1} \in \mathcal{R}^n$ be the points that are generated by the $\ell$-th step of the Krylov method and by the $\ell$-th step of the conjugate gradient method augmented by boundary searches, respectively, starting at the trust region centre $\widehat{\underline{p}}_1 = \check{\underline{p}}_1 = \underline{x}_k$, but the greatest values of $\ell$ for the two methods may be different. Because $\check{\underline{p}}_{\ell+1} - \underline{x}_k$ is in the linear space spanned by the gradients $\underline{\nabla} Q_k(\check{\underline{p}}_j)$, $j = 1, 2, \ldots, \ell$, it is also in the Krylov subspace $\mathcal{K}_\ell$, defined in the complete paragraph between expressions (2.8) and (2.9), even if the dimension of $\mathcal{K}_\ell$ is less than $\ell$. Also the choice of $\widehat{\underline{p}}_{\ell+1}$ by the $\ell$-th step of the Krylov method satisfies $\widehat{\underline{p}}_{\ell+1} - \underline{x}_k \in \mathcal{K}_\ell$, with the additional property that $Q_k(\widehat{\underline{p}}_{\ell+1})$ is the least value of $Q_k(\underline{x})$, $\|\underline{x} - \underline{x}_k\| \leq \Delta_k$, subject to $\underline{x} - \underline{x}_k \in \mathcal{K}_\ell$. Thus the Krylov method enjoys the advantage $Q_k(\widehat{\underline{p}}_{\ell+1}) \leq Q_k(\check{\underline{p}}_{\ell+1})$ for every $\ell$ that occurs for both methods. Moreover, the Krylov method terminates when $\mathcal{K}_{\ell+1} = \mathcal{K}_\ell$ holds, which gives $\ell \leq n$ even if the other conditions for termination are ignored. Usually, however, the number of boundary searches in Sect. 6 can be made arbitrarily large by letting the parameter $\eta_1$ in the test (6.4) be sufficiently small. These remarks suggest that the very poor results for searches in the last three columns of Tables 1 and 2 may be avoided if the Krylov method is applied.

Much of the effort in the development of LINCOA was spent on attempts to include the Krylov subspace method of Sect. 4 when there are linear constraints on the variables. Careful attention was given to the situation when, having chosen an active set at the point $\underline{p}_1 \in \mathcal{R}^n$, which need not be the trust region centre $\underline{x}_k$, the method generates the sequence $\underline{p}_{j+1}$, $j = 1, 2, \ldots, \ell$, in the trust region, and $\underline{p}_{\ell+1}$ is the first point in the sequence that violates a linear constraint. If the function $\phi(\alpha) = Q_k(\underline{p}_\ell + \alpha\{\underline{p}_{\ell+1} - \underline{p}_\ell\})$, $0 \leq \alpha \leq 1$, decreases monotonically, then the method in the first paragraph of Sect. 5 is recommended, which allows either termination with $\underline{x}_k^+ = \underline{p}_{\text{new}}$, or a change to $\mathcal{A}$ with $\underline{p}_{\text{new}}$ becoming the starting point $\underline{p}_1$ of the Krylov method for the new active set, $\underline{p}_{\text{new}}$ being the vector (5.1).

Examples in Sect. 4 show, however, that occasionally the first derivative $\phi'(0)$ is positive, although the Krylov method gives $\phi(1) < \phi(0)$ and $\phi'(1) \le 0$, where $\phi(\cdot)$ is defined above. A way of making further progress in this case is to pick $\theta > 0$, and to let $\underline{p}_{\ell+1}(\theta)$ be the vector $\underline{x}$ that minimizes the extended quadratic function

$$Q_k^+(\underline{x}, \theta) \;=\; Q_k(\underline{x}) \;+\; \theta \left\| \underline{x} - \underline{p}_\ell \right\|^2, \quad \underline{x} \in \mathcal{R}^n, \tag{7.1}$$

subject to $\|\underline{x} - \underline{x}_k\| \le \Delta_k$ and to $\underline{x} - \underline{p}_1$ being in the current Krylov subspace. For every $\theta > 0$, the calculation of $\underline{p}_{\ell+1}(\theta)$ is like the calculation of $\underline{p}_{\ell+1} = \underline{p}_{\ell+1}(0)$, due to $\nabla^2 Q_k^+(\cdot, \theta) - \nabla^2 Q_k$ being a multiple of the unit matrix. The function $\phi(\alpha, \theta) = Q_k^+(\underline{p}_\ell + \alpha\{\underline{p}_{\ell+1}(\theta) - \underline{p}_\ell\}, \theta), 0 \le \alpha \le 1$, does decrease monotonically if $\theta$ is sufficiently large, in particular when $Q_k^+(\cdot, \theta)$ is convex, because of the property $\phi'(1, \theta) \le 0$. The author has considered finding a relatively small value of $\theta$ that supplies the downhill condition $\phi'(0, \theta) \le 0$, followed by a change in $\underline{p}_{\ell+1}$ from $\underline{p}_{\ell+1}(0)$ to $\underline{p}_{\ell+1}(\theta)$. Equation (7.1) with the Krylov method show that the move from $\underline{p}_\ell$ to the changed $\underline{p}_{\ell+1}$ achieves the reduction

$$Q_k(\underline{p}_{\ell+1}) \;\le\; Q_k^+(\underline{p}_{\ell+1}, \theta) \;=\; Q_k^+\left(\underline{p}_{\ell+1}(\theta), \theta\right) \;\le\; Q_k^+(\underline{p}_\ell, \theta) \;=\; Q_k(\underline{p}_\ell). \tag{7.2}$$

Having replaced $\underline{p}_{\ell+1}$ by $\underline{p}_{\ell+1}(\theta)$, where $\theta$ gives the required monotonicity, the method in the second paragraph of this section is applied if $\underline{p}_{\ell+1}$ is infeasible. Otherwise, termination with $\underline{x}_k^+ = \underline{p}_{\ell+1}$ is suitable if condition (2.6) holds, the alternative being to continue the steps of the Krylov method for the calculation of $\underline{p}_{\ell+2}$, with the current active set and the original quadratic model. These techniques provide an iterative procedure that terminates for the current $\mathcal{A}$, due to the argument in the paragraph that includes inequality (5.9).

We give further consideration to the Krylov method when the active set is chosen at a point $\underline{p}_1$ that is different from the centre of the trust region. Then attention is restricted to vectors $\underline{x} \in \mathcal{R}^n$ that satisfy $\underline{a}_j^T(\underline{x} - \underline{p}_1) = 0$, $j \in \mathcal{A}$, which allows the trust region bound $\|\underline{x} - \underline{x}_k\| \le \Delta_k$ to be written as the inequality

$$\|\underline{x} - \widehat{\underline{x}}_k\| \;\le\; \left\{\Delta_k^2 - \|\widehat{\underline{x}}_k - \underline{x}_k\|^2\right\}^{1/2} \;=\; \widehat{\Delta}_k, \tag{7.3}$$

say, where $\widehat{\underline{x}}_k$ is now the shifted trust region centre

$$\widehat{\underline{x}}_k \;=\; \underline{p}_1 + \check{Q}\,\check{Q}^T(\underline{x}_k - \underline{p}_1) \;=\; \underline{x}_k + \widehat{Q}\,\widehat{Q}^T(\underline{p}_1 - \underline{x}_k). \tag{7.4}$$

Indeed, as $\underline{x} - \underline{p}_1$ is restricted to the column space of $\check{Q}$, Eq. (7.4) shows that $\underline{x} - \widehat{\underline{x}}_k$ and $\widehat{\underline{x}}_k - \underline{x}_k$ are in the column spaces of $\check{Q}$ and $\widehat{Q}$, respectively. Thus $\underline{x} - \widehat{\underline{x}}_k$ is orthogonal to $\widehat{\underline{x}}_k - \underline{x}_k$, so the identity $\underline{x} - \underline{x}_k = (\underline{x} - \widehat{\underline{x}}_k) + (\widehat{\underline{x}}_k - \underline{x}_k)$ gives the form (7.3) of the trust region bound. Further, because Eq. (7.4) implies $\|\widehat{\underline{x}}_k - \underline{x}_k\| \le \|\underline{p}_1 - \underline{x}_k\|$, and because active sets are chosen only at points $\underline{p}_1$ that satisfy $\|\underline{p}_1 - \underline{x}_k\| < \Delta_k$, the trust region radius $\widehat{\Delta}_k$ in expression (7.3) is positive.

The form (7.3) of the trust region bound exposes some deficiencies of the Krylov method in the case $\underline{p}_1 \ne \underline{x}_k$. The description of the method in Sect. 4 is not convenient, however, because some of the details of the computation, like the use of the Arnoldi formula (4.13), are not relevant to our discussion. Instead, we employ a definition of the Krylov subspace $\mathcal{K}_\ell$, $\ell \ge 1$, that depends on the sequence of points $\underline{p}_j$, $j = 1, 2, \ldots, \ell$. It can be shown to be equivalent to the previous one, because the Krylov and conjugate gradient methods generate the same points while strictly inside the trust region, the direction (5.7) being in the Krylov subspace $\mathcal{K}_\ell$. Thus $\mathcal{K}_1$ is the subspace

$$\mathcal{K}_\ell \;=\; \text{span}\left\{ \check{Q}\,\check{Q}^T\,\underline{\nabla} Q_k(\underline{p}_j) \;:\; j = 1, 2, \ldots, \ell \right\} \tag{7.5}$$

in the case $\ell = 1$. Having found $\mathcal{K}_1$, we let $\underline{p}_2$ be the point given by the usual construction of $\underline{p}_{\ell+1}$ in the case $\ell = 1$. Indeed, $\underline{p}_{\ell+1}$ is the vector $\underline{x}$ that minimizes $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, subject to $\underline{x} - \underline{p}_1 \in \mathcal{K}_\ell$ and to $\|\underline{x} - \widehat{\underline{x}}_k\| \le \widehat{\Delta}_k$. The definition (7.5) with this construction are applied iteratively for $\ell = 1, 2, 3, \ldots$, until a condition for termination is satisfied, one of them being $\mathcal{K}_\ell = \mathcal{K}_{\ell-1}$. Two obvious features of formula (7.5) are that it provides $\mathcal{K}_1 \subset \mathcal{K}_2 \subset \cdots \subset \mathcal{K}_\ell$, and that it puts the step $\underline{p}_{\ell+1} - \underline{p}_\ell$ into the column space of $\check{Q}$, as required by the constraints (3.2). The termination condition (2.6) is recommended. We have addressed already the situation when $\underline{p}_{\ell+1}$ is infeasible.

We compare this choice of $\underline{p}_{\ell+1}$ to the one that is optimal if $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, is replaced by the linear approximation

$$\Lambda_{k\ell}(\underline{x}) \;=\; Q_k(\underline{p}_\ell) \;+\; (\underline{x} - \underline{p}_\ell)^T\,\underline{\nabla} Q_k(\underline{p}_\ell), \quad \underline{x} \in \mathcal{R}^n, \tag{7.6}$$

which occurs also in Eq. (2.15). The vector $-\check{Q}\check{Q}^T\underline{\nabla} Q_k(\underline{p}_\ell)$ is the steepest descent direction of $\Lambda_{k\ell}(\cdot)$ that is allowed by the active constraints, so the least value of $\Lambda_{k\ell}(\cdot)$ subject to these constraints and the trust region bound (7.3) is at the point

$$\widehat{\underline{p}}_{\ell+1} \;=\; \widehat{\underline{x}}_k \;-\; \widehat{\Delta}_k\,\check{Q}\,\check{Q}^T\,\underline{\nabla} Q_k(\underline{p}_\ell) \Big/ \left\| \check{Q}\,\check{Q}^T\,\underline{\nabla} Q_k(\underline{p}_\ell) \right\|. \tag{7.7}$$

Assuming for the moment that the second derivatives of $Q_k(\cdot)$ are small enough for the approximation $\Lambda_{k\ell}(\underline{x}) \approx Q_k(\underline{x})$, $\|\underline{x} - \widehat{\underline{x}}\| \le \widehat{\Delta}_k$, to be useful, we would like the reduction $Q_k(\underline{p}_\ell) - Q_k(\underline{p}_{\ell+1})$ to compare favourably with $Q_k(\underline{p}_\ell) - \Lambda_{k\ell}(\widehat{\underline{p}}_{\ell+1})$. This hope is achieved if $\widehat{\underline{p}}_{\ell+1} - \underline{p}_1$ is in $\mathcal{K}_\ell$, because then the calculation of $\underline{p}_{\ell+1}$ by the Krylov method provides $Q_k(\underline{p}_{\ell+1}) \le Q_k(\widehat{\underline{p}}_{\ell+1}) \approx \Lambda_{k\ell}(\widehat{\underline{p}}_{\ell+1})$. Equation (7.7) shows that $\widehat{\underline{p}}_{\ell+1} - \underline{p}_1$ is a linear combination of $\widehat{\underline{x}}_k - \underline{p}_1$ and $\check{Q}\check{Q}^T\underline{\nabla} Q_k(\underline{p}_\ell)$, and the definition (7.5) gives $\check{Q}\check{Q}^T\underline{\nabla} Q_k(\underline{p}_\ell) \in \mathcal{K}_\ell$. It follows that the Krylov method is suitable in the case $\widehat{\underline{x}}_k - \underline{p}_1 = 0$, which means that the starting point of the Krylov method for the current $\mathcal{A}$ is at the centre of the trust region constraint (7.3). Otherwise, the Krylov method may be disadvantageous.

We apply the remarks above to the last example of Sect. 4, where we seek a relatively small value of the function (4.19) subject to the constraints (4.20), the trust region centre $\underline{x}_k$ being at the origin. The initial active set is empty, so the move from $\underline{x}_k$ goes along the direction $-\underline{\nabla} Q_k(\underline{x}_k)$, and it reaches the point $\underline{p}_1 = (0, 4, 1)^T$ on the boundary of the linear constraint $x_3 \leq 1$. This description is in Sect. 4, except that our present notation requires the point $(0, 4, 1)^T$ to be called $\underline{p}_1$ instead of $\underline{p}_2$, because a new active set is generated there to prevent violations of the linear constraint. The new set $\{\underline{a}_j : j \in \mathcal{A}\}$ contains only the vector $(0, 0, 1)^T$, which gives the matrices

$$\widehat{Q}\,\widehat{Q}^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \check{Q}\,\check{Q}^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{7.8}$$

Now we are working in $\mathcal{R}^3$, although the reduced space $\mathcal{R}^2$ is employed in the penultimate paragraph of Sect. 4. Expressions (4.19), (7.8), (7.4) and (7.3) supply $\underline{\nabla} Q_k(\underline{p}_1) = (-10, 0, -21.8)^T$, $\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_1) = (-10, 0, 0)^T$, $\widehat{\underline{x}}_k = (0, 0, 1)^T$ and $\widehat{\Delta}_k = 5$. Therefore the Krylov method and Eq. (7.7) provide $\underline{p}_2 = (3, 4, 1)^T$ and $\widehat{\underline{p}}_2 = (5, 0, 1)^T$, respectively. The Krylov method is inferior due to $\underline{p}_1 \neq \widehat{\underline{x}}_k$, the new values of $Q_k(\cdot)$ being $Q_k(\underline{p}_2) = -19.4$ and $Q_k(\widehat{\underline{p}}_2) = -25$. Furthermore, the Krylov method is unable to generate another step that yields the reduction $Q_k(\underline{p}_3) < Q_k(\underline{p}_2)$. Indeed, the projected gradient $\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_2) = (-4, 0, 0)^T$ is parallel to $\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_1)$, so $\mathcal{K}_2 = \mathcal{K}_1$ occurs in the definition (7.5), which is a condition for termination.

There are three excellent reasons for starting the Krylov method for the new active set $\mathcal{A}$ at $\underline{p}_1$ instead of at $\widehat{\underline{x}}_k$ when $\underline{p}_1$ is not a trust region centre. The point $\widehat{\underline{x}}_k$ may be infeasible, the value $Q_k(\underline{p}_1)$ is the least known value of $Q_k(\underline{x}), \underline{x} \in \mathcal{R}^n$, so far subject to the linear constraints and the trust region bound, and the new $\mathcal{A}$ has been chosen carefully so that a move from $\underline{p}_1$ along the direction $-\check{Q}\check{Q}^T \underline{\nabla} Q_k(\underline{p}_1)$ does not violate a constraint until the length of the move is greater than $\eta_2 \Delta_k$. Moreover, while the sequence of Krylov steps is strictly inside the trust region, the steps are suitable, because they are the same as the conjugate gradient steps in Sect. 5. When the Krylov steps move round the boundary of the trust region, however, there is the very strong objection that the definition (7.5) of the Krylov subspaces is without any attention to the actual trust region boundary, this deficiency being shown clearly by the property $\mathcal{K}_1 = \mathcal{K}_2$ in the example of the previous paragraph. Therefore, although it is argued in the first paragraph of this section that boundary searches by the Krylov method are superior to those of Sect. 6 in the case $\underline{p}_1 = \underline{x}_k$, and although this argument is valid too in the unusual situation $\underline{p}_1 = \widehat{\underline{x}}_k \neq \underline{x}_k$, we expect the crude searches of Sect. 6 to be better in the cases $\underline{p}_1 \neq \widehat{\underline{x}}_k$. We recall also that, when $\underline{p}_{\ell+1}$ is generated by the Krylov method, the uphill property $(\underline{p}_{\ell+1} - \underline{p}_\ell)^T \underline{\nabla} Q_k(\underline{p}_\ell) > 0$ is possible, which causes difficulties if $\underline{p}_{\ell+1}$ is infeasible. These disadvantages led to the rejection of the Krylov method from the LINCOA software, as mentioned earlier. Nevetheless, the description of the Krylov method with linear constraints in Sect. 4 may be useful, because, in many applications of LINCOA, most of the changes to $\mathcal{A}$ occur at the beginning of an iteration, and then $\underline{p}_1$ is at the trust region centre $\underline{x}_k$.

The choice of the quadratic model $Q_k(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, for each iteration number $k$ is important, but it is outside the range of our work. Nevertheless, because Tables 1 and 2 in Sect. 6 compare NPT $= n+6$ with NPT $= 2n+1$, we comment briefly on the number of interpolation conditions. When the author began to investigate the symmetric Broyden method for minimization without derivatives, as reported in the last three paragraphs of [8], NPT was chosen to be $\mathcal{O}(n)$ for large $n$, in order to allow the routine work of each iteration to be only $\mathcal{O}(n^2)$. Comparisons were made with NPT $= \frac{1}{2}(n+1)(n+2)$, which is the number of degrees of freedom in a quadratic function. The finding that smaller values of NPT often provide much lower values of $\#F$ was a welcome surprise. For the smaller values, the second derivative matrix $\nabla^2 Q_k$ is usually very different from $\nabla^2 F(\underline{x}_k)$ at the end of the calculation, even when $F(\cdot)$ is quadratic. It seems, therefore, that quadratic models without good accuracy can be helpful to the choice of $\underline{x}_{k+1}$. This view is supported by the following advantage of $Q_k(\cdot)$ over a linear approximation to $F(\cdot)$.

We suppose that there are no linear constraints on the variables, and that we wish to predict whether the reduction $F(\underline{x}) < F(\underline{x}_k)$ is going to occur for some $\underline{x}$ in the trust region of the $k$-th iteration. If a linear approximation $\Lambda_k(\cdot) \approx F(\cdot)$, say, is employed for the prediction, and if $\underline{\nabla}\Lambda_k(\underline{x}_k)$ is nonzero, then the answer is affirmative for all vectors $\underline{x}$ in the set $\{\underline{x} : \Lambda_k(\underline{x}) < \Lambda_k(\underline{x}_k)\} \cap \{\underline{x} : \|\underline{x} - \underline{x}_k\| \leq \Delta_k\}$, which is half of the trust region on one side of the plane $\{\underline{x} : (\underline{x} - \underline{x}_k)^T \underline{\nabla}\Lambda_k(\underline{x}_k) = 0\}$. On the other hand, a typical quadratic model $Q_k(\cdot)$ is subject to the interpolation conditions

$$Q_k(\underline{y}_i) = F(\underline{y}_i), \quad i = 1, 2, \ldots, \mathrm{NPT}, \tag{7.9}$$

with NPT $> n+1$, and we expect $\underline{x}_k$ to be a best interpolation point, which means $\underline{x}_k = \underline{y}_t$, where $t$ is an integer in [1, NPT] that satisfies $F(\underline{y}_t) \leq F(\underline{y}_i)$, $i = 1, 2, \ldots, \mathrm{NPT}$. Thus the set $\{\underline{x} : Q_k(\underline{x}) < Q_k(\underline{x}_k)\}$ is usually very different from a half plane, especially if $\underline{x}_k$ is a strictly interior point of the convex hull of the interpolation points. Indeed, the set excludes a neighbourhood of every $\underline{y}_i$ with $F(\underline{y}_i) > F(\underline{x}_k)$, and searches for relatively small values of $Q_k(\cdot)$ stay away automatically from the current interpolation points. Quadratic models with NPT of magnitude $n$ are obvious candidates for providing this useful feature. Furthermore, because symmetric Broyden updating takes up the freedom in each new model by minimizing the change to the model in a particular way, some helpful properties of the old model can be inherited by the new one, although $\nabla^2 Q_k$ may be a very bad estimate of $\nabla^2 F(\underline{x}_k)$. The author is enthusiastic about such models, because of their success in his software for optimization without derivatives when there are hundreds of variables.

Updating the quadratic model is an example of a subject that is fundamental to the development of algorithms such as LINCOA, but the subject is separate from our present work. Another fundamental subject that has not received our attention is the choice of vectors $\underline{x}$ for the calculation of new values of $F(\underline{x})$ on iterations that are designed to improve the quadratic model, instead of trying to achieve the reduction $F(\underline{x}) < F(\underline{x}_k)$. The number of these "model iterations" is about ($\#F - \#\mathrm{TRI} - \mathrm{NPT}$) in Tables 1 and 2. Therefore our paper is definitely not intended to be a description of

LINCOA, although it may be welcomed by users of LINCOA, because that description has not been written yet. Instead, we have studied the investigations for LINCOA into the construction of feasible vectors $\underline{x}$ that provide a sufficiently small value of $Q_k(\underline{x})$, $\|\underline{x} - \underline{x}_k\| \leq \Delta_k$, subject to linear constraints. Most of the efforts of those investigations, which have taken about 2 years, were spent on promising techniques that have not been included in the software. The prime example is the Krylov subspace method, which was expected to perform better than truncated conjugate gradients, due to its attractive way of taking steps round the trust region boundary in the unconstrained case. The reason for giving so much attention to failures of the Krylov method is that our findings may be helpful to future research.

**Acknowledgments** The author is very grateful to the Liu Bie Ju Centre and the Department of Mathematics at the City University of Hong Kong for excellent facilities and support, while he was investigating versions of the Krylov subspace method for the LINCOA software.

## References

1. Broyden, C.G., Dennis, J.E., Moré, J.J.: On the local and superlinear convergence of quasi-Newton methods. J. Inst. Math. Appl. **12**, 223–245 (1973)
2. Conn, A.R., Gould, N.I.M., Toint, Ph.L.: Trust-Region Methods. MPS/SIAM Series on Optimization, SIAM (Philadelphia) (2000)
3. Fletcher, R.: Practical Methods of Optimization. Wiley, Chichester (1987)
4. Gill, P.E., Murray, W.: Numerical Methods for Constrained Optimization. Academic Press, London (1974)
5. Goldfarb, D., Idnani, A.: A numerically stable dual method for solving strictly quadratic programs. Math. Program. **27**, 1–33 (1983)
6. Moré, J.J., Sorensen, D.C.: Computing a trust region step. SIAM J. Sci. Stat. Comput. **4**, 553–572 (1983)
7. Powell, M.J.D.: A tolerant algorithm for linearly constrained optimization calculations. Math. Program. B **45**, 547–566 (1989)
8. Powell, M.J.D.: On trust region methods for unconstrained minimization without derivatives. Math. Program. B **97**, 605–623 (2003)
9. Powell, M.J.D.: Least Frobenius norm updating of quadratic models that satisfy interpolation conditions. Math. Program. B **100**, 183–215 (2004)
10. Powell, M.J.D.: The NEWUOA software for unconstrained optimization without derivatives. In: Di Pillo, G., Roma, M. (eds.) Large-Scale Optimization, pp. 255–297. Springer, New York (2006)
11. Powell, M.J.D.: Beyond symmetric Broyden for updating quadratic models in minimization without derivatives. Math. Program. B **138**, 475–500 (2013)
12. Powell, M.J.D.: LINCOA. http://en.wikipedia.org/wiki/LINCOA (2013). Accessed 22 May 2015