INDEFINITELY PRECONDITIONED TRUNCATED NEWTON METHOD FOR LARGE SPARSE EQUALITY CONSTRAINED NONLINEAR PROGRAMMING PROBLEMS [1]

Ladislav Lukšan, Jan Vlček
Institute of Computer Science, Academy of Sciences of the Czech Republic,
Pod vodárenskou věží 2, 182 07 Prague 8, Czech Republic

Abstract. A truncated Newton algorithm for large sparse equality constrained nonlinear programming problems is proposed. This algorithm is based on an indefinitely preconditioned smoothed conjugate gradient method applied to the linear KKT system and uses a simple augmented Lagrangian merit function for Armijo type stepsize selection. Most attention is devoted to the termination of the CG method and, especially, to the choice of a suitable preconditioner. Termination criteria guaranteeing sufficient descent in every iteration and decreasing the number of required CG iterations are proposed in Section 2. In Section 3 we investigate four preconditioners, which have 2×2 block structure and prove theoretically their good properties. The efficiency of the truncated Newton algorithm is demonstrated in Section 4 where a comparison of various preconditioners and strategies is also presented.

Keywords. Nonlinear programming, sparse problems, equality constraints, truncated Newton method, augmented Lagrangian function, indefinite systems, indefinite preconditioners, conjugate gradient method, residual smoothing.

1. Introduction

Consider the problem of finding a point $x^* \in R^n$, such that

$$x^* = \arg\min_{x \in \mathcal{F}} F(x), \tag{1}$$

where $\mathcal{F} \subset R^n$ is a feasible set defined by the system of equations

$$\mathcal{F} = \{x \in R^n : c_k(x) = 0, 1 \le k \le m\}. \tag{2}$$

where $m \le n$ (in fact we consider only local minima). Here $F : R^n \to R$ and $c_k : R^n \to R$, $1 \le k \le m$, are twice continuously differentiable functions, whose gradients and Hessian matrices will be denoted by $\nabla F(x)$, $\nabla c_k(x)$, $1 \le k \le m$, and $\nabla^2 F(x)$, $\nabla^2 c_k(x)$, $1 \le k \le m$, respectively. Furthermore, we use the notation

[1] This work was supported under grant No. 201/96/0918 given by the Czech Republic Grant Agency

$c(x) = [c_1(x), \dots, c_m(x)]^T$ and $A(x) = [a_1(x), \dots, a_m(x)] = [\nabla c_1(x), \dots, \nabla c_m(x)]$ and we suppose that the matrix $A(x)$ has a full column rank. Then the solution $x^* \in R^n$ of the problem (1)-(2) satisfies the Karush-Kuhn-Tucker (KKT) conditions, i.e. there exists a vector $u^* \in R^m$, such that

$$\begin{aligned} \nabla_x L(x^*, u^*) &= \nabla F(x^*) + A(x^*)u^* = 0, && (3) \\ \nabla_u L(x^*, u^*) &= c(x^*) = 0, && (4) \end{aligned}$$

where

$$L(x, u) = F(x) + u^T c(x) \tag{5}$$

is the Lagrangian function, whose gradient and Hessian matrix will be denoted by

$$\begin{aligned} g(x, u) &= \nabla_x L(x, u) = \nabla F(x) + \sum_{k=1}^{m} u_k \nabla c_k(x), \\ G(x, u) &= \nabla_x^2 L(x, u) = \nabla^2 F(x) + \sum_{k=1}^{m} u_k \nabla^2 c_k(x), \end{aligned}$$

and $(x^*, u^*) \in R^{n+m}$ is the KKT pair (first order necessary conditions). Let $Z(x)$ be a matrix whose columns form an orthonormal basis in the null space of $A^T(x)$ so that $A^T(x)Z(x) = 0$ and $Z^T(x)Z(x) = I$. If, in addition to (3)-(4), the matrix $Z^T(x^*)G(x^*, u^*)Z(x^*)$ is positive definite, then the point $x^* \in R^n$ is a solution of the problem (1)-(2) (second order sufficient conditions).

Basic methods for a solution of the problem (1)-(2) are iterative and their iteration step has the form

$$\begin{aligned} x^+ &= x + \alpha d, && (6) \\ u^+ &= u + \alpha v, && (7) \end{aligned}$$

where $(d, v) \in R^{n+m}$ is a direction pair ($d \in R^n$ is a direction vector) and $\alpha > 0$ is a stepsize. In this contribution, we confine our attention to methods derived from the Newton method applied to the solution of the KKT system (3)-(4). The iteration step of the Newton method has the form (6)-(7), where $\alpha = 1$ and

$$\begin{bmatrix} G(x, u) & A(x) \\ A^T(x) & 0 \end{bmatrix} \begin{bmatrix} d \\ v \end{bmatrix} = - \begin{bmatrix} g(x, u) \\ c(x) \end{bmatrix}. \tag{8}$$

This is a system of $n + m$ linear equations with $n + m$ unknowns $(d, v) \in R^{n+m}$, whose matrix is always indefinite. Moreover, the matrix $G(x, u)$ is not positive definite in general even if the matrix $Z(x)^T G(x, u) Z(x)$ is. This fact can lead to some difficulties when standard positive definite preconditioners are used. In this case, it is advantageous to transform the linear KKT system (8) in such a way as to contain, if possible, a positive definite matrix in the left-upper corner. This can often be done by adding the second equation, multiplied by ρA, to the first equation (cf. Theorem 2), which yields

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} d \\ v \end{bmatrix} = - \begin{bmatrix} b \\ c \end{bmatrix}, \tag{9}$$

where

$$B = G + \rho A A^T \tag{10}$$

and

$$b = g + \rho A c = \nabla F + A u + \rho A c. \tag{11}$$

If the matrices A and B are large and sparse, then we can solve the system (9) either directly using the sparse Bunch-Parlett type [5] decomposition or iteratively using a preconditioned conjugate gradient type method. Another possibility consists in using a partial elimination of variables. In this case (9) can be transformed to the form

$$\begin{aligned} Bd &= -(b + Av), && (12) \\ A^T B^{-1} A v &= c - A^T B^{-1} b. && (13) \end{aligned}$$

Equation (12) can be solved directly by sparse Choleski type decomposition. We assume that the matrix $A^T B^{-1} A$, the (negative) Schur complement, is large and dense so that the equation (13) is solved iteratively, using the conjugate gradient method, and the matrix $A^T B^{-1} A$ is not built explicitly. Multiplication by this matrix is replaced by sucessive multiplication by the matrices A, B^{-1} after the sparse Choleski type decomposition and A^T.

The main purpose of this contribution is to show that the indefinitely preconditioned smoothed conjugate gradient method applied directly to the system (9) is very robust and efficient in connection with the truncated Newton method for nonlinear equality constrained optimization. Several reasons exist which explain this fact:

- If we are far from the minimum point $x^* \in R^n$, then the system (9) can be solved approximately. Therefore, only a few CG iterations are necessary in most of the Newton iterations so that good iterative methods are competitive with direct methods based on sparse Bunch-Parlet type decomposition. Moreover, direct methods have greater storage requirements and they can fail due to large fill-in.

- Multiplication by the matrix

$$K = \begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \tag{14}$$

  (i.e. successive multiplication by the matrices A, B and A^T) requires the same (or less) number of operations in comparison with the multiplication by the matrix $A^T B^{-1} A$ (i.e. sucessive multiplication by the matrices A, B^{-1} after the sparse Choleski type decomposition and A^T). Therefore, even if the system (9) has the dimension $n+m$, $n \geq m$, we require approximately the same number of operations per iteration as in the case of the system (13) which has the dimension m.

- An application of the Choleski decomposition to (12) requires positive definiteness of the matrix B. Therefore, a nonzero value of the parameter ρ has to be used frequently. On the other hand, the value of the parameter ρ influences the value of the parameter σ in the augmented Lagrangian merit function (Theorem 4), which should not be too large for a good balance between the objective function and the constraints. Therefore, the approach based on equations (12)-(13) is sensitive to the suitable choice of the value ρ. This value is not a priori known, it has to be sought carefully. In Section 4, we show that required values of the parameter ρ differ considerably depending on the problem solved and that the efficiency of the CG method applied to (13) strongly depends on them. This situation does not appear if we solve the original system (9). In this case, the indefinitely preconditioned smoothed CG method works well even if $\rho = 0$.

- The choice $\rho = 0$ has an additional advantage. If $\rho \neq 0$, then the matrix $B = G + \rho A A^T$ can have many more nonzero elements than the matrix G (e.g. if, for some $1 \leq k \leq m$, $\nabla c_k(x)$ is dense but $\nabla^2 c_k(x)$ is sparse). Even if the matrix $A^T D^{-1} A$, used in (30), can have many nonzero elements in a similar case, its dimension is less (often significantly) than the dimension of B in all practical problems.

- The smoothed CG method applied to system (9) can be preconditioned successfully by indefinite preconditioners as it was confirmed by our numerical experiments. The number of smoothed CG iterations was reduced by many times when these preconditioners were used. Such a reduction did not appear when the CG method, applied to the system (13), was preconditioned by a similar way.

The contribution is organized as follows. In Section 2, we present some results concerning direction determination based on system (9) and prove the correctness of the Armijo type line search procedure based on simple augmented Lagrangian merit function. Theorem 3 contains a new result concerning positive definiteness of the Schur complement $A^T B^{-1} A$. The new sufficient conditions guaranteeing descent are formulated in Theorem 4 and correctness of the Armijo type stepsize selection is confirmed by Theorem 5. Conditions given by Theorem 4 are applied in Algorithm 1 and they cannot be weakened without deterioration of its computational efficiency as it was indicated by our numerical experiments. Algorithm 1 is based on the smoothed conjugate gradient method which is mathematically equivalent to the MINRES method in the unpreconditioned case (see [16]). Even if the standard MINRES method is considered to be more stable, it has a disadvantage since the residual vector, required in Algorithm 1, has to be computed by an extra three term recurrence. We tested both of these approaches and found the smoothed conjugate gradient method more suitable for our purpose. In Section 3, we investigate four preconditioners related to the system (9). The indefinite preconditioners C_2 - C_4, which have a 2×2 block structure, are based on the replacement of particular matrices in the expression of K^{-1} by positive definite diagonal matrices. Most attention is devoted to the preconditioner C_4, which has nice properties demonstrated by Theorem 8 - Theorem 10. Efficiency and robustness of the preconditioner C_4 is demonstrated in Section 4 which contains a computational comparison obtained by extensive numerical experiments.

In this contribution, we denote by $\|.\|$ the Euclidean (or spectral) norm.

2. Direction determination and stepsize selection

In this section we consider the system (9) with an arbitrary value $\rho \geq 0$ even if suitable indefinite preconditioners allow us to use the value $\rho = 0$. For a given value $\rho \geq 0$ we denote by $K(\rho)$ the matrix (14) with B given by (10). The following theorems demonstrate the influence of the value ρ on properties of the matrix $K(\rho)$.

Theorem 1. *Let k_+, k_- and k_0 be the number of positive, negative and zero eigenvalues of the matrix $K(0)$ and let l_+, l_- and l_0 be the number of positive, negative and zero eigenvalues of the matrix Z^TGZ. Then $k_- = l_- + m$, $k_+ = l_+ + m$ and $k_0 = l_0$.*

Proof. See [9].

Theorem 2. *Let the matrix Z^TGZ be positive definite. Then a number $\overline{\rho} \geq 0$ exists, such that the matrix B is positive definite whenever $\rho \geq \overline{\rho}$.*

Proof. See [4].

Theorem 3. *Let the matrix $K(0)$ be nonsingular. Then a number $\overline{\rho} \geq 0$ exists, such that the matrix $A^TB^{-1}A$ is positive definite whenever $\rho \geq \overline{\rho}$.*

Proof. If G is positive definite, then $A^TG^{-1}A$ is also positive definite (A has a full rank) so that we can set $\overline{\rho} = 0$. Suppose now that G is not positive definite.
(a) First we prove that a number $\rho_0 > 0$ exists such that the matrix $G + \rho AA^T$ is nonsingular whenever $\rho \geq \rho_0$. We can deduce from Theorem 1 that nonsingularity of the matrix $K(0)$ implies nonsingularity of the matrix Z^TGZ. Therefore, a number $\underline{G} > 0$ exists such that $\|Z^TGZz\| \geq \underline{G}\|z\| \; \forall z \in R^{n-m}$. Denote $Y = A(A^TA)^{-1}$ so that $A^TY = I$, $Z^TY = 0$ and $\|Y\| \leq \overline{A}/\underline{A}^2$, where $\overline{A} = \|A\|$ and $\underline{A}$ is the lowest singular value of the matrix A. Then every vector $x \in R^n$ can be uniquely expressed in the form $x = Yy + Zz$, where $y \in R^m$ and $z \in R^{n-m}$. Suppose that $G + \rho AA^T$ is singular for some value $\rho > (\overline{GA}^2/\underline{A}^4)(1 + (\overline{G}/\underline{G}))$, where $\overline{G} = \|G\|$, so that

$$(G + \rho AA^T)x = GYy + GZz + \rho Ay = 0$$

for some nonzero vector $x \in R^n$. Then necessarily

$$Z^T GYy + Z^T GZz = 0 \tag{15}$$

and

$$y^T Y^T GYy + y^T Y^T GZz + \rho y^T y = 0. \tag{16}$$

From (15) we obtain

$$\frac{\overline{GA}}{\underline{A}^2}\|y\| \geq \|Z^T GYy\| = \|Z^T GZz\| \geq \underline{G}\|z\|$$

so that $\|z\| \leq (\overline{GA})/(\underline{GA}^2)\|y\|$ and, therefore, $\|y\| > 0$ if $\|x\| > 0$. On the other hand, we can write

$$\begin{aligned} y^T Y^T GYy + y^T Y^T GZz + \rho y^T y &\geq \rho\|y\|^2 - \frac{\overline{GA}^2}{\underline{A}^4}\|y\|^2 - \frac{\overline{GA}}{\underline{A}^2}\|y\|\|z\| \\ &\geq \left[\rho - \frac{\overline{GA}^2}{\underline{A}^4}\left(1 + \frac{\overline{G}}{\underline{G}}\right)\right]\|y\|^2 > 0, \end{aligned}$$

which is a contradiction to (16). Therefore, $G + \rho AA^T$ is nonsingular whenever $\rho \geq \rho_0 > (\overline{GA}^2/\underline{A}^4)(1 + (\overline{G}/\underline{G}))$.

(b) Denote $B_0 = G + \rho_0 AA^T$. Since the matrix B_0 is nonsingular by (a), its (negative) Schur complement $A^T B_0^{-1} A$ in the matrix $K(\rho_0)$ is also nonsingular. If $A^T B_0^{-1} A$ is positive definite, then we can set $\overline{\rho} = \rho_0$. Suppose now, that $A^T B_0^{-1} A$ has at least one negative eigenvalue. Let μ be an eigenvalue of the matrix $A^T B_0^{-1} A$ and w be a corresponding eigenvector. Then we obtain successively

$$\begin{gathered} A^T B_0^{-1} Aw = \mu w \\ B_0^{-1} AA^T B_0^{-1} Aw = \mu B_0^{-1} Aw \\ (I + (\rho - \rho_0) B_0^{-1} AA^T) B_0^{-1} Aw = (1 + (\rho - \rho_0)\mu) B_0^{-1} Aw \\ (1 + (\rho - \rho_0)\mu)^{-1} B_0^{-1} Aw = (I + (\rho - \rho_0) B_0^{-1} AA^T)^{-1} B_0^{-1} Aw \\ (1 + (\rho - \rho_0)\mu)^{-1} A^T B_0^{-1} Aw = A^T (B_0 + (\rho - \rho_0) AA^T)^{-1} Aw \\ \mu(1 + (\rho - \rho_0)\mu)^{-1} w = A^T B^{-1} Aw \end{gathered}$$

provided $\rho - \rho_0 \neq -1/\mu$. Consider the function $\lambda(\mu) = \mu/(1 + (\rho - \rho_0)\mu)$ for a given $\rho \geq \rho_0$. If $\mu > 0$, then $\lambda(\mu) > 0$ for an arbitrary $\rho \geq \rho_0$. If $\mu < -1/(\rho - \rho_0) < 0$ then again $\lambda(\mu) > 0$. Therefore, if either $\mu > 0$ or $\mu < -1/(\rho - \rho_0) < 0$ for all eigenvalues of the matrix $A^T B_0^{-1} A$, then all eigenvalues $\lambda(\mu)$ of the matrix $A^T B^{-1} A$ are positive. This situation appears if $\rho \geq \overline{\rho} > \rho_0 - 1/\mu_0$, where $\mu_0 < 0$ is the greatest negative eigenvalue of the matrix $A^T B_0^{-1} A$. □

The last theorem is a useful generalization of Theorem 2, since it only requires nonsingularity of the matrix $K(0)$. This is a very important fact for an iterative solution of the system (13) by the conjugate gradient method, since it guarantees an existence of the value $\rho \geq 0$ which makes the Schur complement $A^T B^{-1} A$ positive definite (also far from the minimum point $x^* \in R^n$ where the matrix $Z^T G Z$ may not be positive definite). Theorem 3 can also be applied to building preconditioners based on Bunch-Parlett decomposition since positive definiteness of the Schur complement used in Theorem 6 and Theorem 7 can be guaranteed.

Now we focus our attention on conditions which assure that the value of a suitable merit function is decreased when the direction vector d, obtained as an inexact solution to the system (9), is used for the stepsize selection. We have chosen the following augmented Lagrangian merit function

$$P(\alpha) = F(x + \alpha d) + (u + v)^T c(x + \alpha d) + \frac{\sigma}{2} \|c(x + \alpha d)\|^2, \tag{17}$$

with $\sigma \geq \rho$, which was preferred since it led to the most efficient and robust algorithm among many other (sometimes more sophisticated) choices we have tested. Note that the term $u + v$ in (17) is the best current approximation of the Lagrange multiplier vector, which is usually used in the next iteration (if $\alpha = 1$).

The derivative of the function (17), for $\alpha = 0$, is given by the formula

$$P'(0) \triangleq \frac{dP(\alpha)}{d\alpha}|_{\alpha=0} = d^T(b + Av) + (\sigma - \rho) d^T A c. \tag{18}$$

The following theorem gives sufficient conditions for a descent property of the direction vector.

Theorem 4. *Let $(d, v) \in R^{n+m}$ be an inexact solution to the system (9) such that*

$$\sigma - \rho \geq \frac{2\underline{B}\|d\|^2 - d^T B d}{c^T c} \tag{19}$$

and

$$d^T h + (\sigma - \rho) c^T r \leq \frac{1}{2}(d^T B d + (\sigma - \rho) c^T c) \tag{20}$$

holds, where

$$h = Bd + Av + b \tag{21}$$

and

$$r = A^T d + c \tag{22}$$

are corresponding residuals. Then $P'(0) \leq -\underline{B}\|d\|^2$.

Proof. Using (18)-(22) we get

$$\begin{aligned} P'(0) &= d^T(b + Av) + (\sigma - \rho) d^T A c = -d^T B d + d^T h + (\sigma - \rho)(c^T r - c^T c) \\ &\leq -\frac{1}{2}(d^T B d + (\sigma - \rho) c^T c) \leq -\underline{B}\|d\|^2. \end{aligned}$$

$\square$

Theorem 4 shows, that if $\sigma \geq \rho$ is sufficiently large and the system (9) is solved sufficiently accurately, then the vector $d \in R^n$ is a descent direction for the merit function (17). Besides (20), we need additional standard conditions guaranteeing superlinear rate of convergence of the inexact Newton method. These conditions should be related to the original system (8) as follows from the theory contained in [3]. Using (10)-(11) and (21)-(22), we obtain

$$Gd + Av + g = Bd + Av + b - \rho AA^T d - \rho Ac = h - \rho Ar,$$

so that (see [3]) a superlinear rate of convergence is assured when

$$\|h - \rho Ar\| \leq \omega \min(\|g\|, \overline{g}) \tag{23}$$

and

$$\|r\| \leq \omega \min(\|c\|, \overline{c}), \tag{24}$$

where $0 \leq \omega < 1$ and $\omega \leq (\|g\| + \|c\|)$ hold simultaneously. The constants $\overline{g}$ and $\overline{c}$ serve as safeguards against unboundedness and only have theoretical importance. Note that conditions (23)-(24) are considered separately since individual residuals of the system (9) can have considerably different norms.

Let $(d, v) \in R^{n+m}$ be an inexact solution to the system (9) satisfying the assumptions of Theorem 4. Then we can use the standard Armijo rule for steplength determination i.e. $\alpha > 0$ in (6)-(7) is chosen so that it is the first member of the sequence $\underline{\beta}^j$, $j = 0, 1, 2, \ldots$, $0 < \underline{\beta} < 1$, such that

$$P(\alpha) - P(0) \leq \underline{\varepsilon} \alpha P'(0), \tag{25}$$

where $0 < \underline{\varepsilon} < 1$.

In the subsequent considerations, we will assume that the matrix B is uniformly bounded and the matrix K is uniformly nonsingular, i.e. constants $\overline{B}$ and $\underline{K}$ exist, independent of the curent iteration, such that $\|B\| \leq \overline{B}$ and $\|Kw\| \geq \underline{K}\|w\|$ $\forall w \in R^{n+m}$ (the matrix B can be preliminarily modified if $\|B\| > \overline{B}$). Furthermore we will suppose that $\sigma \leq \overline{\sigma}$ and that constants $\overline{g}$, $\overline{G}$, $\overline{c}$, $\overline{A}$ exist, independent of the current iteration, such that $\|\nabla F(x + \alpha d)\| \leq \overline{g}$, $\|\nabla^2 F(x + \alpha d)\| \leq \overline{G}$, $\|c(x + \alpha d)\| \leq \overline{c}$, $\|A(x + \alpha d)\| \leq \overline{A}$, $\|\nabla^2 c_k(x + \alpha d)\| \leq \overline{G}$, $1 \leq k \leq m$, hold, respectively, for all $0 \leq \alpha \leq 1$.

Lemma 1. *Let the assumptions of Theorem 4 be satisfied (together with (23)-(24) and the assumptions of boundedness given above). Then a constant $\overline{K}$ exists, independent of the current iteration, such that*

$$P(\alpha) \leq P(0) + \alpha P'(0) + \alpha^2 \overline{K} \|d\|^2 \tag{26}$$

$\forall 0 \leq \alpha \leq 1$.

Proof. Using (23)-(24) we get

$$\|r\| \leq \overline{c}$$

and

$$\|h\| \leq \|h - \rho Ar\| + \|\rho Ar\| \leq \overline{g} + \rho\overline{A}\overline{c}.$$

Since (21)-(22) imply that

$$K\begin{bmatrix} d \\ u+v \end{bmatrix} + \begin{bmatrix} \nabla F + \rho Ac \\ c \end{bmatrix} = \begin{bmatrix} h \\ r \end{bmatrix},$$

we can write

$$\underline{K}\|u+v\| \leq \left\| K\begin{bmatrix} d \\ u+v \end{bmatrix} \right\| \leq 2(\overline{g} + \rho\overline{A}\overline{c} + \overline{c}) \triangleq \underline{K}\overline{U}$$

(ρ is assumed to be constant). Applying the Taylor expansion to (17) and using (18), we get

$$\begin{aligned} P(\alpha) &\leq P(0) + \alpha P'(0) + \frac{1}{2}\alpha^2\overline{G}\|d\|^2 + \frac{1}{2}\alpha^2\sum_{k=1}^{m}|u_k + v_k|\overline{G}\|d\|^2 \\ &\quad + \frac{1}{2}\sigma\alpha^2\overline{A}^2\|d\|^2 + \frac{1}{2}\sigma\alpha^2\sum_{k=1}^{m}|c_k|\overline{G}\|d\|^2 \\ &\leq P(0) + \alpha P'(0) + \frac{1}{2}\alpha^2\left[(1 + \overline{U}\sqrt{m} + \overline{\sigma c}\sqrt{m})\overline{G} + \sigma\overline{A}^2\right]\|d\|^2 \\ &\triangleq P(0) + \alpha P'(0) + \alpha^2\overline{K}\|d\|^2 \end{aligned}$$

$\forall 0 \leq \alpha \leq 1.$ □

Theorem 5. *Let the assumptions of Lemma 1 hold and let $d \neq 0$. Then an integer $k \geq 0$ and a number $\underline{\alpha} > 0$ exist, independent of the current iteration, such that the Armijo rule gives the value $\alpha = \underline{\beta}^j$, satisfying (25), with $j \leq k$ and $\alpha \geq \underline{\alpha}$. Moreover*

$$P(\alpha) - P(0) \leq -\underline{\alpha\varepsilon B}\|d\|^2. \tag{27}$$

Proof. Using Lemma 1 and Theorem 4, we can write

$$P(\alpha) - P(0) \leq \alpha(P'(0) + \alpha\overline{K}\|d\|^2) \leq \alpha P'(0)(1 - \alpha\frac{\overline{K}}{\underline{B}}),$$

so that (25) holds whenever $\alpha \leq (\underline{B}/\overline{K})(1 - \underline{\varepsilon})$. Let $k \geq 0$ be chosen so that it is the lowest integer such that $\underline{\beta}^k \leq (\underline{B}/\overline{K})(1 - \underline{\varepsilon})$ and let $\alpha = \underline{\beta}^j$ be given by the Armijo rule to satisfy (25). Then

$$\alpha = \underline{\beta}^j \geq \underline{\beta}^k \geq \underline{\beta}\frac{\underline{B}}{\overline{K}}(1 - \underline{\varepsilon}) \triangleq \underline{\alpha}. \tag{28}$$

Using (28) and Theorem 4, we get

$$P(\alpha) - P(0) \leq \alpha\underline{\varepsilon}P'(0) \leq -\underline{\alpha\varepsilon B}\|d\|^2.$$ □

Now we focus our attention on the inexact solution of the system (9). To simplify the notation we put

$$y = \begin{bmatrix} d \\ v \end{bmatrix}, \; z = \begin{bmatrix} b \\ c \end{bmatrix}, \; s = \begin{bmatrix} h \\ r \end{bmatrix},$$

i.e. the system (9) will be written in the form $s = Ky + z = 0$. To solve this system, we use the smoothed conjugate gradient method preconditioned by the matrix C. The resulting algorithm is based on the following philosophy. Step 2 realizes the basic preconditioned conjugate gradient method. In Step 3, the residual vector is smoothed. Step 4 is devoted to testing a required accuracy. It serves as a switch for using additional decisions. In Step 5, we compute the penalty parameter satisfying the conditions (19). Step 6 contains a condition for descent (20) as a termination criterion.

Algorithm 1. Direction determination.

Data: $\rho \geq 0$, $0 < \underline{\sigma} < \overline{\sigma}$, $\underline{B} > 0$, $0 < \omega < 1$, $\overline{c} > 0$, $\overline{g} > 0$.

Step 1: Initiation. Set $\tilde{y}_0 := 0$, $\tilde{s}_0 := z$, $y_0 := \tilde{y}_0$, $s_0 := \tilde{s}_0$, $\omega := \min(\omega, \|s_0\|)$, and $j := 0$.

Step 2: CG iteration. If $j \geq n + m + 3$, then go to Step 6, otherwise set $j := j + 1$. Compute $\tilde{p}_{j-1} := C^{-1}\tilde{s}_{j-1}$ and $\beta_{j-1} := \tilde{p}_{j-1}^T \tilde{s}_{j-1}$. If $j = 1$, then set $p_{j-1} := -\tilde{p}_{j-1}$, otherwise set $p_{j-1} := -\tilde{p}_{j-1} + (\beta_{j-1}/\beta_{j-2})p_{j-2}$. Compute $q_{j-1} := Kp_{j-1}$ and $\gamma_{j-1} := \beta_{j-1}/p_{j-1}^T q_{j-1}$ and set $\tilde{y}_j := \tilde{y}_{j-1} + \gamma_{j-1}p_{j-1}$, $\tilde{s}_j := \tilde{s}_{j-1} + \gamma_{j-1}q_{j-1}$.

Step 3: Residual smoothing. Compute $\lambda_j := -(s_{j-1} - \tilde{s}_j)^T \tilde{s}_j / \|s_{j-1} - \tilde{s}_j\|^2$ and set $y_j := \tilde{y}_j + \lambda_j(y_{j-1} - \tilde{y}_j)$, $s_j := \tilde{s}_j + \lambda_j(s_{j-1} - \tilde{s}_j)$.

Step 4: Test for sufficient precision. If $\|r_j\| > \omega \min(\|c\|, \overline{c})$, then go to Step 2. Otherwise compute $w_j := h_j - \rho A r_j$. If $\|w_j\| > \omega \min(\|g\|, \overline{g})$, then go to Step 2.

Step 5: Determination of the penalty parameter. Compute the value $\kappa_j := d_j^T B d_j$ and set $\sigma_j := \min(\overline{\sigma}, \max(\underline{\sigma}, \rho, \rho + (2\underline{B}\|d\|^2 - \kappa_j)/\|c\|^2))$. Set $\kappa_j := \kappa_j + (\sigma_j - \rho)\|c\|^2$.

Step 6: Test for sufficient descent. Set $\mu_j = d_j^T h_j + (\sigma_j - \rho)c^T r_j$. If $\mu_j > \kappa_j/2$, then go to Step 2. Otherwise set $y := y_j$, $\sigma := \sigma_j$, $P'(0) := \mu_j - \kappa_j$ and terminate the computation.

Note that the main reasons for residual smoothing in Step 3 are conditions (23)-(24) which require the corresponding norms to be as small as possible.

3. Preconditioning

The main purpose of the preconditioner C is to change a spectrum of the matrix K to obtain more clustered eigenvalues, which usually leads to acceleration of convergence. Ideally, the matrix C should be as close to the matrix K as possible but a multiplication by the matrix C^{-1} should not be expensive. If the matrix B is nonsingular, then we can write

$$K^{-1} = \begin{bmatrix} B^{-1} - B^{-1}A(A^TB^{-1}A)^{-1}A^TB^{-1} & B^{-1}A(A^TB^{-1}A)^{-1} \\ (A^TB^{-1}A)^{-1}A^TB^{-1} & -(A^TB^{-1}A)^{-1} \end{bmatrix}, \tag{29}$$

so that the matrix C^{-1} should have a similar structure but it should be realized by sparse decompositions. Two basic possibilities exist. First, a positive definite preconditioner can be used. In this case, the matrix KC^{-1} can be transformed to a symmetric matrix by using similarity transformations. In such a way we obtain symmetric system and we can use standard estimates for the covergence rate of both the CG and the MINRES methods. Unfortunately, indefiniteness of the resulting system brings some difficulties (as positive as negative eigenvalues exist so that two intervals bounded by four "extremal" eigenvalues have to be considered). Moreover, the convergence rate often results rather slow. Secondly, we can utilize indefinite preconditioner which assures that all eigenvalues of the matrix KC^{-1} are positive. Unfortunately, this matrix is not symmetrizable by similarity transformations in general so that standard estimates for the convergence rate of the CG method cannot be used. Nevertheless, some theoretical results are presented in this section, which explain good properties of the selected indefinite preconditioners.

We concentrate our attention on four preconditioners C_1 - C_4 which have a 2×2 block structure. The preconditioners C_1 - C_2 are intended for systems with positive definite Hessian matrices while the preconditioners C_3 - C_4 are more general (C_1 is always positive definite, C_2 can either be positive definite or indefinite and C_3 - C_4 are always indefinite). In this section we assume that the matrix K is nonsingular and, therefore, the matrix A has a full column rank.

1) Block diagonal positive definite preconditioner

$$C_1 = \begin{bmatrix} LL^T & 0 \\ 0 & D_1 \end{bmatrix},$$

where LL^T is either complete or incomplete sparse Choleski decomposition of the matrix $B + E$ (with the possible Gill-Murray [8] correction E) and D_1 is a positive definite diagonal matrix. In this case

$$C_1^{-1} = \begin{bmatrix} (LL^T)^{-1} & 0 \\ 0 & D_1^{-1} \end{bmatrix}.$$

The matrix D_1 is advantageous for scaling the second group of equations in (9). If these equations are well-scaled, then $D_1 = \varepsilon I$ is a suitable choice, where $\varepsilon > 0$

serves as a balance between the first and the second groups of equations in (9). The preconditioner C_1 with the choice $D_1 = \varepsilon I$ is studied in [15], where more details (e.g. estimates for extremal eigenvalues of KC_1^{-1}) can be found. In our computational experiments, the value $\varepsilon = 1$ (i.e. $D_1 = I$) appeared to be suitable (other values gave worse results).

2) An indefinite preconditioner based on a diagonal approximation of the Schur complement

$$C_2 = \begin{bmatrix} LL^T & A \\ A^T & A^T(LL^T)^{-1}A - D_2 \end{bmatrix},$$

where LL^T is either complete or incomplete sparse Choleski decomposition of the matrix $B+E$ (with the possible Gill-Murray [8] correction E) and D_2 is a positive definite diagonal matrix. Then

$$C_2^{-1} = \begin{bmatrix} (LL^T)^{-1} - (LL^T)^{-1}AD_2^{-1}A^T(LL^T)^{-1} & (LL^T)^{-1}AD_2^{-1} \\ D_2^{-1}A^T(LL^T)^{-1} & -D_2^{-1} \end{bmatrix}.$$

If $LL^T = B$ (i.e. if LL^T is a complete Choleski decomposition of the positive definite matrix B), then

$$C_2^{-1} = \begin{bmatrix} B^{-1} - B^{-1}AD_2^{-1}A^TB^{-1} & B^{-1}AD_2^{-1} \\ D_2^{-1}A^TB^{-1} & -D_2^{-1} \end{bmatrix},$$

which arises from (29) after replacing the Schur complement $A^TB^{-1}A$ by the diagonal matrix D_2. Moreover

$$KC_2^{-1} = \begin{bmatrix} I & 0 \\ (I-M_2)A^TB^{-1} & M_2 \end{bmatrix},$$

where $M_2 = A^TB^{-1}AD_2^{-1}$.

Theorem 6. *Consider the preconditioner C_2 with $LL^T = B$ (nonsingular) and D_2 positive definite. Then the matrix KC_2^{-1} has at least n unit eigenvalues and a full system of n linearly independent eigenvectors corresponding to these eigenvalues exists. Other eigenvalues of the matrix KC_2^{-1} are positive and if $v^TD_2v = v^TA^TB^{-1}Av$ for some $v \in R^m$ (e.g. if D_2 and $A^TB^{-1}A$ have at least one common diagonal element), then all eigenvalues of the matrix KC_2^{-1} are contained in the interval determined by the extremal eigenvalues of the matrix $A^TB^{-1}AD_2^{-1}$.*

Proof. The matrix KC_2^{-1} is lower block triangular so that it has at least n unit eigenvalues corresponding to the first diagonal block I. Since

$$\begin{bmatrix} I & 0 \\ (I-M_2)A^TB^{-1} & M_2 \end{bmatrix} \begin{bmatrix} e_j \\ A^TB^{-1}e_j \end{bmatrix} = \begin{bmatrix} e_j \\ A^TB^{-1}e_j \end{bmatrix},$$

holds for n linearly independent vectors $e_j \in R^n$, $1 \le j \le n$, then a full system of n linearly independent eigenvectors corresponding to the unit eigenvalues exists. Other eigenvalues of the matrix KC_2^{-1} are eigenvalues of the second diagonal block $M_2 = A^T B^{-1} A D_2^{-1}$. These eigenvalues are positive since the matrices B and D_2 are assumed to be positive definite (so that $A^T B^{-1} A$ is also positive definite). If $v^T D_2 v = v^T A^T B^{-1} A v$, then a unit lies in the interval determined by the (positive) extremal eigenvalues of the matrix $A^T B^{-1} A D_2^{-1}$. Therefore, all eigenvalues of the matrix KC_2^{-1} are contained in the interval determined by the extremal eigenvalues of the matrix $A^T B^{-1} A D_2^{-1}$. $\square$

It is clear that the preconditioner C_2 should be very efficient especially when D_2 is a good approximation of $A^T B^{-1} A$. We tested the choice $D_2 = diag(A^T B^{-1} A)$ which works well when it is measured by the total number of CG iterations. Unfortunately, computation of the matrix $D_2 = diag(A^T B^{-1} A)$ is time consuming (it requires m back substitutions) so that it is impractical in the case of large scale problems. Therefore, we have to use simpler diagonal approximations of the Schur complement. Relatively good results were obtained by the choice $D_2 = \varepsilon I$, especially if $\varepsilon = 1$ (i.e. $D_2 = I$). Notice that we can also use the choice $D_2 = -\varepsilon I$, which leads to a positive definite preconditioner. This positive definite preconditioner is studied in [15], where more details (e.g. estimates for extremal eigenvalues of KC_2^{-1}) can be found. We have tested both of these possibilities. The indefinite preconditioner with $D_2 = \varepsilon I$ appeared more efficient.

3) An indefinite preconditioner based on a diagonal perturbation of the Schur complement

$$C_3^{-1} = \begin{bmatrix} (LL^T)^{-1} & (LL^T)^{-1} A D_3^{-1} \\ D_3^{-1} A^T (LL^T)^{-1} & D_3^{-1} A^T (LL^T)^{-1} A D_3^{-1} - D_3^{-1} \end{bmatrix},$$

where LL^T is either complete or incomplete sparse Choleski decomposition of the matrix $B + A D_3^{-1} A^T + E$ (with the possible Gill-Murray [8] correction E) and D_3 is a positive definite diagonal matrix. If $LL^T = B + A D_3^{-1} A^T$ (i.e. if LL^T is a complete Choleski decomposition of the positive definite matrix $B + A D_3^{-1} A^T$), then

$$C_3 = \begin{bmatrix} B & A \\ A^T & -D_3 \end{bmatrix}.$$

and

$$C_3^{-1} = \begin{bmatrix} B^{-1} - B^{-1} A (A^T B^{-1} A + D_3)^{-1} A^T B^{-1} & B^{-1} A (A^T B^{-1} A + D_3)^{-1} \\ (A^T B^{-1} A + D_3)^{-1} A^T B^{-1} & -(A^T B^{-1} A + D_3)^{-1} \end{bmatrix},$$

which arises from (29) after replacing the Schur complement $A^T B^{-1} A$ by the matrix $A^T B^{-1} A + D_3$. Moreover

$$KC_3^{-1} = \begin{bmatrix} I & 0 \\ (I - M_3) A^T B^{-1} & M_3 \end{bmatrix},$$

where $M_3 = A^T B^{-1} A (A^T B^{-1} A + D_3)^{-1}$.

Theorem 7. *Consider the preconditioner C_3 with $LL^T = B + AD_3^{-1}A^T$ (nonsingular) and D_3 positive definite. Then the matrix KC_3^{-1} has at least n unit eigenvalues and a full system of n linearly independent eigenvectors corresponding to these eigenvalues exists. If the matrix $A^TB^{-1}A$ is positive definite, then all eigenvalues of the matrix KC_3^{-1} satisfy the inequality $0 < \underline{\lambda}(A^TB^{-1}A)/(\underline{\lambda}(A^TB^{-1}A)+\|D_3\|) \le \lambda \le 1$, where $\underline{\lambda}(A^TB^{-1}A)$ is the lowest eigenvalue of the matrix $A^TB^{-1}A$.*

Proof. The first part of the theorem can be proved in the same way as that of Theorem 6. Let the matrix $A^TB^{-1}A$ be positive definite and λ be an eigenvalue of the matrix $A^TB^{-1}A(A^TB^{-1}A + D_3)^{-1}$. Then a vector $\tilde{y} \in R^m$ exists such that $\lambda = \tilde{y}^TA^TB^{-1}A\tilde{y}/(\tilde{y}^TA^TB^{-1}A\tilde{y} + \tilde{y}^TD_3\tilde{y})$. This expression implies $0 < \underline{\lambda}(A^TB^{-1}A)/(\underline{\lambda}(A^TB^{-1}A) + \|D_3\|) \le \lambda \le 1$. □

The preconditioner C_3 with the choice $D_3 = \varepsilon I$ was introduced in [1]. Our computational experiences confirmed a good efficiency of this preconditioner even if it is not as robust as the preconditioner C_4.

4) An indefinite preconditioner based on a diagonal approximation of the Hessian matrix

$$C_4^{-1} = \begin{bmatrix} D_4^{-1} - D_4^{-1}A(R^TR)^{-1}A^TD_4^{-1} & D_4^{-1}A(R^TR)^{-1} \\ (R^TR)^{-1}A^TD_4^{-1} & -(R^TR)^{-1} \end{bmatrix}, \tag{30}$$

where R^TR is either complete or an incomplete sparse Choleski decomposition of the matrix $A^TD_4^{-1}A$ and D_4 is a positive definite diagonal matrix. If $R^TR = A^TD_4^{-1}A$ (i.e. if R^TR is a complete Choleski decomposition of the matrix $A^TD_4^{-1}A$), then

$$C_4 = \begin{bmatrix} D_4 & A \\ A^T & 0 \end{bmatrix}.$$

and

$$C_4^{-1} = \begin{bmatrix} D_4^{-1} - D_4^{-1}A(A^TD_4^{-1}A)^{-1}A^TD_4^{-1} & D_4^{-1}A(A^TD_4^{-1}A)^{-1} \\ (A^TD_4^{-1}A)^{-1}A^TD_4^{-1} & -(A^TD_4^{-1}A)^{-1} \end{bmatrix},$$

which arises from (29) after replacing the matrix B by the diagonal matrix D_4. Moreover,

$$KC_4^{-1} = \begin{bmatrix} I + M_4P_4 & M_4A(A^TD_4^{-1}A)^{-1} \\ 0 & I \end{bmatrix}, \tag{31}$$

where $M_4 = BD_4^{-1} - I$ and $P_4 = I - A(A^TD_4^{-1}A)^{-1}A^TD_4^{-1}$ so that $P_4A = 0$.

The preconditioner C_4 was used in [7]. In the subsequent text, we propose new results, which explain its excellent efficiency.

Theorem 8. *Consider the preconditioner C_4 with $R^TR = A^TD_4^{-1}A$ (nonsingular) and D_4 positive definite. Then the matrix KC_4^{-1} has at least $2m$ unit eigenvalues but, if M_4 is nonsingular, only m linearly independent eigenvectors corresponding to these eigenvalues exist. Other eigenvalues of the matrix KC_4^{-1} are exactly eigenvalues of the matrix $Z^TBZ(Z^TD_4Z)^{-1}$. If Z^TBZ is positive definite, then all* *eigenvalues of the matrix KC_4^{-1} are positive. If, moreover, $v^TZ^TD_4Zv = v^TZ^TBZv$ for some $v \in R^n$, then all eigenvalues of the matrix KC_4^{-1} are contained in the interval determined by the extremal eigenvalues of the matrix $Z^TBZ(Z^TD_4Z)^{-1}$.*

Proof. The matrix KC_4^{-1} is upper block triangular so that it has at least m unit eigenvalues corresponding to the second diagonal block I. Other eigenvalues of the matrix KC_4^{-1} are eigenvalues of the first diagonal block $I + M_4P_4$. Therefore, for such eigenvalue λ, we can write

$$(I + M_4P_4)x = \lambda x, \tag{32}$$

which is an eigenvalue problem of the dimension n. Since the matrix A has a full column rank with the assumption, then m linearly independent vectors of the form $x = Au$ exist, for which $P_4x = 0$, i.e. which are eigenvectors of the matrix $I + M_4P_4$ corresponding to the unit eigenvalue. This implies that the original problem has at least $2m$ unit eigenvalues. Now suppose that

$$\begin{bmatrix} I + M_4P_4 & M_4A(A^TD_4^{-1}A)^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

(i.e. $[x^T, y^T]^T$ is an eigenvector corresponding to the unit eigenvalue). Then necessarily $P_4x + A(A^TD_4^{-1}A)^{-1}y = 0$ (if M_4 is nonsingular) so that $x = Au$ for some $u \in R^m$. Therefore, $P_4x = 0$, which gives $A(A^TD_4^{-1}A)^{-1}y = 0$. But A has a full column rank and $A^TD_4^{-1}A$ is nonsingular so that $y = 0$. Thus we obtain $[x^T, y^T]^T = [(Au)^T, 0]^T$ which gives exactly m linearly independent eigenvectors. Now suppose that $\lambda \neq 1$ in (32). Let us write $x = Au + D_4Zv$. This expression is unique since $Au + D_4Zv = 0$ implies $A^TD_4^{-1}Au = 0$ and $Z^TD_4Zv = 0$ so that $u = 0$ and $v = 0$. But

$$\begin{aligned} M_4P_4x &= M_4P_4(Au + D_4Zv) = M_4D_4Zv \\ &= (BD_4^{-1} - I)D_4Zv = BZv - D_4Zv \end{aligned}$$

and (32) implies

$$M_4P_4x = (\lambda - 1)(Au + D_4Zv).$$

These equalities together give $Z^TBZv = \lambda Z^TD_4Zv$, i.e. λ is an eigenvalue of the matrix $Z^TBZ(Z^TD_4Z)^{-1}$. If Z^TBZ is positive definite, then, since Z^TD_4Z is also positive definite, all eigenvalues of the matrices $Z^TBZ(Z^TD_4Z)^{-1}$ and KC^{-1} are positive. The last part of the theorem can be proved in the same way as that of Theorem 6. □

Theorem 8 explains good properties of the preconditioner C_4. First, since $Z^TBZ = Z^T(G + \rho AA^T)Z = Z^TGZ$, then eigenvalues of the matrix KC_4^{-1} do not depend on the parameter ρ. This explains a low efficiency dependence of the preconditioner C_4 on the parameter ρ, which was confirmed by numerical experiments (Table 4b). Secondly, since Z^TGZ is usually positive definite in the neighborhood of the solution point $x^\star \in R^n$ (second order sufficient conditions), then all eigenvalues of KC_4^{-1} are usually positive. On the other hand, the matrix KC_4^{-1} does not have a full system of linearly independent eigenvectors. This is a consequence of indefiniteness of the matrix C_4 which does not have a square root. For this reason, we cannot apply the standard estimates to the convergence rate of the conjugate gradient method. Nevertheless, the conjugate gradient method with the preconditioner C_4 terminates after at most $n-m+2$ iterations.

Theorem 9. *Consider the conjugate gradient method with the preconditioner C_4, where $R^TR = A^TD_4^{-1}A$ is nonsingular and D_4 is positive definite, applied to the system $Ky = z$. If a breakdown does not occur, then we obtain the solution $y^\star \in R^{n+m}$ after of at most $n-m+2$ iterations.*

Proof. We refer to Algorithm 1 for the notation. (a) It is a well-known fact that the vector $C_4y^\star$ lies in the Krylov space $\mathcal{K} = \text{span}\{\tilde{s}_0, KC_4^{-1}\tilde{s}_0, (KC_4^{-1})^2\tilde{s}_0, \ldots\}$ and that the preconditioned conjugate gradient method generates the vectors $C_4\tilde{y}_i \in \mathcal{K}$, $\tilde{s}_i \in \mathcal{K}$, $0 \le i \le k$, where k is a dimension of $\mathcal{K}$. If a breakdown does not occur, then the vectors $\tilde{s}_i$, $0 \le i \le k-1$, are linearly independent (so that they form a basis in $\mathcal{K}$) and $\tilde{s}_i^TC_4^{-1}\tilde{s}_k = 0$ for all $0 \le i \le k-1$. Let $s \in \mathcal{K}$, so that also $C_4y \in \mathcal{K}$. Consider the quadratic function

$$s^TK^{-1}s = [C_4(y-y^\star)]^TC_4^{-1}KC_4^{-1}[C_4(y-y^\star)] = (\sum_{j=0}^{k-1}\alpha_j\tilde{s}_j)^TC_4^{-1}KC_4^{-1}(\sum_{j=0}^{k-1}\alpha_j\tilde{s}_j)$$

defined on $\mathcal{K}$. A stationary point of this quadratic function is characterized by the following conditions

$$\tilde{s}_i^TC_4^{-1}KC_4^{-1}\left(\sum_{j=0}^{k-1}\alpha_j\tilde{s}_j\right) = \tilde{s}_i^TC_4^{-1}s = 0, \quad 0 \le i \le k-1.$$

These conditions are fulfiled by the vector $\tilde{s}_k \in \mathcal{K}$ obtained by the conjugate gradient method and also by the vector $s^\star = 0$. If a breakdown does not occur, then the above system has an unique solution so that $\tilde{y}_k = y^\star$. Therefore $y^\star$ is obtained after of at most k iterations.
(b) Now we prove by induction that $\mathcal{K}$ has a dimension of at most $n-m+2$. For this purpose we use the expressions

$$\tilde{h}_0 = \sum_{i=1}^m \alpha_i a_i + \sum_{j=1}^{n-m}\beta_j^0 w_j$$

$$M_4A(A^TD_4^{-1}A)^{-1}\tilde{r}_0 = \sum_{i=1}^m \tilde{\alpha}_i a_i + \sum_{j=1}^{n-m}\tilde{\beta}_j w_j$$

(see Theorem 8), where $a_i \in R^n$, $1 \le i \le m$, are columns of the matrix A (eigenvectors corresponding to unit eigenvalues of the matrix $I + M_4P_4$) and $w_j \in R^n$,

$1 \leq j \leq n-m$, are basic vectors in the invariant subspace corresponding to other eigenvalues of the matrix $I + M_4 P_4$. Suppose now that

$$(KC_4^{-1})^k \tilde{s}_0 = \begin{bmatrix} \sum_{i=1}^m \alpha_i a_i \\ \tilde{r}_0 \end{bmatrix} + k \begin{bmatrix} \sum_{i=1}^m \tilde{\alpha}_i a_i \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{j=1}^{n-m} \beta_j^k w_j \\ 0 \end{bmatrix}$$

for some $k \geq 0$ (it is obvious for $k = 0$). After premultiplying this expression by the matrix KC_4^{-1} we can write

$$\begin{aligned}(KC_4^{-1})^{k+1} \tilde{s}_0 &= \begin{bmatrix} \sum_{i=1}^m \alpha_i a_i \\ \tilde{r}_0 \end{bmatrix} + k \begin{bmatrix} \sum_{i=1}^m \tilde{\alpha}_i a_i \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{j=1}^{n-m} \gamma_j^k w_j \\ 0 \end{bmatrix} + \\ &\quad \begin{bmatrix} \sum_{i=1}^m \tilde{\alpha}_i a_i \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{j=1}^{n-m} \tilde{\beta}_j w_j \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \sum_{i=1}^m \alpha_i a_i \\ \tilde{r}_0 \end{bmatrix} + (k+1) \begin{bmatrix} \sum_{i=1}^m \tilde{\alpha}_i a_i \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{j=1}^{n-m} \beta_j^{k+1} w_j \\ 0 \end{bmatrix}\end{aligned}$$

where γ_j^k, $1 \leq j \leq n-m$, are new coordinates in the invariant subspace spanned by the vectors $w_j \in R^n$, $1 \leq j \leq n-m$, and $\beta_j^{k+1} = \gamma_j^k + \tilde{\beta}_j$, $1 \leq j \leq n-m$. Thus we have proved that all the vectors $\tilde{s}_0, KC_4^{-1}\tilde{s}_0, (KC_4^{-1})^2\tilde{s}_0, \ldots$ are combinations of $n-m+2$ vectors

$$\begin{bmatrix} \sum_{i=1}^m \alpha_i a_i \\ \tilde{r}_0 \end{bmatrix}, \quad \begin{bmatrix} \sum_{i=1}^m \tilde{\alpha}_i a_i \\ 0 \end{bmatrix}, \quad \begin{bmatrix} w_j \\ 0 \end{bmatrix} \quad 1 \leq j \leq n-m.$$

so that the Krylov space $\mathcal{K}$ has a dimension of at most $n-m+2$. □

A very interesting result can be obtained in a special case when $c = 0$ in (9). This case occurs if an active set strategy is used for linearly constrained problems (see [4]). Another important application of this case concerns a solution to the Navier-Stokes problem (see [10]).

Theorem 10. *Consider the conjugate gradient method with the preconditioner C_4, where $R^T R = A^T D_4^{-1} A$ is nonsingular and D_4 is positive definite, applied to the system (9), where $c = 0$. Let the matrix $Z^T B Z$ be positive definite. Then the conjugate gradient method finds the vector $d^\star \in R^n$ after of at most $n-m$ iterations and for $1 \leq i \leq n-m$ the following estimation holds*

$$\|\tilde{d}_i - d^\star\| \leq 2\sqrt{\kappa} \left(\frac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}} \right)^i \|\tilde{d}_0 - d^\star\|, \tag{33}$$

where $\kappa = \kappa(Z^T B Z (Z^T D_4 Z)^{-1})$.

Proof. We refer to Algorithm 1 for the notation. (a) Since $\tilde{y}_0 = 0$, we can write

$$\tilde{s}_0 = \begin{bmatrix} b \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{h}_0 \\ 0 \end{bmatrix} = \begin{bmatrix} D_4 Z (Z^T D_4 Z)^{-1} \hat{s}_0 + A \check{s}_0 \\ 0 \end{bmatrix}$$

and

$$p_0 = -C_4^{-1}\tilde{s}_0 = -\begin{bmatrix} Z(Z^T D_4 Z)^{-1}\hat{s}_0 \\ \check{s}_0 \end{bmatrix} \triangleq \begin{bmatrix} Z\hat{p}_0 \\ \check{p}_0 \end{bmatrix}$$

($\hat{s}_0$ and $\check{s}_0$ are determined uniquely as it is pointed out in the proof of Theorem 8). Suppose now that

$$\tilde{s}_i = \begin{bmatrix} \tilde{h}_i \\ 0 \end{bmatrix} = \begin{bmatrix} D_4 Z(Z^T D_4 Z)^{-1}\hat{s}_i + A\check{s}_i \\ 0 \end{bmatrix}, \quad p_i = \begin{bmatrix} Z\hat{p}_i \\ \check{p}_i \end{bmatrix}$$

for some $i \geq 0$ (it is obvious for $i = 0$). Using these expressions we obtain

$$Kp_i = \begin{bmatrix} BZ\hat{p}_i + A\check{p}_i \\ 0 \end{bmatrix}$$

and

$$\begin{aligned} \tilde{s}_i^T C_4^{-1}\tilde{s}_i &= \hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i, \\ p_i^T K p_i &= \hat{p}_i^T Z^T B Z \hat{p}_i \end{aligned}$$

after some manipulations. Therefore

$$\tilde{s}_{i+1} = \tilde{s}_i + \frac{\hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i}{\hat{p}_i^T Z^T B Z\hat{p}_i} K p_i = \begin{bmatrix} \tilde{h}_{i+1} \\ 0 \end{bmatrix},$$

where

$$\begin{aligned} \tilde{h}_{i+1} &= D_4 Z(Z^T D_4 Z)^{-1}\hat{s}_i + A\check{s}_i + \frac{\hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i}{\hat{p}_i^T Z^T B Z\hat{p}_i}(BZ\hat{p}_i + A\check{p}_i) \\ &= D_4 Z(Z^T D_4 Z)^{-1}\hat{s}_{i+1} + A\check{s}_{i+1} \end{aligned}$$

so that

$$\hat{s}_{i+1} = Z^T \tilde{h}_{i+1} = \hat{s}_i + \frac{\hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i}{\hat{p}_i^T Z^T B Z\hat{p}_i} Z^T B Z\hat{p}_i.$$

Similarly, we obtain

$$p_{i+1} = -C_4^{-1}\tilde{s}_{i+1} + \frac{\hat{s}_{i+1}^T (Z^T D_4 Z)^{-1}\hat{s}_{i+1}}{\hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i} p_i = \begin{bmatrix} Z\hat{p}_{i+1} \\ \check{p}_{i+1} \end{bmatrix},$$

so that

$$\hat{p}_{i+1} = -(Z^T D_4 Z)^{-1}\hat{s}_{i+1} + \frac{\hat{s}_{i+1}^T (Z^T D_4 Z)^{-1}\hat{s}_{i+1}}{\hat{s}_i^T (Z^T D_4 Z)^{-1}\hat{s}_i}\hat{p}_i.$$

In this way, we have proved by induction that $\hat{s}_i$, $1 \le i \le n-m$ are residual vectors obtained by the conjugate gradient method with the preconditioner $Z^T D_4 Z$ applied to the system

$$Z^T B Z \hat{d} = Z^T b. \tag{34}$$

(b) Using (9) we can write

$$\begin{bmatrix} B & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} \tilde{d}_i - d^\star \\ \tilde{v}_i - v^\star \end{bmatrix} = \begin{bmatrix} D_4 Z (Z^T D_4 Z)^{-1} \hat{s}_i + A \check{s}_i \\ 0 \end{bmatrix}$$

so that $A^T(\tilde{d}_i - d^\star) = 0$ (second equation) and, therefore, $\tilde{d}_i - d^\star = Z(\hat{d}_i - \hat{d}^\star)$ for some $\hat{d}_i \in R^{n-m}$. On the other hand, $Z^T B Z(\hat{d}_i - \hat{d}^\star) = \hat{s}_i$ (first equation premultiplied by Z^T), i.e. $\hat{d}_i - \hat{d}^\star$ is the vector obtained by the conjugate gradient method with the preconditioner $Z^T D_4 Z$ applied to the system (34). This in turn implies that $\tilde{d}_{n-m} = d^\star$, since $\hat{d}_{n-m} = \hat{d}^\star$ (we suppose without a loss of generality that the conjugate gradient method terminates after exactly $n-m$ iterations), and that the estimation (33) holds, since $\|\tilde{d}_i - d^\star\|^2 = (\hat{d}_i - \hat{d}^\star)^T Z^T Z (\hat{d}_i - \hat{d}^\star) = \|\hat{d}_i - \hat{d}^\star\|^2$. $\square$

Theorem 10 gives very strong results, but it cannot immediately be applied to Algorithm 1. Although $\tilde{d}_{n-m} = d^\star$, the equality $\tilde{v}_{n-m} = v^\star$ may not hold. The solution would be obtained in the next iteration (since $\mathcal{K}$ has a dimension of at most $n-m+1$ when $\tilde{r}_0 = 0$), but $\hat{s}_{n-m} = 0$ implies breakdown (since $\tilde{s}_{n-m}^T C_4^{-1} \tilde{s}_{n-m} = \hat{s}_{n-m}^T (Z^T D Z)^{-1} \hat{s}_{n-m} = 0$). This effect is very unusual, since if $c = 0$, then breakdown appears almost always (the cases where $\tilde{v}_{n-m} = v^\star$ are exceptional). Moreover, if this kind of breakdown appears, then a solution can easily be obtained by using the following procedure. Since $\tilde{d}_{n-m} - d^\star = 0$, we can write $A(\tilde{v}_{n-m} - v^\star) = \tilde{h}_{n-m}$ so that

$$A^T D_4^{-1} A(\tilde{v}_{n-m} - v^\star) = A^T D_4^{-1} \tilde{h}_{n-m}. \tag{35}$$

Having the decomposition $R^T R = A^T D_4^{-1} A$ available, we simply set $v_{n-m} = v^\star = \tilde{v}_{n-m} - (R^T R)^{-1} A^T D_4^{-1} \tilde{h}_{n-m}$ as it follows from the equation (35). Notice that $d_{n-m} = \tilde{d}_{n-m}$ does usually not hold so that we have to set $d_{n-m} = \tilde{d}_{n-m}$ in the algorithm (residual smoothing has to be removed if a breakdown occurs). The above considerations imply that only a "happy" breakdown leading to the solution can occur. At preceeding iterations, a breakdown cannot occur since the system (34) has a positive definite matrix and the preconditioner $Z^T D_4 Z$ is also positive definite.

The above effect really occured in our computations. When we solved the first problem from the Appendix (where $n - m = 2$), we usually obtained an acceptable solution in the fourth iteration. If we used $\|c\| = 10^{-8}$, then an approximate solution was found in the third iteration, but an acceptable one in the sixth iteration (due to roundoff errors implied by a near breakdown). For $c = 0$, the original CG method did not converge, but when we used the procedure based on the equation (35), then we obtained an acceptable solution already in the second iteration.

We have examined the indefinite preconditioner C_4 with various diagonal matrices $D_4 = diag\{D_{ii}\}$. The best results were obtained with the choice

$$\begin{aligned} D_{ii} &= \underline{\Delta}, \quad \text{if} \quad |B_{ii}| < \underline{\Delta}, \\ D_{ii} &= |B_{ii}|, \quad \text{if} \quad \underline{\Delta} \le |B_{ii}| \le \overline{\Delta}, \\ D_{ii} &= \overline{\Delta}, \quad \text{if} \quad \overline{\Delta} < |B_{ii}|, \end{aligned} \tag{36}$$

where $\underline{\Delta} = 10^{-3}$ and $\overline{\Delta} = 10^{6}$.

4. Numerical experiments

Now we summarize results from the previous sections and give a detailed description of an algorithm for equality constrained nonlinear programming problems. This algorithm applies a preconditioned smoothed conjugate gradient method to the system (9) for direction determination and uses the classical Armijo rule for stepsize selection. Before the algorithm is described, we have to note that it is advantageous to use a restart procedure, which treats cases when the matrix K is unsuitable for direction determination. These cases are characterized by a large value of the derivative $P'(0)$. For this purpose we set $\tau = 10^{-4}$, if $\sigma = \max(\underline{\sigma}, \rho)$, or $\tau = 10^{-1}$, otherwise. If $-P'(0) < \tau \|d\| \|g\|$, then we repeat the computation of the direction pair by using the diagonal matrix $\tilde{B} = diag\{\tilde{B}_{ii}\}$ instead of B, where

$$\begin{aligned} \tilde{B}_{ii} &= \underline{\Gamma}, \quad \text{if} \quad \frac{\|g\|}{10}|B_{ii}| < \underline{\Gamma}, \\ \tilde{B}_{ii} &= \frac{\|g\|}{10}|B_{ii}|, \quad \text{if} \quad \underline{\Gamma} \le \frac{\|g\|}{10}|B_{ii}| \le \overline{\Gamma}, \\ \tilde{B}_{ii} &= \overline{\Gamma}, \quad \text{if} \quad \overline{\Gamma} < \frac{\|g\|}{10}|B_{ii}|, \end{aligned} \tag{37}$$

where $\underline{\Gamma} = 0.005$ and $\overline{\Gamma} = 500.0$. This procedure was obtained experimentally.

Algorithm 2. Equality constrained optimization (CG).

Data: $\rho \ge 0$, $0 < \underline{\sigma} < \overline{\sigma}$, $\underline{B} > 0$, $\overline{B} > 0$, $0 < \overline{\omega} < 1$, $0 < \underline{\beta} < 1$, $0 < \underline{\varepsilon} < 1$, $\overline{\delta} > 0$, $\overline{c} > 0$, $\overline{g} > 0$.

Input: Sparsity pattern of the matrices $\nabla^2 F$ and A. Initial choice of the vector x.

Step 1: Initiation. Determine sparsity pattern of the matrix B. Select a suitable preconditioning technique based on either complete or incomplete sparse Choleski decomposition. In the complete case, carry out the symbolic decomposition. Compute the value $F := F(x)$ and the vector $c := c(x)$. Set $u := 0$ and $i := 0$.

Step 2: Termination. Compute the matrix $A := A(x)$ and the vector $g := g(x, u)$. If $\|c\| \le \overline{\delta}$ and $\|g\| \le \overline{\delta}$, then terminate the computation (the solution is found). Otherwise set $i = i + 1$.

Step 3: Approximation of the Hessian matrix. Compute an approximation G of the Hessian matrix $G(x, u)$, by using the differences of gradient $g(x, u)$ as in [2]. Compute the matrix $B := G + \rho AA^T$ and modify it when its elements are larger than $\overline{B}$.

Step 4: Direction determination. Carry out the numerical sparse Choleski decomposition to obtain the inverse preconditioner C^{-1}. Set $\omega = \min(\|g\| + \|c\|, 1/i, \overline{\omega})$. Determine the direction pair (d, v) and the derivative $P'(0)$ by using Algorithm 1.

Step 5: Restart. Set $\tau = 10^{-4}$ if $\sigma = \max(\underline{\sigma}, \rho)$, or $\tau = 10^{-1}$, otherwise. If $-P'(0) < \tau\|d\|\|g\|$, then determine the diagonal matrix $\tilde{B}$ by (37), set $B := \tilde{B}$ and go to Step 4. Otherwise set $\alpha := 1$ and compute the value of the merit function $P(\alpha)$.

Step 6: Termination of the stepsize selection. If $P(\alpha) - P(0) \le \underline{\varepsilon}\alpha P'(0)$, then set $x := x + \alpha d$, $u := u + \alpha v$ and go to Step 2.

Step 7: Continuation of the stepsize selection. Set $\alpha := \underline{\beta}\alpha$, compute the value of the merit function $P(\alpha)$ and go to Step 6.

The computational efficiency of Algorithm 2 was tested by using 18 sparse problems, given in the Appendix. Parameters of these problems are shown in Table 1 where n is the number of variables, m is the number of constraints, $n(A)$ is the number of nonzero elements in A, $n(B)$ is the number of nonzero elements in B and $n(K)$ is the number of nonzero elements in K.

Problem	n	m	$n(A)$	$n(B)$	$n(K)$
1	1000	998	2994	2997	8985
2	1000	993	6951	6975	20877
3	1000	2	4	2498	2506
4	1000	998	2994	2997	8985
5	1000	996	4980	4990	14950
6	999	499	1497	6972	9966
7	1000	4	14	1012	1040
8	1000	998	2994	3597	9585
9	1000	6	30	1524	1584
10	1000	998	2994	2997	8985
11	998	664	1992	2991	6975
12	997	747	1992	2740	6724
13	998	664	2656	3987	9299
14	998	664	1992	3322	7306
15	997	747	2241	2740	7222
16	997	747	1743	2491	5977
17	997	747	1743	2491	5977
18	997	747	1743	2491	5977

Table 1

We chose parameters $\underline{\sigma} = 1.5$, $\overline{\sigma} = 10^{16}$, $\underline{B} = 10^{-16}$, $\overline{B} = \overline{c} = \overline{g} = 10^{60}$, $\underline{\beta} = 0.5$, $\underline{\varepsilon} = 10^{-4}$, $\overline{\omega} = 0.9$, $\overline{\delta} = 10^{-6}$, in all numerical experiments. Values of the parameter ρ depended on the method and the preconditioner used. The value $\rho = 0$ was utilized advantageously when indefinite preconditioners C_3 and C_4 were applied. In all other cases, suitable nonzero values of the parameter ρ had to be sought in order to give sufficiently good results.

For comparison, we tested the fully iterative method represented by Algorithm 2 (CG) either unpreconditioned or preconditioned by various preconditioners as shown in Table 2. Furthermore, we tested the direct method based on sparse Bunch-Parlett decomposition (BP) applied to the system (9) and realized by the subroutine MA27 [6]. Finally, we tested the range-space method based on equations (12)-(13) (GM+CG) (see [12] for more details). The first equation was solved directly by the complete sparse Choleski decomposition $B + E = LL^T$ (with the Gill-Murray [8] correction E) and the second equation was solved iteratively by the smoothed conjugate gradient method. The smoothed conjugate gradient method was realized either as unpreconditioned or preconditioned by the matrices $C_5 = A^TA(LL^T)^{-1}A^TA$ (see [14]) and $C_6 = R^TR$ with R^TR defined as in the previous section (notation is given in Table 2).

Notation	preconditioner	decomposition
P0	none	none
P1	C_1 with $D_1 = \varepsilon I$, $\varepsilon = 1$	complete
P1a	C_1 with $D_1 = \varepsilon I$, $\varepsilon = 1$	incomplete
P2	C_2 with $D_2 = \varepsilon I$, $\varepsilon = 1$	complete
P2a	C_2 with $D_2 = \varepsilon I$, $\varepsilon = 1$	incomplete
P3	C_3 with $D_3 = \varepsilon I$, $\rho = 0$	complete
P3a	C_3 with $D_3 = \varepsilon I$, $\rho = 0$	incomplete
P4	C_4 with (36), $\rho = 0$	complete
P4a	C_4 with (36), $\rho = 0$	incomplete
P5	C_5 applied to (13)	complete
P5a	C_5 applied to (13)	incomplete
P6	C_6 with (36) applied to (13)	complete
P6a	C_6 with (36) applied to (13)	incomplete

Table 2

Unfortunately, most of methods did not solve all 18 problems. The direct BP method failed for Problem 6 (enormous fill-in) and Problem 13 (nonconvergence). Unpreconditioned iterative methods as well as the preconditioners C_1, C_2, C_3 failed for Problem 8 (enormous number of CG iterations) and the preconditioners C_1, C_2 could not be used for Problem 13 (nonconvergence). The summary of results corresponding to methods that solved all 18 problems is given in Table 3. This table contains the total number of the Newton iterations NIT, the total number of function evaluations NFV, the total number of gradient evaluations NGR, the total number of conjugate gradient iterations NCG, the total number of restarts NRS and the total CPU time (in minutes) on a Pentium PC (166 MHz) computer in a double precision arithmetic implementation.

Method	NIT	NFV	NGR	NCG	NRS	CPU time
CG/P4	310	350	2310	985	17	1:03.11
CG/P4a	308	348	2298	1020	17	1:11.24
GM+CG/P5	386	461	2779	2632	0	2:04.35
GM+CG/P5a	384	450	2766	2599	0	2:05.83
GM+CG/P6	359	422	2599	4146	0	1:57.26
GM+CG/P6a	349	411	2530	3954	0	1:57.16

Table 3

For a better comparison of all of the methods mentioned, we introduce three additional tables. These tables contain results obtained by using 15 sparse problems (problems 6, 8, 13 were excluded). The rows that belong to the upper parts of the tables contain the total number of conjugate gradient iterations NCG (or matrix decompositions NDC) together with suitable values of the parameter ρ (or ε) that correspond to individual problems. The rows that belong to the lower parts of the tables contain summary values (denoted as in Table 3). Moreover MB denotes the total storage requirements in megabytes (in our implementation).

Method	CG/P0		CG/P1		CG/P1a		CG/P2		CG/P2a	
Problem	NCG	ρ	NCG	ρ	NCG	ρ	NCG	ρ	NCG	ρ
1	3284	0.01	31	50	28	50	21	50	26	50
2	495	0	56	50	56	50	53	10	52	50
3	114	0	24	50	35	50	15	50	30	50
4	958	0.1	87	50	155	50	74	50	80	50
5	3590	0	107	0.5	424	0.8	53	1	234	2
7	3255	0.1	1047	100	1047	100	54	10	53	10
9	720	0.01	342	0.1	425	0.1	282	0.01	350	0.0005
10	176	0	578	0.01	197	0.01	69	1	52	10
11	237	0	26	50	148	50	16	50	131	10
12	3925	100	257	50	1080	50	151	100	601	500
14	468	0	372	1	1097	1	57	50	57	40
15	1159	0	846	1000	2890	1000	28	100	28	100
16	677	0	144	5	151	5	45	10	75	10
17	313	0	294	0.5	178	0.5	43	10	43	10
18	154	0	54	10	49	10	33	10	61	10
NIT	227		231		265		208		220	
NFV	294		353		511		265		280	
NGR	1556		1566		1823		1426		1544	
NCG	19525		4265		7960		994		1873	
CPU	3:06.41		1:13.99		2:13.96		0:46.46		1:11.67	
MB	0.500		0.584		0.508		0.584		0.508	

Table 4a

Method	CG/P3		CG/P3a		CG/P4		CG/P4a		BP	
problem	NCG	$1/\varepsilon$	NCG	$1/\varepsilon$	NCG	ρ	NCG	ρ	NDC	ρ
1	7	500	7	500	17	0	17	0	6	0
2	18	500	15	500	44	0	44	0	9	0
3	13	100	14	100	26	0	26	0	9	0
4	41	50	34	500	40	0	40	0	21	1
5	34	5	47	5	44	0	44	0	20	0.1
7	114	1	71	10	30	0	30	0	18	0
9	330	1	166	1	217	0	217	0	30	0.001
10	34	500	18	500	27	0	27	0	12	0
11	9	500	94	500	32	0	45	0	9	0
12	202	50	563	50	21	0	40	0	6	0
14	65	100	34	100	30	0	30	0	11	0
15	25	100	39	100	95	0	87	0	40	0
16	11	500	10	500	10	0	10	0	8	0
17	11	500	11	500	13	0	13	0	8	0
18	14	10	14	10	15	0	15	0	9	0
NIT	221		194		212		210		213	
NFV	298		225		248		244		265	
NGR	1509		1332		1468		1456		1436	
NCG	928		1137		661		685		0	
CPU	0:39.32		0:50.25		0:38.12		0:42.45		0:35.60	
MB	0.584		0.508		0.772		0.596		1.200	

Table 4b

Method	GM+CG/P0		GM+CG/P5		GM+CG/P5a		GM+CG/P6		GM+CG/P6a	
Problem	NCG	ρ	NCG	ρ	NCG	ρ	NCG	ρ	NCG	ρ
1	11	50	8	50	8	50	12	50	12	50
2	34	50	13	50	13	50	46	50	46	50
3	15	0	15	0	15	0	15	0	15	0
4	48	50	25	50	25	50	203	80	203	80
5	20	50	18	50	18	50	141	4	141	4
7	20	50	16	50	16	50	20	50	20	50
9	230	0	244	0	225	0	228	0	234	0
10	33	1	21	1	21	1	42	1	42	1
11	10	50	12	50	20	50	75	50	51	50
12	122	50	14	50	38	50	70	50	69	50
14	16	20	19	20	19	20	92	5	111	5
15	34	10	160	15	153	15	123	10	133	10
16	25	5000	18	5000	18	5000	448	1000	448	1000
17	14	100	9	100	9	100	27	100	27	100
18	19	100	15	100	15	100	33	100	33	100
NIT	220		233		230		231		232	
NFV	243		308		296		294		294	
NGR	1504		1580		1560		1575		1583	
NCG	651		607		613		1575		1585	
CPU	0:34.05		0:45.26		0:48.72		0:53.83		0:57:78	
MB	0.536		0.800		0.624		0.800		0.624	

Table 4c

Results presented in the above tables imply several conclusions:

- A very important observation is the fact that the indefinitely preconditioned CG method, applied directly to the system (9), is competitive with the methods BP and GM+CG (unpreconditioned) which use direct solvers. The methods BP and GM+CG are slightly more efficient for the restricted set of 15 test problems, but the BP method has greater storage requirements and the GM+CG method requires suitable values of the parameter ρ which have to be carefully sought. When we uniformly chose $\rho = 0$ for the restricted set of 15 test problems, then the GM+CG method led to 4 failures with NCG=67149. Similarly, when we uniformly chose $\rho = 50$, then the GM+CG method led to 2 failures with NCG=29418.

- The positive definite preconditioner C_1 slightly improves the efficiency of the smoothed CG method applied to the system (9), but this improvement strongly depends on the choice of the parameter ρ. The indefinite preconditioners C_2 and C_3 are more suitable but, again, the efficiency of C_2 strongly depends on the choice of the parameter ρ and C_3 is sensitive to the choice of the value ϵ. The indefinite preconditioner C_4 is the most efficient. It decreases the total number of CG iterations about 25 times. This number is comparable with the number of unpreconditioned CG iterations required for solving the smaller system (13). Moreover, no significant improvement was reached by the preconditioners C_5 and C_6, applied to the system (13).

- The smoothed conjugate gradient method with the indefinite preconditioner C_4 applied to the system (9) is very robust in connection with the nonlinear equality constrained optimization. It never failed on the complete set of 18 sparse test problems and, moreover, it required the lowest computational time in this case. A very useful property of this method is the fact, that it is not sensitive to ill conditioning or indefiniteness of the matrix G and, therefore, it works well with the choice $\rho = 0$. Possible improvement, reached by the optimal choice of the parameter ρ, is not substantial. Another good property of the preconditioner C_4 is the fact that it does not require accurate decomposition of the matrix $A^T D_4^{-1} A$ (it is very efficient even if the incomplete Choleski decomposition is used).

All methods whose results are presented in the above tables were implemented by using the modular interactive system for universal functional optimization UFO [13].

References

[1] O.Axelsson: Preconditioning of indefinite problems by regularization. SIAM J. Numerical Analysis 16 (1979) 58-69.

[2] A.R.Curtis, M.J.D.Powell, J.K.Reid: On the estimation of sparse Jacobian matrices. IMA Journal of Aplied Mathematics 13 (1974) 117-119.

[3] R.S.Dembo, S.C.Eisenstat, T.Steihaug: Inexact Newton methods. SIAM J. Numerical Analysis 19 (1982) 400-408.

[4] R.Fletcher: Practical methods of optimization. Wiley, Chichester, 1987.

[5] I.S.Duff, M.Munksgaard, H.B.Nielsen, J.K.Reid: Direct solution of sets of linear equations whose matrix is sparse and indefinite. J. Inst. Maths. Applics. 23 (1979) 235-250.

[6] I.S.Duff, J.K.Reid: MA27 - A set of Fortran subroutines for solving sparse symmetric sets of linear equations. Tech. Report AERE R10533, HMSO, London, 1982.

[7] R.E.Ewing, R.D.Lazarov, P. Lu, P.S.Vassilevski: Preconditioning indefinite systems arising from mixed finite element discretization of second-order elliptic problems. Lecture Notes in Mathematics No. 1457, 28-43, Springer Verlag, Berlin 1990.

[8] P.E.Gill, W.Murray: Newton type methods for nconstrained and linearly constrained optimization. Mathematical Programming 7 (1974) 311-350.

[9] N.I.M.Gould: On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. Mathematical Programming 32 (1985) 90-99.

[10] A.Ramage, A.J.Wathen: Iterative solution techniques for the Stokes and Navier-Stokes equarions. International J. for Numerical Methods in Fluids 19 (1994) 67-83.

[11] W.Hock, K.Schitkowski: Test Examples for Nonlinear Programming Codes. Lecture Notes in Economics and Mathematical Systems, Vol. 187. Springer-Verlag, Berlin, 1981.

[12] L.Lukšan, J.Vlček: Efficient algorithm for large sparse equality constrained nonlinear programming problem. Research Report No. V-652, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague, Czech Republic, 1996.

[13] L.Lukšan, M.Šiška, M.Tůma, J.Vlček, N.Ramešová: Interactive System for Universal Functional Optimization (UFO), Version 1996. Research Report No. V-701, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague, Czech Republic, 1996.

[14] S.G.Nash, A.Sofer: Preconditioning reduced matrices. SIAM J. Matrix Anal. Appl. 17 (1996) 47-68.

[15] P.S.Vassilevski, D.Lazarov: Preconditioning mixed finite element saddle-point elliptic problems. Numerical Linear Algebra with Applications 3 (1996) 1-20.

[16] R.Weiss: Parameter-free iterative linear solvers. Akademie Verlag, Berlin, 1996.

Appendix

This Appendix contains 18 original sparse problems for equality constrained optimization. We use, for prime k and l, the notation $\mathrm{div}(k,l)$ for integer division, i.e., maximum integer not greater than k/l, and $\mathrm{mod}(k,l)$ for remainder after integer division, i.e., $\mathrm{mod}(k,l) = l(k/l - \mathrm{div}(k,l))$. The starting point is $\bar{x}$. Dense problems HS46 - HS53 can be found in [11].

Problem 1. Chained Rosenbrock function with trigonometric-exponential constrains.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n-1}[100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2] \\
c_k(x) &= 3x_{k+1}^3 + 2x_{k+2} - 5 + \sin(x_{k+1} - x_{k+2})\sin(x_{k+1} + x_{k+2}) + 4x_{k+1} \\
&- x_k \exp(x_k - x_{k+1}) - 3
\end{aligned}
$$

$1 \le k \le m = n - 2$

$\overline{x}_i = -1.2, \quad \mathrm{mod}(i,2) = 1$

$\overline{x}_i = \quad 1.0, \quad \mathrm{mod}(i,2) = 0$

Problem 2. Chained Wood function with Broyden banded constraints.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n/2}[100(x_{2i-1}^2 - x_{2i})^2 + (x_{2i-1} - 1)^2 + 90(x_{2i+1}^2 - x_{2i+2})^2 + (x_{2i+1} - 1)^2 \\
&+ 10(x_{2i} + x_{2i+2} - 2)^2 + (x_{2i} - x_{2i+2})^2/10] \\
c_k(x) &= (2 + 5x_{k+5}^2)x_{k+5} + 1 + \sum_{i=k-5}^{k+1} x_i(1 + x_i)
\end{aligned}
$$

$1 \le k \le m = n - 7$

$\overline{x}_i = -2, \quad \mathrm{mod}(i,2) = 1$

$\overline{x}_i = \quad 1, \quad \mathrm{mod}(i,2) = 0$

Problem 3. Chained Powell singular function with simplified trigonometric-exponential constraints.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n/2}[(x_{2i-1} + 10x_{2i})^2 + 5(x_{2i+1} - x_{2i+2})^2 + (x_{2i} - 2x_{2i+1})^4 + 10(x_{2i-1} - x_{2i+2})^4] \\
c_1(x) &= 3x_1^3 + 2x_2 - 5 + \sin(x_1 - x_2)\sin(x_1 + x_2) \\
c_2(x) &= 4x_n - x_{n-1}\exp(x_{n-1} - x_n) - 3
\end{aligned}
$$

$\overline{x}_i = \quad 3, \quad \mathrm{mod}(i,4) = 1$

$$\overline{x}_i = -1, \quad \mathrm{mod}(i,4) = 2$$
$$\overline{x}_i = \ \ 0, \quad \mathrm{mod}(i,4) = 3$$
$$\overline{x}_i = \ \ 1, \quad \mathrm{mod}(i,4) = 0$$

Problem 4. Chained Cragg-Levy function with tridiagonal constraints.

$$\begin{aligned}
F(x) &= \sum_{i=1}^{n/2}[(\exp(x_{2i-1}) - x_{2i})^4 + 100(x_{2i} - x_{2i+1})^6 + \tan^4(x_{2i+1} - x_{2i+2}) + x_{2i-1}^8 \\
&+ (x_{2i+2} - 1)^2] \\
c_k(x) &= 8x_{k+1}(x_{k+1}^2 - x_k) - 2(1 - x_{k+1}) + 4(x_{k+1} - x_{k+2}^2)
\end{aligned}$$

$$1 \leq k \leq m = n-2$$

$$\overline{x}_i = 1, \quad \mathrm{mod}(i,4) = 1$$
$$\overline{x}_i = 2, \quad \mathrm{mod}(i,4) \neq 1$$

Problem 5. Generalized Broyden tridiagonal function with five diagonal constraints.

$$\begin{aligned}
F(x) &= \sum_{i=1}^{n} |(3 - 2x_i)x_i - x_{i-1} - x_{i+1} + 1|^p \\
c_k(x) &= 8x_{k+2}(x_{k+2}^2 - x_{k+1}) - 2(1 - x_{k+2}) + 4(x_{k+2} - x_{k+3}^2) + x_{k+1}^2 - x_k \\
&+ x_{k+3} - x_{k+4}^2
\end{aligned}$$

$$p = 7/3, \quad x_0 = x_{n+1} = 0, \quad 1 \leq k \leq m = n-4$$

$$\overline{x}_i = -1, \quad \forall i$$

Problem 6. Generalized Broyden banded function with exponential constraints.

$$\begin{aligned}
F(x) &= \sum_{i=1}^{n} |(2 + 5x_i^2)x_i + 1 + \sum_{j=\max(1,i-5)}^{\min(n,i+1)} x_j(1 + x_j)|^p \\
c_k(x) &= 4x_{2k} - (x_{2k-1} - x_{2k+1})\exp(x_{2k-1} - x_{2k} - x_{2k+1}) - 3
\end{aligned}$$

$$p = 7/3, \quad 1 \leq k \leq m = n/2$$

$$\overline{x}_i = 3, \quad \forall i$$

Problem 7. Trigonometric tridiagonal function with simplified five-diagonal constraints.

$$\begin{aligned}
F(x) &= \sum_{i=1}^{n} |n + i(1 - \cos x_i) - \sin x_{i+1} + \sin x_{i-1}| \\
c_1(x) &= 4(x_1 - x_2^2) + x_2 - x_3^2
\end{aligned}$$

$$
\begin{aligned}
c_2(x) &= 8x_2(x_2^2 - x_1) - 2(1 - x_2) + 4(x_2 - x_3^2) + x_3 - x_4^2 \\
c_3(x) &= 8x_{n-1}(x_{n-1}^2 - x_{n-2}) - 2(1 - x_{n-1}) + 4(x_{n-1} - x_n^2) + x_{n-2}^2 - x_{n-3} \\
c_4(x) &= 8x_n(x_n^2 - x_{n-1}) - 2(1 - x_n) + x_{n-1}^2 - x_{n-2}
\end{aligned}
$$

$\overline{x}_i = 1, \ \forall i$

Problem 8. Augmented Lagrangian function with discrete boundary value constraints.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n/5}[\exp(\prod_{j=1}^{5} x_{5i+1-j}) + 10((\sum_{j=1}^{5} x_{5i+1-j}^2 - 10 - \lambda_1)^2 \\
&+ (x_{5i-3}x_{5i-2} - 5x_{5i-1}x_{5i} - \lambda_2)^2 + (x_{5i-4}^3 + x_{5i-3}^3 + 1 - \lambda_3)^2)] \\
c_k(x) &= 2x_{k+1} + h^2(x_{k+1} + h(k+1) + 1)^3/2 - x_k - x_{k+2}
\end{aligned}
$$

$\lambda_1 = -0.002008, \ \lambda_2 = -0.001900, \ \lambda_3 = -0.000261, \ h = 1/(n+1), \ 1 \le k \le m = n-2$

$\overline{x}_i = -1, \ \mathrm{mod}(i,2) = 1$

$\overline{x}_i = \ \ 2, \ \mathrm{mod}(i,2) = 0$

Problem 9. Modified Brown function with simplified seven-diagonal constraints.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n/2}[(x_{2i-1} - 3)^2/1000 - (x_{2i-1} - x_{2i}) + \exp(20(x_{2i-1} - x_{2i}))] \\
c_1(x) &= 4(x_1 - x_2^2) + x_2 - x_3^2 + x_3 - x_4^2 \\
c_2(x) &= 8x_2(x_2^2 - x_1) - 2(1 - x_2) + 4(x_2 - x_3^2) + x_1^2 + x_3 - x_4^2 + x_4 - x_5^2 \\
c_3(x) &= 8x_3(x_3^2 - x_2) - 2(1 - x_3) + 4(x_3 - x_4^2) + x_2^2 - x_1 + x_4 - x_5^2 + x_1^2 + x_5 - x_6^2 \\
c_4(x) &= 8x_{n-2}(x_{n-2}^2 - x_{n-3}) - 2(1 - x_{n-2}) + 4(x_{n-2} - x_{k+1}^2) + x_{n-3}^2 - x_{n-4} \\
&+ x_{n-1} - x_n^2 + x_{n-4}^2 + x_n - x_{n-5} \\
c_5(x) &= 8x_{n-1}(x_{n-1}^2 - x_{n-2}) - 2(1 - x_{n-1}) + 4(x_{n-1} - x_n^2) + x_{n-2}^2 - x_{n-3} \\
&+ x_n + x_{k-2}^2 - x_{k-3} \\
c_6(x) &= 8x_n(x_n^2 - x_{n-1}) - 2(1 - x_n) + x_{n-1}^2 - x_{n-2} + x_{n-2}^2 - x_{n-3}
\end{aligned}
$$

$\overline{x}_i = -1, \ \forall i$

Problem 10. Generalized Brown function with Broyden tridiagonal constraints.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{n/2}[(x_{2i-1}^2)^{(x_{2i}^2+1)} + (x_{2i}^2)^{(x_{2i-1}^2+1)}] \\
c_k(x) &= (3 - 2x_{k+1})x_{k+1} + 1 - x_k - 2x_{k+2}
\end{aligned}
$$

$1 \le k \le m = n-2$

$\overline{x}_i = -1, \quad \mathrm{mod}(i,2) = 1$

$\overline{x}_i = \quad 1, \quad \mathrm{mod}(i,2) = 0$

Problem 11. Chained HS46 problem.

$$\begin{aligned} F(x) &= \sum_{i=1}^{(n-2)/3} [(x_{j+1} - x_{j+2})^2 + (x_{j+3} - 1)^2 + (x_{j+4} - 1)^4 + (x_{j+5} - 1)^6] \\ c_k(x) &= x_{l+1}^2 x_{l+4} + \sin(x_{l+4} - x_{l+5}) - 1 \quad , \mathrm{mod}\ (k,2) = 1 \\ c_k(x) &= x_{l+2} + x_{l+3}^4 x_{l+4}^2 - 2 \quad , \mathrm{mod}(k,2) = 0 \end{aligned}$$

$j = 3(i-1), \;\; l = 3\ \mathrm{div}(k-1,2), \;\; 1 \le k \le m = 2(n-2)/3$

$\overline{x}_i = \quad 2.0, \quad \mathrm{mod}(i,3) = 1$

$\overline{x}_i = \quad 1.5, \quad \mathrm{mod}(i,3) = 2$

$\overline{x}_i = \quad 0.5, \quad \mathrm{mod}(i,3) = 0$

Problem 12. Chained HS47 problem.

$$\begin{aligned} F(x) &= \sum_{i=1}^{(n-1)/4} [(x_{j+1} - x_{j+2})^2 + (x_{j+2} - x_{j+3})^2 + (x_{j+3} - x_{j+4})^4 + (x_{j+4} - x_{j+5})^4] \\ c_k(x) &= x_{l+1} + x_{l+2}^2 + x_{l+3}^2 - 3 \quad , \mathrm{mod}(k,3) = 1 \\ c_k(x) &= x_{l+2} + x_{l+3}^2 + x_{l+4} - 1 \quad , \mathrm{mod}(k,3) = 2 \\ c_k(x) &= x_{l+1} x_{l+5} - 1 \quad , \mathrm{mod}(k,3) = 0 \end{aligned}$$

$j = 4(i-1), \;\; l = 4\ \mathrm{div}(k-1,3), \;\; 1 \le k \le m = 3(n-1)/4$

$\overline{x}_i = \quad 2.0, \quad \mathrm{mod}(i,4) = 1$

$\overline{x}_i = \quad 1.5, \quad \mathrm{mod}(i,4) = 2$

$\overline{x}_i = -1.0, \quad \mathrm{mod}(i,4) = 3$

$\overline{x}_i = \quad 0.5, \quad \mathrm{mod}(i,4) = 0$

Problem 13. Chained modified HS48 problem.

$$\begin{aligned} F(x) &= \sum_{i=1}^{(n-2)/3} [(x_{j+1} - 1)^2 + (x_{j+2} - x_{j+3})^2 + (x_{j+4} - x_{j+5})^4] \\ c_k(x) &= x_{l+1} + x_{l+2}^2 + x_{l+3} + x_{l+4} + x_{l+5} - 5 \quad , \mathrm{mod}(k,2) = 1 \\ c_k(x) &= x_{l+3}^2 - 2(x_{l+4} + x_{l+5}) - 3 \quad , \mathrm{mod}(k,3) = 0 \end{aligned}$$

$j = 3(i-1), \;\; l = 3\ \mathrm{div}(k-1,2), \;\; 1 \le k \le m = 2(n-2)/3$

$\overline{x}_i = 3.0, \quad \mathrm{mod}(i,3) = 1$

$\overline{x}_i = 5.0, \quad \mathrm{mod}(i,3) = 2$

$\overline{x}_i = -3.0, \quad \mathrm{mod}(i,3) = 0$

Problem 14. Chained modified HS49 problem.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{(n-2)/3} [(x_{j+1} - x_{j+2})^2 + (x_{j+3} - 1)^2 + (x_{j+4} - 1)^4 + (x_{j+5} - 1)^6] \\
c_k(x) &= x_{l+1}^2 + x_{l+2} + x_{l+3} + 4x_{l+4} - 7 \quad , \mathrm{mod}(k,2) = 1 \\
c_k(x) &= x_{l+3}^2 - 5x_{l+5} - 6 \quad , \mathrm{mod}(k,3) = 0
\end{aligned}
$$

$j = 3(i-1), \quad l = 3\,\mathrm{div}(k-1,2), \quad 1 \le k \le m = 2(n-2)/3$

$\overline{x}_i = 10., \quad \mathrm{mod}(i,3) = 1$

$\overline{x}_i = 7.0, \quad \mathrm{mod}(i,3) = 2$

$\overline{x}_i = -3.0, \quad \mathrm{mod}(i,3) = 0$

Problem 15. Chained modified HS50 problem.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{(n-1)/4} [(x_{j+1} - x_{j+2})^2 + (x_{j+2} - x_{j+3})^2 + (x_{j+3} - x_{j+4})^4 + (x_{j+4} - x_{j+5})^4] \\
c_k(x) &= x_{l+1}^2 + 2x_{l+2} + 3x_{l+3} - 6 \quad , \mathrm{mod}(k,3) = 1 \\
c_k(x) &= x_{l+2}^2 + 2x_{l+3} + 3x_{l+4} - 6 \quad , \mathrm{mod}(k,3) = 2 \\
c_k(x) &= x_{l+3}^2 + 2x_{l+4} + 3x_{l+5} - 6 \quad , \mathrm{mod}(k,3) = 0
\end{aligned}
$$

$j = 4(i-1), \quad l = 4\,\mathrm{div}(k-1,3), \quad 1 \le k \le m = 3(n-1)/4$

$\overline{x}_i = 35., \quad \mathrm{mod}(i,4) = 1$

$\overline{x}_i = 11., \quad \mathrm{mod}(i,4) = 2$

$\overline{x}_i = 5.0, \quad \mathrm{mod}(i,4) = 3$

$\overline{x}_i = -5.0, \quad \mathrm{mod}(i,4) = 0$

Problem 16. Chained modified HS51 problem.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{(n-1)/4} [(x_{j+1} - x_{j+2})^4 + (x_{j+2} + x_{j+3} - 2)^2 + (x_{j+4} - 1)^2 + (x_{j+5} - 1)^2] \\
c_k(x) &= x_{l+1}^2 + 3x_{l+2} - 4 \quad , \mathrm{mod}(k,3) = 1
\end{aligned}
$$

$$
\begin{aligned}
c_k(x) &= x_{l+3}^2 + x_{l+4} - 2x_{l+5} &&, \bmod(k,3) = 2 \\
c_k(x) &= x_{l+2}^2 - x_{l+5} &&, \bmod(k,3) = 0
\end{aligned}
$$

$$j = 4(i-1), \quad l = 4\,\mathrm{div}(k-1,3), \quad 1 \le k \le m = 3(n-1)/4$$

$$
\begin{aligned}
\overline{x}_i &= 2.5, \quad \bmod(i,4) = 1 \\
\overline{x}_i &= 0.5, \quad \bmod(i,4) = 2 \\
\overline{x}_i &= 2.0, \quad \bmod(i,4) = 3 \\
\overline{x}_i &= -1.0, \quad \bmod(i,4) = 0
\end{aligned}
$$

Problem 17. Chained modified HS52 problem.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{(n-1)/4} \left[(4x_{j+1} - x_{j+2})^2 + (x_{j+2} + x_{j+3} - 2)^4 + (x_{j+4} - 1)^2 + (x_{j+5} - 1)^2\right] \\
c_k(x) &= x_{l+1}^2 + 3x_{l+2} &&, \bmod(k,3) = 1 \\
c_k(x) &= x_{l+3}^2 + x_{l+4} - 2x_{l+5} &&, \bmod(k,3) = 2 \\
c_k(x) &= x_{l+2}^2 - x_{l+5} &&, \bmod(k,3) = 0
\end{aligned}
$$

$$j = 4(i-1), \quad l = 4\,\mathrm{div}(k-1,3), \quad 1 \le k \le m = 3(n-1)/4$$

$$\overline{x}_i = 2, \quad \forall i$$

Problem 18. Chained modified HS53 problem.

$$
\begin{aligned}
F(x) &= \sum_{i=1}^{(n-1)/4} \left[(x_{j+1} - x_{j+2})^4 + (x_{j+2} + x_{j+3} - 2)^2 + (x_{j+4} - 1)^2 + (x_{j+5} - 1)^2\right] \\
c_k(x) &= x_{l+1}^2 + 3x_{l+2} &&, \bmod(k,3) = 1 \\
c_k(x) &= x_{l+3}^2 + x_{l+4} - 2x_{l+5} &&, \bmod(k,3) = 2 \\
c_k(x) &= x_{l+2}^2 - x_{l+5} &&, \bmod(k,3) = 0
\end{aligned}
$$

$$j = 4(i-1), \quad l = 4\,\mathrm{div}(k-1,3), \quad 1 \le k \le m = 3(n-1)/4$$

$$\overline{x}_i = 2, \quad \forall i$$