Institute of Computer Science
Academy of Sciences of the Czech Republic

A limited-memory optimization method using the infinitely many times repeated BNS update and conjugate directions

Jan Vlček[a], Ladislav Lukšan[a, b]

[a]Institute of Computer Science, Czech Academy of Sciences,
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic and
[b]Technical University of Liberec, Hálkova 6, 461 17 Liberec, Czech Republic

Technical report No. V 1245

March 2018

Pod Vodárenskou věží 2, 182 07 Prague 8 phone: +420 266 052 083, fax: +420 286 585 789,
e-mail:luksan@cs.cas.cz, vlcek@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

A limited-memory optimization method using the infinitely many times repeated BNS update and conjugate directions[1]

Jan Vlček[a], Ladislav Lukšan[a, b]

[a]Institute of Computer Science, Czech Academy of Sciences,
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic and
[b]Technical University of Liberec, Hálkova 6, 461 17 Liberec, Czech Republic

Technical report No. V 1245

March 2018

Abstract:

To improve the performance of the limited-memory variable metric L-BFGS method for large scale unconstrained optimization, repeating of some BFGS updates was proposed in [1, 2]. But the suitable extra updates need to be selected carefully, since the repeating process can be time consuming. We show that for the limited-memory variable metric BNS method, matrix updating can be efficiently repeated infinitely many times under some conditions, with only a small increase of the number of arithmetic operations. The limit variable metric matrix can be written as a block BFGS update [22], which can be obtained by solving of some low-order Lyapunov matrix equation. The resulting method can be advantageously combined with methods based on vector corrections for conjugacy, see e.g. [21]. Global convergence of the proposed algorithm is established for convex and sufficiently smooth functions. Numerical experiments demonstrate the efficiency of the new method.

Keywords:
Unconstrained minimization, variable metric methods, limited-memory methods, the repeated BFGS update, global convergence, numerical results

[1]This work was supported by the Institute of Computer Science of the CAS (RVO: 67985807).

1 Introduction

In this report we assume that the problem function $f : \mathcal{R}^N \to \mathcal{R}$ is differentiable and propose a new limited-memory variable metric (VM) method for large scale unconstrained optimization

$$\min f(x) : \; x \in \mathcal{R}^N,$$

based on the widely used BNS method [4] and on vector corrections for conjugacy [20, 21].

The best known limited-memory VM methods are the L-BFGS (see [11, 18], [13] - subroutine PLIS) and BNS methods. A brief description of these methods is given in Section 2. Their performance can be improved e.g. by the methods based on vector corrections for conjugacy [20, 21]. The method in [21] uses correction vectors from more previous iterations and gives better results, but the conditions for the choice of these vectors are complicated, since the corrections can deteriorate stability and require extra arithmetic operations.

Another way how to improve the performance of the L-BFGS (or equivalently BNS) method was proposed in [1, 2], where some BFGS updates are computed several times. However, such repeating process can be time consuming and thus the suitable extra updates need to be selected carefully.

We show in Section 3 that the BNS updating can be repeated infinitely many times under some conditions in such a way that the comparative increase in the computational time is very small for N large. The limit VM matrix can be written as the block BFGS update, investigated in [22]:

$$H_+ = S\,(A\Theta)^{-1}S^T + \left(I - SA^{-T}Y^T\right)H^I\left(I - YA^{-1}S^T\right), \quad A = S^TY, \qquad (1.1)$$

$\Theta \in \mathcal{R}^{m\times m}$, A, Θ nonsingular, $\Theta = I$ for A symmetric. For quadratic functions this approach gives the same result as methods based on vector corrections for conjugacy, which for H^I symmetric positive definite and $\Theta = I$ represent the best improvement of convergence in some sense [21, 22]. For general functions, the matrix $(A\Theta)^{-1}$ is the solution to a suitable Lyapunov matrix equation, see Section 3. In Section 4 we show that the degree of this equation can be decreased of unit always, or more by combination with methods based on vector corrections for conjugacy. Vice versa, the combination of these methods with the repeated BNS update enables us to reduce the number of correction vectors and to simplify the conditions for their choice significantly to obtain comparable numerical results. In comparison with method [22], we need no complicated formation of suitable blocks.

In Section 5 we present an efficient method for solving of the corresponding low-order Lyapunov equations numerically. The application to the limited-memory VM methods and the corresponding algorithm are described in Section 6. Global convergence of the algorithm is established in Section 7 and numerical results are reported in Section 8.

We will denote by $\|\cdot\|_F$ the Frobenius matrix norm, by $\|\cdot\|$ the spectral matrix norm, by $|\cdot|$ the size of both scalars and vectors (the Euclidean vector norm) and by $[A]_{n_1}^{n_2}$ the principal submatrix of A with both row and column indices of entries from n_1 to n_2.

2 The standard BNS method

The BNS method belongs to the VM or quasi-Newton (QN) line search iterative methods [8, 16]. They start with an initial point $x_0 \in \mathcal{R}^N$ and generate iterations $x_{k+1} \in \mathcal{R}^N$ by the process $x_{k+1} = x_k + s_k$, $s_k = t_k d_k$, $k \ge 0$, where d_k is the direction vector and the stepsize $t_k > 0$ is chosen in such a way that

$$f_{k+1} - f_k \le \varepsilon_1 t_k g_k^T d_k, \qquad g_{k+1}^T d_k \ge \varepsilon_2 g_k^T d_k \tag{2.1}$$

(the Wolfe line search conditions, see e.g. [19]), $0 < \varepsilon_1 < 1/2$, $\varepsilon_1 < \varepsilon_2 < 1$, $f_k = f(x_k)$ and $g_k = \nabla f(x_k)$. Usually $d_k = -H_k g_k$ with a symmetric positive definite matrix H_k; typically H_0 is a multiple of I and H_{k+1} is obtained from H_k by a VM update to satisfy the QN condition (secant equation)

$$H_{k+1} y_k = s_k \tag{2.2}$$

(see [8, 16]), where $y_k = g_{k+1} - g_k$, $k \ge 0$. For $k \ge 0$ we denote

$$B_k = H_k^{-1}, \qquad b_k = s_k^T y_k,$$

(note that $b_k > 0$ for $g_k \ne 0$ by (2.1)). To simplify notation we frequently omit index k and replace index $k+1$ by symbol $+$, index $k-1$ by symbol $-$ and index $k-2$ by symbol $=$.

Among VM methods, the BFGS method [8, 16, 19], belongs to the most efficient ones; the update formula preserves positive definite VM matrices and can be written in the following quasi-product form

$$H_+ = (1/b) s s^T + \Big(I - (1/b) s y^T\Big) H \Big(I - (1/b) y s^T\Big). \tag{2.3}$$

The BFGS method can be easily modified for large-scale optimization; the BNS and L-BFGS methods represent its well-known limited-memory adaptations. In every iteration we choose $H_k^I \in \mathcal{R}^{N \times N}$ (usually $H_k^I = \zeta_k I$, $\zeta_k > 0$) and recurrently update H_k^I (without forming an approximation of the inverse Hessian matrix explicitly) by the BFGS method, using m couples of vectors $(s_{k-\tilde{m}}, y_{k-\tilde{m}}), \dots, (s_k, y_k)$ successively, where

$$\tilde{m} = \min(k, \hat{m} - 1), \quad m = \tilde{m} + 1, \quad k \ge 0 \tag{2.4}$$

and $\hat{m} > 1$ is a given parameter. In case of the BNS method, the BNS update can be expressed either in the form [4],

$$H_+ = H^I + \Big[S, H^I Y\Big] \begin{bmatrix} R^{-T}(D + Y^T H^I Y) R^{-1} & -R^{-T} \\ -R^{-1} & 0 \end{bmatrix} \begin{bmatrix} S^T \\ Y^T H^I \end{bmatrix},$$

where $S_k = [s_{k-\tilde{m}}, \dots, s_k]$, $Y_k = [y_{k-\tilde{m}}, \dots, y_k]$, $D_k = \mathrm{diag}[b_{k-\tilde{m}}, \dots, b_k]$, $(R_k)_{i,j} = (S_k^T Y_k)_{i,j}$ for $i \le j$, $(R_k)_{i,j} = 0$ otherwise (an upper triangular matrix), $k \ge 0$, or in the form

$$H_+ = S R^{-T} D R^{-1} S^T + \Big(I - S R^{-T} Y^T\Big) H^I \Big(I - Y R^{-1} S^T\Big). \tag{2.5}$$

This indicates that for $H^I = \zeta I$ the direction vector (and subsequently an auxiliary vector $Y^T H_+ g_+$) can be calculated efficiently without computing of H_+ explicitly by

$$-H_+ g_+ = -\zeta g_+ - S\Big[R^{-T}\Big((D + \zeta Y^T Y) R^{-1} S^T g_+ - \zeta Y^T g_+\Big)\Big] + Y\Big[\zeta R^{-1} S^T g_+\Big], \tag{2.6}$$

$$Y^T H_+ g_+ = \zeta Y^T g_+ + Y^T S\Big[R^{-T}\Big((D + \zeta Y^T Y) R^{-1} S^T g_+ - \zeta Y^T g_+\Big)\Big] - Y^T Y\Big[\zeta R^{-1} S^T g_+\Big], \tag{2.7}$$

see [4], where in the square brackets we multiply by low-order matrices.

3 The repeated BNS update

We will derive the infinitely many times repeated standard BNS update (2.5) of an arbitrary matrix H^I and describe its properties and relations to various forms of the discrete and continuous Lyapunov matrix equations, see e.g. [9].

Theorem 3.1. *Let $H^I \in \mathcal{R}^{N\times N}$, $A = S^TY$, $C = AR^{-1} - I$, $\bar{H}_0 = H^I$ and*

$$\bar{H}_{i+1} = SR^{-T}DR^{-1}S^T + \Big(I - SR^{-T}Y^T\Big)\bar{H}_i\Big(I - YR^{-1}S^T\Big), \quad i = 0, 1, \dots \tag{3.1}$$

Suppose that the spectral radius $\varrho(C) < 1$. Then the matrices $I + C$, A are nonsingular and the repeated BNS update H_+ of H^I can be defined by $H_+ = \lim_{i\to\infty} \bar{H}_i$ and satisfies

$$H_+ = SX^*S^T + \Big(I - SA^{-T}Y^T\Big)H^I\Big(I - YA^{-1}S^T\Big), \tag{3.2}$$

where X^* is the unique and symmetric positive definite solution to the discrete Lyapunov (or Stein) matrix equation

$$X^* = C^TX^*C + M, \quad M = R^{-T}DR^{-1}, \tag{3.3}$$

which can be equivalently written as

$$A^TX^*(2R - A) + (2R - A)^TX^*A = 2D, \tag{3.4}$$

or, denoting $\tilde{X} = R^TX^*R$,

$$\tilde{X} = \tilde{C}^T\tilde{X}\tilde{C} + D, \quad \tilde{C} = R^{-1}CR = R^{-1}(A - R). \tag{3.5}$$

Proof. First we prove

$$\bar{H}_i = SX_iS^T + \Big(I - SV_i^TY^T\Big)H^I\Big(I - YV_iS^T\Big), \tag{3.6}$$

where

$$X_i = M + C^TMC + \dots + (C^T)^{i-1}MC^{i-1}, \tag{3.7}$$

$$V_i = R^{-1}\Big(I - C + C^2 + \dots + (-C)^{i-1}\Big), \tag{3.8}$$

$i = 1, 2, \dots$, by induction. For $i = 1$ it is true by (3.1). Suppose that (3.6) holds for some $i \ge 1$. From (3.1) and (3.6) we obtain

$$\bar{H}_{i+1} = SR^{-T}DR^{-1}S^T + \Big(I - SR^{-T}Y^T\Big)SX_iS^T\Big(I - YR^{-1}S^T\Big) + K^TH^IK, \tag{3.9}$$

where

$$\begin{aligned} K &= \Big(I - YV_iS^T\Big)\Big(I - YR^{-1}S^T\Big) = I - Y\Big(R^{-1} + V_i(I - AR^{-1})\Big)S^T \\ &= I - Y\Big(R^{-1} - V_iC\Big)S^T = I - YV_{i+1}S^T \end{aligned}$$

by (3.8). Using (3.7), relation (3.9) can be rewritten as

$$\bar{H}_{i+1} - K^TH^IK = S\Big[M + \Big(I - R^{-T}A^T\Big)X_i\Big(I - AR^{-1}\Big)\Big]S^T = S\Big[M + C^TX_iC\Big]S^T = SX_{i+1}S^T,$$

which gives (3.6) for $i + 1$ and completes the induction.

Let $\varrho(C) < 1$. Then the matrices $I + C = AR^{-1}$, A are obviously nonsingular and using Theorems 1.7.13–1.7.14 in [7], we obtain $\lim_{i\to\infty} C^i = 0$ and $\lim_{i\to\infty} V_i = R^{-1}(I + C)^{-1} = A^{-1}$ by (3.8). Since M is symmetric positive definite, we can use Theorem 7.6.2 in [7] to get that the series $M + C^TMC + (C^T)^2MC^2 + \dots$ in (3.7) converges to some X^*, which obviously satisfies (3.3). Subsequently, (3.6) yields (3.2).

Since all matrices $(C^T)^i M C^i$, $i = 1, 2, \ldots$, are symmetric positive semidefinite and M is symmetric positive definite, also $X^* = M + \sum_{i=1}^{\infty} (C^T)^i M C^i$ is symmetric positive definite. If $X' = C^T X' C + M$ with some $X' \in \mathcal{R}^{m \times m}$, then for $\Delta = X^* - X'$ we have $\Delta = C^T \Delta C$, thus also $\Delta = (C^T)^i \Delta C^i$, $i = 1, 2, \ldots$, which implies $\Delta = 0$ by $\lim_{i \to \infty} C^i = 0$.

From (3.3) we obtain equivalently

$$\begin{aligned} D &= R^T M R = R^T X^* R - R^T C^T X^* C R = R^T X^* R - (A - R)^T X^* (A - R) \\ &= A^T X^* R + R^T X^* A - A^T X^* A = A^T X^* \Big(R - \frac{1}{2} A \Big) + \Big(R - \frac{1}{2} A \Big)^T X^* A, \end{aligned}$$

by $C = AR^{-1} - I$, which gives (3.4). Clearly equations (3.3) and (3.5) are equivalent. □

The following lemma indicates why we expect better results for the repeated BNS update compared with the standard BNS update in case that A is near to a symmetric positive definite matrix, e.g. close to a local minimum.

Lemma 3.1. *Let the assumptions of Theorem 3.1 be satisfied, the repeated BNS update H_+ of H be given by (3.2). Then $H_+ Y = S X^* A$. If the matrix A is symmetric, then it is positive definite, $X^* = A^{-1}$ and $H_+ Y = S$, i.e. the QN conditions with all stored difference vectors are satisfied.*

Proof. The first assertion follows from (3.2). Let $A = A^T$, i.e. $A - R = R^T - D$. Setting $X^* A = I$ to (3.4), we get

$$A^T X^* (R + R - A) + (R + R - A)^T X^* A = (R - R^T + D) + (R^T - R + D) = 2D,$$

thus the matrix $X^* = A^{-1}$ is the solution to (3.4), unique and symmetric positive definite by Theorem 3.1. Furthermore, from $H_+ Y = S X^* A$ we obtain $H_+ Y = S$. □

Most of the numerical methods to solve the discrete Lyapunov equation use some transformation to the continuous Lyapunov equation [9]. E.g. (3.4) can be immediately rewritten as

$$E \tilde{Z} + \tilde{Z}^T E = 2D, \quad E = A^T X^* A, \quad \tilde{Z} = 2A^{-1} R - I, \tag{3.10}$$

but the following transformations appear to be more advantageous. In this section we will suppose that there is the unique factorization $A = UL$, where U is an upper triangular matrix with nonzero diagonal entries and L a lower triangular matrix with unit diagonal entries. A sufficient condition for the existence of this factorization is $\det[A]_i^m \neq 0$ (principal minors, see the end of Section 1), $i = 1, \ldots, m$, by Theorem 1.4.3 in [7], considering the rows and columns of A arranged in reverse order.

Using this factorization, we can equivalently rewrite equation (3.4) successively as

$$\begin{aligned} U^T X^* (2R - UL) L^{-1} + L^{-T} (2R - UL)^T X^* U &= 2 L^{-T} D L^{-1}, \\ U^T X^* U (2U^{-1} R L^{-1} - I) + (2 L^{-T} R^T U^{-T} - I) U^T X^* U &= 2 L^{-T} D L^{-1}, \end{aligned}$$

i.e. as the Lyapunov equation

$$XZ + Z^T X = 2W, \quad X = U^T X^* U, \quad Z = 2U^{-1} R L^{-1} - I, \quad W = L^{-T} D L^{-1}. \tag{3.11}$$

Since $X = U^T R^{-T} \tilde{X} R^{-1} U$ by $\tilde{X} = R^T X^* R$, we can rewrite (3.11) also as

$$\tilde{X} R^{-1} U Z U^{-1} R + R^T U^{-T} Z^T U^T R^{-T} \tilde{X} = 2 R^T A^{-T} D A^{-1} R,$$

i.e. as the Lyapunov equation

$$\tilde{X}\tilde{Z} + \tilde{Z}^T\tilde{X} = 2\tilde{W}, \quad \tilde{W} = R^T A^{-T} D A^{-1} R, \tag{3.12}$$

in view of

$$R^{-1}UZU^{-1}R = R^{-1}U(2U^{-1}RL^{-1} - I)U^{-1}R = 2A^{-1}R - I = \tilde{Z}.$$

The order of equations (3.10) – (3.12) can be always decreased, see Section 4.

Some properties of solutions to (3.10) – (3.12) are given by the following lemma. Note that property $X = D^U$ for A symmetric is numerically advantageous e.g. when the matrix A is almost symmetric, thus we prefer to solve (3.11).

Lemma 3.2. *Let the Lyapunov equation (3.11) has a solution X (or (3.12) has a solution $\tilde{X}$, or (3.10) has a solution E), which is symmetric positive definite. Then this solution is unique. Moreover, let A be symmetric. Then $X = D^U$, where D^U is a diagonal matrix with the same diagonal entries as U, $\tilde{X} = R^T A^{-1} R$ and $E = A^T$.*

Proof. Since W is a symmetric positive definite matrix, the matrix Z is positive stable by Theorem 7.6.2 in [7], which yields the uniqueness of the solution to the Lyapunov equation (3.11) by Theorem 6.1.9 in [7], similarly for (3.12) and (3.10).

If $A = A^T$, Lemma 3.1 implies $X^* = A^{-1} = L^{-1}U^{-1}$, which yields $\tilde{X} = R^T X^* R = R^T A^{-1} R$, $E = A^T X^* A = A^T$ and $X = U^T L^{-1} \triangleq D^U$ by (3.11). The symmetry of A gives $UL = L^T U^T$, thus $L^{-T}U = U^T L^{-1}$, i.e. the upper triangular matrix $L^{-T}U$ is equal to the lower triangular matrix $U^T L^{-1} = D^U$, thus D^U is a diagonal matrix. Since L, L^{-1} have unit diagonal entries, $D^U = U^T L^{-1}$ has the same diagonal entries as U^T (or U). ☐

4 Relations to methods based on vector corrections

In this section we will show (Theorem 4.1) how the order of equations (3.10) – (3.12) can be decreased, if some lower-right-corner principal submatrix of order $\mu \geq 1$ of A is diagonal, e.g. by using vector corrections for conjugacy, which we describe in Section 4.1. Since every such submatrix of order 1 can be considered to be diagonal, we can always decrease the order of these equations and choose $\mu \geq 1$.

Moreover, in the following lemma we will show how the assumption $\varrho(C) < 1$ in Theorem 3.1 can be equivalently written in another form if some lower-right-corner principal submatrix of A is upper triangular.

Note that we can also always assume that $\mu < m$, since for $A = R$ we have $C = AR^{-1} - I = 0$, thus $X^* = M$ and update (3.2) is identical to (2.5).

Lemma 4.1. *Let the matrix R be given as in Section 2, $A = S^T Y$, $C = AR^{-1} - I$, $\tilde{C}$ by (3.5) and let A be partitioned in the form*

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{4.1}$$

with $A_{22} \in \mathcal{R}^{\mu \times \mu}$, $0 < \mu < m$. Let $\tilde{C}, R$ be partitioned in the same way and let $A_{22} = R_{22}$. Then $\tilde{C}_{12}$ and $\tilde{C}_{22}$ are null matrices and $\varrho(C) = \varrho(\tilde{C}) = \varrho(\tilde{C}_{11})$.

Proof. The matrices $C, \tilde{C}$ are similar, thus $\varrho(C) = \varrho(\tilde{C})$. Since μ of the latest columns of the matrices $A - R$, $\tilde{C} = R^{-1}(A - R)$ are null vectors by $A_{22} = R_{22}$ and (3.5), we deduce that $\tilde{C}_{12}$ and $\tilde{C}_{22}$ are null matrices, therefore $\varrho(\tilde{C}) = \varrho(\tilde{C}_{11})$. $\square$

Theorem 4.1. *Let the assumptions of Lemma 4.1 be satisfied and suppose that* $A = UL$, *where* U *is an upper triangular matrix with nonzero diagonal entries and* L *a lower triangular matrix with unit diagonal entries. Let the matrices* S, Y, D *be given as in Section 2,* X^* *by (3.3),* $\tilde{X}$ *by (3.5),* $E, W, \tilde{W}, X, Z, \tilde{Z}$ *by (3.10) – (3.12),* D^U *by Lemma 3.2 and denote* $X^I = (X^*)^{-1}$. *Let the matrices* $D, E, L, U, W, \tilde{W}, X, \tilde{X}, X^I, Z, \tilde{Z}$ *be partitioned in the same way as* A *in Lemma 4.1 and suppose that* $A_{22} = D_{22}$, $\varrho(C) < 1$ *and the columns of* S *are linearly independent. Then*

(a) X_{12}, X_{21}, $\tilde{X}_{12}$, $\tilde{X}_{21}$ *are null matrices,* $X_{22} = \tilde{X}_{22} = U_{22} = D_{22}$, $L_{22} = I$,

(b) $X^I_{12} = (X^I_{21})^T = A_{12}$, $E^T_{12} = E_{21} = A_{21}$, $X^I_{22} = E_{22} = D_{22}$,

(c) $X_{11}, \tilde{X}_{11}$ *are the unique solutions to the Lyapunov equations*

$$X_{11}Z_{11} + Z_{11}^T X_{11} = 2W_{11}, \quad \tilde{X}_{11}\tilde{Z}_{11} + \tilde{Z}_{11}^T \tilde{X}_{11} = 2\tilde{W}_{11}. \tag{4.2}$$

Proof. Let $J = [0, I]^T \in \mathcal{R}^{m\times\mu}$. The matrix H_+ given by (3.2) is always created by the successive BFGS updating of some matrix $H^A \in \mathcal{R}^{N\times N}$, using columns of SJ, YJ. Since $(SJ)^T YJ = A_{22} = D_{22}$ by assumption, by analogy with (2.5) we can write

$$H_+ = SJD_{22}^{-1}J^TS^T + (I - SJD_{22}^{-1}J^TY^T)H^A(I - YJD_{22}^{-1}J^TS^T)\,, \tag{4.3}$$

which yields $H_+YJ = SJ$. Using Lemma 3.1, we obtain $SJ = H_+YJ = SX^*AJ$ and thus

$$X^*AJ = J \tag{4.4}$$

by linear independency of columns of S. In view of $U^{-T}XL = (U^{-T}XU^{-1})(UL) = X^*A$ by the second relation in (3.11), we get $U^{-T}XLJ = X^*AJ = J$, i.e. $XLJ = U^TJ$. Since

$$U^TJ = \begin{bmatrix} U_{11}^T & 0 \\ U_{12}^T & U_{22}^T \end{bmatrix} \begin{bmatrix} 0 \\ I \end{bmatrix} = \begin{bmatrix} 0 \\ U_{22}^T \end{bmatrix}, \quad X(LJ) = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \begin{bmatrix} 0 \\ L_{22} \end{bmatrix} = \begin{bmatrix} X_{12}L_{22} \\ X_{22}L_{22} \end{bmatrix},$$

we get $X_{12} = 0$, $X_{21} = 0$ and $X_{22} = U_{22}^T L_{22}^{-1}$. From $A = UL$ and $A_{22} = D_{22}$ we have $U_{22}L_{22} = D_{22}$, i.e. the lower triangular matrix L_{22} is equal to the upper triangular matrix $U_{22}^{-1}D_{22}$, thus L_{22} is a diagonal matrix. But L_{22} has unit diagonal entries, therefore $L_{22} = I$, $U_{22} = D_{22}$ and $X_{22} = D_{22}$. Further, since μ of the latest columns of $\tilde{C}$, $\tilde{C}^T\tilde{X}\tilde{C} = \tilde{X} - D$ are null vectors by Lemma 4.1 and (3.5), $\tilde{X}_{12}$ and $\tilde{X}_{21}$ are null matrices and $\tilde{X}_{22} = D_{22}$.

Using (3.11), we obtain

$$2W = XZ + Z^TX = \begin{bmatrix} X_{11}Z_{11} & X_{11}Z_{12} \\ X_{22}Z_{21} & X_{22}Z_{22} \end{bmatrix} + \begin{bmatrix} Z_{11}^TX_{11} & Z_{21}^TX_{22} \\ Z_{12}^TX_{11} & Z_{22}^TX_{22} \end{bmatrix},$$

which gives the first equation in (4.2); the second equation can be obtained similarly, using (3.12). The uniqueness of solutions to the both equations follows in the same way as for the solution to (3.11) in the proof of Lemma 3.2.

Finally, from (4.4) we obtain on the one hand $X^IJ = AJ$ by $X^I = (X^*)^{-1}$ and on the other hand $EJ = A^TJ$ by $E = A^TX^*A$, which together gives (b) by $A_{22} = D_{22}$. $\square$

Corollary 4.1. *Let the assumptions of Theorems 3.1 and 4.1 be satisfied, the matrix H_+ given by (3.2) be symmetric nonsingular and $B_+ = H_+^{-1}$. Then*

$$(H_+Y - S)^T B_+(H_+Y - S) = \begin{bmatrix} E_{11}+X_{11}^I-A_{11}-A_{11}^T & 0 \\ 0 & 0 \end{bmatrix}. \tag{4.5}$$

Proof. Using Lemma 3.1 and (3.10), we obtain $H_+Y = SX^*A = SA^{-T}E$, and thus also $Y = B_+SX^*A$, i.e. $B_+S = YA^{-1}(X^*)^{-1}$, which gives $Y^TH_+Y = E$ and $S^TB_+S = (X_*)^{-1} = X^I$, therefore

$$(H_+Y - S)^T B_+(H_+Y - S) = E + X^I - A - A^T$$

and it suffices to use Theorem 4.1*(b)*. □

4.1 Application of corrections for conjugacy

Our numerical experiments indicate that the repeated BNS update (3.2) can improve the performance of the L-BFGS method and that this improvement can be increased if we also use vector corrections for conjugacy, see Section 1. Moreover, Theorem 4.1 shows that this combination can decrease the order of the corresponding Lyapunov matrix equation.

As we mentioned in Section 1, corrections for conjugacy are advantageous for quadratic functions, thus we should use these corrections whenever an objective function is close to a quadratic function, see Section 6. On the other hand, in [21] it was shown that correction vectors only from two preceding iterations can be sufficient, considering that these corrections can deteriorate stability and require additional arithmetic operations.

Since these corrections are performed before updating, we will consider all columns of S, Y to be possibly corrected and write $S_k = [\tilde{s}_{k-\tilde{m}}, \ldots, \tilde{s}_k]$, $Y_k = [\tilde{y}_{k-\tilde{m}}, \ldots, \tilde{y}_k]$, where we set $\tilde{s}_i = s_i$ and $\tilde{y}_i = y_i$, $i = k-\tilde{m}, \ldots, k$, if the corrections for conjugacy are not used.

To calculate corrected $\tilde{s}, \tilde{y}$, we define projections

$$P_1 = I - (1/\tilde{b}_-)\tilde{y}_-\tilde{s}_-^T, \quad P_2 = I - (1/\tilde{b}_=)\tilde{y}_=\tilde{s}_=^T, \quad P_{12} = P_1P_2, \tag{4.6}$$

for $\tilde{b}_- = \tilde{s}_-^T\tilde{y}_- \neq 0$, $\tilde{b}_= = \tilde{s}_=^T\tilde{y}_= \neq 0$. In view of the assumption $A_{22} = D_{22}$ of Theorem 4.1 we use projections P_2, P_{12} only if $\tilde{s}_-^T\tilde{y}_= = \tilde{s}_=^T\tilde{y}_- = 0$ (when $\tilde{s}_-, \tilde{y}_-$ were corrected in the previous iteration); in this case we have

$$P_1P_2 = \left(I - \frac{\tilde{y}_-\tilde{s}_-^T}{\tilde{b}_-}\right)\left(I - \frac{\tilde{y}_=\tilde{s}_=^T}{\tilde{b}_=}\right) = I - \frac{\tilde{y}_-\tilde{s}_-^T}{\tilde{b}_-} - \frac{\tilde{y}_=\tilde{s}_=^T}{\tilde{b}_=} = P_2P_1. \tag{4.7}$$

Subsequently we set

$$\tilde{s} = P_1^Ts = s - (s^T\tilde{y}_-/\tilde{b}_-)\tilde{s}_-, \quad \tilde{y} = P_1y = y - (\tilde{s}_-^Ty/\tilde{b}_-)\tilde{y}_- \tag{4.8}$$

in case of correction vectors only from one preceding iteration, or

$$\tilde{s} = P_{12}^Ts = s - \frac{s^T\tilde{y}_-}{\tilde{b}_-}\tilde{s}_- - \frac{s^T\tilde{y}_=}{\tilde{b}_=}\tilde{s}_=, \quad \tilde{y} = P_{12}y = y - \frac{\tilde{s}_-^Ty}{\tilde{b}_-}\tilde{y}_- - \frac{\tilde{s}_=^Ty}{\tilde{b}_=}\tilde{y}_= \tag{4.9}$$

in case of correction vectors from two preceding iterations. Both corrected and uncorrected difference vectors can be uniformly written in the form $\tilde{s} = s + S_P\sigma$, $\tilde{y} = y + Y_P\eta$, where $\sigma, \eta \in \mathcal{R}^{\tilde{m}}$ are suitable vectors and by S_P, Y_P (or by S_k^P, Y_k^P) we denote the submatrices of S, Y with columns from previous iterations, i.e. $S \triangleq [S_P, \tilde{s}]$, $Y \triangleq [Y_P, \tilde{y}]$.

The following lemma gives expressions for $\tilde{b} = \tilde{s}^T\tilde{y}$ and a damage of the QN conditions with non-corrected vectors caused by corrections (4.8)–(4.9) and shows that for this choice, the assumptions of Lemma 4.1 and Theorem 4.1 are satisfied. More precisely, we can choose $\mu = 3$ in case of correction vectors from two preceding iterations, $\mu = 2$ in case of correction vectors only from one preceding iteration, or $\mu = 1$ otherwise.

Lemma 4.2. *We have*

$$\tilde{b} = b - s^T\tilde{y}_- \, \tilde{s}_-^T y/\tilde{b}_-, \quad \tilde{s}^T\tilde{y}_- = \tilde{s}_-^T\tilde{y} = 0 \tag{4.10}$$

for $\tilde{s}, \tilde{y}$ given by (4.8), or

$$\tilde{b} = b - s^T\tilde{y}_- \, \tilde{s}_-^T y/\tilde{b}_- - s^T\tilde{y}_= \, \tilde{s}_=^T y/\tilde{b}_=, \quad \tilde{s}^T\tilde{y}_- = \tilde{s}^T\tilde{y}_= = \tilde{s}_-^T\tilde{y} = \tilde{s}_=^T\tilde{y} = 0 \tag{4.11}$$

for $\tilde{s}, \tilde{y}$ given by (4.9) with $\tilde{s}_-^T\tilde{y}_= = \tilde{s}_=^T\tilde{y}_- = 0$. Moreover, let $\tilde{H}_+$ be a matrix given by (2.5) with vectors s_i, y_i replaced by $\tilde{s}_i, \tilde{y}_i$, $i = k{-}\tilde{m}, \dots, k$. Then $\tilde{H}_+\tilde{y} = \tilde{s}$, $\tilde{H}_+\tilde{y}_- = \tilde{s}_-$ and

$$(\tilde{H}_+y - s)^T\tilde{H}_+^{-1}(\tilde{H}_+y - s) = (s^T\tilde{y}_- - \tilde{s}_-^T y)^2/\tilde{b}_- \tag{4.12}$$

for $\tilde{s}, \tilde{y}$ given by (4.8), or $\tilde{H}_+\tilde{y} = \tilde{s}$, $\tilde{H}_+\tilde{y}_- = \tilde{s}_-$, $\tilde{H}_+\tilde{y}_= = \tilde{s}_=$ and

$$(\tilde{H}_+y - s)^T\tilde{H}_+^{-1}(\tilde{H}_+y - s) = (s^T\tilde{y}_- - \tilde{s}_-^T y)^2/\tilde{b}_- + (s^T\tilde{y}_= - \tilde{s}_=^T y)^2/\tilde{b}_= \tag{4.13}$$

for $\tilde{s}, \tilde{y}$ given by (4.9) with $\tilde{s}_-^T\tilde{y}_= = \tilde{s}_=^T\tilde{y}_- = 0$.

Proof. Let $\tilde{s}, \tilde{y}$ be given by (4.8). Then $\tilde{b} = s^TP_1^2y = s^TP_1y = b - s^T\tilde{y}_- \, \tilde{s}_-^T y/\tilde{b}_-$ by $P_1^2 = P_1$ and the rest of (4.10) follows from $P_1\tilde{y}_- = P_1^T\tilde{s}_- = 0$.

Let $\tilde{s}, \tilde{y}$ be given by (4.9) with $\tilde{s}_-^T\tilde{y}_= = \tilde{s}_=^T\tilde{y}_- = 0$. Then $P_{12}^2 = P_1(P_2P_1)P_2 = P_1P_2 = P_{12}$ by (4.7), $P_1^2 = P_1$ and $P_2^2 = P_2$, thus $\tilde{b} = s^TP_{12}y = b - s^T\tilde{y}_- \, \tilde{s}_-^T y/\tilde{b}_- - s^T\tilde{y}_= \, \tilde{s}_=^T y/\tilde{b}_=$ and the rest of (4.11) follows from $P_1\tilde{y}_- = P_1^T\tilde{s}_- = P_2\tilde{y}_= = P_2^T\tilde{s}_= = 0$ and (4.7).

Similarly as we got $H_+YJ = SJ$ from (4.3) in the proof of Theorem 4.1, we prove that all QN conditions with corrected vectors are satisfied by (4.10)–(4.11). From $\tilde{H}_+\tilde{y} = \tilde{s}$ we directly obtain $\tilde{H}_+y - s = \tilde{s} - s - \tilde{H}_+(\tilde{y} - y)$. For $\tilde{s}, \tilde{y}$ given by (4.8) this yields $\tilde{H}_+y - s = [(s^T\tilde{y}_- - \tilde{s}_-^T y)/\tilde{b}_-]\tilde{s}_-$ by $\tilde{H}_+\tilde{y}_- = \tilde{s}_-$, which gives (4.12); similarly we get (4.13). □

5 Solution to the Lyapunov equation

Denoting $n = m - \mu$ (the order of e.g. A_{11}, see (4.1)), in this section we show how equations (4.2), which we can solve instead of the Lyapunov equations (3.11), (3.12), can be solved efficiently for small n; without loss of generality, we restrict ourselves to the first equation (4.2). We consider only the case $1 \le n < m$, see the comments before Lemma 4.1.

There are many methods how to solve the Lyapunov equations, see e.g. [9]. Since we want to have rounding errors small, we attempt to solve only linear systems whose order is as small as possible. To achieve this, we use a similar idea as in [10]. For $n > 2$ we choose some block of entries of X_{11}, calculate the remaining entries and then we obtain the appropriate values of chosen entries as the solution to some linear system. We will follow up here the case $m \le 5$, i.e. $n \le 4$, although our approach could be adapted also for greater n. If some partial computations are not feasible, we use the BNS update instead of the repeated BNS update. We will suppose here that $\varrho(C) = \varrho(\tilde{C}_{11}) < 1$ and thus equations (4.2) have the unique solution by Theorem 4.1.

The case $n \leq 2$ is described in Section 5.1, $n = 3$ and $n = 4$ in Section 5.2. In Section 5.3 we mention briefly the spectral radius checking of matrices of order $n \leq 4$. We will denote the entries of matrices X_{11}, W_{11}, Z_{11} (partitions of X, W, Z, see Theorem 4.1) in the following way: $X_{11} = [x_{ij}]_{i,j=1}^n$, $W_{11} = [w_{ij}]_{i,j=1}^n$, $Z_{11} = [z_{ij}]_{i,j=1}^n$.

5.1 Solution for $n = 1$ and $n = 2$

For $n = 1$ we have $x_{11} = w_{11}/z_{11}$ by (4.2). For $n = 2$, the first equation in (4.2) can be written as

$$2W_{11} = X_{11}Z_{11} + Z_{11}^T X_{11} = \begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} + \begin{bmatrix} z_{11} & z_{21} \\ z_{12} & z_{22} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix},$$

which yields the system

$$\begin{aligned} x_{11}z_{11} + x_{12}z_{21} &= w_{11}\,, \\ x_{11}z_{12} + x_{12}z_{22} + z_{11}x_{12} + z_{21}x_{22} &= 2w_{12}\,, \\ x_{12}z_{12} + x_{22}z_{22} &= w_{22}\,. \end{aligned} \tag{5.1}$$

From the first and third equation we obtain

$$x_{11} = (w_{11} - x_{12}z_{21})/z_{11}, \quad x_{22} = (w_{22} - x_{12}z_{12})/z_{22}, \tag{5.2}$$

which together with the second equation (5.1) gives

$$x_{12} = \frac{2w_{12}z_{11}z_{22} - w_{11}z_{12}z_{22} - w_{22}z_{21}z_{11}}{(z_{11} + z_{22})(z_{11}z_{22} - z_{12}z_{21})}. \tag{5.3}$$

The values x_{11}, x_{22} can be obtained from (5.2).

5.2 Solution for $n = 3$ and $n = 4$

In a similar way as (5.1), for $n = 3$ from the first equation in (4.2) we obtain the system

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} + \begin{bmatrix} z_{11} & z_{21} \\ z_{12} & z_{22} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} + \begin{bmatrix} x_{13} \\ x_{23} \end{bmatrix} \begin{bmatrix} z_{31} & z_{32} \end{bmatrix} + \begin{bmatrix} z_{31} \\ z_{32} \end{bmatrix} \begin{bmatrix} x_{13} & x_{23} \end{bmatrix} = 2 \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & w_{22} \end{bmatrix}, \tag{5.4}$$

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \begin{bmatrix} z_{13} \\ z_{23} \end{bmatrix} + \begin{bmatrix} x_{13} \\ x_{23} \end{bmatrix} z_{33} + \begin{bmatrix} z_{11} & z_{21} \\ z_{12} & z_{22} \end{bmatrix} \begin{bmatrix} x_{13} \\ x_{23} \end{bmatrix} + \begin{bmatrix} z_{31} \\ z_{32} \end{bmatrix} x_{33} = 2 \begin{bmatrix} w_{13} \\ w_{23} \end{bmatrix}, \tag{5.5}$$

$$\begin{bmatrix} x_{13} & x_{23} \end{bmatrix} \begin{bmatrix} z_{13} \\ z_{23} \end{bmatrix} + 2x_{33}z_{33} + \begin{bmatrix} z_{13} & z_{23} \end{bmatrix} \begin{bmatrix} x_{13} \\ x_{23} \end{bmatrix} = 2w_{33}\,. \tag{5.6}$$

Choosing values x_{13}, x_{23}, we can calculate the coresponding x_{11}, x_{12}, x_{22} from (5.4) and x_{33} from (5.6), using results in Section 5.1. We denote $W_{(i)}$, $i = 0, 1, 2$, vectors equal to the left side of (5.5) for the following choices of x_{13}, x_{23}: $W_{(0)}$ for $x_{13} = x_{23} = 0$, $W_{(1)}$ for $x_{13} = 1$, $x_{23} = 0$, $W_{(2)}$ for $x_{13} = 0$, $x_{23} = 1$. Then the appropriate values x_{13}, x_{23} satisfy the linear system (of order two)

$$\left(W_{(1)} - W_{(0)}\right)x_{13} + \left(W_{(2)} - W_{(0)}\right)x_{23} = 2\begin{bmatrix} w_{13} & w_{23} \end{bmatrix}^T - W_{(0)} \tag{5.7}$$

by the linearity of the problem. Note that the vectors $W_{(1)}-W_{(0)}$, $W_{(2)}-W_{(0)}$, $W_{(0)}$ can be calculated directly (i.e. without computing $W_{(1)}$, $W_{(2)}$), utilizing (5.2)–(5.3).

For $n=4$ we can proceed in a similar way, considering that the matrices X_{11}, Z_{11}, W_{11} are partitioned in the form

$$X_{11} = \begin{bmatrix} \bar{X}_1 & \bar{X}_2 \\ \bar{X}_2^T & \bar{X}_3 \end{bmatrix}, \quad Z_{11} = \begin{bmatrix} \bar{Z}_1 & \bar{Z}_2 \\ \bar{Z}_3 & \bar{Z}_4 \end{bmatrix}, \quad W_{11} = \begin{bmatrix} \bar{W}_1 & \bar{W}_2 \\ \bar{W}_2^T & \bar{W}_3 \end{bmatrix}, \tag{5.8}$$

where all these partitions are square blocks of order two. We choose entries of $\bar{X}_2$, i.e. the values $x_{13}, x_{14}, x_{23}, x_{24}$, and calculate $\bar{X}_1, \bar{X}_3$ from

$$\bar{X}_1\bar{Z}_1 + \bar{X}_2\bar{Z}_3 + \bar{Z}_1^T\bar{X}_1 + \bar{Z}_3^T\bar{X}_2^T = 2\bar{W}_1, \tag{5.9}$$

$$\bar{X}_2^T\bar{Z}_2 + \bar{X}_3\bar{Z}_4 + \bar{Z}_2^T\bar{X}_2 + \bar{Z}_4^T\bar{X}_3^T = 2\bar{W}_3 \tag{5.10}$$

by (4.2), using results in Section 5.1. Then similarly as above we calculate vectors $W_{(0)}, \ldots, W_{(4)}$ from

$$\bar{X}_1\bar{Z}_2 + \bar{X}_2\bar{Z}_4 + \bar{Z}_1^T\bar{X}_2 + \bar{Z}_3^T\bar{X}_3 = 2\bar{W}_2 \tag{5.11}$$

and values $x_{13}, x_{14}, x_{23}, x_{24}$ from the system of order 4 similar to (5.7).

5.3 Spectral radius checking for low-order matrices

Our proof of global convergence for the repeated BNS update is based on the assumption $\|R_{11}\tilde{C}_{11}R_{11}^{-1}\|_F \le \rho$, $\rho \in (0,1)$, see Section 7, which implies $\varrho(C) < 1$ by Lemma 4.1. Still, the spectral radius can be efficiently tested for matrices of order $n \le 4$, e.g. by calculating coefficients of a characteristic polynomial $a_0\lambda^4 + a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4$ as sums of all its principal minors of the corresponding order by Theorem 1.6.2 in [7] and then using the transformation $\lambda = (z+1)/(z-1)$ and Theorem 7.6.3 (Routh-Hurwitz) in [7].

6 Implementation

In this section we assume that $H^I = \zeta I$, $\zeta = s^Ty/y^Ty > 0$, and use results from the previous sections to implement a modified BNS method. It consists in replacing some difference vectors s, y by the corrected vectors $\tilde{s} = s + S_P\sigma$, $\tilde{y} = y + Y_P\eta$, $\sigma, \eta \in \mathcal{R}^{\tilde{m}}$, using values from one or two preceding iterations, see Section 4.1, and then replacing some BNS updates (2.5) by the repeated BNS updates (3.2), where X^* is given by (3.11), thus by the first equation in (4.2). We also denote by S_P, Y_P (or S_k^P, Y_k^P) the submatrices of S, Y with columns from previous iterations.

As we mentioned in Section 4.1, we should use corrections for conjugacy whenever an objective function is close to a quadratic function. As a measure of the deviation in points $x_{k-\tilde{m}}, \ldots, x_k, k > 0$, e.g. the values (zero for quadratic functions)

$$\Delta_i^k = (\tilde{s}_i^T y_k - s_k^T \tilde{y}_i)^2 / (\tilde{b}_i\, b_k), \quad i \in \{k-\tilde{m}, \ldots, k-1\}, \quad k > 0, \tag{6.1}$$

could serve. In view of Lemma 4.2, these values also indicate whether the QN condition with corrected difference vectors is a good substitute for the QN condition with non-corrected vectors, i.e. they give a relative damage of the QN condition for the non-corrected vectors s, y caused by these corrections.

Principally, we do not use the vectors $\tilde{s}_i, \tilde{y}_i$, $i \in \{k-2, k-1\}$, for the correction of s, y if $\tilde{b} < \delta_1 b$, $\delta_1 \in (0,1)$, since satisfaction of $\tilde{b} > 0$ is required for the BFGS method with corrected difference vectors to have VM matrices symmetric positive definite, see e.g. [19], and too small $\tilde{b}$ can deteriorate stability. Besides, we do not use the vectors $\tilde{s}_-, \tilde{y}_-$ if $\Delta^k_{k-1} > \delta_2$, $\delta_2 \in (0,1)$, to have a relative damage of the QN condition small or if $\max[|\tilde{s}_-|/|s_-|, |\tilde{y}_-|/|y_-|] > \Delta$, $\Delta > 1$, due to our proof of global convergence. As for the vectors $\tilde{s}_=, \tilde{y}_=$, we do not use them if $\Delta^k_{k-1} + \Delta^k_{k-2} > \delta_2$, if the vectors $\tilde{s}_-, \tilde{y}_-$ was not corrected, see Section 4.1, or if the benefit of the corrections would be too small, to increase the efficiency of our method. It was shown in [21] that the ratio $b/\tilde{b}$ is a good indicator of this benefit. We regard the benefit of corrections using $\tilde{s}_=, \tilde{y}_=$ as sufficient if $\tilde{b}^{(1)}/\tilde{b}^{(2)} > 1 + \delta_3$, $\delta_3 \in (0,1)$, where $\tilde{b}^{(i)}$ is the value of $\tilde{b}$ (given by (4.10) or (4.11)), corresponding to the correction with values from i preceding iterations, $i \in \{1, 2\}$.

To have A sufficiently symmetric in view of Lemmas 3.1–3.2, the repeated BNS update is not used if $\sum_{i,j\in\mathcal{I}_k} \tilde{\Delta}^j_i > \delta_4$, $\delta_4 \in (0,1)$, where

$$\tilde{\Delta}^j_i = (\tilde{s}_i^T \tilde{y}_j - \tilde{s}_j^T \tilde{y}_i)^2 / (\tilde{b}_i \, \tilde{b}_j), \quad i, j \in \{k - \tilde{m}, \ldots, k\} \triangleq \mathcal{I}_k, \; i \neq j, \quad k > 0, \tag{6.2}$$

or if absolute value of some diagonal entry of U would be smaller than $\delta_5 \operatorname{Tr} A$, $\delta_5 \in (0,1)$, during the factorization $A = UL$, see Procedure 6.1 (this case occurred very rarely in our numerical experiments).

We use this update only if $\mu < m$, see comments before Lemma 4.1. Due to our proof of global convergence, also only if all $\tilde{b}_i$, $i \in \mathcal{I}_k$, are greater than $\varepsilon_D \|A\|_F$ and $\|R_{11} \tilde{C}_{11} R_{11}^{-1}\|_F \le \rho$, $\rho \in (0,1)$ (we use the notation as in Lemma 4.1), which replaces the condition $\varrho(C) < 1$ and obviously implies $\varrho(\tilde{C}_{11}) \le \rho$. Note that the condition $\|R_{11} \tilde{C}_{11} R_{11}^{-1}\|_F \le \rho$ appears to be slightly more advantageous than e.g. $\|\tilde{C}_{11}\|_F \le \rho$. Finally, we found empirically that this update should not be used for $m < \hat{m}$ (initially or after contingent restarts), since this can deteriorate the robustness of the method.

For the repeated BNS update with $H^I = \zeta I$, the direction vector and an auxiliary vector $Y^T H_+ g_+$ can be calculated efficiently by

$$-H_+ g_+ = -\zeta g_+ - S\Big[U^{-T}\Big((X + \zeta L^{-T} Y^T Y L^{-1}) q - \zeta L^{-T} Y^T g_+\Big)\Big] + Y\Big[\zeta L^{-1} q\Big], \tag{6.3}$$

$$Y^T H_+ g_+ = \zeta Y^T g_+ + Y^T S\Big[U^{-T}\Big((X + \zeta L^{-T} Y^T Y L^{-1}) q - \zeta L^{-T} Y^T g_+\Big)\Big] - Y^T Y\Big[\zeta L^{-1} q\Big], \tag{6.4}$$

$q = U^{-1} S^T g_+$, in view of (3.2) and (3.11).

To improve the readability of the main algorithm, we first present two auxiliary procedures. Procedure 6.1, based on Lemma 4.1 in [22], is used for the factorization $A = UL$, see Section 3. Procedure 6.2 serves for updating of basic matrices $S^T Y = A$, $Y^T Y$ and is similar to the algorithm given in [4] for updating of matrices $D, R, Y^T Y$ in (2.5). In comparison with the standard BNS method, where only the diagonal of A and the part of A above the diagonal is used, we need the whole matrix A here, therefore we use an additional vector $Y_P^T s = -t \, Y_P^T H g$ (see also Algorithm 6.1) to have the number of arithmetic operations approximately the same as for the corresponding algorithm in [4].

Procedure 6.1 *(UL factorization of A)*

Given: a tolerance parameter $\delta_5 > 0$ and the $m \times m$ matrix A.

(i): Set $Q := A$ and $\nu := m$.

(ii): If $|Q_{\nu,\nu}| < \delta_5 \operatorname{Tr} A$ then the factorization fails and return.

(iii): Set $Q_{\nu,j} := Q_{\nu,j}/Q_{\nu,\nu}$, $j = 1, \ldots, \nu-1$ and $Q_{i,j} := Q_{i,j} - Q_{i,\nu} Q_{\nu,j}$, $i = 1, \ldots, \nu-1$, $j = 1, \ldots, \nu-1$. Set $\nu := \nu - 1$. If $\nu > 1$ go to *(ii)*.

(iv): Set $L_{i,j} := Q_{i,j}$ for $1 \le j < i \le m$, $L_{i,j} := 1$ for $1 \le j = i \le m$, $U_{i,j} := Q_{i,j}$ for $1 \le i \le j \le m$, $L_{i,j} := U_{i,j} := 0$ otherwise. Return.

Procedure 6.2 *(Updating of basic matrices)*

Given: $t > 0$, matrices $S_P, Y_P, S_P^T Y_P, Y_P^T Y_P$ and vectors $s, y, \tilde{s}, \tilde{y}, g_+, S_P^T g, Y_P^T g, Y_P^T H g, \sigma, \eta$.

(i): Compute $S_P^T g_+$, $Y_P^T g_+$, $\tilde{s}^T g_+$, $\tilde{y}^T g_+$.

(ii): Set $S := [\,S_P, \tilde{s}]$, $Y := [\,Y_P, \tilde{y}]$, $S^T g_+ := [S_P^T g_+, \tilde{s}^T g_+]$, $Y^T g_+ := [Y_P^T g_+, \tilde{y}^T g_+]$.

(iii): Compute $S_P^T y = S_P^T g_+ - S_P^T g$, $Y_P^T y = Y_P^T g_+ - Y_P^T g$, $Y_P^T s = -t\, Y_P^T H g$.

(iv): Compute $S_P^T \tilde{y} = S_P^T y + S_P^T Y_P \eta$, $Y_P^T \tilde{s} = Y_P^T s + Y_P^T S_P \sigma$, $Y_P^T \tilde{y} = Y_P^T y + Y_P^T Y_P \eta$, $\tilde{s}^T \tilde{y}$, $\tilde{y}^T \tilde{y}$.

(v): Set $S^T Y := \begin{bmatrix} S_P^T Y_P & S_P^T \tilde{y} \\ \tilde{s}^T Y_P & \tilde{s}^T \tilde{y} \end{bmatrix}$, $Y^T Y := \begin{bmatrix} Y_P^T Y_P & Y_P^T \tilde{y} \\ \tilde{y}^T Y_P & \tilde{y}^T \tilde{y} \end{bmatrix}$ and return.

We now state the method in details. For simplicity, we omit stopping criteria and a contingent restart when some computed direction vector is not sufficiently a descent direction.

Algorithm 6.1

Data: A maximum number $\hat{m}$, $1 < \hat{m} \le 5$ (see Section 5) of columns S, Y, line search parameters $\varepsilon_1, \varepsilon_2$, $0 < \varepsilon_1 < 1/2$, $\varepsilon_1 < \varepsilon_2 < 1$, tolerance parameters $\delta_1, \ldots, \delta_5$, $\delta_i \in (0, 1)$, $i \in \{1, \ldots, 5\}$, and global convergence parameters $\rho, \varepsilon_D \in (0, 1)$, $\Delta > 1$.

Step 0: Initiation. Choose starting point $x_0 \in \mathcal{R}^N$, define the starting matrix $H_0 = I$ and the direction vector $d_0 = -g_0$ and initiate the iteration counter k, the correction indicator i_C and the update indicator i_U to zero.

Step 1: Line search. Compute $x_{k+1} = x_k + t_k d_k$, where t_k satisfies (2.1), $g_{k+1} = \nabla f(x_{k+1})$, $s_k = t_k d_k$, $y_k = g_{k+1} - g_k$, $b_k = s_k^T y_k$, $\zeta_k = b_k / y_k^T y_k$ and set $\tilde{m} := \min(k, \hat{m}-1)$, $m := \tilde{m} + 1$ and define $H_k^I := \zeta_k I$. If $k = 0$ set $S_k := [s_k]$, $Y_k := [y_k]$, $S_k^T Y_k := [s_k^T y_k]$, $Y_k^T Y_k := [y_k^T y_k]$, compute $S_k^T g_{k+1}$, $Y_k^T g_{k+1}$ and go to Step 7.

Step 2: Correction preparation. Set $i_P := i_C$, $i_C := 0$. If $m > 1$ compute $\tilde{b}_k^{(1)}$ by (4.10) and Δ_{k-1}^k by (6.1). If $\Delta_{k-1}^k \le \delta_2$ and $\tilde{b}_k^{(1)} > \delta_1 b_k$ and $\max[|\tilde{s}_{k-1}|/|s_{k-1}|, |\tilde{y}_{k-1}|/|y_{k-1}|] \le \Delta$ set $i_C := 1$. If $i_C\, i_P = 0$ or $m \le 2$ go to Step 3. Compute $\tilde{b}_k^{(2)}$ by (4.11) and Δ_{k-2}^k by (6.1). If $\Delta_{k-1}^k + \Delta_{k-2}^k \le \delta_2$ and $\tilde{b}_k^{(2)} > \delta_1 b_k$ and $\tilde{b}_k^{(1)} / \tilde{b}_k^{(2)} > 1 + \delta_3$ set $i_C := 2$.

Step 3: Correction. If $i_C = 0$ set $\tilde{s}_k := s_k$, $\tilde{y}_k := y_k$, otherwise compute $\tilde{s}_k, \tilde{y}_k$ by (4.8) for $i_C = 1$ or by (4.9) for $i_C = 2$, i.e. σ_k, η_k such that $\tilde{s}_k = s_k + S_k^P \sigma_k$, $\tilde{y}_k = y_k + Y_k^P \eta_k$.

Step 4: Basic matrices updating. Using Procedure 6.2, form the matrices $S_k, Y_k, S_k^T Y_k$, $Y_k^T Y_k$ and set $A_k = S_k^T Y_k$, $\tilde{b}_k := \tilde{s}_k^T \tilde{y}_k$.

Step 5: Update selection. Define $\tilde{C}_{11}^k$ and R_{11}^k as $\tilde{C}_{11}$ and R_{11} in Lemma 4.1 and set $i_U := 0$. If $m = \hat{m}$ and $m \ge 2 + i_C$ and $\min_{i \in \mathcal{I}_k} \tilde{b}_i \ge \varepsilon_D \|A_k\|_F$ and $\|R_{11}^k \tilde{C}_{11}^k (R_{11}^k)^{-1}\|_F \le \rho$ and $\sum_{i,j \in \mathcal{I}_k} \tilde{\Delta}_i^j \le \delta_4$, where $\tilde{\Delta}_i^j$ is given by (6.2), set $i_U := 1$. If $i_U = 1$ use Procedure 6.1 to factorize $A_k = U_k L_k$; if the factorization fails, set $i_U := 0$.

Step 6: VM update. If $i_U = 1$ solve the Lyapunov equation (4.2) according to Section 5 and define update H_{k+1} of H_k^I by (3.2), otherwise define update H_{k+1} by (2.5).

Step 7: Direction vector. Compute $d_{k+1} = -H_{k+1}g_{k+1}$ and an auxiliary vector $Y_k H_{k+1} g_{k+1}$ by (2.6)–(2.7) for $i_U = 0$ or by (6.3)–(6.4) for $i_U = 1$. Set $k := k+1$. If $k \geq \hat{m}$ delete the first column of S_{k-1}, Y_{k-1} and the first row and column of $S_{k-1}^T Y_{k-1}$, $Y_{k-1}^T Y_{k-1}$ to form matrices S_k^P, Y_k^P, $(S_k^P)^T Y_k^P$, $(Y_k^P)^T Y_k^P$. Go to Step 1.

7 Global convergence

In this section, we establish global convergence of Algorithm 6.1. The following assumption and Lemma 7.1 are presented in [20], Lemma 7.2 in [22] and Lemma 7.3 in [12].

Assumption 7.1. *The objective function $f : \mathcal{R}^N \to \mathcal{R}$ is bounded from below and uniformly convex with bounded second-order derivatives (i.e. $0 < \underline{G} \leq \underline{\lambda}(G(x)) \leq \overline{\lambda}(G(x)) \leq \overline{G} < \infty$, $x \in \mathcal{R}^N$, where $\underline{\lambda}(G(x))$ and $\overline{\lambda}(G(x))$ are the lowest and the greatest eigenvalues of the Hessian matrix $G(x)$).*

Lemma 7.1. *Let objective function f satisfy Assumption 7.1. Then $\underline{G} \leq |y|^2/b \leq \overline{G}$ and $b/|s|^2 \geq \underline{G}$.*

Lemma 7.2. *Let $K_1, K_2 \in \mathcal{R}^{\nu\times\nu}$, $\nu > 0$, be symmetric positive semidefinite matrices. Then $0 \leq \mathrm{Tr}(K_1K_2) \leq \mathrm{Tr}\, K_1\, \mathrm{Tr}\, K_2$. Moreover, if K_2 is symmetric positive definite, then $\mathrm{Tr}(K_1K_2^{-1}) \leq \mathrm{Tr}\, K_1\, (\mathrm{Tr}\, K_2)^{\nu-1}/\det K_2$.*

Lemma 7.3. *Let K_1, K_2 be symmetric positive definite matrices. Then $K_1 - K_2$ is positive semidefinite if and only if $K_2^{-1} - K_1^{-1}$ is positive semidefinite.*

Lemma 7.4. *Let R_{11} be given as in Lemma 4.1, $n = m - \mu$, $\varepsilon_D \in (0,1)$ and $\min_{i\in\mathcal{I}} \tilde{b}_i \geq \varepsilon_D \|A\|_F$, where $\mathcal{I}$ is given by (6.2). Then*

$$\det R^T R \geq (\varepsilon_D^2\, \mathrm{Tr}\, R^T R)^m, \quad \det R_{11}^T R_{11} \geq (\varepsilon_D^2\, \mathrm{Tr}\, R_{11}^T R_{11})^n. \tag{7.1}$$

Proof. We have $\det R^T R = \det D^2 \geq (\varepsilon_D^2 \|A\|_F^2)^m \geq (\varepsilon_D^2 \|R\|_F^2)^m = (\varepsilon_D^2 \mathrm{Tr}\, R^T R)^m$ and the rest can be proved in a similar way. □

Lemma 7.5. *Let the assumptions of Theorem 4.1 and Lemma 7.4 be satisfied and $\|\hat{C}_{11}\|_F \leq \rho$, $\rho \in (0,1)$, where $\hat{C}_{11} = R_{11}\tilde{C}_{11}R_{11}^{-1}$. Then $\|\tilde{C}\|_F^2 < m\, \varepsilon_D^{-2n}$ and $\mathrm{Tr}\, X^* < \omega^2/\mathrm{Tr}\, D$, where $\omega = (1 + m\varepsilon_D^{-2m})/(1-\rho^2)$.*

Proof. We first show that $\|\tilde{C}_{11}\|_F < \varepsilon_D^{-n}$. By $\tilde{C}_{11} = R_{11}^{-1}\hat{C}_{11}R_{11}$ we get $\|\tilde{C}_{11}\|_F^2 = \mathrm{Tr}(R_{11}^T \hat{C}_{11}^T R_{11}^{-T} R_{11}^{-1} \hat{C}_{11} R_{11})$ and since the trace of a product of two square matrices is independent of the order of multiplication, Lemma 7.2 and Lemma 7.4 give

$$\|\tilde{C}_{11}\|_F^2 \leq \mathrm{Tr}(\hat{C}_{11} R_{11} R_{11}^T \hat{C}_{11}^T) \frac{(\mathrm{Tr}\, R_{11}R_{11}^T)^{n-1}}{\det R_{11}R_{11}^T} \leq \frac{\mathrm{Tr}(R_{11}R_{11}^T \hat{C}_{11}^T \hat{C}_{11})}{\varepsilon_D^{2n}\, \mathrm{Tr}\, R_{11}R_{11}^T} \leq \frac{\|\hat{C}_{11}\|_F^2}{\varepsilon_D^{2n}} < \frac{1}{\varepsilon_D^{2n}}. \tag{7.2}$$

We can now prove the first assertion. Observing that

$$R^{-1} = \begin{bmatrix} R_{11}^{-1} & -R_{11}^{-1}R_{12}D_{22}^{-1} \\ 0 & D_{22}^{-1} \end{bmatrix}, \quad A - R = \begin{bmatrix} A_{11} - R_{11} & 0 \\ A_{21} & 0 \end{bmatrix},$$

we can see that $\tilde{C}_{21} = D_{22}^{-1}A_{21}$ by $\tilde{C} = R^{-1}(A - R)$, which implies

$$\|\tilde{C}_{21}\|_F^2 \le \|D_{22}^{-1}\|_F^2 \|A_{21}\|_F^2 \le \mu(\varepsilon_D \|A\|_F)^{-2} \, \|A\|_F^2 = \mu \varepsilon_D^{-2} \tag{7.3}$$

by assumption. Since $\tilde{C}_{12}, \tilde{C}_{22}$ are null matrices by Lemma 4.1, from (7.2)–(7.3) we get

$$\|\tilde{C}\|_F^2 = \|\tilde{C}_{11}\|_F^2 + \|\tilde{C}_{21}\|_F^2 \le \varepsilon_D^{-2n} + \mu \varepsilon_D^{-2} < (1+\mu)\varepsilon_D^{-2n},$$

which gives $\|\tilde{C}\|_F^2 < m\, \varepsilon_D^{-2n}$ by $\mu < m$.

It remains to prove that $\operatorname{Tr} X^* < \omega^2 / \operatorname{Tr} D$. Using Theorem 4.1 , from (3.5) we obtain

$$\tilde{X} - D = \begin{bmatrix} \tilde{X}_{11} - D_{11} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \tilde{C}_{11}^T & \tilde{C}_{21}^T \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{X}_{11} & 0 \\ 0 & D_{22} \end{bmatrix} \begin{bmatrix} \tilde{C}_{11} & 0 \\ \tilde{C}_{21} & 0 \end{bmatrix},$$

which yields

$$\tilde{X}_{11} - D_{11} = \tilde{C}_{11}^T \tilde{X}_{11} \tilde{C}_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21} = R_{11}^T \hat{C}_{11}^T R_{11}^{-T} \tilde{X}_{11} R_{11}^{-1} \hat{C}_{11} R_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21}. \tag{7.4}$$

by $\tilde{C}_{11} = R_{11}^{-1} \hat{C}_{11} R_{11}$. Denoting $\hat{X}_{11} = R_{11}^{-T} \tilde{X}_{11} R_{11}^{-1}$, we can rewrite (7.4) in the form

$$\hat{X}_{11} = \hat{C}_{11}^T \hat{X}_{11} \hat{C}_{11} + R_{11}^{-T} (D_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21}) R_{11}^{-1}. \tag{7.5}$$

Using Lemma 7.2 more times, we first get

$$\operatorname{Tr}(D_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21}) = \operatorname{Tr} D_{11} + \operatorname{Tr}(\tilde{C}_{21} \tilde{C}_{21}^T D_{22}) < (1 + \mu \varepsilon_D^{-2}) \operatorname{Tr} D \tag{7.6}$$

by (7.3). Further $\operatorname{Tr}(\hat{C}_{11}^T \hat{X}_{11} \hat{C}_{11}) = \operatorname{Tr}(\hat{C}_{11} \hat{C}_{11}^T \hat{X}_{11}) \le \rho^2 \operatorname{Tr} \hat{X}_{11}$ by $\|\hat{C}_{11}\|_F \le \rho$, thus (7.5) gives

$$\begin{aligned} (1-\rho^2) \operatorname{Tr} \hat{X}_{11} &\le \operatorname{Tr}\Big((D_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21}) (R_{11}^T R_{11})^{-1} \Big) \\ &\le \operatorname{Tr}(D_{11} + \tilde{C}_{21}^T D_{22} \tilde{C}_{21}) \, (\operatorname{Tr} R_{11}^T R_{11})^{n-1} / \det R_{11}^T R_{11} \\ &\le (1 + \mu \varepsilon_D^{-2}) \operatorname{Tr} D / (\varepsilon_D^{2n} \operatorname{Tr} R_{11}^T R_{11}) < (1+\mu) \varepsilon_D^{-2n-2} \operatorname{Tr} D / \operatorname{Tr} R_{11}^T R_{11} \end{aligned}$$

by (7.6) and Lemma 7.4. From the last inequality we obtain

$$\operatorname{Tr} \tilde{X}_{11} = \operatorname{Tr}(R_{11}^T \hat{X}_{11} R_{11}) \le \operatorname{Tr} \hat{X}_{11} \operatorname{Tr} R_{11} R_{11}^T < \frac{(1+\mu) \varepsilon_D^{-2n-2}}{1-\rho^2} \operatorname{Tr} D \le \frac{m \varepsilon_D^{-2m}}{1-\rho^2} \operatorname{Tr} D, \tag{7.7}$$

which yields

$$\operatorname{Tr} \tilde{X} = \operatorname{Tr}(\tilde{X}_{11} + D_{22}) < \operatorname{Tr}(\tilde{X}_{11} + D) < \omega \operatorname{Tr} D \tag{7.8}$$

by Theorem 4.1. Finally, by $\tilde{X} = R^T X^* R$, Lemma 7.4 and the Schwarz inequality we have

$$\begin{aligned} \operatorname{Tr} X^* &= \operatorname{Tr} [\tilde{X} (R^T R)^{-1}] \le \operatorname{Tr} \tilde{X} \, (\operatorname{Tr} R^T R)^{m-1} / \det R^T R \\ &\le \frac{\operatorname{Tr} \tilde{X}}{\varepsilon_D^{2m} \operatorname{Tr} R^T R} < \frac{\omega \operatorname{Tr} D}{\varepsilon_D^{2m} \operatorname{Tr} D^2} \le \frac{m\, \omega}{\varepsilon_D^{2m} \operatorname{Tr} D} < \frac{\omega^2}{\operatorname{Tr} D}. \end{aligned}$$

□

Theorem 7.1. *Let objective function f satisfy Assumption 7.1. Then, Algorithm 6.1 generates a sequence $\{g_k\}$ that either satisfies $\lim_{k \to \infty} |g_k| = 0$ or terminates with $g_k = 0$ for some k.*

Proof. Denote $\tilde{\mathcal{I}}_k = \{k-1\}$ for $i_C = 1$ in Step 2 of Algorithm 6.1, $\tilde{\mathcal{I}}_k = \{k-2, k-1\}$ for $i_C = 2$ and $\tilde{\mathcal{I}}_k = \emptyset$ otherwise, $k > 0$. The safeguarding technique in this step guarantees

$$\tilde{b}_k \ge \delta_1 b_k, \qquad |\tilde{s}_i| \le \Delta |s_i|, \; |\tilde{y}_i| \le \Delta |y_i|, \; \Delta > 1, \quad i \in \tilde{\mathcal{I}}_k, \tag{7.9}$$

by $i_P \ne 0$ for $i_C = 2$ (see Step 2), therefore from (4.6)–(4.9) we get

$$|\tilde{s}_k| \le \Big\| \prod_{i\in\tilde{\mathcal{I}}_k} \Big(I - \frac{\tilde{s}_i\,\tilde{y}_i^T}{\tilde{b}_i}\Big) \Big\| \, |s_k| \le |s_k| \prod_{i\in\tilde{\mathcal{I}}_k} \frac{|\tilde{s}_i|\,|\tilde{y}_i|}{\tilde{b}_i} \le \Big(\frac{\Delta^4}{\delta_1^2} \prod_{i\in\tilde{\mathcal{I}}_k} \frac{|s_i|\,|y_i|}{b_i}\Big) |s_k| \le \theta_0 |s_k| \qquad (7.10)$$

for $\tilde{\mathcal{I}}_k \ne \emptyset$ and $k > 0$, with $\theta_0 = (\Delta^2/\delta_1)^2\, \overline{G}/\underline{G} > 1$ by Lemma 7.1. Similarly we obtain $|\tilde{y}_k| \le \theta_0 |y_k|$, which together with (7.10) gives

$$\begin{aligned} |\tilde{s}_k|^2/\tilde{b}_k &\le (\theta_0^2/\delta_1)|s_k|^2/b_k \le (\theta_0^2/\delta_1)/\underline{G} \triangleq \theta_1, && (7.11)\\ |\tilde{y}_k|^2/\tilde{b}_k &\le (\theta_0^2/\delta_1)|y_k|^2/b_k \le (\theta_0^2/\delta_1)\overline{G} \triangleq \theta_2 && (7.12) \end{aligned}$$

for $\tilde{\mathcal{I}}_k \ne \emptyset$ and $k > 0$. Writing $\tilde{s}_k = s_k, \tilde{y}_k = y_k$ for difference vectors without correction, we can use (7.11)–(7.12) for all $k \ge 0$ by Lemma 7.1 and $\theta_0^2/\delta_1 > 1$.

As we mentioned in Section 6, in all iterations we choose $H_k^I = \zeta_k I$, $\zeta_k = b_k/|y_k|^2$, see Step 1. Denoting $B_k^I = (H_k^I)^{-1}$, Lemma 7.1 gives

$$\operatorname{Tr} B_k^I = (|y_k|^2/b_k) \operatorname{Tr} I \le N\overline{G}, \quad \det B_k^I = (|y_k|^2/b_k)^N \ge \underline{G}^N, \quad k \ge 0. \qquad (7.13)$$

(i) Suppose first that $i_U = 0$, i.e. the BNS update (2.5) of H_k^I is used, where columns of S, Y are $\tilde{s}_i, \tilde{y}_i$, see above and Section 4.1. It is equivalent to the recurrent updating H_k^I by the BFGS method (2.3)

$$H_{i+1}^{k+1} = (1/\tilde{b}_i)\tilde{s}_i\tilde{s}_i^T + \Big(I - (1/\tilde{b}_i)\tilde{s}_i\tilde{y}_i^T\Big) H_i^{k+1} \Big(I - (1/\tilde{b}_i)\tilde{y}_i\tilde{s}_i^T\Big), \qquad (7.14)$$

$i \in \mathcal{I}_k$, $H_0^{k+1} = H_k^I = \zeta_k I$, $H_{k+1} = H_{k+1}^{k+1}$. These updates satisfy (see [19])

$$\begin{aligned} \operatorname{Tr} B_{i+1}^{k+1} &= \operatorname{Tr} B_i^{k+1} + |\tilde{y}_i|^2/\tilde{b}_i - |B_i^{k+1}\tilde{s}_i|^2/\tilde{s}_i^T B_i^{k+1}\tilde{s}_i, && (7.15)\\ \det B_{i+1}^{k+1} &= \Big(\tilde{b}_i/\tilde{s}_i^T B_i^{k+1}\tilde{s}_i\Big) \det B_i^{k+1}, && (7.16) \end{aligned}$$

where $B_i^{k+1} = (H_i^{k+1})^{-1}$, $i = k - \tilde{m}, \dots, k+1$, which yields

$$\operatorname{Tr} B_i^{k+1} \le N\overline{G} + m\,\theta_2 \triangleq \theta_3 \qquad (7.17)$$

by (7.12)–(7.13). Since $\tilde{b}_i/\tilde{s}_i^T B_i^{k+1}\tilde{s}_i = (\tilde{b}_i/|\tilde{s}_i|^2)(|\tilde{s}_i|^2/\tilde{s}_i^T B_i^{k+1}\tilde{s}_i) \ge 1/(\theta_1\theta_3)$ by (7.11) and (7.17), for all $k > 0$ in view of (7.16)–(7.17) and (7.13) we get

$$\begin{aligned} \operatorname{Tr} B_{k+1} &= \operatorname{Tr} B_{k+1}^{k+1} \le \theta_3, && (7.18)\\ \det B_{k+1} &= \det B_{k+1}^{k+1} \ge \underline{G}^N/(\theta_1\theta_3)^m \triangleq \theta_4. && (7.19) \end{aligned}$$

(ii) Let $i_U = 1$, i.e. the repeated BNS update (3.2) is used, where again columns of S, Y are $\tilde{s}_i, \tilde{y}_i$. Using Theorem 2.3 in [22], we can write for $k > 0$

$$\begin{aligned} B_{k+1} &= (1/\zeta_k)I - (1/\zeta_k)S_k(S_k^TS_k)^{-1}S_k^T + Y_kA_k^{-1}(X_k^*)^{-1}A_k^{-T}Y_k^T, && (7.20)\\ \det B_{k+1} &= 1/(\det X_k^*\, \det S_k^TS_k)\,, && (7.21) \end{aligned}$$

where X_k^* is symmetric positive definite by $\varrho(C_k) < 1$, see Step 5, Lemma 4.1 and Theorem 3.1. Therefore

$$\operatorname{Tr} B_{k+1} \le N/\zeta_k + \operatorname{Tr}(Y_kA_k^{-1}(X_k^*)^{-1}A_k^{-T}Y_k^T) \le N/\zeta_k + \operatorname{Tr}(Y_kA_k^{-1}M_k^{-1}A_k^{-T}Y_k^T) \qquad (7.22)$$

by Lemma 7.3, (3.3) and the positive definiteness of M_k. Since $M_k^{-1} = R_k D_k^{-1} R_k^T$, see (3.3), (7.22) implies

$$\mathrm{Tr}\, B_{k+1} - N/\zeta_k \le \mathrm{Tr}(Y_k K_k D_k^{-1} K_k^T Y_k^T) = \mathrm{Tr}(J_k D_k^{-1}) \le \sum_{i \in \mathcal{I}_k} (J_k)_{ii}/\tilde{b}_i, \tag{7.23}$$

denoting $K_k = A_k^{-1} R_k = (I + \tilde{C}_k)^{-1}$, see (3.5), and $J_k = K_k^T Y_k^T Y_k K_k$. Denoting further by λ_i, $i = 1, \ldots, m$, the eigenvalues of $\tilde{C}_k$, in view of Step 5 we can use Lemma 7.2 and Lemma 7.5 to have

$$\begin{aligned} \mathrm{Tr}\, J_k &= \mathrm{Tr}\Big(Y_k^T Y_k (K_k^{-T} K_k^{-1})^{-1}\Big) \le \mathrm{Tr}(Y_k^T Y_k) \mathrm{Tr}(K_k^{-T} K_k^{-1})^{m-1} / \det(K_k^{-T} K_k^{-1}) \\ &= \mathrm{Tr}(Y_k^T Y_k) \frac{\|I + \tilde{C}_k\|_F^{2m-2}}{\det(I + \tilde{C}_k)^2} \le \mathrm{Tr}(Y_k^T Y_k) \frac{(N^{1/2} + \sqrt{m}\, \varepsilon_D^{-n})^{2m}}{\prod_{i=1}^m |1 + \lambda_i|^2} \le \theta_5 \sum_{i \in \mathcal{I}_k} |\tilde{y}_i|^2 \end{aligned} \tag{7.24}$$

with $\theta_5 = [(N^{1/2} + m\, \varepsilon_D^{-m})/(1-\rho)]^{2m} > 1$. Since $\tilde{b}_i \ge \varepsilon_D \|A_k\|_F$, $i \in \mathcal{I}_k$, see Step 5, we obtain

$$\tilde{b}_i \ge \varepsilon_D \|D_k\|_F = \varepsilon_D (\mathrm{Tr}\, D_k^2)^{1/2} \ge \varepsilon_D\, \mathrm{Tr}\, D_k / \sqrt{m} \tag{7.25}$$

by the Schwarz inequality. From (7.23)–(7.25) and $\sqrt{m} < m$ we get

$$\mathrm{Tr}\, B_{k+1} \le \frac{N}{\zeta_k} + \frac{\sqrt{m}}{\varepsilon_D} \frac{\mathrm{Tr}\, J_k}{\mathrm{Tr}\, D_k} < \frac{N}{\zeta_k} + \frac{m\theta_5}{\varepsilon_D} \frac{\sum_{i \in \mathcal{I}_k} |\tilde{y}_i|^2}{\sum_{i \in \mathcal{I}_k} \tilde{b}_i} \le N\overline{G} + \frac{m\theta_5}{\varepsilon_D} \sum_{i \in \mathcal{I}_k} \frac{|\tilde{y}_i|^2}{\tilde{b}_i} \le \theta_6 \tag{7.26}$$

with $\theta_6 = N\overline{G} + m^2 \theta_5 \theta_2 / \varepsilon_D > \theta_3$ by $\zeta_k = b_k / |y_k|^2$, see Step 1, Lemma 7.1 and (7.12).

We proceed to estimate $\det B_{k+1}$. Using geometric/arithmetic mean inequality and Lemma 7.5, from (7.21) we get

$$\det B_{k+1}^{-1/m} = (\det S_k^T S_k \, \det X_k^*)^{1/m} \le \frac{\mathrm{Tr}\, S_k^T S_k}{m} \frac{\mathrm{Tr}\, X_k^*}{m} < \frac{\omega^2\, \mathrm{Tr}\, S_k^T S_k}{m^2\, \mathrm{Tr}\, D_k} \le \frac{\omega^2}{m^2} \sum_{i \in \mathcal{I}_k} \frac{|\tilde{s}_i|^2}{\tilde{b}_i} \le \frac{\omega^2 \theta_1}{m}$$

by (7.11), i.e.

$$\det B_{k+1} > \Big(m/(\omega^2 \theta_1)\Big)^m \ge \theta_7 \tag{7.27}$$

with $\theta_7 = \min[\theta_4, (m/(\omega^2\theta_1))^m] \le \theta_4$.

(iii) The lowest eigenvalue $\underline{\lambda}(B_k)$ of B_k satisfies $\underline{\lambda}(B_k) \ge \det B_k / (\mathrm{Tr}\, B_k)^{N-1}$, $k \ge 0$. Setting $q_k = H_k^{1/2} g_k$, from (7.18) – (7.19) and (7.26) – (7.27) we get

$$\frac{(s_k^T g_k)^2}{|s_k|^2 |g_k|^2} = \frac{s_k^T B_k s_k}{s_k^T s_k} \frac{g_k^T H_k g_k}{g_k^T g_k} = \frac{s_k^T B_k s_k}{s_k^T s_k} \frac{q_k^T q_k}{q_k^T B_k q_k} \ge \frac{\det B_k}{(\mathrm{Tr}\, B_k)^{N-1}} \frac{1}{\mathrm{Tr}\, B_k} \ge \frac{\theta_7}{\theta_6^N}, \; k > 1, \tag{7.28}$$

which implies $\lim_{k \to \infty} |g_k| = 0$, see Theorem 3.2 in [19] and relations (3.17) – (3.18) ibid. □

One can show in the same way as in [11] that inequality (7.28), line search conditions (2.1) and Assumption 7.1 imply that the sequence $\{x_k\}$ is R-linearly convergent.

8 Numerical experiments

In this section, we compare our results with the results obtained by the L-BFGS method [11, 18] and by our two latest limited-memory methods [21, 22]. All methods are implemented in the optimization software system UFO [17], which can be downloaded from `www.cs.cas.cz/luksan/ufo.html`. We use the following collections of test problems (problems which were not solved by the L-BFGS method were excluded from our numerical experiments):

- **Test 11** – 55 chosen problems from [15] (computed repeatedly ten times for a better comparison), which are problems from the CUTE collection [5], some of them modified; used N are given in Table 1, where the modified problems are marked with '*',
- **Test 12** – 73 problems from [3], $N = 10\,000$,
- **Test 25** – 68 chosen problems from [14], which are sparse test problems for unconstrained optimization, contained in the system UFO, $N = 10\,000$.

The source texts and the reports corresponding to these test collections can be downloaded from the web page `www.cs.cas.cz/luksan/test.html`.

Problem	N	Problem	N	Problem	N	Problem	N
ARWHEAD	5000	DIXMAANI	3000	EXTROSNB	1000	NONDIA	5000
BDQRTIC	5000	DIXMAANJ	3000	FLETCBV3*	1000	NONDQUAR	5000
BROYDN7D	2000	DIXMAANK	3000	FLETCBV2	1000	PENALTY3	1000
BRYBND	5000	DIXMAANL	3000	FLETCHCR	1000	POWELLSG	5000
CHAINWOO	1000	DIXMAANM	3000	FMINSRF2	5625	SCHMVETT	5000
COSINE	5000	DIXMAANN	3000	FREUROTH	5000	SINQUAD	5000
CRAGGLVY	5000	DIXMAANO	3000	GENHUMPS	1000	SPARSINE	1000
CURLY10	1000	DIXMAANP	3000	GENROSE	1000	SPARSQUR	1000
CURLY20	1000	DQRTIC	5000	INDEF*	1000	SPMSRTLS	4999
CURLY30	1000	EDENSCH	5000	LIARWHD	5000	SROSENBR	5000
DIXMAANE	3000	EG2	1000	MOREBV*	5000	TOINTGSS	5000
DIXMAANF	3000	ENGVAL1	5000	NCB20*	1010	TQUARTIC*	5000
DIXMAANG	3000	CHNROSNB*	1000	NCB20B*	1000	WOODS	4000
DIXMAANH	3000	ERRINROS*	1000	NONCVXU2	1000		

Table 1: Dimensions for Test 11 – the modified CUTE collection.

We have chosen $\hat{m}$=5, which is an often used value in comparisons of limited-memory methods. In [19] the results for the L-BFGS method with $\hat{m} = 3, 5, 17, 29$ are compared and it is stated that the best CPU time is often obtained for small values of $\hat{m}$, but the algorithm tends to be less robust when $\hat{m}$ is small; it is also confirmed by our numerical experiments. Note that the required amount of storage is $2(\hat{m} + 1)N$.

Furthermore, we have used $\delta_1 = 10^{-4}$, $\delta_2 = 10^{-2}$, $\delta_3 = \delta_4 = 0.2$, $\delta_5 = 10^{-7}$, $\varepsilon_D = 10^{-6}$, $\rho = 0.99$, $\Delta = 10^3$, $\varepsilon_1 = 10^{-4}$, $\varepsilon_2 = 0.8$ and the final precision $\|g(x^\star)\|_\infty \le 10^{-6}$.

Table 2 contains the total number of function and also gradient evaluations (NFV) and the total computational time in seconds (Time).

Method	Test 11		Test 12		Test 25	
	NFV	Time	NFV	Time	NFV	Time
L-BFGS	80539	10.494	119338	51.51	502966	438.58
Alg. 4.2 in [21]	63987	9.062	66244	30.15	309650	305.88
Alg. 1 in [22]	65228	8.745	96748	40.13	371830	345.88
Alg. 6.1	63162	9.080	66941	30.46	299736	323.95

Table 2: Comparison of the selected methods.

For a better demonstration of both the efficiency and the reliability, we compare selected optimization methods by using performance profiles introduced in [6]. The performance profile $\rho_M(\tau)$ is defined by the formula

$$\rho_M(\tau) = \frac{\text{number of problems where } \log_2(\tau_{P,M}) \le \tau}{\text{total number of problems}}$$

with $\tau \ge 0$, where $\tau_{P,M}$ is the performance ratio of the number of function evaluations (or the time) required to solve problem P by method M to the lowest number of function evaluations (or the time) required to solve problem P. The ratio $\tau_{P,M}$ is set to infinity (or some large number) if method M fails to solve problem P.

The value of $\rho_M(\tau)$ at $\tau = 0$ gives the percentage of test problems for which the method M is the best and the value for τ large enough is the percentage of test problems that method M can solve. The relative efficiency and reliability of each method can be directly seen from the performance profiles: the higher is the particular curve, the better is the corresponding method. Figures 1–3, based on results in Table 2, reveal the performance profiles for tested methods graphically. They demonstrate the efficiency of our method in comparison with the L-BFGS method. We can also see that the numerical results for the new method and the results for our methods [21, 22] are comparable.

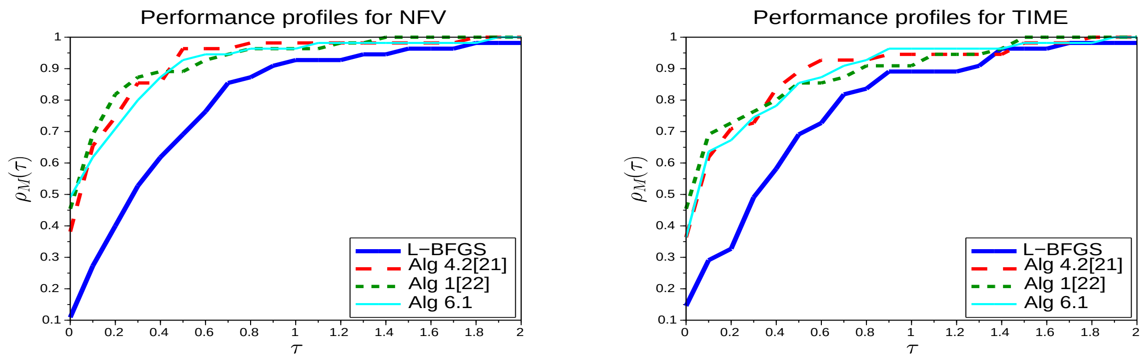

Figure 8.1: Comparison of $\rho_M(\tau)$ for Test 11 and various methods for NFV and TIME.

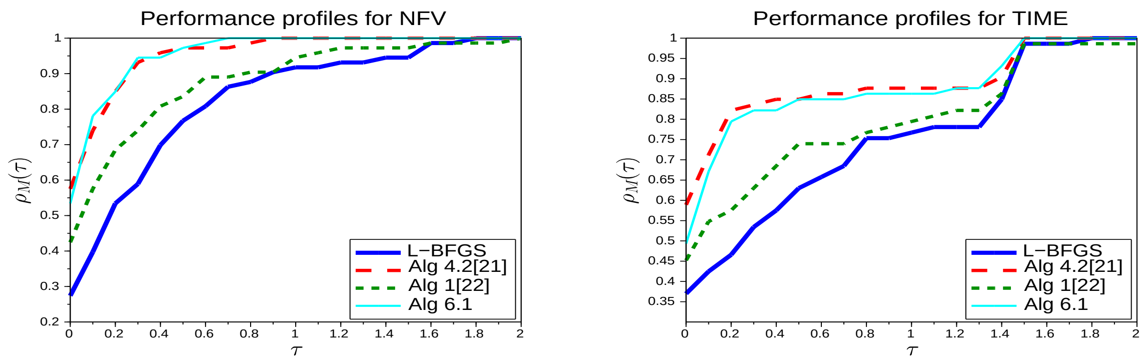

Figure 8.2: Comparison of $\rho_M(\tau)$ for Test 12 and various methods for NFV and TIME.

9 Conclusions

In this contribution, we derive the infinitely times repeated BNS update for general functions, describe its properties and relations to various forms of the discrete and continuous

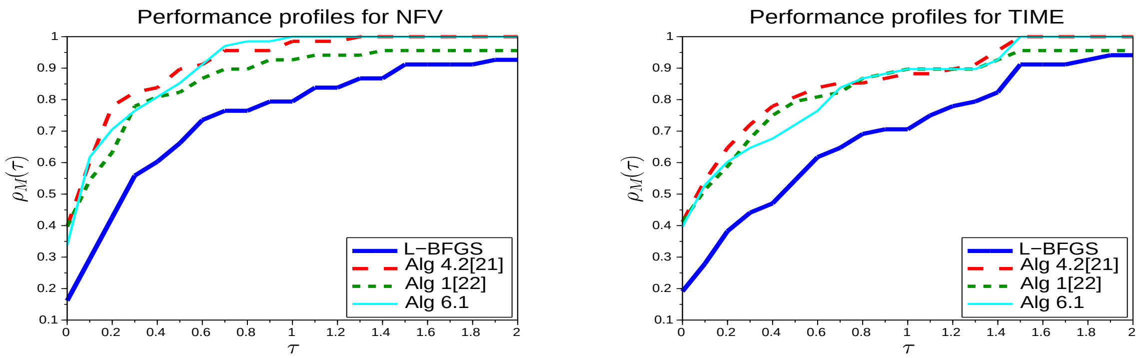

Figure 8.3: Comparison of $\rho_M(\tau)$ for Test 25 and various methods for NFV and TIME.

Lyapunov matrix equations and show how the order of these equations can be decreased by combination with methods [20, 21] based on vector corrections for conjugacy.

Our experiments indicate that this approach can improve unconstrained large-scale minimization results significantly compared with the frequently used L-BFGS method.

References

[1] M. Al-Baali, *Improved Hessian approximations for the limited memory BFGS method*, Numerical Algorithms, 22 (1999) 99-112.

[2] M. Al-Baali, *Extra-Updates Criterion for the Limited Memory BFGS Algorithm for Large Scale Nonlinear Optimization*, Journal of Complexity, 18 (2002) 557-572.

[3] N. Andrei, *An unconstrained optimization test functions collection*, Advanced Modeling and Optimization, 10 (2008) 147-161.

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

[5] 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.

[6] E.D. Dolan, J.J. Moré: *Benchmarking optimization software with performance profiles*, Mathematical Programming 91 (2002) 201-213.

[7] Fiedler, M.: Special matrices and their applications in numerical mathematics, 2nd ed. Mineola, N.Y., Dover Publications, 2008.

[8] R. Fletcher: *Practical Methods of Optimization*, John Wiley & Sons, Chichester, 1987.

[9] R. Gajić: *The Lyapunov matrix equation in system stability and control*, Academic Press, Inc., London, 1995.

[10] J.L. Howland, J.A. Senez: *A Constructive Method for the Solution of the Stability Problem*, Numer. Math. 16 (1970) 1-7.

[11] D.C. Liu, J. Nocedal: *On the limited memory BFGS method for large scale optimization*, Math. Prog. 45 (1989) 503-528.

[12] L. Lukšan, C. Matonoha, J. Vlček: *On Lagrange multipliers of trust-region subproblems*, BIT Numerical Mathematics 48 (2008) 763-768.

[13] L. Lukšan, C. Matonoha, J. Vlček: *Algorithm 896: LSA - Algorithms for Large-Scale Optimization*, ACM Transactions on Mathematical Software 36 (2009) 16:1-16:29.

[14] L. Lukšan, C. Matonoha, J. Vlček: *Sparse Test Problems for Unconstrained Optimization*, Report V-1064, Prague, ICS AS CR, 2010.

[15] L. Lukšan, C. Matonoha, J. Vlček: *Modified CUTE Problems for Sparse Unconstrained Optimization*, Report V-1081, Prague, ICS AS CR, 2010.

[16] L. Lukšan, E. Spedicato: *Variable metric methods for unconstrained optimization and nonlinear least squares*, J. Comput. Appl. Math. 124 (2000) 61-95.

[17] L. Lukšan, M. Tůma, C. Matonoha, J. Vlček, N. Ramešová, M. Šiška, J. Hartman: *UFO 2014. Interactive System for Universal Functional Optimization*, Report V-1218, Prague, ICS AS CR, 2014.

[18] J. Nocedal: *Updating quasi-Newton matrices with limited storage*, Math. Comp. 35 (1980) 773-782.

[19] J. Nocedal, S.J. Wright: *Numerical optimization*, Springer-Verlag, New York, 1999.

[20] J. Vlček, L. Lukšan: *A conjugate directions approach to improve the limited-memory BFGS method*, Appl. Math. Comput. 219 (2012) 800-809.

[21] J. Vlček, L. Lukšan: *A modified limited-memory BNS method for unconstrained minimization based on conjugate directions idea*, Optimization Methods & Software 30 (2015) 616-633.

[22] J. Vlček, L. Lukšan: *Properties of the block BFGS update and its application to the limited-memory block BNS method for unconstrained minimization*, Report V-1244, Prague, ICS CAS, 2017, to appear in Numerical Algorithms.