Jan Vlček; Ladislav Lukšan

Application of the infinitely many times repeated BNS update and conjugate directions to limited-memory optimization methods

In: Jan Chleboun and Pavel Kůs and Petr Přikryl and Miroslav Rozložník and Karel Segeth and Jakub Šístek and Tomáš Vejchodský (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Hejnice, June 24-29, 2018. Institute of Mathematics CAS, Prague, 2019. pp. 177–185.

Persistent URL: http://dml.cz/dmlcz/703083

Programs and Algorithms of Numerical Mathematics 19
J. Chleboun, P. Kůs, P. Přikryl, M. Rozložník, K. Segeth, J. Šístek, T. Vejchodský (Eds.)
Institute of Mathematics CAS, Prague 2019

APPLICATION OF THE INFINITELY MANY TIMES REPEATED BNS UPDATE AND CONJUGATE DIRECTIONS TO LIMITED-MEMORY OPTIMIZATION METHODS

Jan Vlček[1], Ladislav Lukšan[1,2]
[1]Institute of Computer Science, Czech Academy of Sciences,
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic,
vlcek@cs.cas.cz, luksan@cs.cas.cz,
[2]Technical University of Liberec, Hálkova 6, 461 17 Liberec, Czech Republic

Abstract: To improve the performance of the L-BFGS method for large scale unconstrained optimization, repeating of some BFGS updates was proposed e.g. in [1]. Since this can be time consuming, the extra updates need to be selected carefully. We show that groups of these updates can be repeated infinitely many times under some conditions, without a noticeable increase of the computational time; the limit update is a block BFGS update [17]. It can be obtained by solving of some Lyapunov matrix equation whose order can be decreased by application of vector corrections for conjugacy [16]. Global convergence of the proposed algorithm is established for convex and sufficiently smooth functions. Numerical results indicate the efficiency of the new method.

Keywords: Unconstrained minimization, limited-memory variable metric methods, the repeated Byrd-Nocedal-Schnabel update, the Lyapunov matrix equation, the conjugate directions, global convergence, numerical results.

MSC: 65K10, 65F30

1. Introduction

In this contribution we assume that the problem function $f\colon \mathcal{R}^N \to \mathcal{R}$ 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 well known method [3] with the BFGS update, denoted by the BNS method here, and on vector corrections for conjugacy [15], [16].

The best known limited-memory VM methods are the L-BFGS [8] (implemented as subroutine PLIS in [9]) and BNS methods, described briefly in Section 2. Their performance can be improved, e.g. using vector corrections for conjugacy. But since

DOI: 10.21136/panm.2018.19

they can deteriorate stability and require extra arithmetic operations, the conditions for its application are complicated.

Another way was proposed e.g. in [1], where some BFGS updates are computed several times, which however can be time consuming. In Section 3 we will derive a limit update formula for the infinitely many times repeated BNS update, which can be written as the block BFGS update [17] and which can be obtained by solving of some low-order Lyapunov matrix equation. For quadratic functions this update represents the best improvement of convergence in some sense under some conditions [16], [17]. Note that the relative increase in the computational time for one iteration in comparison with the BNS update is very small for N large.

In Section 4 we show that the order of this equation can be decreased of unit always, or more by application of vector corrections for conjugacy. Vice versa, the combination of those methods with the repeated BNS update enables us to reduce the number of correction vectors and to simplify the conditions for their choice.

In Section 5 we outline 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 presented in Section 6. Global convergence is established in Section 7 and numerical results are reported in Section 8.

We refer to report [18] for details and proofs of assertions, here we briefly present only the main results. We will denote by $\|\cdot\|_F$ the Frobenius matrix norm.

2. The standard BNS method

The BNS method belongs to the VM or quasi-Newton (QN) line search iterative methods [12]. 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 usually the direction vector is $d_k = -H_k g_k$, matrix $H_k \in \mathcal{R}^{N\times N}$ is symmetric positive definite and the stepsize $t_k > 0$ is chosen in such a way that

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

(the Wolfe line search conditions [14]), $0 < \varepsilon_1 < 1/2$, $\varepsilon_1 < \varepsilon_2 < 1$, $f_k = f(x_k)$ and $g_k = \nabla f(x_k)$. 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)

$$H_{k+1} y_k = s_k, \qquad y_k = g_{k+1} - g_k \tag{2}$$

(see [12]), $k \ge 0$. To simplify the 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 [12, 14] belongs to the most efficient; the update preserves positive definite VM matrices and can be written in the form

$$H_+ = (1/b) s s^T + \Big(I - (1/b) s y^T\Big) H \Big(I - (1/b) y s^T\Big), \quad b = s^T y, \tag{3}$$

where $b > 0$ by (1). 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 $(s_{k-\widetilde{m}}, y_{k-\widetilde{m}}), \ldots, (s_k, y_k)$ successively, where $\widetilde{m} = \min(k, \hat{m}-1)$, $m = \widetilde{m} + 1$, $k \geq 0$ and $\hat{m} > 1$ is a given parameter. In case of the BNS method, the update formula can be expressed in the form [3]

$$H_+ = SR^{-T}DR^{-1}S^T + \left(I - SR^{-T}Y^T\right)H^I\left(I - YR^{-1}S^T\right), \tag{4}$$

where $S_k = [s_{k-\widetilde{m}}, \ldots, s_k]$, $Y_k = [y_{k-\widetilde{m}}, \ldots, y_k]$, $D_k = \mathrm{diag}[b_{k-\widetilde{m}}, \ldots, b_k]$, $(R_k)_{i,j} = (S_k^T Y_k)_{i,j}$ for $i \leq j$, $(R_k)_{i,j} = 0$ otherwise (an upper triangular matrix), $k \geq 0$.

3. The repeated BNS update

Repeating the standard BNS update (4) in the following way

$$\bar{H}_{i+1} = SR^{-T}DR^{-1}S^T + \left(I - SR^{-T}Y^T\right)\bar{H}_i\left(I - YR^{-1}S^T\right), \quad i = 0, 1, \ldots \tag{5}$$

for an arbitrary matrix $\bar{H}_0$, we will derive the infinitely many times repeated BNS (RBNS) update and describe its properties and relations to various forms of the discrete and continuous Lyapunov matrix equations, see e.g. [6].

Theorem 1. *Let $H^I \in \mathcal{R}^{N\times N}$, $A = S^T Y$, $C = AR^{-1} - I$, $\bar{H}_0 = H^I$ and the sequence $\{\bar{H}_i\}_{i=1}^{\infty}$ be given by (5). If the spectral radius $\varrho(C) < 1$, then the matrices $I + C$, A are nonsingular and the RBNS update H_+ of H^I defined by $H_+ = \lim_{i\to\infty} \bar{H}_i$ satisfies*

$$H_+ = SX^*S^T + \left(I - SA^{-T}Y^T\right)H^I\left(I - YA^{-1}S^T\right), \tag{6}$$

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

$$X^* = C^T X^* C + R^{-T}DR^{-1}. \tag{7}$$

Moreover, if A is symmetric, then it is positive definite, $X^* = A^{-1}$ and $H_+Y = S$, i.e. the QN conditions (2) with all stored difference vectors are satisfied.

The property $H_+Y = S$ for A symmetric 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.

Most of the numerical methods to solve the discrete Lyapunov equation use some transformation to the continuous Lyapunov equation [6]. Here 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. We refer to report [18] for details and conditions for existence of this factorization.

Using this factorization, we can equivalently rewrite (7) as the Lyapunov equation

$$XZ + Z^T X = 2W, \quad X = U^T X^* U, \quad Z = 2U^{-1}RL^{-1} - I, \quad W = L^{-T}DL^{-1}. \tag{8}$$

The order of equation (8) can be always decreased, see Section 4. Some properties of solutions to (8) are given by the following lemma.

Theorem 2. *Let $\varrho(C)<1$. Then $X = U^T X^* U$ is a unique and symmetric positive definite solution to the Lyapunov equation (8). Moreover, let A be symmetric. Then $X = D^U$, where D^U is a diagonal matrix with the same diagonal entries as U.*

Note that the property $X = D^U$ for A symmetric is numerically advantageous e.g. when the matrix A is almost symmetric.

4. Relations to methods based on vector corrections

The following theorem shows how the order of equation (8) 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, see Section 4.1. Since every such submatrix of order one can be considered to be diagonal, we can always decrease the order of these equations and choose $\mu \geq 1$. Besides, we show how the assumption $\varrho(C)<1$ in Theorems 1–3 can be equivalently written in another form. Note that we can also always assume that $\mu < m$, since the definitions of A in Theorem 1 and D, R after (4) imply that D is the diagonal part and R upper triangular part of A and thus for $A=D$ (i.e. $R = D = A$) we have $C = AR^{-1} - I = 0$, therefore $X^* = R^{-T} D R^{-1}$ by (7) and update (6) is identical to (4).

Theorem 3. *Let the matrices D, R, S, Y be given as in Section 2, W, X, Z by (8), D^U by Theorem 2, X^* by (7), $C = AR^{-1} - I$, $\widetilde{C} = R^{-1} C R$ and suppose that $A = S^T Y = UL$, where U is an upper triangular matrix with nonzero diagonal entries and L a lower triangular matrix with unit diagonal entries. Let A be partitioned in the form*

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{9}$$

with $A_{22} \in \mathcal{R}^{\mu \times \mu}$, $0 < \mu \leq m$, and $\widetilde{C}, D, L, U, W, X, Z$ be partitioned in the same way. If $A_{22} = D_{22}$ then $\widetilde{C}_{12}$, $\widetilde{C}_{22}$ are null matrices and $\varrho(C) = \varrho(\widetilde{C}) = \varrho(\widetilde{C}_{11})$. Moreover, let $\varrho(C)<1$ and the columns of S be linearly independent. Then
(a) X_{12}, X_{21} are null matrices and $L_{22} = I$,
(b) $X_{22} = U_{22} = D_{22}$ and X_{11} is the unique solution to the Lyapunov equation

$$X_{11} Z_{11} + Z_{11}^T X_{11} = 2W_{11}. \tag{10}$$

4.1. Application of corrections for conjugacy

Our numerical experiments indicate that the RBNS update (6) can improve the performance of the L-BFGS method and that this improvement can be increased if we also use vector corrections for conjugacy. In [16] it was shown that correction vectors from only 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 $S_k = [\widetilde{s}_{k-\widetilde{m}}, \ldots, \widetilde{s}_k]$, $Y_k = [\widetilde{y}_{k-\widetilde{m}}, \ldots, \widetilde{y}_k]$, $\widetilde{b}_i = \widetilde{s}_i^T \widetilde{y}_i$, where $\widetilde{s}_i = s_i$, $\widetilde{y}_i = y_i$, $k - \widetilde{m} \leq i \leq k$, if the corrections are not used.

In case of one correction vector from the preceding iteration and $\widetilde{b}_- \neq 0$ we set

$$\widetilde{s} = P_1^T s, \quad \widetilde{y} = P_1 y, \quad P_1 = I - (1/\widetilde{b}_-)\widetilde{y}_- \widetilde{s}_-^T\,, \tag{11}$$

which yields $\widetilde{s}^T \widetilde{y}_- = \widetilde{s}_-^T \widetilde{y} = 0$, for two correction vectors and $\widetilde{b}_- \widetilde{b}_= \neq 0$ we set

$$\widetilde{s} = P_2^T s, \quad \widetilde{y} = P_2 y, \quad P_2 = I - (1/\widetilde{b}_-)\widetilde{y}_- \widetilde{s}_-^T - (1/\widetilde{b}_=)\widetilde{y}_= \widetilde{s}_=^T, \tag{12}$$

which yields $\widetilde{s}^T \widetilde{y}_- = \widetilde{s}^T \widetilde{y}_= = \widetilde{s}_-^T \widetilde{y} = \widetilde{s}_=^T \widetilde{y} = 0$. In view of the assumption $A_{22} = D_{22}$ of Theorem 3 we use the projection P_2 only if $\widetilde{s}_-^T \widetilde{y}_= = \widetilde{s}_=^T \widetilde{y}_- = 0$ (if $\widetilde{s}_-, \widetilde{y}_-$ were corrected).

5. Solution to the Lyapunov equation

Denoting $n = m - \mu$ (the order of the Lyapunov equation (10)), we consider here only the case $1 \leq n < m$, see the comments before Theorem 3. Many methods for solving the Lyapunov equations can be found in [6]. Since we want to have rounding errors small, we attempt to solve only linear systems whose order is maximally four for $m \leq 5$ (i.e. $n \leq 4$) (the usual order is $n(n+1)/2$, i.e. 10 for $n = 4$). Thus for $n > 2$ we generalize the idea used in [7] for special matrices Z, W. We choose some block of entries of X_{11}, calculate the remaining entries and then we obtain the appropriate values of chosen entries as a solution to some linear system, see [18] for details. Note that equation (10) can be solved directly for $n \leq 2$.

6. Implementation

We assume that $H^I = \zeta I$, $\zeta = s^T y / y^T y > 0$ and implement a modified BNS method, which replaces some difference vectors s, y by the corrected vectors $\widetilde{s}$, $\widetilde{y}$ and then some BNS updates (4) by the repeated BNS updates (6) with X^* given by (8) and Theorem 3. A more detailed description can be found in [18].

We use corrections for conjugacy when the value $\widetilde{b}$ is sufficiently great, the deviation of A from symmetry is small and if the values $|\widetilde{s}_-|/|s_-|, |\widetilde{y}_-|/|y_-|$ are not too great. We do not use correction vectors from two preceding iterations, if the benefit of the corrections would be too small (as an indicator, the ratio $b/\widetilde{b}$ is used, see [16]).

We use the repeated BNS update if $\|R_{11} \widetilde{C}_{11} R_{11}^{-1}\|_F \leq \rho < 1$, if diagonal entries of A, U are sufficiently great and if the deviation of A from symmetry is not too great.

For the repeated BNS update with $H^I = \zeta I$, the direction vector and an auxiliary vector $Y^T H_+ g_+$, see below, can be calculated efficiently, similarly as in [3]. The procedure for updating the basic matrices $S^T Y = A$, $Y^T Y$ is similar to the algorithm given in [3] for updating the matrices $D, R, Y^T Y$ in (4), see Procedure 6.2 in [18]. Since we need the whole matrix A, we use an auxiliary vector $Y_-^T s = -t\, Y_-^T H g$ for computation of the last row of A to have the number of arithmetic operations approximately the same.

Algorithm 6.1 (simplified, without stopping criteria)

Data: A maximum number $\hat{m}$, $1 < \hat{m} \le 5$, of columns S, Y, line search parameters $\varepsilon_1, \varepsilon_2$ and the global convergence parameter ρ.

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 to zero.

Step 1: Line search. Compute $x_{k+1} = x_k + t_k d_k$, where t_k satisfies (1), $g_{k+1} = \nabla f(x_{k+1})$, $s_k = t_k d_k$, $y_k = g_{k+1} - g_k$, $b_k = s_k^T y_k$, $\zeta_k = b_k / y_k^T y_k$, set $\widetilde{m} := \min(k, \hat{m}-1)$, $m := \widetilde{m} + 1$ 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 $S_k^T g_{k+1}$, $Y_k^T g_{k+1}$, define H_{k+1} by (4) and go to Step 5.

Step 2: Corrections. If conditions for corrections are satisfied, see above, compute $\widetilde{s}_k, \widetilde{y}_k$ by (11) or (12), otherwise set $\widetilde{s}_k := s_k$, $\widetilde{y}_k := y_k$.

Step 3: Basic matrices update. Similarly as in [3] form the matrices $S_k, Y_k, S_k^T Y_k$, $Y_k^T Y_k$ and set $A_k = S_k^T Y_k$, $\widetilde{b}_k := \widetilde{s}_k^T \widetilde{y}_k$.

Step 4: VM update. If conditions for the RBNS update are satisfied, see above, solve the Lyapunov equation (10) according to Section 5 and define update H_{k+1} of H_k^I by (6), otherwise define H_{k+1} by (4).

Step 5: Direction vector. Compute $d_{k+1} = -H_{k+1} g_{k+1}$ and an auxiliary vector $Y_k^T H_{k+1} g_{k+1}$. Set $k := k + 1$. If $k \ge \hat{m}$ delete the first column of S_{k-1}, Y_{k-1} and the first row and column of $S_{k-1}^T Y_{k-1}$, $Y_{k-1}^T Y_{k-1}$. Go to Step 1.

7. Global convergence

Assumption 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. $0 < \underline{G} \le \underline{\lambda}(G(x)) \le \overline{\lambda}(G(x)) \le \overline{G} < \infty$, $x \in \mathcal{R}^N$, where $\underline{\lambda}(G(x))$ and $\overline{\lambda}(G(x))$ are the lowest and the greatest eigenvalues of the Hessian matrix $G(x)$).*

Theorem 4. *If the objective function f satisfies Assumption 1, 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.*

8. Numerical experiments

We compare our results with the results obtained by the L-BFGS method [8] and by our two latest limited-memory methods [16, 17], all implemented in the system UFO [13], using the following collections of test problems:

- **Test 11** – 55 modified problems [11] from the CUTE collection [4] with various dimensions N from 1000 to 5000 (prescribed for the given problem),
- **Test 12** – 73 problems from collection [2], $N = 10\,000$,
- **Test 25** – 68 problems from collection [10], $N = 10\,000$.

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 [16]	63987	9.062	66244	30.15	309650	305.88
Alg. 1 in [17]	65228	8.745	96748	40.13	371830	345.88
Alg. 6.1	63162	9.080	66941	30.46	299736	323.95

Table 1: Comparison of the selected methods

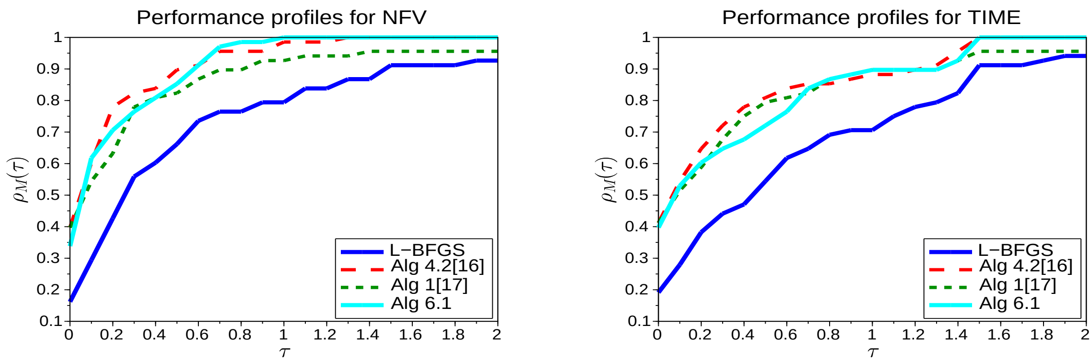

Figure 1: Comparison of $\rho_M(\tau)$ for Test 25 and various methods for NFV and TIME.

The source texts and the reports corresponding to test collections Test 11 and Test 25 can be downloaded from the web page `www.cs.cas.cz/luksan/test.html`.

We have chosen $\hat{m} = 5$, $\rho = 0.99$, $\varepsilon_1 = 10^{-4}$, $\varepsilon_2 = 0.8$ and the final precision $\|g(x^\star)\|_\infty \le 10^{-6}$.

Table 1 contains the total number of function and also gradient evaluations (NFV) and the total computational time in seconds (Time).

For Test 25, we also compare these methods by using performance profiles [5]. Value $\rho_M(0)$ is the percentage of the test problems for which method M is the best and value $\rho_M(\tau)$ for τ large enough is the percentage of the problems that method M can solve. Performance profiles show the relative efficiency and reliability of the methods: the higher is the particular curve, the better is the corresponding method.

Figure 1, based on the results in Table 1, demonstrates 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 [16], [17] are comparable.

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 Lyapunov matrix equations and show how the order of these equations can be decreased by combination with methods [15, 16] 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.

Acknowledgements

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

References

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

[2] Andrei, N.: An unconstrained optimization test functions collection. Adv. Model. Optim. **10** (2008), 147–161.

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

[4] Bongartz, I., Conn, A.R., Gould, N., and Toint, P.L.: CUTE: constrained and unconstrained testing environment. ACM Trans. Math. Software **21** (1995), 123–160.

[5] Dolan, E.D. and Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program. **91** (2002), 201–213.

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

[7] Howland, J.L. and Senez, J.A.: A constructive method for the solution of the stability problem. Numer. Math. **16** (1970), 1–7.

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

[9] Lukšan, L., Matonoha, C., and Vlček, J.: Algorithm 896: LSA-algorithms for large-scale optimization. ACM Trans. Math. Software **36** (2009), 16:1–16:29.

[10] Lukšan, L., Matonoha, C., and Vlček, J.: Sparse test problems for unconstrained optimization. Report V-1064, ICS AS CR, Prague, 2010.

[11] Lukšan, L., Matonoha, C., and Vlček, J.: Modified CUTE problems for sparse unconstrained optimization. Report V-1081, ICS AS CR, Prague, 2010.

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

[13] Lukšan, L., Tůma, M., Matonoha, C., Vlček, J., Ramešová, N., Šiška, M., and Hartman, J.: UFO 2017. Interactive system for universal functional optimization. Report V-1252, ICS AS CR, Prague, 2017, [online] `www.cs.cas.cz/luksan/ufo.html`.

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

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

[16] Vlček, J. and Lukšan, L.: A modified limited-memory BNS method for unconstrained minimization based on conjugate directions idea. Optim. Methods Softw. **30** (2015), 616–633.

[17] Vlček, J. and Lukšan, L.: Properties of the block BFGS update and its application to the limited-memory block BNS method for unconstrained minimization. Numer. Algorithms (2018) [online], https://doi.org/10.1007/s11075-018-0513-3.

[18] Vlček, J. and Lukšan, L.: A limited-memory optimization method using the infinitely many times repeated BNS update and conjugate directions. Report V-1245, ICS ASCR, Prague, 2018.