Institute of Computer Science
Academy of Sciences of the Czech Republic

Band preconditioners for the matrix-free truncated Newton method

Ladislav Lukšan, Ctirad Matonoha, Jan Vlček

Technical report No. 1079

September 2010

Pod Vodárenskou věží 2, 182 07 Prague 8 phone: +420 2 688 42 44, fax: +420 2 858 57 89,
e-mail:e-mail:ics@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

Band preconditioners for the matrix-free truncated Newton method

Ladislav Lukšan, Ctirad Matonoha, Jan Vlček [1]

Technical report No. 1079

September 2010

Abstract:

This report is devoted to preconditioning techniques for the matrix-free truncated Newton method. After a review of basic known approaches, we propose new results concerning tridiagonal and pentadiagonal preconditioners based on the standard BFGS updates and on numerical differentiation. Furthermore, we present results of extensive numerical experiments serving for the careful comparison of suitable preconditioning techniques and confirming efficiency of the band preconditioners.

Keywords:
Unconstrained optimization, large scale optimization, truncated Newton method, matrix-free Newton method, band preconditioners, algorithms.

[1]This work was supported by the Grant Agency of the Czech Republic, project No. 201/09/1957, and the institutional research plan No. AVOZ10300504

1 Introduction

We consider the unconstrained minimization problem

$$x^* = \arg\min_{x \in R^n} F(x),$$

where function $F : \mathcal{D}(F) \subset R^n \to R$ is twice continuously differentiable and n is large. We use the notation

$$g(x) = \nabla F(x), \qquad G(x) = \nabla^2 F(x)$$

and the assumption that $\|G(x)\| \leq \overline{G}$, $\forall x \in \mathcal{D}(F)$. Numerical methods for unconstrained minimization are usually iterative and their iteration step has the form

$$x_{k+1} = x_k + \alpha_k s_k, \quad k \in N,$$

where s_k is a direction vector and α_k is a step-length. In this report, we will deal with the Newton method, which uses the quadratic model

$$F(x_k + s) \approx Q(x_k + s) = F(x_k) + g^T(x_k)s + \frac{1}{2}s^T G(x_k)s \tag{1}$$

for direction determination in such a way that

$$s_k = \arg\min_{s \in \mathcal{M}_k} Q(x_k + s). \tag{2}$$

There are two basic possibilities for direction determination: the line-search method, where

$$\mathcal{M}_k = R^n,$$

and the trust-region method, where

$$\mathcal{M}_k = \{s \in R^n : \|s\| \leq \Delta_k\}$$

(here $\Delta_k > 0$ is the trust region radius). Properties of line search and trust region methods together with comments concerning their implementation are perfectly introduced in [4], [22], so no more details are given here.

In this report, we assume that neither matrix $G_k = G(x_k)$ nor its sparsity pattern are explicitly known. In this case, direct methods based on Gaussian elimination cannot be used, so it is necessary to compute the direction vector (2) iteratively. There are many various iterative methods making use of a symmetry of the Hessian matrix, see [26]. Some of them, e.g. [7], [8], [24] allow us to consider indefinite Hessian matrices. Even if these methods are of theoretical interest and lead to nontraditional preconditioners, see [9], we restrict our attention to modifications of the conjugate gradient method [27], [28], [29], which are simple and very efficient (also in the indefinite case).

To make the subsequent investigations clear, we first introduce two basic iterative algorithms for direction determination utilizing the preconditioned conjugate gradient (PCG) method: the line search algorithm proposed in [27] and the trust region algorithm proposed in [28] and [29] (the outer index k is for the sake of simplicity omitted).

Algorithm 1 *Direction determination by the PCG method (the line-search strategy)*

$s_1 = 0, \quad g_1 = g, \quad h_1 = C^{-1}g_1, \quad \rho_1 = g_1^T h_1, \quad p_1 = -h_1.$
Do $i = 1$ **to** $n + 3$
$q_i = Gp_i, \quad \sigma_i = p_i^T q_i.$
If $\sigma_i < \underline{\varepsilon}\|p_i\|$ **then** $s = s_i,$ stop.
$\alpha_i = \rho_i/\sigma_i, \quad s_{i+1} = s_i + \alpha_i p_i, \quad g_{i+1} = g_i + \alpha_i q_i,$
$h_{i+1} = C^{-1}g_{i+1}, \quad \rho_{i+1} = g_{i+1}^T h_{i+1}.$
If $\|g_{i+1}\| \le \omega\|g_1\|$ **or** $i = m$ **then** $s = s_i,$ stop.
$\beta_i = \rho_{i+1}/\rho_i, \quad p_{i+1} = -h_{i+1} + \beta_i p_i.$
End do

Algorithm 2 *Direction determination by the PCG method (the trust-region strategy)*

$s_1 = 0, \quad g_1 = g, \quad h_1 = C^{-1}g_1, \quad \rho_1 = g_1^T h_1, \quad p_1 = -h_1.$
Do $i = 1$ **to** $n + 3$
$q_i = Gp_i, \quad \sigma_i = p_i^T q_i.$
If $\sigma_i \le 0$ **then** $s = s_i + \lambda_i p_i, \ \ \|s\| = \Delta_i,$ stop.
$\alpha_i = \rho_i/\sigma_i.$
If $\|s_i + \alpha_i p_i\| \ge \Delta_i$ **then** $s = s_i + \lambda_i p_i, \ \ \|s\| = \Delta_i,$ stop.
$s_{i+1} = s_i + \alpha_i p_i, \quad g_{i+1} = g_i + \alpha_i q_i,$
$h_{i+1} = C^{-1}g_{i+1}, \quad \rho_{i+1} = g_{i+1}^T h_{i+1}.$
If $\|g_{i+1}\| \le \omega\|g_1\|$ **or** $i = m$ **then** $s = s_i,$ stop.
$\beta_i = \rho_{i+1}/\rho_i, \quad p_{i+1} = -h_{i+1} + \beta_i p_i.$
End do

Since matrix G is not given explicitly, we use numerical differentiation instead of matrix multiplication. Thus the product $q = Gp$ is replaced by the difference

$$G(x)p \approx \frac{g(x + \delta p) - g(x)}{\delta} \tag{3}$$

where $\delta = \varepsilon/\|p\|$ (usually $\varepsilon = \sqrt{\varepsilon_M}$ and ε_M is a machine precision). An optimization method, where the direction vector is computed iteratively by one of the above algorithms and where product $q = Gp$ is replaced by difference (3), is called the truncated Newton method. This method has been theoretically studied in many papers, see [5], [6], [19], [23]. The following theorem, which easily follows from the mean value theorem, confirms the choice (3).

Theorem 1 *Let function $F : R^n \to R$ have Lipschitz continuous second order derivatives (with the constant $\overline{L}$). Let $q = G(x)p$ and*

$$\tilde{q} = \frac{g(x + \delta p) - g(x)}{\delta}, \quad \delta = \frac{\varepsilon}{\|p\|}.$$

Then it holds

$$\|\tilde{q} - q\| \leq \frac{1}{2}\varepsilon\overline{L}\|p\|.$$

A disadvantage of the truncated Newton method consists in the fact that it requires a large number of inner iterations (i.e. a large number of gradient evaluations) if matrix $G = G(x)$ is ill-conditioned. Therefore, the conjugate gradient method need to be suitably preconditioned. Unfortunately, the sparsity pattern of G is not known, so the usual preconditioning methods (e.g. methods based on the incomplete Choleski decomposition) cannot be used. In this report we confine to the following particular approaches (see [18], [19], [20], [25]).

(A1) Preconditioners based on the limited memory BFGS method.

(A2) Preconditioners obtained by the standard BFGS method equivalent to the preconditioned conjugate gradient method.

(A3) Preconditioners obtained by the numerical differentiation.

(A4) Preconditioners determined by the Lanczos method equivalent to the unpreconditioned conjugate gradient method.

These possibilities are described in Section 2. In Section 3, we focus on band preconditioners based on approach (A2) and prove some new results concerning their positive definiteness. Section 4 is devoted to band preconditioners based on approach (A3), where we propose some theoretical results confirming their efficiency. The last section contains some important comments concerning implementation of band preconditioners and results of computational experiments comparing various preconditioning techniques.

2 Building preconditioners for the truncated Newton method

In this section, we describe four approaches mentioned above in more details. This description is necessary for understanding new results introduced in the subsequent sections and for an extensive comparison of selected preconditioners presented in Section 5.

The idea of approach (A1), the use of the limited memory BFGS updates, is very simple (see [18]). Matrix $C_k^{-1} = H_k = H_k^k$, used as a preconditioner in the k-th step of the truncated Newton method, is determined recurrently in such a way that $H_{k-l}^k = \gamma_{k-l}I$, where l is the number of updates (usually $l = \min(k-1, 3)$) and

$$\begin{aligned} H_{j+1}^k &= H_j^k + \left(\frac{y_j^T H_j^k y_j}{y_j^T d_j} + 1\right)\frac{d_j d_j^T}{y_j^T d_j} - \frac{H_j^k y_j d_j^T + d_j (H_j^k y_j)^T}{y_j^T d_j} \\ &= V_j^T H_j^k V_j + \frac{d_j d_j^T}{y_j^T d_j} \end{aligned}$$

for $k - l \leq j \leq k - 1$ with

$$V_j = I - \frac{y_j d_j^T}{y_j^T d_j}, \quad d_j = x_{j+1} - x_j, \quad y_j = g_{j+1} - g_j.$$

Matrix H_k is not computed explicitly. In the i-th inner step of the conjugate gradient method, which is used in the k-th outer step of the Newton method, a vector $h_i = C_k^{-1} g_i = H_k\, g_i$ is determined by the Strang recurrences [17]. First, we set $u_k = g_i$ and compute numbers and vectors

$$\sigma_j = \frac{d_j^T u_{j+1}}{y_j^T d_j} \quad \text{and} \quad u_j = u_{j+1} - \sigma_j y_j, \quad k-l \le j \le k-1,$$

respectively, using backward recurrences. Then we set $v_{k-l} = \gamma_{k-l} u_{k-l}$ and compute vectors

$$v_{j+1} = v_j + \left(\sigma_j - \frac{y_j^T v_j}{y_j^T d_j}\right) d_j, \quad k-l \le j \le k-1,$$

using forward recurrence. Finally, we set $h_i = v_k$. Scaling parameter γ_{k-l} serves for improving the efficiency of the preconditioner. A suitable choice is $\gamma_{k-l} = y_k^T d_k / y_k^T y_k$. The use of the Strang recurrences is the oldest (and simplest) possibility for implementing the limited memory BFGS method. There are other useful procedures based on explicit [2] or recursive [16] matrix formulations.

Approach (A2), introduced in [20], is based on the fact that the BFGS method with perfect line search applied to a strictly convex quadratic function is equivalent to the conjugate gradient method with the same step-size selection. Assume that the conjugate gradient method, used in the current step of the Newton method, is preconditioned by the matrix C_k. Applying the equivalent BFGS updates, we construct matrix B_k, whose elements serves for the determination of preconditioner C_{k+1} utilized in the next step of the Newton method. To simplify the notation, we omit index k in the subsequent considerations.

The BFGS method equivalent to the PCG method generates a sequence of matrices B_i, $1 \le i \le m$, in such a way that $B_1 = C$ and

$$B_{i+1} = B_i + \frac{y_i y_i^T}{d_i^T y_i} - \frac{B_i d_i (B_i d_i)^T}{d_i^T B_i d_i} = B_i + \frac{G p_i (G p_i)^T}{p_i^T G p_i} + \frac{g_i g_i^T}{p_i^T g_i}$$

for $1 \le i \le m$, where $d_i = s_{i+1} - s_i = \alpha_i p_i$ and $y_i = g_{i+1} - g_i = G d_i$. Vectors p_i and g_i are byproducts of the conjugate gradient method preconditioned by matrix C. If we use vectors $\tilde{q}_i$ (given by the numerical differentiation) and $\tilde{g}_i$ instead of vectors $q_i = G p_i$ and g_i, respectively, we can write

$$B_1 = C, \quad B_{i+1} = B_i + \frac{\tilde{q}_i \tilde{q}_i^T}{p_i^T \tilde{q}_i} + \frac{\tilde{g}_i \tilde{g}_i^T}{p_i^T \tilde{g}_i}, \quad 1 \le i \le m. \tag{4}$$

From the above formulation, it is evident that only vectors generated by the preconditioned conjugate gradient method (with matrix multiplication replaced by the numerical differentiation) are used for the determination of matrices B_i, $1 \le i \le m$. These matrices do not occur in correction terms, so we can update and save only their selected parts (which has the same effect as the deletion of superfluous elements in the final matrix $B = B_{m+1}$). If

the vectors $\tilde{q}_i$ and $\tilde{g}_i$ are good approximations of the vectors q_i and g_i, then the matrices B_i, $1 \le i \le m$, are positive definite. Further, if the number of steps of the conjugate gradient method is sufficiently large, the matrix $B = B_{m+1}$ is a good approximation of matrix G so we can use its saved part as a preconditioner in the next step of the Newton method.

The described idea is mentioned in [20], where the author recommends to use the main diagonal of the matrix B to define the diagonal preconditioner. If $C = D$, where D is a diagonal matrix containing diagonal elements of B, no problem arises because positive definite matrix B has positive numbers on the main diagonal. Diagonal preconditioning for problems with sparse Hessian matrices justifies the following theorem proved in [13].

Theorem 2 *Let $\mathcal{D}_n$ be the set of all diagonal matrices of order n and let D be a diagonal matrix containing diagonal elements of matrix G. Then it holds*

$$\kappa(GD^{-1}) \le l \min_{M \in \mathcal{D}_n} \kappa(GM^{-1})$$

where κ is a spectral condition number and l is a maximum number of nonzero elements in rows of matrix G.

Approach (A3) is based on the assumption that the Hessian matrix has a simple pattern (even if this assumption is not really satisfied). If the fictive Hessian matrix is diagonal, then all its elements can be approximated by using one gradient difference

$$G(x)v \approx g(x+v) - g(x), \qquad v = [\delta_1, \ldots, \delta_n]^T,$$

where $\delta_1, \ldots, \delta_n$ are suitable differences, see [3]. Diagonal matrix $C = D = \mathrm{diag}(\alpha_1, \ldots, \alpha_n)$, where $Dv = g(x+v) - g(x)$, is then used as a preconditioner. One has $\alpha_i \delta_i = g_i(x+v) - g_i(x)$, so

$$\alpha_i = \frac{g_i(x+v) - g_i(x)}{\delta_i}, \qquad 1 \le i \le n. \tag{5}$$

Remark 1 The differences δ_i, $1 \le i \le n$, can be chosen in two different ways:

(1) We set $\delta_i = \delta$, $1 \le i \le n$, so $v = \delta e$, where e is a vector with all elements equal to one. Usually $\delta = \sqrt{\varepsilon_M}/\|e\| = \sqrt{\varepsilon_M/n}$ (similarly as in Theorem 1).

(2) We set $\delta_i = \sqrt{\varepsilon_M}\max(|x_i|, 1)$, $1 \le i \le n$. This choice is less sensitive to rounding errors.

In both cases one can write $\delta_i = \varepsilon \overline{\delta}_i$, $1 \le i \le n$, where $\varepsilon = \sqrt{\varepsilon_M}$ and either $\overline{\delta}_i = 1/\sqrt{n}$ or $\overline{\delta}_i = \max(|x_i|, 1)$ for $1 \le i \le n$.

A disadvantage of preconditioners based on the numerical differentiation consists in the fact that they may not be positive definite. Consider a strictly convex quadratic function $F : R^2 \to R$:

$$F(x) = \frac{1}{2}\, x^T \begin{bmatrix} 1 & -2 \\ -2 & 6 \end{bmatrix} x, \qquad g(x) = \begin{bmatrix} 1 & -2 \\ -2 & 6 \end{bmatrix} x.$$

Then it holds

$$\frac{g(x+\delta e)-g(x)}{\delta} = \begin{bmatrix} 1 & -2 \\ -2 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 4 \end{bmatrix},$$

thus

$$De = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 4 \end{bmatrix},$$

which gives $\alpha_1 = -1$, $\alpha_2 = 4$, so matrix D is not positive definite. This drawback can be removed by setting

$$\alpha_i = \frac{|g_i(x+v)-g_i(x)|}{\delta_i}, \qquad 1 \le i \le n,$$

instead of (5). This modification is justified by the following theorem proved in [13] (see also [25]).

Theorem 3 *Let $\mathcal{D}_n$ be the set of all diagonal matrices of order n and let $D = \mathrm{diag}(\alpha_1, \ldots, \alpha_n)$ be a diagonal matrix such that*

$$\alpha_i = \sum_{j=1}^{n} |G_{ij}|, \qquad 1 \le j \le n,$$

where G_{ij}, $1 \le j \le n$, are the elements of the i-th row of matrix G. Then it holds

$$\kappa_1(GD^{-1}) = \min_{M \in \mathcal{D}_n} \kappa_1(GM^{-1}),$$

where κ_1 is an l_1 condition number (the product of l_1 norms of a matrix and its inverse).

If all elements of Hessian matrix G are positive and if we set $v = \delta e$, then we can write $De = (g(x+\delta e) - g(x))/\delta \approx Ge$, so

$$\alpha_i \approx \sum_{j=1}^{n} G_{ij} = \sum_{j=1}^{n} |G_{ij}|$$

and matrix D is according to Theorem 3 an ideal preconditioner (in l_1 norm) for the system of equations $Gs + g = 0$. If matrix G does not contain only positive numbers, one has

$$|\alpha_i| \approx \left| \sum_{j=1}^{n} G_{ij} \right| \le \sum_{j=1}^{n} |G_{ij}|,$$

so the elements of modified matrix D form the lower bound for the elements of an ideal preconditioner.

Approach (A4) is based on the use of the symmetric Lanczos method, which is equivalent to the conjugate gradient method. The elements of a tridiagonal matrix $\bar{T}_m$ obtained by the Lanczos method can be determined from the coefficients of the conjugate gradient method (Algorithm 1 and Algorithm 2) by transformations $\bar{\alpha}_1 = 1/\alpha_1$ and

$$\bar{\beta}_i^2 = \frac{\beta_i}{\alpha_i^2}, \quad \bar{\alpha}_{i+1} = \frac{\beta_i}{\alpha_i} + \frac{1}{\alpha_{i+1}}, \quad 1 \le i \le m,$$

(see [12]), where m is the number such that $\alpha_i > 0$ for $1 \leq i \leq m$. Tridiagonal matrix $\bar{T}_m$ obtained by this way is positive definite (it follows from the proof of Theorem 4 introduced below). This matrix has dimension $m \leq n$.

In order to obtain a preconditioner with the dimension n, we set

$$C = [Q_m, Q_{n-m}] \begin{bmatrix} \bar{T}_m & 0 \\ 0 & I_{n-m} \end{bmatrix} [Q_m, Q_{n-m}]^T = (I - Q_m Q_m^T) + Q_m \bar{T}_m Q_m^T, \tag{6}$$

where Q_m is a matrix with m orthonormal columns obtained by the symmetric Lanczos method and Q_{n-m} is a matrix with $n-m$ orthonormal columns such that matrix $[Q_m, Q_{n-m}]$ is square and orthogonal. Matrix C is also positive definite and its inverse can be computed using the simple formula

$$C^{-1} = (I - Q_m Q_m^T) + Q_m \bar{T}_m^{-1} Q_m^T, \tag{7}$$

see [9].

The disadvantage of preconditioner (6) lies in that this matrix can be defined only in the unpreconditioned step of the Newton method. If we use a preconditioner, then the columns of matrix Q_m are not orthonormal (see [4]) and matrix (6) does not have the required properties (its inverse cannot be computed by (7)). In order to avoid such difficulties, we would have to include the used preconditioner into expression (6) (in place of the unit matrix). It means that we would have to store preconditioners from all previous steps which is impractical. Thus we proceed as follows. We perform $m \ll n$ steps of the unpreconditioned conjugate gradient method and construct preconditioner (6), which is used in the next steps of the conjugate gradient method (we can also go back to the beginning of the iteration process or use a more complicated strategy described in [9]).

3 Band preconditioners obtained by the standard BFGS updates

Now we focus our attention on approach (A2) considering band preconditioners with the bandwidth greater than one. Let $C = T$, where T is a tridiagonal matrix containing elements of three main diagonals of positive definite matrix $B = B_{m+1}$ obtained by (4). In this case, matrix C may not be positive definite (even if B is positive definite). As an example, consider matrices

$$B = \begin{bmatrix} 2 & -2 & 2 \\ -2 & 3 & -3 \\ 2 & -3 & 4 \end{bmatrix}, \qquad T = \begin{bmatrix} 2 & -2 & 0 \\ -2 & 3 & -3 \\ 0 & -3 & 4 \end{bmatrix}.$$

Both these matrices have positive elements on the main diagonal and positive main subdeterminants of the second order. But it holds that $\det B = 2$ and $\det T = -10$ so T is not positive definite, although B is positive definite. In order to remove this drawback, we modify matrix T to be positive definite (see below).

First we introduce the following well-known lemma, see [11] (for convenience we denote elements of T by α_i and β_i even if they have different meaning than step-sizes α_i and coefficients β_i used in conjugate gradient algorithms).

Lemma 1 *Consider a tridiagonal matrix*

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & \dots & 0 & 0 \\ \beta_1 & \alpha_2 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \alpha_{n-1} & \beta_{n-1} \\ 0 & 0 & \dots & \beta_{n-1} & \alpha_n \end{bmatrix} \tag{8}$$

and denote Δ_i a main subdeterminant of the i-th order of matrix T containing rows and columns with indices $1, 2, \dots, i$. Then we can write $\Delta_1 = \alpha_1$ and

$$\Delta_i = \alpha_i \Delta_{i-1} - \beta_{i-1}^2 \Delta_{i-2}, \qquad 2 \le i \le n, \tag{9}$$

where $\Delta_0 = 1$.

This lemma can be used in the proof of the next theorem (assertion of this theorem also follows from the Cholesky decomposition of symmetric tridiagonal matrix described in [11]).

Theorem 4 *Let $\gamma_1 = \alpha_1$ and*

$$\gamma_i = \alpha_i - \frac{\beta_{i-1}^2}{\gamma_{i-1}}, \qquad 2 \le i \le n. \tag{10}$$

Then tridiagonal matrix (8) is positive definite if and only if $\gamma_i > 0$ for $1 \le i \le n$.

Proof We prove by induction that $\Delta_i = \gamma_i \Delta_{i-1}$ for $1 \le i \le n$, where again $\Delta_0 = 1$. This assertion is obvious for $i = 1$. Assume that $\Delta_{i-1} = \gamma_{i-1}\Delta_{i-2}$ for some index $i > 1$. Using (9) and (10), we obtain

$$\begin{aligned} \Delta_i &= \alpha_i \Delta_{i-1} - \beta_{i-1}^2 \Delta_{i-2} = \alpha_i \Delta_{i-1} + \gamma_{i-1}(\gamma_i - \alpha_i)\Delta_{i-2} \\ &= (\Delta_{i-1} - \gamma_{i-1}\Delta_{i-2})\alpha_i + \gamma_{i-1}\gamma_i \Delta_{i-2} = \gamma_i \Delta_{i-1}, \end{aligned}$$

so the induction step is finished. Since $\Delta_i = \gamma_i \Delta_{i-1}$ for $1 \le i \le n$, then $\Delta_i > 0$ holds if and only if $\gamma_i > 0$ (for $1 \le i \le n$). □

Theorem 4 can be utilized in such a way that we compute numbers γ_i, $1 < i \le n$, and as soon as $\gamma_i \le 0$ for some index i, we decrease the off-diagonal element β_{i-1} so that $\beta_{i-1}^2 < \gamma_{i-1}\alpha_i$ (e.g. we set $\beta_{i-1}^2 = \lambda_{i-1}\gamma_{i-1}\alpha_i$, where $0 < \lambda_{i-1} < 1$). The trouble is that if we choose λ_{i-1} unsuitably, the resulting tridiagonal matrix can be ill-conditioned. For practical purposes it is more convenient to use the following theorem and its corollary.

Theorem 5 *Consider a tridiagonal matrix (8) with positive numbers on the main diagonal. If matrices*

$$\begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix}, \qquad 1 \le i < n - 1, \tag{11}$$

are positive semidefinite then matrix T is positive definite.

Proof For an arbitrary vector $v \in R^n$, we can write

$$\begin{aligned} v^T T v &= \sum_{i=1}^{n} \alpha_i v_i^2 + 2 \sum_{i=1}^{n-1} \beta_i v_i v_{i+1} \\ &= \frac{1}{2}\alpha_1 v_1^2 + \frac{1}{2} \sum_{i=1}^{n-1} \left(\alpha_i v_i^2 + \alpha_{i+1} v_{i+1}^2 + 4\beta_i v_i v_{i+1} \right) + \frac{1}{2}\alpha_n v_n^2 \\ &= \frac{1}{2}\alpha_1 v_1^2 + \frac{1}{2} \sum_{i=1}^{n-1} [v_i, v_{i+1}] \begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \end{bmatrix} + \frac{1}{2}\alpha_n v_n^2 \end{aligned} \tag{12}$$

Since matrices (11) appearing in this equality are positive semidefinite by the assumption, one has $v^T T v \geq 0$. Assume that $v^T T v = 0$. We prove by induction that $v = 0$, which implies positive definiteness of T. Since $\alpha_1 > 0$, then necessarily $v_1 = 0$. Assume that $v_j = 0$ for $1 \leq j \leq i$, where $i < n$. Since

$$[v_i, v_{i+1}] \begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \end{bmatrix} = \alpha_{i+1} v_{i+1}^2$$

and $\alpha_{i+1} > 0$, one has $v_{i+1} = 0$, which finishes the induction step. □

Remark 2 Theorem 5 can be slightly weakened. As we can see, terms $\alpha_1 v_1^2$ and $\alpha_n v_n^2$ can be added to the first and the last terms of the sum in formula (12), respectively. Thus matrix T is positive definite if matrices

$$\begin{bmatrix} 2\alpha_1 & 2\beta_1 \\ 2\beta_1 & \alpha_2 \end{bmatrix}, \qquad \begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix}, \qquad \begin{bmatrix} \alpha_{n-1} & 2\beta_{n-1} \\ 2\beta_{n-1} & 2\alpha_n \end{bmatrix},$$

where $2 \leq i < n-2$, are positive semidefinite and at least one of them is positive definite. This condition is useful, since elements α_1 and α_n are often smaller than we need (see Theorem 10).

Corollary 1 *Let tridiagonal matrix T contain the main diagonal and halves of co-diagonals of the positive definite matrix B (thus $\alpha_i = b_{i,i}$, $1 \leq i \leq n$, and $\beta_i = b_{i,i+1}/2$, $1 \leq i \leq n-1$). Then T is positive definite.*

Proof Substituting $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$ and $\beta_i = b_{i,i+1}/2$, we obtain

$$\begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix} = \begin{bmatrix} b_{i,i} & b_{i,i+1} \\ b_{i,i+1} & b_{i+1,i+1} \end{bmatrix}, \qquad 1 \leq i \leq n-1.$$

These matrices are positive definite, since matrix B is positive definite. □

Remark 3 Theorem 5 and Corollary 1 can be utilized in three particular ways.

(1) We choose elements of matrix T by Corollary 1. Thus for $1 \leq i \leq n-1$, we set $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$ and $\beta_i = b_{i,i+1}/2$.

(2) For $1 \leq i \leq n-1$, we set $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$, $\beta_i = b_{i,i+1}$, and compute determinant $\alpha_i\alpha_{i+1} - 4\beta_i^2$ of matrix (11). If $\alpha_i\alpha_{i+1} - 4\beta_i^2 \geq 0$, then β_i remains unchanged, else we divide β_i by two.

(3) For $1 \leq i \leq n-1$, we set $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$, $\beta_i = b_{i,i+1}$, and compute determinant $\alpha_i\alpha_{i+1} - 4\beta_i^2$ of matrix (11). If $\alpha_i\alpha_{i+1} - 4\beta_i^2 \geq 0$, then β_i remains unchanged, else we set $\beta_i = (1/2)\sqrt{\alpha_i\alpha_{i+1}}$.

In all these cases, the resulting matrix is positive definite.

Assertions of Theorem 5 and Corollary 1 can be generalized for further band matrices. We first show the corresponding procedure in the case of the following pentadiagonal matrix

$$P = \begin{bmatrix} \alpha_1 & \beta_1 & \gamma_1 & \dots & 0 & 0 & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \dots & 0 & 0 & 0 \\ \gamma_1 & \beta_2 & \alpha_3 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \alpha_{n-2} & \beta_{n-2} & \gamma_{n-2} \\ 0 & 0 & 0 & \dots & \beta_{n-2} & \alpha_{n-1} & \beta_{n-1} \\ 0 & 0 & 0 & \dots & \gamma_{n-2} & \beta_{n-1} & \alpha_n \end{bmatrix} \tag{13}$$

Theorem 6 *Consider a pentadiagonal matrix P with positive elements on the main diagonal. If matrices*

$$\begin{bmatrix} \alpha_i & (3/2)\beta_i & 3\gamma_i \\ (3/2)\beta_i & \alpha_{i+1} & (3/2)\beta_{i+1} \\ 3\gamma_i & (3/2)\beta_{i+1} & \alpha_{i+2} \end{bmatrix}, \qquad 1 \leq i < n-2, \tag{14}$$

are positive semidefinite, then matrix P is positive definite.

Proof For an arbitrary vector $v \in R^n$, we can write

$$\begin{aligned} v^T P v &= \sum_{i=1}^{n} \alpha_i v_i^2 + 2\sum_{i=1}^{n-1} \beta_i v_i v_{i+1} + 2\sum_{i=1}^{n-2} \gamma_i v_i v_{i+2} \\ &= \frac{1}{3}\alpha_1 v_1^2 + \frac{1}{3}(\alpha_1 v_1^2 + \alpha_2 v_2^2) + \beta_1 v_1 v_2 \\ &+ \frac{1}{3}\sum_{i=1}^{n-2} \left(\alpha_i v_i^2 + \alpha_{i+1} v_{i+1}^2 + \alpha_{i+2} v_{i+2}^2 + 3\beta_i v_i v_{i+1} + 3\beta_{i+1} v_{i+1} v_{i+2} + 6\gamma_i v_i v_{i+2}\right) \\ &+ \frac{1}{3}(\alpha_{n-1} v_{n-1}^2 + \alpha_n v_n^2) + \beta_{n-1} v_{n-1} v_n + \frac{1}{3}\alpha_n v_n^2 \\ &= \frac{1}{3}\alpha_1 v_1^2 + \frac{1}{3}\,[v_1, v_2] \begin{bmatrix} \alpha_1 & (3/2)\beta_1 \\ (3/2)\beta_1 & \alpha_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\ &+ \frac{1}{3}\sum_{i=1}^{n-2} [v_i, v_{i+1}, v_{i+2}] \begin{bmatrix} \alpha_i & (3/2)\beta_i & 3\gamma_i \\ (3/2)\beta_i & \alpha_{i+1} & (3/2)\beta_{i+1} \\ 3\gamma_i & (3/2)\beta_{i+1} & \alpha_{i+2} \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \\ v_{i+2} \end{bmatrix} \\ &+ \frac{1}{3}\,[v_{n-1}, v_n] \begin{bmatrix} \alpha_{n-1} & (3/2)\beta_{n-1} \\ (3/2)\beta_{n-1} & \alpha_n \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix} + \frac{1}{3}\alpha_n v_n^2 \end{aligned}$$

Since matrices appearing in this equality are positive semidefinite by the assumption, one has $v^T P v \geq 0$. Assume that $v^T P v = 0$. We prove by induction that $v = 0$, which implies positive definiteness of P. Similarly as in the proof of Theorem 5 we obtain $v_1 = 0$ and $v_2 = 0$. Assume that $v_j = 0$ for $1 \leq j \leq i$, where $i < n - 1$. Since

$$[v_i, v_{i+1}, v_{i+2}] \begin{bmatrix} \alpha_i & (3/2)\beta_i & 3\gamma_i \\ (3/2)\beta_i & \alpha_{i+1} & (3/2)\beta_{i+1} \\ 3\gamma_i & (3/2)\beta_{i+1} & \alpha_{i+2} \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \\ v_{i+2} \end{bmatrix} = \alpha_{i+2} v_{i+2}^2$$

and $\alpha_{i+2} > 0$, one has $v_{i+2} = 0$, which finishes the induction step. □

Corollary 2 *Let a pentadiagonal matrix P contain the main diagonal, two thirds of the first co-diagonals, and one third of the second co-diagonals of the positive definite matrix B (thus $\alpha_i = b_{i,i}$, $1 \leq i \leq n$, $\beta_i = 2b_{i,i+1}/3$, $1 \leq i \leq n-1$, and $\gamma_i = b_{i,i+2}/3$, $1 \leq i \leq n-2$). Then P is positive definite.*

Proof Substituting $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$, $\alpha_{i+2} = b_{i+2,i+2}$, $\beta_i = 2b_{i,i+1}/3$, $\beta_{i+1} = 2b_{i+1,i+2}/3$ and $\gamma_i = b_{i,i+2}/3$, we obtain

$$\begin{bmatrix} \alpha_i & (3/2)\beta_i & 3\gamma_i \\ (3/2)\beta_i & \alpha_{i+1} & (3/2)\beta_{i+1} \\ 3\gamma_i & (3/2)\beta_{i+1} & \alpha_{i+2} \end{bmatrix} = \begin{bmatrix} b_{i,i} & b_{i,i+1} & b_{i,i+2} \\ b_{i,i+1} & b_{i+1,i+1} & b_{i+1,i+2} \\ b_{i,i+2} & b_{i+1,i+2} & b_{i+2,i+2} \end{bmatrix}, \qquad 1 \leq i \leq n-2.$$

These matrices are positive definite, since matrix B is positive definite. □

Theorem 7 *Let assumptions of Theorem 6 be satisfied. Then determinants Δ_i of matrices (14) can be computed according to the formula*

$$\Delta_i = \alpha_{i+1}\left(\alpha_i \alpha_{i+2} - 9\gamma_i^2\right) - \frac{9}{4}\left(\alpha_i \beta_{i+1}^2 + \alpha_{i+2}\beta_i^2 - 6\beta_i \beta_{i+1}\gamma_i\right). \tag{15}$$

The determinant Δ_i is nonnegative if and only if $\underline{\gamma}_i \leq \gamma_i \leq \overline{\gamma}_i$ where

$$\begin{aligned} \underline{\gamma}_i &= \frac{1}{3\alpha_{i+1}}\left(\frac{9}{4}\beta_i\beta_{i+1} - \sqrt{D_i}\right), \\ \overline{\gamma}_i &= \frac{1}{3\alpha_{i+1}}\left(\frac{9}{4}\beta_i\beta_{i+1} + \sqrt{D_i}\right) \end{aligned}$$

are the roots of the quadratic equation $\Delta_i = 0$ and

$$D_i = \left(\alpha_i\alpha_{i+1} - \frac{9}{4}\beta_i^2\right)\left(\alpha_{i+1}\alpha_{i+2} - \frac{9}{4}\beta_{i+1}^2\right)$$

is the discriminant, divided by 36, of this equation, which is nonnegative provided that both multipliers are nonnegative.

Proof Relation (15) follows immediately by evaluation of the pertinent determinant. Since quadratic term $-9\gamma_i^2$ in (15) has the negative sign and $\alpha_{i+1} > 0$ by the assumption, determinant Δ_i is nonnegative if and only if $\underline{\gamma}_i \leq \gamma_i \leq \overline{\gamma}_i$, where $\underline{\gamma}_i$, $\overline{\gamma}_i$ are the roots of quadratic equation $\Delta_i = 0$. By (15), the discriminant of this equation (divided by number 36) is given by the relation

$$\begin{aligned} D_i &= \frac{81}{16}\beta_i^2\beta_{i+1}^2 - \frac{9}{4}\alpha_i\alpha_{i+1}\beta_{i+1}^2 - \frac{9}{4}\alpha_{i+1}\alpha_{i+2}\beta_i^2 + \alpha_i\alpha_{i+1}^2\alpha_{i+2} \\ &= \frac{9}{4}\beta_{i+1}^2\left(\frac{9}{4}\beta_i^2 - \alpha_i\alpha_{i+1}\right) - \alpha_{i+1}\alpha_{i+2}\left(\frac{9}{4}\beta_i^2 - \alpha_i\alpha_{i+1}\right) \\ &= \left(\alpha_i\alpha_{i+1} - \frac{9}{4}\beta_i^2\right)\left(\alpha_{i+1}\alpha_{i+2} - \frac{9}{4}\beta_{i+1}^2\right). \end{aligned}$$

□

Theorem 7 offers two possibilities how to choose a new element γ_i in case that $\Delta_i < 0$. If $\gamma_i < \underline{\gamma}_i$, we set $\gamma_i := \underline{\gamma}_i$. If $\gamma_i > \overline{\gamma}_i$, we set $\gamma_i := \overline{\gamma}_i$. However, more advantageous is to set

$$\gamma_i = \frac{1}{2}(\underline{\gamma}_i + \overline{\gamma}_i) = \frac{3}{4}\frac{\beta_i\beta_{i+1}}{\alpha_{i+1}}, \tag{16}$$

because this choice is computationally simpler and gives better practical results.

Remark 4 Theorem 6 and Corollary 2 can be utilized in three particular ways.

(1) We choose elements of matrix P by Corollary 2. For $1 \leq i \leq n$, we set $\alpha_i = b_{i,i}$. For $1 \leq i \leq n-1$, we set $\beta_i = 2b_{i,i+1}/3$. For $1 \leq i \leq n-2$, we set $\gamma_i = b_{i,i+2}/3$.

(2) For $1 \leq i \leq n-2$, we set $\alpha_i = b_{i,i}$, $\alpha_{i+1} = b_{i+1,i+1}$, $\alpha_{i+2} = b_{i+2,i+2}$, $\beta_i = b_{i,i+1}$, $\beta_{i+1} = b_{i+1,i+2}$ and $\gamma_i = b_{i,i+2}$. If matrix (14) is positive definite, then β_i, β_{i+1} and γ_i remain unchanged, else we set $\beta_i = 2b_{i,i+1}/3$, $\beta_{i+1} = 2b_{i+1,i+2}/3$ and $\gamma_i = b_{i,i+2}/3$.

(3) For $1 \leq i \leq n$, we set $\alpha_i = b_{i,i}$. For $1 \leq i \leq n-1$, we set $\beta_i = b_{i,i+1}$, and compute leading subdeterminant $\alpha_i\alpha_{i+1} - (9/4)\beta_i^2$ of matrix (14). If $\alpha_i\alpha_{i+1} - (9/4)\beta_i^2 \geq 0$, then β_i remains unchanged, else we set $\beta_i = (2/3)\sqrt{\alpha_i\alpha_{i+1}}$. For $1 \leq i \leq n-2$, we set $\gamma_i = b_{i,i+2}$ and compute determinant Δ_i by (15). If $\Delta_i \geq 0$, then γ_i remains unchanged, else we compute γ_i by (16).

In all these cases, the resulting matrix is positive definite.

So far we have assumed that preconditioner C is at most pentadiagonal, but Corollary 2 can be generalized for other band preconditioners. Let C be a symmetric band matrix with the bandwidth l, so it has the main diagonal and $k-1 = (l-1)/2$ pairs of co-diagonals, which are equal to the corresponding diagonals of positive definite matrix B. Then if we multiply the i-th pair of co-diagonals by number $(k-i)/k$ (for $1 \leq i \leq k-1$), the resulting matrix is positive definite. The proof of this assertion is similar to proof of Corollary 2 (we use an analogy of Theorem 6).

4 Band preconditioners obtained by the numerical differentiation

Now we focus our attention on approach (A3) considering band preconditioners with the bandwidth greater than one. Assume that the Hessian matrix has a band pattern (even if this assumption is not really satisfied). The elements of this fictive matrix, used as a preconditioner, can be determined by the numerical differentiation. This is performed only once at the beginning of the outer step of the Newton method.

In order to determine all elements of the band Hessian matrix which has $k-1$ pairs of co-diagonals (thus $k=(l+1)/2$ where l is a bandwidth), it suffices to use k gradient differences, which means to compute k extra gradients during each outer step of the Newton method.

Theorem 8 *Let the Hessian matrix of function F be tridiagonal (as matrix T in (8)). Set $v_1=[\delta_1,0,\delta_3,0,\delta_5,0,\ldots]$, $v_2=[0,\delta_2,0,\delta_4,0,\delta_6,\ldots]$, where $\delta_i=\varepsilon\overline{\delta}_i$, $1\le i\le n$. Then for $1<i<n$ it holds*

$$
\begin{aligned}
\alpha_1 &= \lim_{\varepsilon\to 0}\frac{g_1(x+v_1)-g_1(x)}{\delta_1}, \quad \beta_1 = \lim_{\varepsilon\to 0}\frac{g_1(x+v_2)-g_1(x)}{\delta_2},\\
\alpha_i &= \lim_{\varepsilon\to 0}\frac{g_i(x+v_1)-g_i(x)}{\delta_i}, \quad \beta_i = \lim_{\varepsilon\to 0}\frac{g_i(x+v_2)-g_i(x)-\delta_{i-1}\beta_{i-1}}{\delta_{i+1}}, \quad \mathrm{mod}(i,2)=1,\\
\alpha_i &= \lim_{\varepsilon\to 0}\frac{g_i(x+v_2)-g_i(x)}{\delta_i}, \quad \beta_i = \lim_{\varepsilon\to 0}\frac{g_i(x+v_1)-g_i(x)-\delta_{i-1}\beta_{i-1}}{\delta_{i+1}}, \quad \mathrm{mod}(i,2)=0,\\
\alpha_n &= \lim_{\varepsilon\to 0}\frac{g_n(x+v_1)-g_n(x)}{\delta_n}, \quad \mathrm{mod}(n,2)=1,\\
\alpha_n &= \lim_{\varepsilon\to 0}\frac{g_n(x+v_2)-g_n(x)}{\delta_n}, \quad \mathrm{mod}(n,2)=0.
\end{aligned}
$$

Proof Theorem 1 implies that $g(x+v_1)-g(x)=G(x)v_1+o(\varepsilon)$, $g(x+v_2)-g(x)=G(x)v_2+o(\varepsilon)$, so after substituting $G(x)=T$, where T is a tridiagonal matrix of the form (8), and rearranging individual elements we obtain

$$
\begin{aligned}
&\frac{g_1(x+v_1)-g_1(x)}{\delta_1}=\alpha_1+o(1), \qquad \frac{g_1(x+v_2)-g_1(x)}{\delta_2}=\beta_1+o(1),\\
&\frac{g_i(x+v_1)-g_i(x)}{\delta_i}=\alpha_i+o(1), \qquad \frac{g_i(x+v_2)-g_i(x)}{\delta_{i+1}}=\beta_{i+1}+\beta_{i-1}\frac{\delta_{i-1}}{\delta_{i+1}}+o(1), \qquad \mathrm{mod}(i,2)=1,\\
&\frac{g_i(x+v_2)-g_i(x)}{\delta_i}=\alpha_i+o(1), \qquad \frac{g_i(x+v_1)-g_i(x)}{\delta_{i+1}}=\beta_{i+1}+\beta_{i-1}\frac{\delta_{i-1}}{\delta_{i+1}}+o(1), \qquad \mathrm{mod}(i,2)=0,\\
&\frac{g_n(x+v_1)-g_i(x)}{\delta_i}=\alpha_n+o(1), \qquad \mathrm{mod}(n,2)=1,\\
&\frac{g_n(x+v_2)-g_i(x)}{\delta_i}=\alpha_n+o(1), \qquad \mathrm{mod}(n,2)=0,
\end{aligned}
$$

where $2\le i\le n-1$. Since ratios $\delta_{i-1}/\delta_{i+1}=\overline{\delta}_{i-1}/\overline{\delta}_{i+1}$ remain constant for $2\le i\le n-1$, the theorem is proved. □

Remark 5 Theorem 8 specifies an efficient way for building a tridiagonal preconditioner. We choose a fixed number ε (e.g. $\varepsilon = \sqrt{\varepsilon_M}$) and compute elements of matrix $C = T$ according to formulas mentioned in Theorem 8 (where limits are omitted).

Matrix $C = T$ obtained by Remark 5 may not be positive definite even if the Hessian matrix is positive definite. Tridiagonal matrix obtained by application of Theorem 8 (with $\overline{\delta}_i = \overline{\delta}$, $1 \le i \le n$) to a strictly convex quadratic function of three variables with the positive definite Hessian matrix

$$G = \begin{bmatrix} 1 & -1 & -2 \\ -1 & 4 & -1 \\ -2 & -1 & 8 \end{bmatrix}$$

can serve as an example. We will state two theorems supporting a choice of tridiagonal preconditioning in cases when the actual Hessian matrix is pentadiagonal.

Theorem 9 *Let the Hessian matrix $G(x)$ be pentadiagonal, positive definite, and diagonally dominant. Then, if $\delta_i = \varepsilon\overline{\delta}$, $1 \le i \le n$, and if the number ε is sufficiently small, matrix $C = T$ obtained by Remark 5 is positive definite and diagonally dominant.*

Proof Consider pentadiagonal Hessian matrix of the form (13) (with elements denoted by tilde), and set $\tilde{\gamma}_{-1} = \tilde{\gamma}_0 = \tilde{\beta}_0 = \beta_0 = 0$, $\tilde{\gamma}_{n-1} = \tilde{\gamma}_n = \tilde{\beta}_n = \beta_n = 0$ to simplify the notation. Using the assumption of diagonal dominance, we obtain

$$\tilde{\alpha}_i > |\tilde{\gamma}_{i-2}| + |\tilde{\beta}_{i-1}| + |\tilde{\beta}_i| + |\tilde{\gamma}_i|$$

for $1 \le i \le n$. Using Theorem 8 and Remark 5, one can write

$$\alpha_i \approx \tilde{\gamma}_{i-2} + \tilde{\alpha}_i + \tilde{\gamma}_i, \qquad \beta_{i-1} + \beta_i \approx \tilde{\beta}_{i-1} + \tilde{\beta}_i \tag{17}$$

for $1 \le i \le n$. Therefore $\beta_i \approx \tilde{\beta}_i$ and if number ε is sufficiently small, the strict inequality is preserved and we can write

$$\alpha_i - |\beta_{i-1}| - |\beta_i| \approx \tilde{\alpha}_i + \tilde{\gamma}_{i-2} + \tilde{\gamma}_i - |\tilde{\beta}_{i-1}| - |\tilde{\beta}_i| \ge \tilde{\alpha}_i - |\tilde{\gamma}_{i-2}| - |\tilde{\beta}_{i-1}| - |\tilde{\beta}_i| - |\tilde{\gamma}_i| > 0$$

for $1 \le i \le n$. Thus matrix $C = T$ is diagonally dominant and, therefore, positive definite if number ε is sufficiently small. □

In Theorem 9, we assume that all differences are equal, which is fulfilled for instance when $\delta_i = \sqrt{2\varepsilon_M/n}$, $1 \le i \le n$. But the numerical experiments show that the choice $\delta_i = \sqrt{\varepsilon}\max(|x_i|, 1)$, $1 \le i \le n$, is usually more advantageous.

Matrix $C = T$ obtained by Remark 5 is positive definite for many practical problems. Consider a boundary value problem for the second order ordinary differential equation

$$y''(t) = \varphi(y(t)), \quad 0 \le t \le 1, \quad y(0) = y_0, \quad y(1) = y_1,$$

where function $\varphi : R \to R$ is twice continuously differentiable. If we divide the interval $[0, 1]$ onto $n + 1$ parts using nodes $t_i = ih$, $0 \le i \le n + 1$, where $h = 1/(n + 1)$ is the mesh-size and if we replace the second order derivatives in nodes with differences

$$y''(t_i) = \frac{y(t_{i-1}) - 2y(t_i) + y(t_{i+1})}{h^2}, \quad 1 \le i \le n,$$

we will obtain a system of n nonlinear equations

$$f_i(x) \triangleq h^2\varphi(x_i) + 2x_i - x_{i-1} - x_{i+1} = 0, \tag{18}$$

where $x_i = y(t_i)$, $0 \le 1 \le n + 1$, so $x_0 = y_0$ and $x_{n+1} = y_1$. If we solve this system by the least squares method, the minimized function has the form

$$F(x) = \frac{1}{2}f^T(x)f(x) = \frac{1}{2}\sum_{i=1}^{n} f_i^2(x) = \frac{1}{2}\sum_{i=1}^{n}\left(h^2\varphi(x_i) + 2x_i - x_{i-1} - x_{i+1}\right)^2, \tag{19}$$

where $x = [x_1, \ldots, x_n]^T$ and $f = [f_1, \ldots, f_n]^T$.

Theorem 10 *Let the truncated Newton method be applied to the sum of squares (19) with a linear function $\varphi : R \to R$. Then, if $\delta_i = \varepsilon\overline{\delta}$, $1 \le i \le n$, and if the number ε is sufficiently small, matrix $C = T$ obtained by Remark 5 is positive definite.*

Proof Obviously,

$$\nabla f_i(x) = \begin{bmatrix} -1 \\ \psi(x_i) \\ -1 \end{bmatrix}, \qquad \nabla^2 f_i(x) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \psi'(x_i) & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where $\psi(x_i) = 2+h^2\varphi'(x_i)$ and $\psi'(x_i) = h^2\varphi''(x_i)$. For a sum of squares, the Hessian matrix $G(x)$ can be expressed in the form $G(x) = J^T(x)J(x) + W(x)$, where $J(x)$ is the Jacobian matrix of mapping $f(x)$ and $W(x)$ is a second order term. Restricting on submatrices of order five, we can write

$$J(x) = \begin{bmatrix} \psi_1 & -1 & 0 & 0 & 0 \\ -1 & \psi_2 & -1 & 0 & 0 \\ 0 & -1 & \psi_3 & -1 & 0 \\ 0 & 0 & -1 & \psi_4 & -1 \\ 0 & 0 & 0 & -1 & \psi_5 \end{bmatrix}, \quad W(x) = \begin{bmatrix} f_1\psi_1' & 0 & 0 & 0 & 0 \\ 0 & f_2\psi_2' & 0 & 0 & 0 \\ 0 & 0 & f_3\psi_3' & 0 & 0 \\ 0 & 0 & 0 & f_4\psi_4' & 0 \\ 0 & 0 & 0 & 0 & f_5\psi_5' \end{bmatrix},$$

$$J^T(x)J(x) = \begin{bmatrix} \psi_1^2 + 1 & -(\psi_1 + \psi_2) & 1 & 0 & 0 \\ -(\psi_1 + \psi_2) & \psi_2^2 + 2 & -(\psi_2 + \psi_3) & 1 & 0 \\ 1 & -(\psi_2 + \psi_3) & \psi_3^2 + 2 & -(\psi_3 + \psi_4) & 1 \\ 0 & 1 & -(\psi_3 + \psi_4) & \psi_4^2 + 2 & -(\psi_4 + \psi_5) \\ 0 & 0 & 1 & -(\psi_4 + \psi_5) & \psi_5^2 + 1 \end{bmatrix},$$

where $\psi_i = \psi(x_i)$, $1 \le i \le n$, which demonstrates that Hessian matrix $G(x) = J^T(x)J(x) + W(x)$ is pentadiagonal. If function $\varphi : R \to R$ is linear (so $\varphi'(x_i) = \varphi'$, $\varphi''(x_i) = 0$, $1 \le i \le n$, where φ' is the constant slope of linear function φ), than one has $W(x) = 0$, so $G(x) = J^T(x)J(x)$. Denoting $P = G(x)$ we can write (as in the proof of Theorem 9) $\tilde{\alpha}_1 = \psi_1^2 + 1$, $\tilde{\alpha}_n = \psi_n^2 + 1$ and

$$\begin{aligned} \tilde{\alpha}_i &= \psi_i^2 + 2, \qquad 2 \le i \le n-1, \\ \tilde{\beta}_i &= -(\psi_i + \psi_{i+1}), \qquad 1 \le i \le n-1, \\ \tilde{\gamma}_i &= 1, \qquad 1 \le i \le n-2. \end{aligned}$$

If T is the matrix obtained by Remark 5, then (17) holds, which gives $\alpha_1 \approx \psi_1^2 + 2$, $\alpha_2 \approx \psi_2^2 + 3$, $\alpha_i \approx \psi_i^2 + 4$, $3 \le i \le n-2$, $\alpha_{n-1} \approx \psi_{n-1}^2 + 3$, $\alpha_n \approx \psi_n^2 + 2$ and $\beta_i \approx -(\psi_i + \psi_{i+1})$, $1 \le i \le n-1$. Now we use the fact that formula (12) can be rewritten (as in Remark 2) in the form

$$\begin{aligned} 2v^T T v &= [v_1, v_2] \begin{bmatrix} 2\alpha_1 & 2\beta_1 \\ 2\beta_1 & \alpha_2 - 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\ &+ [v_2, v_3] \begin{bmatrix} \alpha_2 + 1 & 2\beta_2 \\ 2\beta_2 & \alpha_3 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \end{bmatrix} \\ &+ \sum_{i=3}^{n-3} [v_i, v_{i+1}] \begin{bmatrix} \alpha_i & 2\beta_i \\ 2\beta_i & \alpha_{i+1} \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \end{bmatrix} \\ &+ [v_{n-2}, v_{n-1}] \begin{bmatrix} \alpha_{n-2} & 2\beta_{n-2} \\ 2\beta_{n-2} & \alpha_{n-1} + 1 \end{bmatrix} \begin{bmatrix} v_{n-2} \\ v_{n-1} \end{bmatrix} \\ &+ [v_{n-1}, v_n] \begin{bmatrix} \alpha_{n-1} - 1 & 2\beta_{n-1} \\ 2\beta_{n-1} & 2\alpha_n \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix} \\ &\approx [v_1, v_2] \begin{bmatrix} 2(\psi_1^2 + 2) & -2(\psi_1 + \psi_2) \\ -2(\psi_1 + \psi_2) & \psi_2^2 + 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\ &+ \sum_{i=2}^{n-2} [v_i, v_{i+1}] \begin{bmatrix} \psi_i^2 + 4 & -2(\psi_i + \psi_{i+1}) \\ -2(\psi_i + \psi_{i+1}) & \psi_{i+1}^2 + 4 \end{bmatrix} \begin{bmatrix} v_i \\ v_{i+1} \end{bmatrix} \\ &+ [v_{n-1}, v_n] \begin{bmatrix} \psi_{n-1}^2 + 2 & -2(\psi_{n-1} + \psi_n) \\ -2(\psi_{n-1} + \psi_n) & 2(\psi_n^2 + 2) \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix} \triangleq 2v^T \tilde{T} v. \quad (20) \end{aligned}$$

Since

$$\begin{aligned} 2(\psi_i^2 + 2)(\psi_{i+1}^2 + 2) - 4(\psi_i + \psi_{i+1})^2 &= 2\psi_i^2 \psi_{i+1}^2 + 8 - 8\psi_i \psi_{i+1} \\ &= 2(\psi_i \psi_{i+1} - 2)^2 \ge 0, \qquad i \in \{1, n-1\}, \\ (\psi_i^2 + 4)(\psi_{i+1}^2 + 4) - 4(\psi_i + \psi_{i+1})^2 &= \psi_i^2 \psi_{i+1}^2 + 16 - 8\psi_i \psi_{i+1} \\ &= (\psi_i \psi_{i+1} - 4)^2 \ge 0, \qquad 2 \le i \le n-2, \end{aligned}$$

all matrices used in the right hand side of (20) are positive semidefinite. Since function φ is linear (with the constant slope φ'), one has $\psi_i = 2 + h^2\varphi'$, $1 \le i \le n$. If $(2 + h^2\varphi')^2 \ne 2$,

the first and the last matrices in (20) are positive definite. In the opposite case, all the other matrices are positive definite. Thus matrix $\tilde{T}$ is positive definite by Remark 2. Since eigenvalues of symmetric matrix depend continuously on its elements, the positive definiteness of $\tilde{T}$ is not violated after small changes of its elements. Thus matrix $T \approx \tilde{T}$, whose elements are obtained by using gradient differences, is positive definite if number ε is sufficiently small. □

Assumptions of Theorem 10 are relatively strong. Nevertheless, these assumptions are sufficient (not necessary) for matrix T to be positive definite. Most of all, it is strongly improbable that all determinants in (20) could be zeroes, so we can assume that matrix $\tilde{T}$, appearing in (20), is positive definite. If we are close to the solution, where $F(x) = 0$, one has $f_i \approx 0$, $1 \le i \le n$. Moreover, matrix $\mathrm{diag}(\psi'_1, \ldots, \psi'_n)$ is usually small in comparison with $J(x)^T J(x)$ (if $n \approx 1000$, then $h^2 \approx 10^{-6}$). Since a small change of diagonal elements does not violate the positive definiteness of $\tilde{T}$, we can expect that matrix $\tilde{T}$ is positive definite in a sufficiently small neighborhood of the solution even if function $\varphi : R \to R$ is nonlinear. Then also matrix T is positive definite if number ε is sufficiently small.

If the Hessian matrix is pentadiagonal and positive definite (as in the previous two theorems), it should be advantageous to use the pentadiagonal preconditioner introduced in the following theorem, whose proof is very similar to the proof of Theorem 8, so it is omitted.

Theorem 11 *Let the Hessian matrix of function F be pentadiagonal (as matrix P). Set $v_1 = [\delta_1, 0, 0, \delta_4, 0, 0, \ldots]$, $v_2 = [0, \delta_2, 0, 0, \delta_5, 0, \ldots]$, $v_3 = [0, 0, \delta_3, 0, 0, \delta_6, \ldots]$, where $\delta_i = \varepsilon\bar{\delta}_i$, $1 \le i \le n$. Then it holds*

$$\alpha_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_1) - g_i(x)}{\delta_i}, \qquad \beta_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_2) - g_i(x) - \delta_{i-2}\gamma_{i-2}}{\delta_{i+1}},$$
$$\gamma_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_3) - g_i(x) - \delta_{i-1}\beta_{i-1}}{\delta_{i+2}}, \qquad \mathrm{mod}(i,3) = 1,$$
$$\alpha_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_2) - g_i(x)}{\delta_i}, \qquad \beta_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_3) - g_i(x) - \delta_{i-2}\gamma_{i-2}}{\delta_{i+1}},$$
$$\gamma_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_1) - g_i(x) - \delta_{i-1}\beta_{i-1}}{\delta_{i+2}}, \qquad \mathrm{mod}(i,3) = 2,$$
$$\alpha_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_3) - g_i(x)}{\delta_i}, \qquad \beta_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_1) - g_i(x) - \delta_{i-2}\gamma_{i-2}}{\delta_{i+1}},$$
$$\gamma_i = \lim_{\varepsilon\to 0} \frac{g_i(x+v_2) - g_i(x) - \delta_{i-1}\beta_{i-1}}{\delta_{i+2}}, \qquad \mathrm{mod}(i,3) = 0,$$

where quantities with indices $i < 1$ are assumed to be zeroes and quantities defined by expressions containing indices $i > n$ are not computed.

5 Implementation notes and numerical experiments

In this section we introduce several practical comments concerning preconditioners proposed in Section 3 and Section 4. First it is important to be able to decide whether a preconditioner will be used or rejected, since the obtained preconditioner may not be always suitable for the use. Primarily, it is necessary to emphasize that an indefinite preconditioner is inappropriate also in case when the Hessian matrix is not positive definite. In this case the direction vector usually directs towards a saddle point of a minimized function, so it is not appropriate to obtain such a direction too quickly.

A suitable tool for testing positive definiteness and ill-conditioning of a symmetric matrix is the Gill-Murray decomposition proposed in [10]. If a pivot obtained during the Gill-Murray elimination step is less than $\delta \max(1, \max_{1\leq i\leq n}(|\alpha_i|))$, where δ is a prescribed bound and α_i, $1 \leq i \leq n$, are diagonal elements of the preconditioner, then the decomposition is terminated and the preconditioner is rejected. If such a situation arises, it is not worth performing the complete Gill-Murray decomposition and using the obtained positive definite matrix as a preconditioner (this claim was confirmed by numerical experiments). The number δ is usually chosen very small (e.g. $\delta = 10^{-12}$). Sometimes, however, it is better to choose a larger value (e.g. $\delta = 10^{-2}$).

Band preconditioners obtained by the standard BFGS updates (approach (A2)) need to be corrected in advance, otherwise they are mostly rejected during the Gill-Murray decomposition. Their modifications based on Theorem 5 (statement (3) of Remark 3) and Theorem 6 (statement (3) of Remark 4), reveal to be very successful. However, preconditioners based on approach (A2) require rejecting more often (e.g. by setting $\delta = 10^{-2}$).

Band preconditioners obtained by the numerical differentiation (approach (A3)) do not require sophisticated corrections. It suffices to replace their diagonal elements with their absolute values. Modifications based on Theorem 5 and Theorem 6 decrease effectiveness of preconditioning in this case. Numerical experiments indicate that it suffices to choose $\delta = 10^{-12}$ for rejecting (except for diagonal preconditioners, which are more sensitive to rejecting).

The truncated Newton methods that use various preconditioning techniques were tested using two collections of unconstrained optimization problems. The first collection, TEST25 described in [14], contains 82 test problems with 1000 variables obtained from various sources (we have used 71 problems suitable for non-derivative methods) and the second collection, TEST11 described in [15], contains 58 test problems with 1000 variables obtained from the CUTE collection [1] (we have used 54 problems suitable for non-derivative methods). These collections can be found on `http://www.cs.cas.cz/luksan/test.html` together with reports [14] and [15].

The results of computational experiments are reported in two tables corresponding to two collections TEST25 and TEST11. The tables contain the following data: `NIT` – the total number of iterations, `NFV` – the total number of function evaluations, `NFG` – the total number of gradient evaluations, `NCG` – the total number of inner iterations, `NCN` – the total number of preconditioned outer iterations, `NCP` – the total number of problems with enlarged bound for rejecting, `Time` – the total computational time.

The rows correspond to the methods tested: `TN` – the unpreconditioned truncated Newton method, `TNLM` – preconditioning by the limited memory BFGS method, `TNVM` – band preconditioning based on the standard BFGS updates (1 – diagonal, 2 – tridiagonal, 3 - pentadiagonal), `TNND` – band preconditioning based on the numerical differentiation (1 – diagonal, 2 – tridiagonal, 3 – pentadiagonal), `TNLT` – preconditioning based on the Lanczos method, `LMVM` – the limited memory BFGS method, `CG` – the nonlinear conjugate gradient method. Methods `LMVM` and `CG` are mentioned only for comparison (they are quite different from the truncated Newton methods studied in this report).

Method	NIT	NFV	NFG	NCG	NCN	NCP	Time
TN	7425	11827	372789	359505	-	-	66.08
TNLM	7270	12521	233269	219347	7270	-	42.55
TNVM-1	7095	10303	274344	262855	4335	37	50.43
TNVM-2	6751	9252	139989	129933	4260	37	27.47
TNVM-3	6803	8857	229501	219820	4027	36	51.67
TNND-1	6522	8491	347384	331709	3857	40	59.51
TNND-2	7573	11245	147391	119434	4409	3	25.45
TNND-3	7107	10726	125262	91665	4943	4	24.57
TNLT	7398	11672	352199	339081	6808	1	55.61
LMVM	121314	127189	127189	-	-	-	39.59
CG	109166	325994	325994	-	-	-	75.72

Test 25 – Truncated Newton line search methods

Method	NIT	NFV	NFG	NCG	NCN	NCP	Time
TN	14680	16253	216097	170010	-	-	50.22
TNLM	6888	8106	156903	134956	6888	-	51.23
TNVM-1	14138	15501	222231	178150	12704	9	63.27
TNVM-2	13165	14395	190084	149188	13034	23	53.39
TNVM-3	12166	13411	181085	143146	12008	19	68.94
TNND-1	10298	11508	170121	127525	6139	10	32.42
TNND-2	14389	15960	159446	85337	2526	1	36.92
TNND-3	14303	15834	204090	116125	1544	2	53.01
TNLT	14208	15809	221636	179803	5914	1	61.34
LMVM	73754	77355	77355	-	-	-	31.38
CG	87687	277138	277138	-	-	-	75.81

Test 11 – Truncated Newton trust region methods

References

[1] I. Bongartz, A.R. Conn, N. Gould, P.L. Toint: CUTE: constrained and unconstrained testing environment. ACM Transactions on Mathematical Software **21** (1995), 123-160.

[2] R.H.Byrd, J.Nocedal, R.B.Schnabel: Representation of quasi-Newton matrices and their use in limited memory methods. Mathematical Programming **63** (1994) 129-156.

[3] T.F.Coleman, J.J.Moré: Estimation of sparse Hessian matrices and graph coloring problems. Mathematical Programming **28** (1984) 243-270.

[4] Conn A.R., Gould N.I.M, Toint P.L.: *Trust-Region Methods* SIAM, Philadelphia, 2000.

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

[6] Dembo, R.S, Steihaug T.: Truncated Newton algorithms for large-scale optimization. Math. Programming **26** (1983) 190-212.

[7] Fasano G.: Planar-conjugate gradient algorithm for large scale unconstrained optimization. Part 1: Theory. Journal of Optimization Theory and Applications **125** (2005) 523-541.

[8] Fasano G.: Planar-conjugate gradient algorithm for large scale unconstrained optimization. Part 1: Applications Journal of Optimization Theory and Applications **125** (2005) 543-558.

[9] Fasano G., Roma M.: Preconditioning Newton-Krylov methods in nonconvex large scale optimization. Report DIS 01-2007, Dipartimento di Informatica e Sistemistica "A. Ruberti" SAPIENZA - Universita di Roma, 2007.

[10] P.E.Gill, W.Murray: Newton type methods for unconstrained and linearly constrained optimization. Math. Programming, **7** (1974), 311-350.

[11] Golub G.H., Meurant G.: *Matrices, moments and quadrature with applications*. Princeton University Press, Princeton, 2010.

[12] Golub G.H., Van Loan C.: *Matrix Computations*. Academic Press, New York, 1981.

[13] Higham N.J.: *Accuracy and Stability of Numerical Algorithms*. SIAM, Philadelphia, 2002.

[14] Lukšan L. Matonoha C., Vlček J.: Sparse test problems for unconstrained optimization. Report V-1064, Institute of Computer Science AS CR, Prague, 2010.

[15] Lukšan L. Matonoha C., Vlček J.: Sparse test problems for unconstrained optimization. Report V-1081, Institute of Computer Science AS CR, Prague, 2010.

[16] L.Lukšan, J.Vlček: Recursive formulation of limited memory variable metric methods Report V-1059, Institute of Computer Science AS CR, Prague, 2010.

[17] Matthies H., Strang G.: The solution of nonlinear finite element equations. Int. J. for Numerical Methods in Engineering **14** (1979), 1613-1623.

[18] Morales J.L., Nocedal J.: Automatic preconditioning by limited memory quasi-Newton updating. SIAM J. Optimization **10** (2000), 1079-1096.

[19] Nash S.G.: Newton-type minimization via Lanczos method. SIAM Journal on Numerical Analysis **21** (1984), 770-788.

[20] Nash S.G.: Preconditioning of truncated-Newton methods. SIAM Journal on Scientific and Statistical Computation **6** (1985), 599-616.

[21] Nocedal J.: Updating quasi-Newton matrices with limited storage. Mathematics of Computation **35** (1980), 773-782.

[22] Nocedal J., Wright S.J.: *Numerical Optimization*. Springer, New York, 1999.

[23] O'Leary D.P.: A discrete Newton algorithm for minimizing a function of many variables. Mathematical Programming **23** (1983), 20-33.

[24] Paige C.C., Saunders M.A.: Solution of sparse indefinite systems of linear equations. SIAM Journal on Numerical Analysis **12** (1975), 617-629.

[25] Roma M.: Dynamic scaling based preconditioning for truncated Newton methods in large scale unconstrained optimization. Optimization Methods and Software **20** (2005), 693-713.

[26] Saad Y.: *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, 2003.

[27] Steihaug T.: Damped inexact quasi-Newton methods. Report MASC TR 81-3, Department of Mathematical Sciences, Rice University, Houston, Texas 1984.

[28] Steihaug T.: The conjugate gradient method and trust regions in large-scale optimization. SIAM Journal on Numerical Analysis **20** (1983) 626-637.

[29] Toint P.L.: Towards an efficient sparsity exploiting Newton method for minimization. In: Sparse Matrices and Their Uses (I.S.Duff, ed.), Academic Press, London 1981, 57-88.