# An Adaptive Nonmonotone Trust Region Method Based on a Structured Quasi Newton Equation for the Nonlinear Least Squares Problem[⋆]

Graciela Croceri, Gonzalo Pizarro and Graciela Sottosanto[⋆⋆]

Department of Mathematics, Comahue National University, Santa Fe 1400, 8300
Neuquén, Argentina

**Abstract.** In this work an iterative method to solve the nonlinear least squares problem is presented. The algorithm combines a secant method with a strategy of nonmonotone trust region. In order to define the quadratic model, the Hessian matrix is chosen using a secant approach that takes advantage of the structure of the problem, and the radius of the trust region is updated following an adaptive technique. Moreover, convergence properties of this algorithm are proved. The numerical experimentation, in which several ways of choosing the Hessian matrix are compared, shows the efficiency and robustness of the method.

**Keywords:** Trust region-Least Squares Problem-Structured Secant Approximation

## 1 Introduction

This paper is concerned with an algorithm to solve the unconstrained nonlinear least squares problem.

There are many and well known reasons to study methods for solving the nonlinear least squares problems. These problems appear frequently in various fields of science and they are applied in the analysis of statistical data, in social sciences, in data fitting and parameter estimation, and in more traditional fields like engineering and physics sciences.

The problem to consider is the following:

$$minf(x) = \frac{1}{2}||F(x)||_2^2 = \frac{1}{2}\sum_{i=1}^{m} F_i(x)^2, \tag{1}$$

where the residual function is $F(x) = (F_1(x), \ldots, F_m(x))^T$, the components $F_i : \mathbb{R}^n \to \mathbb{R}$ for $i = 1, \ldots, m$, are twice continuously differentiable and, in general, nonlinear functions, and $||.||$ denotes the Euclidean norm.

[⋆] This work was supported by Comahue National University (Grant E097/13).

[⋆⋆] Corresponding author: graciela.sottosanto@faea.uncoma.edu.ar.

The gradient of the function $f$ is

$$\nabla f(x) = J(x)^T F(x), \tag{2}$$

where $J : \mathbb{R}^{m \times n}$ is the Jacobian matrix of $F$, and the Hessian matrix of $f$ is

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{i=1}^{m} F_i(x) \nabla^2 F_i(x). \tag{3}$$

In the following we denote $f_k = f(x_k)$, $\nabla f_k = \nabla f(x_k)$, $J_k = J(x_k)$, $F_k = F(x_k)$, and $s_k = x_{k+1} - x_k$.

Although it is possible to solve the problem (1) with the traditional algorithms for unconstrained minimization problems, specific algorithms have been developed. The most efficient methods use the special structure of the problem and quasi-Newton strategies.

In particular, we will introduce an algorithm that uses a nonmonotone trust region with an adaptive radius and a structured secant approach of a portion of the Hessian matrix.

This work is organized as follows. In the next section we present the nonmonotone adaptive trust region. Section 3 is devoted to the structured secant approximations, the proposed algorithm is in Section 4. In Section 5 we analyze theoretical properties and global convergence of the method. Numerical experimentation is described in Section 6. Also, for visualizing the numerical results, performance profiles are used to show a good graphical comparison. In the last section, comments and concluding remarks are presented.

## 2 The nonmonotone trust region

There exist lots of iterative methods for solving the unconstrained minimization problem. In many cases they require the addition of a line search or trust region techniques, in order to guarantee the global convergence results.

The classic globalization strategy of trust region is a very popular iterative process that requires in each iteration the solution of the following subproblem

$$\begin{aligned} m_k(s) &= f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T B_k s \\ s.t. \quad & ||s||_2 \leq \delta_k \end{aligned} \tag{4}$$

where $B_k \in \mathbb{R}^{n \times n}$ is a symmetric matrix.

It is well known that $m_k$ is a quadratic model of $f$ around $x_k$ and $\delta_k$ is the radius of the trust region. If the function achieves sufficient reduction at the minimizer of the quadratic model, the trial step $s_k$ is accepted and the new point, $x_{k+1} = x_k + s_k$, is obtained, otherwise the radius of the region is reduced and is updated following a standard scheme [2,3]. Under mild conditions, this approach builds a sequence of iterates $\{x_k\}$ such that $\{f(x_k)\}$ is monotonically decreasing. However, some research suggests that if the objective function falls in very steep and narrow valleys, the method slows down the convergence rate and it can be made very short steps and winding paths.

In the last years, variants of the method have been proposed so that relaxing the requirement of monotonicity, more efficient algorithms are obtained. The idea of building nonmonotone methods dates back to 1986, when Grippo et al. [4] presented a nonmonotone line search technique combined with a Newton method for unconstrained optimization. In 1993 Deng et al. [5] extended nonmonotone techniques of line search to trust region. Subsequently, many authors have proposed variants of the nonmonotone trust region method [6,7,8,9,10]. They have been able to show that the modified algorithm has similar properties of convergence to the usual algorithm of the trust region. The basic objective of the modifications is to decrease the number of inefficient iterations, reducing the amount of subproblems to solve.

Once the subproblem (4) is solved with a *double-dogleg* strategy, and the trial step $s_k$ is found, we need to decide if the new point $x_{k+1} = x_k + s_k$ is accepted. The classical acceptance condition of the step is relaxed so that the new point is compared to the worst in a certain number of previous steps. So the actual reduction of the function

$$Ared_k = f_{l_k} - f_{k+1} \tag{5}$$

is compared with the predicted reduction

$$Pred_k = f_{l_k} - m_k(s_k), \tag{6}$$

where

$$f_{l_k} = f(x_{l(k)}) = \max_{0 \leq j \leq p(k)} f(x_{k-j}), \tag{7}$$

$p(0) = 0$ and $0 \leq p(k) \leq min\{p(k-1)+1, N\}$, $k \geq 1$ and $N$ is a nonnegative integer.

In this way, the new point is compared to the worst point in a certain number of previous steps.

Then, if

$$r_k = \frac{Ared_k}{Pred_k} \geq \eta \tag{8}$$

we accepted the trial step, otherwise, it is rejected. Thus, in the ratio we compare the variation of the function in the two steps and the difference between the quadratic model and the value of the function in the worst point considered in the last $p(k)$ iterations. We observe that the method does not generate a monotone decreasing sequence $\{f(x_k)\}$. However, when $N \geq 1$, the acceptance criterion of the step should guarantee some decrease of $f(x_k + s_k)$. When $N = 0$ the method is reduced to the classic trust region method.

### 2.1 Adaptive trust region radius

In this subsection we discuss a nonmonotone version of the trust region strategy where the radius is selected in an adaptive way. In recent years, several authors, among them [11,12,13,14], have proposed strategies in which the adjustment of the radius in each internal iteration is performed using some procedures that involve values of the function, the gradient or the Hessian matrix in the current point.

In this case we propose a method that adjusts it automatically according to the information of the current point, and we calculate the radius by a simple formula that only involves values of the objective function. In this case, we use

$$\delta_k = \begin{cases} c^q R_k & if\ k = 0 \\ c^q max\{R_k, \delta_{k-1}\} & if\ k \geq 1 \end{cases} \tag{9}$$

with $R_k = \theta_k f_{l_k} + (1 - \theta_k)||F_k||$, for $0 < c < 1$, $q$ is a positive integer which increases in each inner iteration when the relationship between the actual reduction and the predicted reduction is different to the expected. So, q is the smallest nonnegative integer ensuring that the trust region is greater than $\eta \in (0, 1)$. The other parameter is $0 \leq \theta_k \leq 1$, which is adaptive so that larger radios are obtained when the current point is far away from the solution. In this way, an adaptive nonmonotone method is constructed.

## 3 The structured secant approximation

As we have already mentioned, many specific algorithms that exploit their particular structure, have been developed to solve the problem of the nonlinear least squares. The Gauss-Newton and the Levenberg Marquardt methods [1,15], are based on the observation that the second order part of the Hessian matrix tends to zero in the optimum for zero residual problems. In this case, they have a q-quadratically convergence rate, but their performance is poor for nonzero residual problems. To improve this situation, structured quasi-Newton methods were proposed.

From the structured BFGS update proposed by Al Baali et al. [16], in 1989, Dennis et al. [17] derived a principle that takes into account the structure with which they achieved local superlinear convergence. More recently, Huschens [18] proposed a new structured quasi Newton method that converges quadratically for zero residual problems, and superlinear for nonzero residual problems. Other works such as [19,20] have been published later. In the first of them, the authors proposed a structured quasi Newton method that uses a “product structure”, and showed that the resulting algorithm is quadratically convergent for the zero-residual case and superlinearly convergent for the nonzero-residual case. In the last, the authors provide global convergence of a hybrid Gauss-Newton structured BFGS. The structure principle as designated by Zhou and Chen [20], was originally introduced, by Dennis, Martinez and Tapia [17].

All these approximations take advantage of the structure present in the Hessian matrix of the problem. We consider the Hessian matrix composed of a term containing first order information, which is the available part, and the other containing the second order information, that is

$$\nabla^2 f(x) = J(x)^T J(x) + S(x) = C(x) + S(x), \tag{10}$$

where $C : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ and $S(x) = \sum_{i=1}^m F_i(x)\nabla^2 F_i(x)$.

The objective is to construct a secant approximation of the portion containing the second order information.

If $B_{k+1}$ is an approximation of $\nabla^2 f$ and $A_{k+1}$ is an approximation of $S$ then

$$B_{k+1} = C_{k+1} + A_{k+1}, \tag{11}$$

with $A_{k+1} = A_k + \Delta(s_k, y_k^\sharp, A_k, v_k)$.

We use a secant update BFGS type

$$\Delta(s, y^\sharp, A, v) = \frac{(y^\sharp - As)v^T + v(y^\sharp - As)^T}{v^T s} - \frac{(y^\sharp - As)^T s}{(v^T s)^2} vv^T, \tag{12}$$

where

$$v(s, y, B) = y + \left(\frac{y^T s}{s^T B s}\right)^{1/2} Bs, \tag{13}$$

$$y_k = y_k^\sharp + J_{k+1}^T J_{k+1} s_k, \tag{14}$$

and $y_k^\sharp$ is an approximation of $S_{k+1}s_k$. Then

$$B_{k+1} = C_{k+1} + A_k + \Delta(s_k, y_k^\sharp, A_k, v(s_k, y_k, B_k)), \tag{15}$$

and so it is reasonable to define

$$A_{k+1} = A_k + \Delta(s_k, y_k^\sharp, A_k, v(s_k, y_k, B_k)) \tag{16}$$

as the structured secant update of $A$.

### 3.1 Several ways of choosing $y^\sharp$

In this work, we have numerical evaluations for three different choices of $y^\sharp$.

Dennis [23] and Bartholomew-Biggs [24] introduced, independently, the following approximation

$$y_k^\sharp = (J_{k+1} - J_k)^T F_k. \tag{17}$$

They obtained this expression by considering the secant equation of $A_{k+1}$ and observing that

$$\nabla^2 F_i(x_{k+1}) s_k \approx \nabla F_i(x_{k+1}) - \nabla F_i(x_k), \tag{18}$$

for $i = 1, \dots, m$. Then

$$\begin{aligned}
\left[\sum_{i=1}^m F_i(x_{k+1})\nabla^2 F_i(x_{k+1})\right] s_k &= \sum_{i=1}^m F_i(x_{k+1})\left[\nabla^2 F_i(x_{k+1}) s_k\right] && (19)\\
&\approx \sum_{i=1}^m F_i(x_{k+1})\left[\nabla F_i(x_{k+1}) - \nabla F_i(x_k)\right] && (20)\\
&= (J_{k+1} - J_k)^T F_k. && (21)
\end{aligned}$$

On the other hand, Huschens, in his work [18], using a scaling of the Hessian matrix of the problem

$$\nabla^2 f(x) = C(x) + ||F(x)|| \sum_{i=1}^{m} \frac{F_i(x)}{||F(x)||} \nabla^2 F_i(x) \tag{22}$$

and

$$B(x) = C(x) + ||F(x)||A, \tag{23}$$

showed how the structured product technique can be exploited to obtain values of $y^\sharp$ that allow to improve the convergence. From this consideration, in the works of Zhang et al. [19] and Zhou and Chen [20], the following updates were obtained:

$$y_k^\sharp = \frac{(J_{k+1} - J_k)^T F_k}{||F_k||} + \frac{3F_k}{||s_k||^2} \left[ (J_{k+1} - J_k)s_k - 2(F_{k+1} - F_k) \right], \tag{24}$$

$$y_k^\sharp = (J_{k+1} - J_k)^T F_{k+1} \frac{||F_{k+1}||}{||F_k||}. \tag{25}$$

## 4 The algorithm

Now we give a description of our nonmonotone, adaptive and structured trust region method to solve the problem (1).

**Algorithm 1** *Given $x_0 \in \mathbb{R}^n$, the positive constants $c, \eta \in (0, 1)$, $\theta_0 \in [0, 1]$, $\epsilon > 0$, a positive integer $N$ and a symmetric matrix $B_0 \in \mathbb{R}^{n \times n}$:*

**Step 0.** *Set $k = 0$ and calculate $\delta_0 = \frac{1}{2}||F_0||_2^2$.*
**Step 1.** *If $x_k$ is a stationary point of the problem the algorithm terminates, otherwise, set $q = 0$ and go to Step 2.*

**Step 2.** *(Compute the trial step)*
*Solve approximately the problem (4), obtain $s_k$ and define $x_{k+1} = x_k + s_k$.*

**Step 3.** *(Evaluation of the trial step)*
*Calculate $Ared_k$, $Pred_k$ and $r_k$ according to (5) and (6).*

**Step 4.** *(Update of the trust region)*

– *If $r_k < \eta$ then $x_{k+1} = x_k$, assign $q \leftarrow q + 1$, update $\delta_k$ by (9) and go to Step 2.*

- *If $r_k \geq \eta$ then update $\delta_k$ by (9) and go to Step 5.*

**Step 5.** *(Update all the information)*
*Assign $k \leftarrow k+1$, obtain $B_k = J_k^T J_k + \alpha A_k$ according to (16), generate $\theta_k$ by an adaptive formula and go to Step 1.*

*Note 1.* We observe that the value of $\alpha$ is 1 if we use (17) and (25) and $||F_{k+1}||$, if (24) is used.

## 5 Analysis of convergence

In this section we prove the well definition of the algorithm and its global convergence. To do this we need to establish the following assumptions about the problem. The function $f$, the gradient $\nabla f$ and the approximation $B$ are asked to satisfy the following assumptions:

**H1.** There exists an open convex subset $S \subseteq R^n$ such that, for all $k$, $x_k$, $x_k+s_k \in S$.
**H2.** $\nabla f \in Lip_\gamma(S)$.
**H3.** The sequence of Hessian approximation $\{B_k\}$ is uniformly bounded.
**H4.** The set $S_0 = \{x \in R^n : f(x) \leq f(x_0)\}$ is compact.

The next lemmas are crucial in order to obtain the well definition of the algorithm and to analyze the convergence of the method.

**Lemma 1.** *If $s_k$ is the solution of the trust region subproblem (4) then*

$$Pred_k \geq \frac{1}{2}||\nabla f_k|| min \left\{ \delta_k, \frac{||\nabla f_k||}{||B_k||} \right\}. \tag{26}$$

*Proof.*

$$Pred_k = f_{l_k} - m_k(s_k) \geq f_k - m_k(s_k) = m_k(0) - m_k(s_k)$$

By using [25, Lemma 6.1.3] results

$$m_k(0) - m_k(s_k) = \frac{1}{2}||\nabla f_k|| min \left\{ \delta_k, \frac{||\nabla f_k||}{||B_k||} \right\}, \tag{27}$$

what concludes the proof.

**Lemma 2.** *Assuming (H1)-(H3) we have*

$$|Ared_k - Pred_k| \leq \nu ||s_k||_2^2. \tag{28}$$

*Proof.* From the definitions 5 and 6 we have

$$\begin{aligned}|Ared_k - Pred_k| &= |f_{l_k} - f_{k+1} - f_{l_k} + m_k(s_k)| \\ &= | - f_{k+1} + m_k(s_k)| \\ &= | - f_{k+1} + f_k + \nabla f_k^T s_k + \frac{1}{2} s_k^T B_k s_k|.\end{aligned}$$

Applying the Taylor's Theorem it results

$$f(x_k + s_k) = f_k + \nabla f_k^T s_k + \int_0^1 [\nabla f(x_k + ts_k) - \nabla f(x_k)]^T s_k dt. \tag{29}$$

Then,

$$\begin{aligned}|Ared_k - Pred_k| &\leq \left| \int_0^1 [\nabla f(x_k + ts_k) - \nabla f(x_k)]^T s_k dt \right| + \frac{1}{2}|s_k^T B_k s_k| \\ &\leq \int_0^1 \left| (\nabla f(x_k + ts_k) - \nabla f(x_k))^T s_k \right| dt + \frac{1}{2}|s_k^T B_k s_k| \\ &\leq \int_0^1 ||\nabla f(x_k + ts_k) - \nabla f(x_k)|| \; ||s_k|| dt + \frac{1}{2}|s_k^T B_k s_k| \\ &\leq \gamma \int_0^1 ||(x_k + ts_k - x_k)|| \; ||s_k|| dt + \frac{1}{2}|s_k^T B_k s_k| \\ &\leq \frac{\gamma}{2}||s_k||^2 + \frac{1}{2}\beta||s_k||^2 = \nu||s_k||^2,\end{aligned}$$

where $\gamma$ is the Lipschitz constant and $\beta$ is a bound of the norm of the Hessian approximation.
Thus, (28) is obtained with $\nu = \frac{1}{2}(\gamma + \beta)$, and we complete the proof.

The following result guarantees the well definition of the algorithm.

**Theorem 1.** *Suppose that assumptions (H1)-(H3) hold. Then the Algorithm cannot cycle infinitely.*

*Proof.* Suppose by contradiction, that the Algorithm cycles infinitely between step 2 and step 4 at iteration $k$. That is, we assume that the internal iteration cycles infinitely, i. e., $\delta_{k(q)} \to 0$ as $q \to \infty$.
Note that between steps 2 and 4 we have $x_{k(q)} = x_k$, $\nabla f(x_{k(q)}) = \nabla f_k$, $B(x_{k(q)}) = B_k$, $s_{k(q)}$ and $\delta_{k(q)}$.
On the other hand, we have that $r_{k(q)} \leq \eta$ and

$$|r_{k(q)} - 1| = \left| \frac{Ared_{k(q)} - Pred_{k(q)}}{Pred_{k(q)}} \right| = \frac{|Ared_{k(q)} - Pred_{k(q)}|}{|Pred_{k(q)}|}. \tag{30}$$

Using the fact that $x_k$ in not optimum of (1) we can conclude that there exists a constant $\epsilon > 0$ such that $||\nabla f_k|| > \epsilon$.
From (H3) in Lemma 1, it results

$$Pred_k \geq \frac{1}{2}||\nabla f_k||min\left\{\delta_k, \frac{||\nabla f_k||}{\beta}\right\}. \tag{31}$$

Then with our contradiction hypothesis we have

$$Pred_k \geq \frac{1}{2}\epsilon\ min\left\{\delta_k, \frac{\epsilon}{\beta}\right\}. \tag{32}$$

This last result allows us to write in (30)

$$|r_{k(q)} - 1| \leq \frac{\nu||s_{k(q)}||^2}{\frac{1}{2}||\nabla f_k||\min\left\{\delta_{k(q)}, \frac{||\nabla f_k||}{\beta}\right\}} \leq \frac{\nu\delta_{k(q)}^2}{\frac{1}{2}\epsilon\ \min\left\{\delta_k, \frac{\epsilon}{\beta}\right\}} \rightarrow 0,\ as\ q \rightarrow \infty.$$

Therefore, there exists a sufficiently large $q$, called $q_k$, such that $r_{k(q_k)} \geq \eta$ and the inner cycle of the Algorithm 1 is well defined.

In the following theorem we establish the result of global convergence.

**Theorem 2.** *If the hypothesis (H1)-(H4) are satisfied then*

$$\lim_{k\rightarrow\infty} ||\nabla f_k|| = 0. \tag{33}$$

*Proof.* By contradiction, for all sufficiently large $k$, assume that there exists a constant $\epsilon > 0$ and an infinite subset $K \subseteq N \cup 0$ satisfying $||\nabla f_k|| > \epsilon$ for $k \in K$. Also by the definition of $R_k$ and $f_{l_k}$

$$\begin{aligned} ||F_k|| &= (1-\theta)||F_k|| + \theta||F_k|| \\ &\leq (1-\theta)f_{l_k} + \theta||F_k|| = R_k \\ &\leq (1-\theta)f_{l_k} + \theta f_{l_k} = f_{l_k}. \end{aligned}$$

Then

$$||F_k|| \leq R_k \leq f_{l_k}. \tag{34}$$

Furthermore, let us in mind that in the definition of $Ared_k$ the equation (32) allows us to write

$$f_{l_k} - f_{k+1} \geq \eta Pred_k \geq \frac{\eta}{2}\epsilon\ min\left\{\delta_k, \frac{\epsilon}{\beta}\right\}. \tag{35}$$

Since the radius is obtained by the formula (9) and by the definition of $R_k$ we have

$$\begin{aligned} f_{l_k} - f_{k+1} &\geq \frac{\eta}{2}\epsilon\, min\left\{c^q \max\{R_k, \delta_{k-1}\}, \frac{\epsilon}{\beta}\right\} && (36)\\ &\geq \frac{\eta}{2}\epsilon\ \min\left\{c^q R_k, \frac{\epsilon}{\beta}\right\} && (37)\\ &\geq \frac{\eta}{2}\epsilon\ \min\left\{c^q(\theta f_{l_k} + (1-\theta)||F_k||), \frac{\epsilon}{\beta}\right\} && (38)\\ &\geq \frac{\eta}{2}\epsilon\ \min\left\{c^q(\theta||F_k|| + (1-\theta)||F_k||), \frac{\epsilon}{\beta}\right\} && (39)\\ &\geq \frac{\eta}{2}\epsilon\ \min\left\{c^q||F_k||, \frac{\epsilon}{\beta}\right\} = \frac{\eta}{2}\epsilon L_k, && (40) \end{aligned}$$

where $L_k = \min\left\{c^q||F_k||, \frac{\epsilon}{\beta}\right\}$ and the inequality (39) in is obtained by using (34). Then

$$f_{l_k} - f_{k+1} \geq \frac{\eta}{2}\epsilon L_k. \tag{41}$$

Now, by taking a limit from both sides of the last inequality (41), as $k \to \infty$, we have that $\lim_{k\to\infty} L_k = 0$, and $q \to \infty$ for sufficiently large $k \in K$. This fact only is possible if $q \to \infty$ as $k \in K$. This is a contradiction with Theorem 1, where the well definition of the algorithm is proved, and then, the stopping criterion of the Algorithm 1 holds.

## 6 Numerical results

In order to show the behavior of the algorithm, we considered a set of 26 test problems from the literature [26,19]. The dimension of the problems range from 3 to 180 and the component functions range also from 4 to 300. Only 10 of them belong to the group of zero residual. The problems that are indicated in the table 1 with $(*)$ belong to the non-zero residual group. Each problem was considered with the initial points proposed in [19]. The algorithm has been coded in Scilab. The code has been performed on a personal computer with Core i3 processor with 3.8 Gb of RAM memory, using Ubuntu 14.04 Linux operation system.

The purpose of numerical experimentation is to show the behavior of the algorithm with the three different ways of the structure of the matrix. Simultaneously, the objective is to compare between them and against the traditional monotone trust region [1].

The nonmonotone trust region subproblem was approximately solved by using a quasi Newton method with the different approximations of the Hessian matrix that have been described in section 3. The monotone trust region subproblem was solved using standart dogleg procedure and $B_k$, the approximation to the Hessian matrix, was updated by using the approach of Dennis, Martinez and

Tapia [17].
The parameters are chosen as follows: $c = 0.5$ and $\eta = 0.1$. Moreover, all the algorithms are stopped whenever the total number of iterates exceeds 500 or the condition $\|\nabla f_k\| \leq \epsilon$, with $\epsilon = 10^{-6}$is satisfied.

In Table 1, the number of variables of the problem is indicated as $n$, and $m$ is the number of component functions of the residual. The number of iterations for each of the used secant approximations is reported. We indicated with ADMT the algorithm proposed with the secant approximation by Dennis, Martínez and Tapia [17] that corresponds to the choice of $y^\sharp$ (17), AZXZ indicates to the secant approximation due to Zhang, Xue and Zhang [19], with $y^\sharp$ (24), AZCH corresponds to the secant approximation that use the expression (25), proposed by Zhou and Chen [20] and, finally, AMTR designate to the monotone trust region. We have noted with $(-)$ when the problem could not be solved by the algorithm using a given update for the maximum number of established iterations.

Although ADMT and AMTR use the same secant update, the nonmonotone method uses less or equal number of iterations than the monotone in all cases of zero residual problems and in 94% of non-zero residual problems.

The AZCH strategy solves 92% of the all problems and with respect to the monotone method it uses less iterations in 77% of the cases.

We also take the advantages of the performance profile suggested by Dolan and Moré [27], which is a tool to compare the efficiency of algorithms. Therefore, we illustrate the results in Figures 1-3 according to the total number of external iterations for the zero and nonzero residual problems, and all the solved problems, respectively.

Figure 1 shows that the AZCH algorithm is the most efficient: it solves with less or equal number of iterations in 90 % of cases. Clearly, on the non-zero residual problems, the ADMT algorithm is the most efficient because it solve about 87% of the test problems with the least number of iterations (See Figure 2).

If the user does not know *a priori* whether the problem is zero or non-zero residual, Figure 3 shows that ADMT is the most reliable algorithm to solve it.

## 7 Concluding remarks

In this work an algorithm to solve the nonlinear least squares problem has been presented. A nonmonotone trust region strategy and a quasi Newton with several ways of choosing the approximation of the Hessian matrix, have been used to obtain the solution of the problem. The trust radius in each subproblem adjusts itself adaptively. The combination of these techniques allows us to have a method that leads us to decrease the number of subproblems to resolve. Global convergence of the proposed method is established.

The algorithm has been applied to the resolution of a set of standard test problems. Our preliminary computational experiments show that it is promising for solving the unconstrained nonlinear least square problem. The first numerical results indicate that the technique is efficient to solve the zero and nonzero

**Table 1.** Test problems

| Problem | n | m | ADMT | AZXZ | AZCH | AMTR |
|---|---|---|---|---|---|---|
| Chained Rosembrock | 40 | 78 | 113 | 183 | 77 | – |
| Chained Wood | 100 | 186 | 165 | 158 | 116 | – |
| Chained Powell singular | 40 | 76 | 13 | 25 | 10 | 24 |
| Chained Cragg and Levy (*) | 100 | 245 | 34 | 95 | 129 | 23 |
| Generalized Broyden triangular | 60 | 60 | 5 | 11 | 4 | 5 |
| Generalized Broyden banded | 60 | 60 | 6 | 5 | 5 | 7 |
| Chained Freudenstein and Roth (*) | 140 | 248 | 17 | – | 22 | 17 |
| Wright and Holt zero residual | 60 | 300 | 6 | 7 | 6 | – |
| Toint quadratic merging (*) | 60 | 174 | 21 | 33 | 21 | – |
| Chained exponential (*) | 3 | 5 | 24 | – | 196 | 24 |
| Chained serpentine | 80 | 158 | 196 | 203 | 162 | – |
| Chained and modified (HS47) (*) | 11 | 18 | 91 | – | 347 | 365 |
| Chained and modified (HS47) (*) | 83 | 162 | 223 | – | 337 | – |
| Chained and modified (HS48) (*) | 29 | 63 | 53 | – | 52 | 84 |
| Chained and modified (HS48) (*) | 32 | 70 | 56 | – | 43 | 75 |
| Chained and modified (HS48) (*) | 38 | 84 | 62 | – | 62 | 129 |
| Sparse signomial (*) | 4 | 4 | 11 | 14 | 12 | 15 |
| Sparse exponential (*) | 16 | 28 | 5 | – | 5 | 7 |
| Sparse trigonometric (*) | 20 | 36 | 10 | 128 | 28 | – |
| Countercurrent reactors (*) | 40 | 40 | 355 | – | 343 | – |
| Countercurrent reactors (*) | 80 | 80 | 121 | 311 | 193 | – |
| Tridiagonal system | 140 | 140 | 39 | 37 | 28 | 64 |
| Structured Jacobian problem | 60 | 60 | 19 | – | 19 | 19 |
| Shifted discrete boundary value | 180 | 180 | 18 | 29 | 20 | 29 |
| Modified discrete boundary value (*) | 41 | 41 | 22 | – | – | 23 |
| Attracting Repelling (*) | 20 | 38 | 148 | 182 | – | 265 |

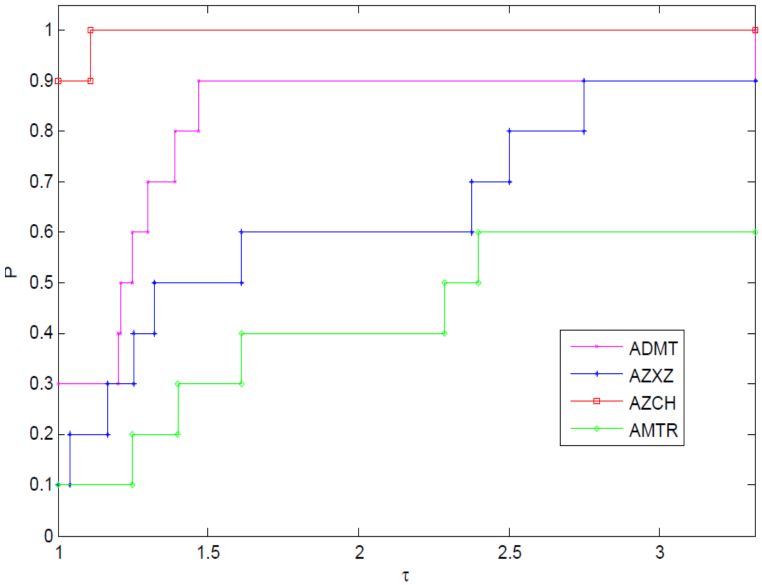

**Fig. 1.** Performance profile for zero residual problems

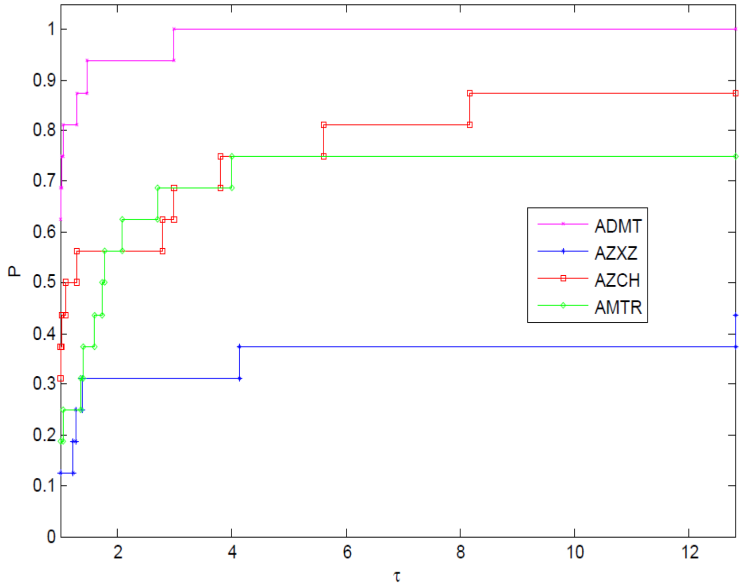

**Fig. 2.** Performance profile for nonzero residual problems

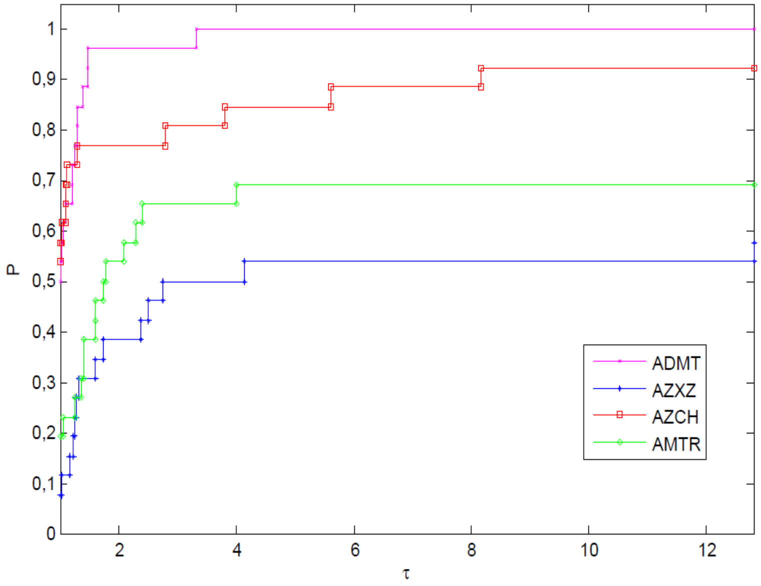

**Fig. 3.** Performance profile for all solved problems

residual problems. The results of numerical experimentation showed a better performance of the nomonotone methods with respect to the classic monotone trust region. The application of the algorithm to solving real world problems is a challenge which we are working.

**Acknowledgements** We thank the anonymous referees who allowed us to improve a previous version of our manuscript.

## References

1. Dennis, J., Schnabel, R.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations. SIAM, Philadelphia (1996)
2. Conn, A., Gould, N., Toint, P.: Trust Region Methods. SIAM, Philadelphia (2000)
3. Moré, J. J.: Recent developments in algorithms and software for trust region methods. In A. Bachem, M. Grotschel and B. Korte, eds., Mathematical Programming: The state of the art, Springer-Verlag, New York, 258–287 (1984)
4. Grippo, L., Lampariello, F., Lucidi, S.: A nonmonotone line search technique for Newton's method. SIAM J. Numer. Anal. 23 (4), 707–716 (1986)
5. Deng, N., Xiao, Y, Zhou, F.: Nonmonotonic trust region algorithm. Journal of Optimization Theory and Applications 76, 259–285 (1993)
6. Sun, W.: Nonmonotone trust region method for solving optimization problems. Applied Mathematics and Computation 156, 159–74 (2004)
7. Mo, J., Zhang, K., Wei, Z.: A nonmonotone trust region method for unconstrained optimization. Applied Mathematics and Computation 171, 371–384 (2005)

8. Ahookhosh, M., Amini, K.: An efficient nonmonotone trust-region method for unconstrained optimization. Numerical Algorithms 59 (4), 523–540 (2012)
9. Chen, J., Sun, W., Yang, Z.: A non-monotone retrospective trust-region method for unconstrained optimization. Journal of Industrial and Management Optimization 9 (4), 919–944 (2013)
10. Maciel, M. C., Mendonça, M., Verdiell, A.: Monotone and nonmonotone trust-region-based algorithms for large scale unconstrained optimization problems. Computational Optimization and Applications 54, 27–43 (2013)
11. Zhang, J., Zhang, X.: A nonmonotone adaptive trust region method and its convergence. Computers and Mathematics with Applications 45, 1469–1477 (2003)
12. Fu, J., Sun, W.: Nonmonotone adaptive trust-region method for unconstrained optimization problems. Appl. Math. Comput. 163, 489–504 (2005)
13. Esmaeili, H., Kimiaei, M.: A new adaptive trust-region method for system of nonlinear equations. Appl. Math. Modelling 38, 3003–3015 (2014)
14. Kamandi, A., Amini, K.: An improved adaptive trust-region algorithm. Optim. Lett. DOI 10.1007/s11590-016-1018-4 (2016)
15. Fletcher, R.: Practical Methods of Optimization. John Wiley, Chichester and New York, (1987)
16. Al-Baali, M., Fletcher, R.: Variational methods for non-linear least squares. J. Oper. Res. Soc. 36, 405–421 (1985)
17. Dennis, J., Martínez, H. J. and Tapia, R.: Convergence theory for the structured BFGS secant methods with an application to nonlinear least squares. Journal of Optimization Theory and Applications 61 (2), 161–178 (1989)
18. Huschens, J.: On the use of product structure in secant methods for nonlinear least square problems. SIAM J. Optim. 4, 108–129 (1994)
19. Zhang, J., Xue, Y., Zhang, K.: A structured secant methods based on a new quasi-Newton equation for nonlinear least squares problems. BIT Numerical Mathematics 43, 217–229 (2003)
20. Zhou, W., Chen, X.: Global Convergence of a New Hybrid Gauss-Newton Structured BFGS Method for Nonlinear Least Squares Problems. SIAM J. Optim. 20 (5), 2422–2441 (2010)
21. Dennis, J.: Toward a unified convergence theory for Newton-like methods. In L. B. Rall, ed., Nonlinear functional analysis and applications, New Yok Academic Press (1971)
22. Broyden, C. G., Dennis, J. E., Moré, J.: On the Local and Superlinear Convergence of Quasi-Newton Methods. IMA Journal of Applied Mathematics 12(3), 223–245 (1973)
23. Dennis, J.: A brief survey of convergence results for quasi Newton methods. Nonlinear Programming SIAM-AMS Proceedings 9, 185–199 (1976)
24. Bartholomew–Biggs, M.: The estimation of the Hessian matrix in nonlinear least square problem with non zero residuals. Mathematical Programming 12, 67–80 (1977)
25. Sun, W., Yuan, Y.: Optimization Theory and Methods. Nonlinear Programming. Springer, New York (2006)
26. Luksan, L., Vlcek, J.: Sparse and partially separable test problems for unconstrained and equality constrained optimization. Technical report No. 767, Institute of Computer Science, Academy of Sciences of the Czech Republic (1999)
27. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program., 91, 201–213 (2002)