NEW METHODS FOR LARGE-SCALE UNCONSTRAINED OPTIMIZATION

Ladislav Lukšan, Jan Vlček
Institute of Computer Science AVČR

Contents:

- Shifted variable metric methods.
- Limited-memory variable metric methods.
- Shifted limited-memory variable metric methods.
- Variable metric methods for dense nonsmooth optimization.
- Variable metric methods for general large-scale nonsmooth optimization.
- Variable metric methods for partially separable nonsmooth optimization.
- Large scale minimax optimization.
- Hybrid methods for large-scale nonlinear least squares.
- Efficient methods for solving large-scale trust-region subproblems.

Shifted variable metric methods:

We consider line-search iterative methods of the form

$$x_+ = x + t\,d, \quad d = -Hg,$$

where H is a positive definite matrix. Step-length $t > 0$ is chosen in such a way that

$$\begin{aligned} F_+ - F &\leq \varepsilon_1\, t\, d^T g, \\ d^T g_+ &\geq \varepsilon_2\, d^T g \end{aligned}$$

(the weak Wolfe conditions), where $F = F(x)$, $g = \nabla F(x)$ and $0 < \varepsilon_1 < 1/2$, $\varepsilon_1 < \varepsilon_2 < 1$.

Shifted variable metric methods use matrix $H = \zeta I + A$, where $\zeta > 0$ and A is positive semidefinite. Starting from the zero matrix, they generate a sequence of positive semidefinite matrices satisfying the (modified) quasi-Newton condition $A_+ y = \varrho \tilde{s}$, where $s = x_+ - x$, $y = g_+ - g$ and $\tilde{s} = s - \zeta_+ y$. Here ϱ is a correction parameter and $\zeta_+ > 0$ is a shift parameter. Update

$$\begin{aligned} A_+ = A &+ \varrho\, \frac{\tilde{s}\tilde{s}^T}{\tilde{b}} - \frac{A y y^T A}{\bar{a}} \\ &+ \frac{\eta}{\bar{a}} \left(\frac{\bar{a}}{\tilde{b}} \tilde{s} - A y \right) \left(\frac{\bar{a}}{\tilde{b}} \tilde{s} - A y \right)^T \end{aligned}$$

is used, where $\bar{a} = y^T A y$ and $\tilde{b} = y^T \tilde{s}$.

Theorem 1. Let A be positive semidefinite and $\eta \geq 0$. If $0 < \zeta_+ < y^T s / y^T y$, then A_+ is positive semidefinite.

Determination of the shift parameter:

Theorem 1 implies condition

$$\zeta_+ = \mu \, b/\hat{a}, \quad 0 < \mu < 1,$$

where $b = y^T s$ and $\hat{a} = y^T y$. If μ is too small, then matrix H is unsuitable, especially in the first n iterations, where A is singular. If μ is too large, the stability is usually lost (numerical explosion). Two basic choices:

- Constant choice. In this case $0 < \mu < 1/2$. If $\mu \to 1/2$, then the shifted BFGS method becomes unstable. Efficient choices lie in the interval $0.20 \leq \mu \leq 0.25$, e.g., $\mu = 0.22$.

- Variable choice. Using a theoretical investigation of stability and global convergence, we have obtained choice

$$\mu = \sqrt{1 - \bar{a}/a} \Big/ \left(1 + \sqrt{1 - b^2/(\hat{a}|s|^2)}\right)$$

(the numerator assures the global convergence and the denominator assures the stability).

Global convergence:

Assumption 1. The objective function $f : R^n \to R$ is uniformly convex and has bounded second-order derivatives, i.e.

$$0 < \underline{G} \leq \underline{\lambda}(G(x)) \leq \overline{\lambda}(G(x)) \leq \overline{G} < \infty$$

for all $x \in R^n$, where $\underline{\lambda}(G(x))$ and $\overline{\lambda}(G(x))$ are the lowest and the greatest eigenvalues of the Hessian matrix $G(x)$.

Assumption 2. Parameters ϱ_k and μ_k of the shifted VM method are uniformly positive and bounded, in the sense that

$$0 < \underline{\varrho} \leq \varrho_k \leq \overline{\varrho},$$

$$0 < \underline{\mu} \leq \mu_k \leq \overline{\mu} < 1,$$

for every $k \geq 1$.

Theorem 2. Consider a shifted variable metric method satisfying Assumption 2 with the line-search fulfilling the weak Wolfe conditions. Let the objective function satisfy Assumption 1. Then, if $0 \leq \eta \leq 1$ and $\mu^2 \leq 1 - \bar{a}/a$, one has

$$\liminf_{k \to \infty} |g_k| = 0.$$

Computational results:

The shifted variable metric methods were tested by using a set of 92 relatively difficult test problems with 50 and 200 variables (subroutine `TEST28` in `www.cs.cas.cz/~luksan/test.html`).

N - the number of variables

MET - the method used

SBFGS - shifted BFGS

SDFP - shifted DFP

BFGS - standard BFGS

DFP - standard DFP

NIT - the total number of iterations

NEV - the total number of function evaluations

NF - the number of failures for a given set

TIME - the total computational time

N	MET	NIT	NEV	NF	TIME
50	SBFGS	11256	12178	-	1.03
	SDFP	46010	48237	8	3.78
	BFGS	14958	16474	1	1.26
	DFP	79486	84215	35	6.66
200	SBFGS	30429	36080	1	25.11
	SDFP	92799	100461	15	74.88
	BFGS	36099	39991	2	27.21
	DFP	146851	158979	32	113.75

Conclusions:

- The shifted VM methods are competitive with the classic VM methods. They are more efficient than standard implementations of the classic VM methods. However, the classic VM methods can be improved by a suitable scaling, which is not true in the case of shifted VM methods.
- The shifted VM methods are not intended for solving problems, which can be successfully solved by the classic VM methods. However, these methods are ideal as starting methods for the shifted limited-memory VM methods, which are based on the same idea.

Variable metric methods with reduced Hessians (Gill + Leonard (2003)):

Let $S_k = [s_{k-m}, \ldots, s_{k-1}]$ and Z_k be a matrix whose columns form an orthonormal basis in the subspace generated by columns of S_k. The variable metric methods with reduced Hessians determine the direction vector from the reduced system

$$d_k = Z_k \tilde{d}_k, \quad Z_k^T B_k Z_k \tilde{d}_k = -\tilde{g}_k, \quad \tilde{g}_k = Z_k^T g_k.$$

The reduced Hessian $Z_k^T B_k Z_k$ is updated by using differences $\tilde{s_k} = t_k \tilde{d}_k$ and $\tilde{y_k} = Z_k^T (g_{k+1} - g_k)$. In the next iteration, Z_k and also the updated $Z_k^T B_k Z_k$ are changed to correspond to S_{k+1}.

Limited-memory variable metric methods:

We set $x_{k+1} = x_k - t_k d_k = x_k - t_k H_k^k g_k$, where $H_{k-m}^k = \gamma_k I$ (usually $\gamma_k = b_{k-1}/a_{k-1}$) and

$$H_{j+1}^k = \gamma_j^k V_j^T H_j^k V_j + \frac{\rho_j}{b_j} s_j s_j^T, \quad V_j = I - \frac{1}{b_j} y_j s_j^T$$

for $i - n \leq j \leq i - 1$. Again $s_j = x_{j+1} - x_j$, $y_j = g_{j+1} - g_j$, $a_j = y_j^T H_j y_j$, $b_j = y_j^T s_j$ and ρ_j are correction parameters.

Strang recurrences (Nocedal (1980)):

Backward recurrence $u_k = -g_k$ and

$$\begin{aligned} \sigma_j &= s_j^T u_{j+1}/b_j, \\ u_j &= u_{j+1} - \sigma_j y_j, \end{aligned}$$

for $k-1 \geq j \geq k-m$. Forward recurrence $v_{k-m} = (b_{k-1}/a_{k-1}) u_{k-m}$ and

$$\begin{aligned} v_{i-m} &= (b_{i-1}/a_{i-1}) u_{i-m}, \\ v_{j+1} &= v_j + (\rho_j \sigma_j - y_j^T v_j/b_j) s_j, \end{aligned}$$

for $k - m \leq j \leq k - 1$. Finally we set $d_k = v_k$. Note that $2m$ vectors s_j, y_j, $k - m \leq j \leq k - 1$ are used.

The compact matrix form (Byrd + Nocedal + Schnabel (1990)):

$$H_k^k = \gamma_k I - [S_k, \gamma_k Y_k] M_k [S_k, \gamma_k Y_k]^T.$$

Here $S_k = [s_{k-m}, \ldots, s_{k-1}]$, $Y_k = [y_{k-m}, \ldots, y_{k-1}]$, and

$$M_k = \begin{bmatrix} (R_k^{-1})^T (C_k + \gamma_k Y_k^T Y_k) R_k^{-1} & -(R_k^{-1})^T \\ -R_k^{-1} & 0 \end{bmatrix},$$

where C_k is a diagonal matrix containing the diagonal part of $S_k^T Y_k$, and R_k is an upper triangular matrix containing the upper triangular part of $S_k^T Y_k$. Again $2m$ vectors s_j, y_j, $k - m \leq j \leq k - 1$ are used.

- Limited-memory VM methods are globally convergent for uniformly convex functions with bounded second-order derivatives.
- Limited-memory VM methods find the minimum of a convex quadratic function after at most n steps (quadratic termination) if the line-search is perfect.
- The method based on the Strang recurrences is more stable. The method with compact matrices can be used for inverse updates ($B_k = H_k^{-1}$ instead of H_k).

Shifted limited-memory variable metric methods (L + V (2004)):

We suppose that $x_+ = x - tHg$ where $H = \zeta I + A = \zeta I + UU^T$ and where $n \times m$ matrix U is updated by formula $U_+ = VU$ with a low rank matrix V chosen in such a way that the (modified) quasi-Newton condition $A_+ y = U_+ U_+^T y = \rho \tilde{s}$ is satisfied. This condition can be replaced by

$$U_+^T y = z, \quad U_+ z = \varrho \tilde{s}, \quad z^T z = \varrho \tilde{b}. \qquad (*)$$

Theorem 3. Let T be a symmetric positive definite matrix and $z \in R^m$. Denote $\mathcal{U}$ a set of $n \times m$ matrices. Then the unique solution to

$$\min\{y^T T y \, \|T^{-1/2}(U_+ - U)\|_F^2 : U_+ \in \mathcal{U}\}$$

s.t. $(*)$ is

$$U_+ = U - \frac{Ty}{y^T T y} y^T U + (\varrho \tilde{s} - Uz + \frac{y^T U z}{y^T T y} Ty) \frac{z^T}{z^T z}$$

(Ty and z are parameters defining a class of shifted limited-memory variable metric methods).

VAR1 - Rank 1 variationally derived method

We assume that Ty and $\rho\tilde{s} - Uz$ are linearly dependent and set

$$z = \vartheta U^T B s, \qquad \vartheta = \pm\sqrt{\varrho\tilde{b}/\bar{c}}. \qquad (*)$$

Then

$$U_+ = U - \frac{\varrho\tilde{s} - \vartheta ABs}{\varrho\tilde{b} - \vartheta\bar{b}} \left(y - \vartheta Bs\right)^T U,$$

which gives the best results for the choice $\mathrm{sgn}(\vartheta\bar{b}) = -1$.

VAR2 - Rank 2 variationally derived method

With z given by $(*)$ and with the simple choice $Ty = \tilde{s}$, we obtain

$$U_+ = U - \frac{\tilde{s}}{\tilde{b}} y^T U + \left(\varrho\frac{\tilde{s}}{\vartheta} - ABs + \frac{\bar{b}}{\tilde{b}}\tilde{s}\right) \frac{s^T BU}{\bar{c}}.$$

The efficiency of both these methods significantly depends on the value of the correction parameter ϱ. Very good results were obtained with choices $\varrho = \nu$, $\varrho = \varepsilon$, $\varrho = \sqrt{\nu\varepsilon}$ and $\varrho = \zeta/(\zeta + \zeta_+)$, where $\nu = \mu/(1-\mu)$, μ is a relative shift parameter and $\varepsilon = \sqrt{1 - \bar{a}/a}$ is the damping factor of μ.

Global convergence:

Theorem 4. Consider a shifted variable metric method VAR1 or VAR2 satisfying Assumption 2 and inequality $\mu^2 \leq \zeta \hat{a}/a$ together with the line search fulfilling the weak Wolfe conditions. Let the objective function satisfy Assumption 1. Then if

$$\vartheta_k = -\operatorname{sgn} \bar{b}_k \min\left(C, \sqrt{\varrho_k \tilde{b}_k / \bar{c}_k}\right)$$

for some $C > 0$ and all $k \in N$, one has

$$\liminf_{k \to \infty} |g_k| = 0$$

(the VAR2 method allows the value $C = \infty$)

General formulas:

$$U_+ = \frac{\tilde{s} z^T}{\tilde{b}} + \left(I - \frac{T y y^T}{y^T T y}\right) U \left(I - \frac{z z^T}{z^T z}\right)$$

$$\begin{aligned} U_+ U_+^T &= \rho \frac{\tilde{s}\tilde{s}^T}{\tilde{b}} + \left(I - \frac{T y y^T}{y^T T y}\right) \\ &\quad U \left(I - \frac{z z^T}{z^T z}\right) U^T \left(I - \frac{y y^T T}{y^T T y}\right) \end{aligned}$$

Usually $Ty = \tilde{s}$. This choice gives the (full) shifted BFGS method if $z = 0$.

Computational results:

The shifted limited-memory variable metric methods were tested by using a set of 22 test problems with 1000 and 5000 variables (subroutine `TEST14` in `www.cs.cas.cz/~luksan/test.html`). Always 10 vectors (or pairs) were stored for $N = 1000$ and 5 vectors (or pairs) were stored for $N = 5000$.

N - the number of variables

MET - the method used

- VAR1 - Rank 1 variationally derived method
- VAR2 - Rank 2 variationally derived method
- LBFGSS - Limited-memory BFGS method with Strang recurrences
- LBFGSC - Limited-memory BFGS method with compact matrices
- LBFGSR - Limited-memory BFGS method with reduced Hessians

NIT - the total number of iterations

NEV - the total number of function evaluations

NF - the number of failures for a given set

TIME - the total computational time

N	Method	NIT	NEV	NF	TIME
1000	VAR1	19317	19680	-	13.86
	VAR2	18227	18546	-	13.76
	LBFGSS	20427	21456	-	15.17
	LBFGSC	20555	26003	1	16.55
	LBFGSR	22385	33181	-	24.09
	CG	20520	41049	-	17.91
5000	VAR1	94801	97858	-	8:02.1
	VAR2	85662	87483	-	7:22.6
	LBFGSS	108315	111456	2	9:33.8
	LBFGSC	102313	105828	1	10:32.6
	LBFGSR	98046	154931	-	10:41.4
	CG	69805	168471	1	6:45.3

Conclusions:

- Methods VAR1 and VAR2 are very efficient, competitive with the LBFGSS method for well-conditioned problems. The LBFGSS method can be better than VAR1 and VAR2 for ill-conditioned problems.

- Shifted limited-memory VM methods are still under development. Our limited computational experience indicates that they could be improved using more suitable choice of parameters. The theory is not yet finished.

Nonsmooth optimization:

We assume that objective function $f : R^n \to R$ is locally Lipschitz and we are able to compute a subgradient $g \in \partial f(x)$ at any point $x \in R^n$. Since a locally Lipschitz function is differentiable almost everywhere by the Rademacher theorem, then usually $g = \nabla f(x)$. A special feature of the nonsmooth problems is the fact that the gradient $\nabla f(x)$ changes discontinuously and is not small in the neighborhood of a local extremum. Thus the standard optimization methods cannot be used efficiently.

Bundle methods:

Values $f(x^k)$, $g(x^k) \in \partial f(x^k)$ at single point x^k do not suffice for describing the local properties of the nonsmooth objective function. A bundle of values $f^j = f(y^j)$, $g^j \in \partial f(y^j)$ obtained at trial points y^j, $j \in \mathcal{J}_k \subset \{1, \ldots, k\}$, gives much better information.

Piecewise linear function:

We define a piecewise linear function

$$\begin{aligned} f_L^k(x) &= \max_{j \in \mathcal{J}_k} \{f^j + (x - y^j)^T g^j\} \\ &= \max_{j \in \mathcal{J}_k} \{f(x^k) + (x - x^k)^T g^j - \alpha_j^k\}, \end{aligned}$$

where $\alpha_j^k = f(x^k) - f_j^k$, are linearization errors and $f_j^k = f^j + (x^k - x^j)^T g^j$. This function is majorized by the objective function and $\alpha_j^k \geq 0$, $j \in \mathcal{J}_k$, in the convex case.

Subgradient locality measures:

To guarantee nonnegativity of numbers α_j^k, $j \in \mathcal{J}_k$, in the nonconvex case, the subgradient locality measures

$$\alpha_j^k = \max \left\{ |f(x^k) - f_j^k|, \gamma (s_j^k)^\nu \right\},$$

where $\gamma > 0$, $\nu \geq 1$, $f_j^k = f_j(x^k)$ and

$$s_j^k = \|x^j - y^j\| + \sum_{i=j}^{k-1} \|x^{i+1} - x^i\|$$

are used instead of linearization errors.

Aggregation:

Since we can only work with limited-size bundles where $|\mathcal{J}_k| \leq m$, the set $\mathcal{J}_k$ is usually determined in such a way that $\mathcal{J}_k = \{1, \ldots, k\}$ for $k \leq m$, and $\mathcal{J}_{k+1} = \mathcal{J}_k \cup \{k+1\} \backslash \{k+1-m\}$ for $k \geq m$. One possibility guaranteeing global convergence is an additional use of transformed aggregate values f_a^k, g_a^k, s_a^k and

$$\alpha_a^k = \max\left\{|f(x^k) - f_a^k|, \gamma(s_a^k)^\nu\right\},$$

which accumulate information from the previous iterations. These values represent a linear function, which is added to the definition of the piecewise linear function.

Piecewise quadratic function:

Direction vector $d^k \in R^n$ is usually obtained as a minimum of the piecewise quadratic function

$$\begin{aligned} f_Q^k(x) &= \frac{1}{2}(x - x^k)^T G^k (x - x^k) \\ &+ \max\{f_L^k(x), f(x^k) + (x - x^k)^T g_a^k - \alpha_a^k\}, \end{aligned}$$

where $(1/2)(x - x^k)^T G^k (x - x^k)$ is the regularizing term with symmetric positive definite matrix G^k. This term restricts the size of the direction vector (in the same way as in the trust region methods).

Primal QP subproblem:

Minimization of the above piecewise quadratic function is equivalent to the solution of the following QP subproblem: Minimize

$$\frac{1}{2}d^T G^k d + v$$

subject to

$$-\alpha_j^k + d^T g^j \leq v, \quad j \in \mathcal{J}_k, \quad -\alpha_a^k + d^T g_a^k \leq v$$

(v is an extra variable). The solution of the primal QP subproblem can be expressed in the form

$$\begin{aligned} d^k &= -(G^k)^{-1}\tilde{g}_a^k, \\ v^k &= -(d^k)^T G^k d^k - \tilde{\alpha}_a^k, \end{aligned}$$

where

$$\tilde{g}_a^k = \sum_{j \in \mathcal{J}_k} \lambda_j^k g^j + \lambda_a^k g_a^k,$$

$$(\tilde{\alpha}_a^k, \tilde{f}_a^k, \tilde{s}_a^k) = \sum_{j \in \mathcal{J}_k} \lambda_j^k (\alpha_j^k, f_j^k, s_j^k) + \lambda_a^k (\alpha_a^k, f_a^k, s_a^k).$$

Here λ_j^k, $j \in \mathcal{J}_k$, λ_a^k, are Lagrange multipliers of the primal QP subproblem.

Dual QP subproblem:

Lagrange multipliers λ_j^k, $j \in \mathcal{J}_k$, λ_a^k, are solutions of the dual QP problem: Minimize

$$\frac{1}{2}\left(\sum_{j\in\mathcal{J}_k}\lambda_j g^j + \lambda_a g_a^k\right)^T (G^k)^{-1}\left(\sum_{j\in\mathcal{J}_k}\lambda_j g^j + \lambda_a g_a^k\right)$$

$$+\sum_{j\in\mathcal{J}_k}\lambda_j\alpha_j^k + \lambda_a\alpha_a^k$$

subject to

$$\begin{array}{l}\lambda_j \geq 0, \quad j \in \mathcal{J}_k, \quad \lambda_a \geq 0,\\ \sum_{j\in\mathcal{J}_k}\lambda_j + \lambda_a = 1.\end{array}$$

The minimum value of the dual function is

$$\begin{aligned} w^k &= \frac{1}{2}(\tilde{g}_a^k)^T(G^k)^{-1}\tilde{g}_a^k + \tilde{\alpha}_a^k \\ &= -v^k - \frac{1}{2}(\tilde{g}_a^k)^T(G^k)^{-1}\tilde{g}_a^k. \end{aligned}$$

Nonsmooth line search:

It is not possible to simply set $x^{k+1} = x^k + d^k$. To guarantee the global convergence, we use a line search procedure which generates two points

$$\begin{aligned} x^{k+1} &= x^k + t_L^k d^k, \\ y^{k+1} &= x^k + t_R^k d^k, \end{aligned}$$

where $0 \leq t_L^k \leq t_R^k \leq 1$ are stepsizes, in such a way that exactly one of the two possibilities, the descent step or the zero step, occurs. The descent step implies the conditions

$$t_R^k = t_L^k > 0, \qquad f(x^k + t_L^k d^k) \leq f(x^k) - \varepsilon_L t_L^k w^k,$$

while the zero step implies the conditions

$$t_R^k > t_L^k = 0, \qquad (d^k)^T g(x^k + t_R^k d^k) \geq \alpha^{k+1} - \varepsilon_R w^k$$

with

$$\begin{aligned} \alpha^{k+1} = \max \{ & |f(x^k) - f(x^k + t_R^k d^k) \\ & + t_R^k (d^k)^T g(x^k + t_R^k d^k)|, \gamma |t_R^k d^k|^\nu \}. \end{aligned}$$

Here $0 < \varepsilon_L < 1/2$ and $\varepsilon_L < \varepsilon_R < 1$.

Bundle update:

Having point x_{k+1} determined, it is necessary to transform all values. One has

$$
\begin{aligned}
f_j^{k+1} &= f_j^k + (x^{k+1} - x^k)^T g^j, \quad j \in J_k \\
f_a^{k+1} &= \tilde{f}_a^k + (x^{k+1} - x^k)^T \tilde{g}_a^k \\
f_{k+1}^{k+1} &= f^{k+1} + (x^{k+1} - y^{k+1}) g^{k+1} \\
g_a^{k+1} &= \tilde{g}_a^k \\
s_j^{k+1} &= s_j^k + \|x^{k+1} - x^k\|, \quad j \in J_k \\
s_a^{k+1} &= \tilde{s}_a^k + \|x^{k+1} - x^k\| \\
s_{k+1}^{k+1} &= \|x^{k+1} - y^{k+1}\|
\end{aligned}
$$

Quadratic function update:

We assume that matrices G^k are uniformly positive definite and uniformly bounded and that $G^{k+1} - G^k$ is positive semidefinite after every zero step. The proximal bundle method uses matrices $G^k = \sigma^k I$.

Global convergence:

One can prove that the number of consecutive zero steps is finite and that every cluster point of the sequence $\{x^k\}$ is a stationary point of the objective function. This follows from the fact that the norms of aggregate subgradients tends to zero implying $0 \in \partial f(x_k)$, if the number of consecutive zero steps is infinite. An infinite sequence of the descent steps can be investigated by the standard way.

Variable metric methods for nonsmooth problems:

- Standard bundle methods require relatively large bundles ($m \sim n$). Thus we need to solve quadratic programming subproblems with a relatively large number of constraints.
- Standard variable metric methods successfully solve many nonsmooth problems. Nonsmooth variable metric methods combine good properties of standard variable metric methods and standard bundle methods.

Basic ideas:

- Variable metric (BFGS) updates are applied to matrix $H^k = (G^k)^{-1}$ after descent steps, which allows us to decrease the bundle dimension significantly. Only three subgradients (old, new and aggregate) are used in every step.
- Variable metric (rank-1) norm-decreasing updates are used after zero steps. This together with the line search described above guarantees the global convergence.
- The direction vector and aggregate values are obtained by solving a simple QP subproblem containing only three constraints.

Initiation of aggregate values:

In the first iteration or after a descent step, we set $\tilde{g}^k = g^k$, $\tilde{\alpha}^k = 0$ and $m = k$.

Direction vector:

The direction vector is determined by formula $d^k = -H^k \tilde{g}^k$. At the same time, we set $w^k = (1/2)(\tilde{g}^k)^T H^k \tilde{g}^k + \tilde{\alpha}^k$. If w^k is sufficiently small, then an approximate solution is found.

Quadratic programming subproblem:

The quadratic programming subproblem reduces to the minimization of the function

$$\frac{1}{2} \left\| (H^k)^{1/2} (\lambda_1 g^k + \lambda_2 g^{k+1} + \lambda_3 \tilde{g}^k) \right\|^2$$

$$+ \lambda_2 \alpha^{k+1} + \lambda_3 \tilde{\alpha}^k,$$

where

$$\lambda_i \geq 0, \quad i \in \{1, 2, 3\}, \quad \lambda_1 + \lambda_2 + \lambda_3 = 1.$$

The optimal values $\lambda_i^k \geq 0$, $i \in \{1, 2, 3\}$ can be computed in a simple way.

Aggregation:

The resulting aggregate subgradient and aggregate subgradient locality measure are given as

$$\tilde{g}^{k+1} = \lambda_1^k g^k + \lambda_2^k g^{k+1} + \lambda_3^k \tilde{g}^k,$$

$$\tilde{\alpha}^{k+1} = \lambda_2^k \alpha^{k+1} + \lambda_3^k \tilde{\alpha}^k.$$

BFGS update after a descent step:

Let $u^k = g^{k+1} - g^k$. If $(u^k)^T d^k > 0$, we set

$$\begin{aligned} H^{k+1} &= H^k + \left(t_L^k + \frac{(u^k)^T H^k u^k}{(u^k)^T d^k} \right) \frac{d^k (d^k)^T}{(u^k)^T d^k} \\ &\quad - \frac{H^k u^k (d^k)^T + d^k (u^k)^T H^k}{(u^k)^T d^k}, \end{aligned}$$

otherwise we set $H^{k+1} = H^k$.

Rank-1 update after a zero step:

Let $v^k = H^k u^k - t_R^k d^k$. If $v_k^T \tilde{g}_k < 0$ (which implies $(u^k)^T v^k > 0$), we set

$$H^{k+1} = H^k - v^k (v^k)^T / (u^k)^T v^k,$$

otherwise we set $H^{k+1} = H^k$. Then $\tilde{g}_k^T H^{k+1} \tilde{g}_k \leq \tilde{g}_k^T H^k \tilde{g}_k$ after the zero step, which is necessary for the global convergence.

Global convergence:

A complete description of nonsmooth variable metric algorithm is given in V + L: JOTA 111 (2001) 407-430. The following result was proved for this algorithm.

Theorem 5. Assume function $F : R^n \to R$ is locally Lipschitz and the level set $\{x \in R^n : f(x) \leq f(x_1)\}$ is bounded. Then every cluster point of $\{x_k\}$ is stationary for f.

Computational experiments:

We give a numerical comparison of two methods for nonsmooth optimization:

- PBM - Proximal bundle method.
- NVM - Nonsmooth variable metric method.

These methods were tested by using a set of 25 dense nonsmooth test problems (subroutine `TEST19` in `www.cs.cas.cz/~luksan/test.html`). The test results are given in the following table, where `NI` is the number of iterations, `NE` is the number of function evaluations and `F` is the minimum function value. The last row contains the summary values and the total computational time (in seconds).

P	NI	NE	PB - F	NI	NE	NVM - F
1	42	45	.38117064D-06	34	34	.27598807D-10
2	18	20	.46154993D-08	15	16	.94894120D-10
3	31	33	1.9522245	17	17	1.9522247
4	14	16	2.0000000	17	17	2.0000000
5	17	19	-3.0000000	20	20	-2.9999996
6	13	15	7.2000014	19	19	7.2000000
7	11	12	-1.4142135	10	10	-1.4142133
8	66	68	-.99999940	55	59	-.99999247
9	13	15	-1.0000000	37	37	-.99999979
10	43	46	-7.9999999	14	14	-7.9999998
11	43	45	-43.999999	38	38	-43.999999
12	27	29	22.600162	40	40	22.600162
13	60	62	-32.348678	52	53	-32.348678
14	154	155	-2.9196975	32	32	-2.9197003
15	92	93	.55981566	81	83	.55981553
16	74	75	-.84140828	89	89	-.84140570
17	160	162	9.7857723	241	241	9.7858732
18	128	143	16.703861	88	89	16.703838
19	150	151	.16712381D-06	123	123	.14683215D-05
20	39	40	.12440972D-12	23	23	.00000000
21	245	251	-638530.48	357	359	-638564.91
22	52	53	.11665945D-11	358	360	.41534959D-05
23	19	20	.51313988D-08	65	66	.32729678D-05
24	27	28	.23412735D-07	67	67	.94570857D-06
25	428	450	32.349182	313	315	32.349159
Σ	1966	2046	TIME = 1.48	2205	2221	TIME = 0.93

Variable metric methods for large-scale nonsmooth problems:

Standard variable metric updates are replaced by special variable metric updates. But the condition $(\tilde{g}^k)^T H^{k+1} \tilde{g}^k \leq (\tilde{g}^k)^T H^k \tilde{g}^k$ has to be satisfied after the zero step.

Criterion $(1/2)(\tilde{g}^k)^T H^k \tilde{g}^k + \tilde{\alpha}^k \leq \varepsilon$ frequently leads to a premature termination. Therefore criterion $(1/2)(\tilde{g}^k)^T \tilde{g}^k + \tilde{\alpha}^k \leq 10^3 \, \varepsilon$ should be satisfied simultaneously.

- Limited-memory variable metric methods in the compact form for general nonsmooth problems (Haarala + Miettinen + Mäkelä (to appear)).
- Shifted limited-memory variable metric methods for general nonsmooth problems.
- Partitioned variable metric methods for partially separable nonsmooth problems.
- Special methods for partially separable minimax problems.

Shifted limited-memory VM methods:

We have tested a simple strategy. The shifted limited-memory VM update is applied after every descent step. It is also used after the zero step if $(\tilde{g}^k)^T H^{k+1} \tilde{g}^k \leq (\tilde{g}^k)^T H^k \tilde{g}^k$. In the opposite case, matrix H^k is kept unchanged.

Partially separable nonsmooth problems:

$$F(x) = \sum_{i=1}^{m} f_i(x)$$

where $f_i(x)$, $1 \leq i \leq m$, are nonsmooth functions depending on a small number of variables (n_i - say) (m is usually large). A typical example is

$$F(x) = \sum_{i=1}^{m} |f_i(x)|$$

(sum of absolute values). Packed quantities $\hat{f}_i : R^{n_i} \to R$, $\hat{x}_i \in R^{n_i}$, $\hat{g}_i \in R^{n_i}$ (gradients), $\hat{G}_i \in R^{n_i \times n_i}$ (Hessian matrices). Packed quasi-Newton conditions $\hat{B}_i^{k+1} \hat{s}_i^k = \hat{y}_i^k$, where $\hat{s}_i^k = \hat{x}_i^{k+1} - \hat{x}_i^k$ and $\hat{y}_i^k = \hat{g}_i^{k+1} - \hat{g}_i^k \Rightarrow$ packed quasi-Newton updates.

Direction vector:

$$B^k s^k = -\tilde{g}^k, \quad B^k = \sum_{i=1}^{m} B_i^k, \quad \tilde{g}^k = \sum_{i=1}^{m} \tilde{g}_i^k.$$

Here B^k is a large sparse matrix. We use a sparse Choleski (or Gill-Murray) decomposition $B^k = L^k D^k (L^k)^T$. Furthermore, $w^k = -(1/2)(s^k)^T \tilde{g}^k + \tilde{\alpha}^k$.

Quadratic programming subproblem:

Matrix H^k is replaced by sparse Choleski (or Gill-Murray) decomposition $B^k = L^k D^k (L^k)^T$. Matrix multiplications are replaced by solutions of systems with triangular matrices (back elimination). Thus we obtain Lagrange multipliers λ_1, λ_2, λ_3. The aggregate subgradients are obtained by the formula

$$\tilde{g}_i^{k+1} = \lambda_1^k g_i^k + \lambda_2^k g_i^{k+1} + \lambda_3^k \tilde{g}_i^k, \quad 1 \le i \le m$$

BFGS update after a descent step:

For $1 \le i \le m$, we set

$$\hat{B}_i^{k+1} = \hat{B}_i^k + \frac{\hat{y}_i^k (\hat{y}_i^k)^T}{(\hat{s}_i^k)^T \hat{y}_i^k} - \frac{\hat{B}_i^k \hat{s}_i^k (\hat{B}_i^k \hat{s}_i^k)^T}{(\hat{s}_i^k)^T \hat{B}_i^k \hat{s}_i^k}, \quad (\hat{s}_i^k)^T \hat{y}_i^k > 0$$

$$\hat{B}_i^{k+1} = \hat{B}_i^k, \quad (\hat{s}_i^k)^T \hat{y}_i^k \le 0$$

Rank-1 update after a zero step:

For $1 \le i \le m$, we set $\hat{v}_i^k = \hat{y}_i^k - \hat{B}_i^k \hat{s}_i^k$ and

$$\hat{B}_i^{k+1} = \hat{B}_i^k + \frac{\hat{v}_i^k (\hat{v}_i^k)^T}{(\hat{s}_i^k)^T \hat{v}_i^k}, \quad (\hat{s}_i^k)^T \hat{v}_i^k > 0$$

$$\hat{B}_i^{k+1} = \hat{B}_i^k, \quad (\hat{s}_i^k)^T \hat{v}_i^k \le 0$$

Computational experiments:

We give a numerical comparison of four methods:

- PBM - Proximal bundle method.
- NVM - Nonsmooth variable metric method.
- SNVM - Shifted limited-memory nonsmooth variable metric method.
- PNVM - Partitioned nonsmooth variable metric method.

These methods were tested by using the set of 22 partially separable nonsmooth test problems (sums of absolute values) with 50, 500 and 1000 variables (subroutine `TEST15` in `www.cs.cas.cz/~luksan/test.html`).

N	MET	NIT	NEV	NF	TIME
50	PBM	55960	56854	3	29.61
	NVM	28325	28405	-	4.06
	SNVM	42243	42326	-	4.50
	PNVM	13421	13557	-	2.53
500	NVM	91832	91973	2	1281.74
	SNVM	88389	88409	3	119.91
	PNVM	15294	15369	-	32.74
1000	PNVM	14951	14976	-	166.13

Partially separable minimax problems:

$$F(x) = \max_{1 \le i \le m} f_i(x)$$

where $f_i(x)$, $1 \le i \le m$, are nonsmooth functions depending on a small number of variables (n_i -say) (m is usually large).

- Let $F(x) = f_i(x)$ for some $1 \le i \le m$. Then any subgradient of $f_i(x)$ is a subgradient of $F(x)$.

- At an arbitrary point $x \in R^n$, we can easily found a sparse subgradient $g(x) = g_i(x)$ containing only n_i nonzero elements.

- Quadratic programming subproblem

$$\frac{1}{2} d^T G^k d + v$$

subject to

$$-\alpha_j^k + d^T g^j \le v, \quad j \in \mathcal{J}_k, \quad -\alpha_a^k + d^T g_a^k \le v$$

has sparse constraints. If $G^k = \sigma^k I$, we obtain a sparse quadratic programming subproblem. Thus having an efficient sparse QP solver, we can use the proximal bundle method.

Nonlinear programming methods:

Let

$$F(x) = \max_{1 \le i \le m} |f_i(x)|$$

where $f_i(x)$, $1 \le i \le m$, are smooth functions depending on a small number of variables. Then minimization of F is equivalent to the sparse nonlinear programming problem with $n+1$ variables $x \in R^n$, $z \in R$: Minimize z subject to

$$-z \le f_i(x) \le z, \quad 1 \le i \le m$$

This problem can be solved by methods utilizing sparsity (SQP, interior point, nonsmooth equation).

- Choosing a suitable initial value of z we obtain a feasible starting point.

- Function $F(x)$ is an ideal merit function for the above problem.

Hybrid methods for large-scale nonlinear least squares:

$$F(x) = f^T(x)f(x) = \sum_{i=1}^{m} f_i^2(x)$$

where $f_i(x)$, $1 \leq i \leq m$, are smooth functions depending on a small number of variables (n_i - say) (m is usually large). Jacobian matrix $J(x) = [J_{ij}(x)] = [\partial f_i(x)/\partial x_j]$. Gradient $g(x) = J^T(x)f(x)$. Hessian matrix

$$G(x) = J^T(x)J(x) + \sum_{i=1}^{m} f_i(x)G_i(x)$$

($G_i(x)$ are Hessian matrices of $f_i(x)$, $1 \leq i \leq m$. Gauss-Newton matrix $J^T(x)J(x)$. Second-order matrix $C(x) = G(x) - J^T(x)J(x)$. We assume that matrices $J(x)$, $C(x)$ and $G(x)$ are sparse.

Gauss-Newton method:

The Hessian matrix G is replaced by J^TJ. Matrix J^TJ is frequently ill-conditioned (even singular) $\Rightarrow$ trust - region realization. If the minimum value $F(x^*)$ is large (large residual problem), the Gauss-Newton method can be inefficient.

Theorem 5. Let $F_k \to F^* = 0$ Q-superlinearly. Then $(F_k - F_{k+1})/F_k \to 1$. Let $F_k \to F^* > 0$. Then $(F_k - F_{k+1})/F_k \to 0$.

Philosophy of hybrid methods: The direction vector is obtained by the trust-region strategy using the quadratic model $(1/2)d^T Bd + f^T Jd$ and the constraint $\|d\| \leq \Delta$. If $F - F_+ > \underline{\vartheta} F$, then $B_+ = J_+^T J_+$ (Gauss-Newton method). If $F - F_+ \leq \underline{\vartheta} F$, then $B_+ = J_+^T J_+ + C_+$, where C_+ is an approximation of the second order term. Usually $\underline{\vartheta} \approx 10^{-4}$

Gauss-Newton method with the Newton corrections:

In the first iteration we use matrix $B = J^T J$. In the subsequent iterations, we set

$$B_+ = J_+^T J_+ \quad , \quad F - F_+ > \underline{\vartheta} F$$

$$B_+ = J_+^T J_+ + \sum_{k=1}^{m} f_k^+ G_k^+ \quad , \quad F - F_+ \leq \underline{\vartheta} F$$

where $J_+ = J(x_+)$ and $f_k^+ = f_k(x_+)$, $G_k^+ = G_k(x_+)$, $1 \leq k \leq m$, ($G_k(x_+)$ is a difference approximation of the Hessian matrix of $f_k(x_+)$).

Gauss-Newton method with the Marwil corrections:

In the first iteration we use matrix $B = J^T J$. In the subsequent iterations, we set

$$B_+ = J_+^T J_+ \quad , \quad F - F_+ > \underline{\vartheta} F$$
$$B_+ = \mathcal{P}_S \mathcal{P}_{QG}(J_+^T J_+) \quad , \quad F - F_+ \leq \underline{\vartheta} F$$

where

$$\mathcal{P}_S W = (W + W^T)/2$$

for a given square matrix W and

$$\mathcal{P}_{QG} B = \mathcal{P}_G(B + us^T)$$

Here $u \in R^n$ solves linear system $Du = y - Bs$ with diagonal matrix D such that

$$D_{ii} = \sum_{B_{ij} \neq 0} s_j^2$$

and

$$\begin{aligned} (\mathcal{P}_G W)_{ij} &= W_{ij}, \quad B_{ij} \neq 0 \\ (\mathcal{P}_G W)_{ij} &= 0, \quad B_{ij} = 0 \end{aligned}$$

(gangster operator).

Computational experiments:

We give a numerical comparison of five methods:

- GN - Gauss-Newton method.
- GNN - Gauss-Newton method with the Newton corrections.
- GNM - Gauss-Newton method with the Marwil corrections.
- DN - Discrete Newton method.
- PVM - Partitioned variable metric method.

These methods were implemented using various step-length (SL) strategies

- MS - Optimum locally constrained step of Moré and Sorensen.
- DL - The dog-leg strategy of Powell.
- LS - Standard line-search.

These algorithms were tested by using a set of 52 sparse least-squares test problems with 1000 variables (subroutines `TEST15` and `TEST18` in `www.cs.cas.cz/~luksan/test.html`). The results are given in the following Table, where `NIT` is the total number of iterations, `NEV` is the total number of function evaluations, NF is the number of failures and `TIME` is the total computational time (in seconds).

SL	MET	NIT	NEV	NF	TIME
MS	GN	8542	8929	1	72.00
	GNN	5499	5801	-	51.94
	GNM	6434	6801	-	62.88
	DN	7804	52398	1	202.07
DL	GN	9244	9602	-	38.84
	GNN	7767	8216	-	35.68
	GNM	6851	7029	-	25.87
	DN	10326	91181	-	171.98
LS	PVM	12093	16285	1	99.17

Large-scale trust-region subproblems:

Let

$$Q(d) = \frac{1}{2} d^T B d + g^T d$$

We seek $d \in R^n$ in such a way that $\|d\| \leq \Delta$ (Euclidean norm), $\|d\| < \Delta \Rightarrow \|Bd + g\| \leq \overline{\omega}\|g\|$ with $0 \leq \omega < 1$ and

$$-Q(d) \geq \sigma \|g\| \min\left(\Delta, \frac{\|g\|}{\|B\|}\right)$$

with $0 < \sigma \leq 1/2$. We assume that matrix B is sparse.

Basic methods for sparse trust-region subproblems:

- Optimum locally constrained step. Vector $d \in R^n$ minimizes $Q(d)$ over a trust-region defined by constraint $\|d\| \leq \Delta$. Necessary and sufficient conditions: $(B + \lambda I)d = -g,\ \lambda \geq 0,\ B + \lambda I$ positive semidefinite, $\|d\| \leq \Delta$, $\lambda(\Delta - \|d\|) = 0$. We solve nonlinear equation $1/\|d(\lambda)\| = 1/\Delta$ with $(B+\lambda I)d(\lambda) = -g$ by the Newton method using sparse Choleski decomposition of $B + \lambda I$ (Moré + Sorensen (1983)). This method is very robust but requires 2-3 sparse Choleski decompositions per iteration.
- Application of the conjugate gradient method. If a negative curvature is indicated or the trust-region is left, we stop on the boundary of the trust-region (Steihaug (1983), Toint (1981)). This method is very efficient especially when suitable preconditioning is used.
- Dog-leg method. A simple combination of the Cauchy step $d_C = (g^T g / g^T B g)g$ (the first CG step) and the Newton step $d_N = -B^{-1}g$ is used (Powell (1970), Denis + Mei (1975)). This is a very simple and efficient method which requires only one sparse Choleski decomposition per iteration.

- Multiple dog-leg method. Let $m << n$ (usually $m = 5$). Let d_{CG} be the step obtained after m CG iterations. If $\|d_{CG}\| < \Delta$, a combination of d_{CG} and $d_N = -B^{-1}g$ is used (Steihaug (1983)).
- Application of the Lanczos process. Initially, the conjugate gradient algorithm is used (the Lanczos tridiagonal matrix is constructed from the CG coefficients). If a negative curvature is indicated or the trust-region is left, we turn to the Lanczos process. In this case, $d = Z\tilde{d}$, where $\tilde{d}$ is obtained by minimizing quadratic function

$$\tilde{Q}(\tilde{d}) = (1/2)\tilde{d}^T T\tilde{d} + \|g\|e_1^T\tilde{d}$$

  subject to $\|\tilde{d}\| \leq \Delta$. Here $T = Z^T BZ$ is the Lanczos tridiagonal matrix and $Z^T Z = I$. (Gould + Lucidi + Roma + Toint (1997)). This method cannot be preconditioned, since preconditioning changes the original trust-region subproblem ($d^T Cd \leq \Delta^2$ for precontitioner C).
- The shifted Steihaug-Toint method. The preconditioned CG iterations are applied to system $(B+\tilde{\lambda}I)d = -g$, where $\tilde{\lambda}$ is an approximation of the optimum Lagrange multiplier obtained from a small-size Lanczos matrix.

The shifted Steihaug-Toint method:

Let $m << n$ (usually $m = 5$). We use the m Lanczos steps with matrix B and initial vector g to obtain tridiagonal matrix T of order m. Then we determine $\tilde{d}$ and $\tilde{\lambda}$ by minimizing

$$\tilde{Q}(\tilde{d}) = (1/2)\tilde{d}T\tilde{d} + \|g\|e_1^T\tilde{d}$$

subject to $\|\tilde{d}\| \leq \Delta$. The solution of this subproblem is inexpensive, since T is tridiagonal. Obtaining $\tilde{\lambda}$, we apply the Steihaug-Toint conjugate gradient method to system $(B + \tilde{\lambda}I)d(\tilde{\lambda}) = -g$ (instead of $Bd = -g$) and use the direction vector $d = d(\tilde{\lambda})$. The following theorem guarantees that $\|d\| < \Delta$ if and only if the optimum locally constrained step lies inside the trust region.

Theorem 6. Let $\tilde{\lambda}$ be the Lagrange multiplier of the above small-size subproblem and λ be the Lagrange multiplier obtained by the Moré-Sorensen method. Then $0 \leq \tilde{\lambda} \leq \lambda$.

- Vector $d(\tilde{\lambda})$ is usually closer to the Moré-Sorensen optimum locally constrained step in comparison with that obtained by the Steihaug-Toint method with the original matrix B.

- If $\tilde{\lambda} > 0$, matrix $B + \tilde{\lambda}I$ has smaller condition number than B.

Computational experiments:

We give a numerical comparison of seven methods for sparse trust-regions:

- MS - Optimum locally constrained step of Moré and Sorensen.
- DL - The dog-leg strategy of Powell.
- MDL - The multiple dog-leg strategy ($m = 5$).
- ST - The basic Steihaug-Toint method.
- GLRT - The method of Gould, Lucidi, Roma and Toint based on the Lanczos process.
- PST - Preconditioned Steihaug-Toint method (with the incomplete Choleski preconditioner).
- PSST - Preconditioned shifted Steihaug-Toint method ($m = 5$).

These algorithms were used for solving trust-region subproblems arising in the realization of the discrete Newton method. They were tested by using a set of 22 sparse least-squares test problems with 5000 variables (subroutine `TEST14` in `www.cs.cas.cz/~luksan/test.html`). The results are given in the following Table, where `NIT` is the total number of iterations, `NEV` is the total number of function evaluations, NCG is the total number of the CG iterations and `TIME` is the total computational time (in seconds).

N	MET	NIT	NEV	NCG	TIME
1000	MS	1918	1955	-	4.65
	DL	2515	2716	-	4.42
	MDL	2292	2456	12203	4.61
	ST	3329	3784	53573	8.20
	GLRT	3107	3444	55632	8.53
	PST	2631	2823	910	5.14
	PSST	1999	2046	1161	4.25
5000	MS	8391	8566	-	2:02.44
	DL	9657	10133	-	1:55.77
	MDL	8938	9276	47236	2:02.84
	ST	16894	19163	358111	6:04:42
	GLRT	14679	16383	366695	6:41.45
	PST	10600	11271	3767	2:25.42
	PSST	8347	8454	4329	1:48.87

Conclusions:

- Direct methods MS and DL based on the sparse Choleski decomposition are very efficient for our set of test problems. Iterative methods require a suitable preconditioning.
- The Moré-Sorensen strategy MS gives the best approximation of the optimum locally constrained step and decreases the number of the major iterations.
- Our new strategy PSST can be efficiently preconditioned. It gives a good approximation of the optimum locally constrained step.