**Int. J. Res. Ind. Eng. Vol. 6, No. 4 (2017) 339–349**

**International Journal of Research in Industrial Engineering**

**www.riejournal.com**


# A Barzilai Borwein Adaptive Trust-Region Method for Solving Systems of Nonlinear Equation

F. Rahpeymaii[1], M. Kimiaei[2*]

[1]Department of Mathematics, Payame Noor University, PO BOX 19395--3697, Tehran, Iran.
[2]Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, A-1090 Vienna, Austria.

**A B S T R A C T**

In this paper, we introduce a new adaptive trust-region approach to solve systems of nonlinear equations. In order to improve the efficiency of adaptive radius strategy proposed by Esmaeili and Kimiaei [8], Barzilai Borwein technique (BB) [3] with low memory is used which can truly control the trust-region radius. In addition, the global convergence of the new approach is proved. Computational experience suggests that the new approach is more effective in practice in comparison with other adaptive trust-region algorithms.

**Keywords**: Nonlinear equations, trust-region framework, adaptive radius, two-point gradient technique.

***Article history:*** Received: 26 October 2017 Accepted: 01 December 2017

## 1. Introduction

Consider the nonlinear system

$$F(x) = 0, \quad x \in R^n \tag{1}$$

which $F: R^n \to R^n$ is a continuously differentiable mapping in the following form: $F(x) = (F_1(x), F_2(x), \ldots, F_n(x))^T$. The nonlinear system (1), in which $F(x)$ has a zero, can be written as the following problem:

$$\min f(x) = \frac{1}{2}\|F(x)\|^2 \qquad \text{s.t} \quad x \in R^n, \tag{2}$$

which $\|\cdot\|$ denotes the Euclidean norm.

An efficient class of global approaches is presented, called the trust-region method. This method first defines a region around the current iterate $x_k$ as follows:

$$\Omega_k := \{x \mid \|x - x_k\| \le \Delta_k\},$$

* Corresponding author
E-mail: kimiaeim83@univie.ac.at
DOI: 10.22105/riej.2017.101854.1027

which $\Delta_k > 0$ is the trust-region radius and then the quadratic model is presented by

$$\min m_k(x_k + d) \coloneqq \frac{1}{2}\|F_k + J_k d\|^2 = f_k + d^T J_k^T F_k + \frac{1}{2} d^T J_k^T J_k d \qquad (3)$$
$$\text{s.t} \quad d \in R^n \text{ and } \|d\| \leq \Delta_k,$$

where $f_k \coloneqq f(x_k)$, $F_k \coloneqq F(x_k)$, and $J_k \coloneqq F'(x_k)$, Jacobian of $F(x)$. In the region $\Omega_k$, the approximate minimizer of the quadratic model is found by the step $d_k$, called the *trust-region step*. The goal of trust-region method is that the quadratic model implies to be an adequate representation of the objective function $f$. Let us now define the ratio

$$r_k \coloneqq \frac{f_k - f(x_k + d_k)}{m_k(x_k) - m(x_k + d_k)}, \qquad (4)$$

which has a key role in selecting new iterate $x_{k+1}$ and in updating the trust-region radius $\Delta_{k+1}$. Whenever $r_k$ is near 1, a good agreement can be obtained which leads to expand the trust-region radius for the next iteration. If $r_k \leq 0$, then we can shrink the trust-region radius, otherwise, the radius of trust-region will be fixed.

One of globalization technique is the traditional trust-region framework (TTR), (see [6, 18]). The updating radius of such method is very sensitive and has some disadvantages, cf. [8-11]. Some researchers [8-11, 23] to overcome these disadvantages, have tried to control the trust-region radius. Recently, in order to produce smaller (larger) radii whenever iterates are near (far away from) the optimizer, Esmaeili and Kimiaei [8] presented an efficient adaptive trust-region to solve Eq. (1) as follows:

$$\Delta_k = \begin{cases} c^{p_k} R_k, & \text{if } k = 0, \\ c^{p_k} \max\{R_k, \Delta_{k-1}\}, & \text{if } k \geq 1, \end{cases} \qquad (5)$$

where $0 < c < 1$, $p_k$ is the smallest nonnegative integer $p$ ensuring the trust-region ratio is greater than a real-valued parameter $\mu \in (0,1)$,

$$R_k \coloneqq \eta_k F_{\ell(k)} + (1 - \eta_k)\|F_k\|, \qquad (6)$$

for which $\eta_k \in [\eta_{\min}, \eta_{\max}]$, $\eta_{\min} \in [0,1]$, $\eta_{\max} \in [\eta_{\min}, 1]$, and

$$F_{\{l(k)\}} = \max_{0 \leq j \leq m(k)} \{\|F_{k-j}\|\}, \quad k \in N_0 \coloneqq N \cup \{0\}, \qquad (7)$$

with $m(0) = 0$ and $0 \leq m(k) \leq \min\{m(k-1)+1, M\}$ in which $M \geq 0$. But this method does not use the information of $J_k$, which can be effective on numerical results.

In this paper, the algorithm described to solve nonlinear systems takes advantages of two-point gradient technique using low memory to produce a new adaptive radius strategy. The global convergence and q-quadratic convergence rate are established. The efficiency of the new method is confirmed to solve systems of nonlinear equations. This paper is organized as follows: Before describing the structure of new algorithm, we introduce a trust-region method for which BB method is used to produce the new adaptive radius in Section 2. In Section 3, the global convergence and the q-quadratic convergence rate of the new algorithm under some suitable assumptions are investigated. Numerical results are reported in Section 4. Finally, some conclusive remarks are given in Section 5.

## 2. Motivation and Algorithmic Structure

In order to obtain the smaller (bigger) radii near (far away from) the optimizer, we first introduce the Barzilai-Borwein (BB) technique and then by using its advantages a new adaptive radius strategy is produced.

It is well known that one of methods for which computational cost is trivial is the steepest descent method to solve Eq. (2) whose exact step-length is presented by

$$\theta_k := \operatorname*{argmin}_{\theta>0} f\,(x_k - \theta g_k),$$

where $g_k := J_k^T F_k$. The steepest descent direction $-g_k$ for which the speed convergence is slow cannot produce the smaller step-lengths near the optimizer. An efficient technique to overcome this drawback, using few storage locations and inexpensive computations is the BB method whose step-lengths $\theta_k$ are computed by

$$\theta_k^1 := \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}} \quad \text{and} \quad \theta_k^2 := \frac{y_{k-1}^T y_{k-1}}{s_{k-1}^T y_{k-1}}, \tag{8}$$

where $y_{k-1} := g_k - g_{k-1}$ and $s_{k-1} := x_k - x_{k-1}$, for more details see [3].

At each iteration, in order to take advantages of both $\theta_k^1$ and $\theta_k^2$, we introduce the new adaptive radius strategy by

$$\Delta_k := \begin{cases} c^{p_k} R_k, & \text{if } k = 0, \\ c^{p_k} \max\left\{\tilde{\theta}_k R_k, \Delta_{k-1}\right\}, & \text{if } k \geq 1, \end{cases} \tag{9}$$

in which $0 < c < 1$, $p_k$ is the smallest nonnegative integer $p$ ensuring the trust-region ratio is greater than a real-valued parameter $\mu \in (0,1)$, and

$$\tilde{\theta}_k := \begin{cases} \max\left\{\theta_{\min}, \min\left\{\max\left\{\theta_k^1, \theta_k^2\right\}, \theta_{\max}\right\}\right\}, & \text{if } \theta_k^1 > 0,\ \theta_k^2 > 0, \\ \lambda, & \text{else}, \end{cases} \tag{10}$$

where $0 < \theta_{\min} < \theta_{\max} < \infty$ and $0 < \lambda < \theta_{\max}$.

Based on BB strategy, the new method tries to produce the bigger (smaller) trust-region radius for cases where iterations are far away from (near) the optimizer by using $\tilde{\theta}_k$. This can truly control the radius of trust-region and hence it will decrease the total number of iterates.

Here, we add the new adaptive radius term into the trust-region algorithm as follows:

**Algorithm 1. BBATR** (Barzilai Borwein Adaptive Trust-Region Algorithm)

**Input:** An initial point $x_0 \in R^n, c, \mu \in (0,1), M > 0, \varepsilon > 0$ *and* $k_{\max}$.

**Output:** $x_b, f_b$;

```
begin
    Δ_0 := ‖F_0‖; F_ℓ(0) := ‖F_0‖; R_0 = F_ℓ(0); m(0) := 0; k := 0;
    while ‖F_k‖ ≥ ε & k ≤ k_max do
        p := 0; r_k := 0;
        while r_k < μ do
            specify the trial point d_k by solving the subproblem (3);
            compute F(x_k + d_k);
            f(x_k + d_k) := 1/2 ‖F(x_k + d_k)‖^2;
            determine r_k using (4);
            if r_k < μ then
                set p := p + 1 and determine Δ_k by (9);
            end
        end
        x_{k+1} := x_k + d_k; F_{k+1} := F(x_{k+1}); f_{k+1} := f(x_{k+1}); J_{k+1} := J(x_{k+1});
        let m(k+1) := min{m(k)+1, M};
        calculate F_ℓ(k+1) by (7) and R_k by (6);
        compute θ^1_{k+1} and θ^2_{k+1} by (8) and then θ̃_{k+1} obtain by (10);
        determine Δ_{k+1} by (9);
        k ← k + 1;
    end
    x_b := x_k; f_b := f_k;
end
```

To prove the global convergence of new method, we present the following assumptions:

(A1) The level set $L(x_0) := \{x \in R^n \mid f(x) \le f(x_0)\}$ is bounded for any given $x_0 \in R^n$ and $F(x)$ is continuously differentiable on compact convex set $\Omega$ containing the level set $L(x_0)$.

(A2) The matrix $\{J_k\}_{k\ge 0}$ is bounded and uniformly nonsingular on $\Omega$, i.e. there exist constants $0 \le M_0 \le 1 \le M_1$ such that

$$\|J_k\| \le M_1 \quad and \quad M_0\|F_k\| \le \|g_k\| \quad \forall\, k \in N_0 \tag{11}$$

(A3) The decrease on the model $m_k$ is at least as much as a fraction of the one obtained by the Cauchy point, i.e. there exists a constant $\beta \in (0,1)$ such that

$$m_k(x_k) - m_k(x_k + d_k) \ge \beta \|g_k\| \min\left[\Delta_k, \frac{\|g_k\|}{\|J_k^T J_k\|}\right], \tag{12}$$

for all $k \in N_0$, see [6,8].

(A4) The matrix $J(x)$ is Lipschitz continuous in $L(x_0)$, with Lipschitz constant $\gamma_L$, i.e.

$$\|J(x) - J(y)\| \le \gamma_L \|x - y\| \quad \forall\, x, y \in L(x_0).$$

**Lemma 1.** Assume that (A4) holds and the sequence $\{x_k\}_{k\geq 0}$ is generated by Algorithm 1. Let $d_k$ be a solution of the sub problem (3) such that $\|F(x_k)+J(x_k)d_k\| \leq \|F(x_k)\|$. Then, we have

$$|f(x_k+d_k)-m_k(x_k+d_k)| \leq O\left(\|d_k\|^2\right). \tag{13}$$

*Proof.* See [8].

The following lemma indicates that BBATR has infinitely many successful iterations.

**Lemma 2.** Suppose that (A2)-(A4) hold and the sequence $\{x_k\}_{k\geq 0}$ is generated by BBATR. Then, there exist infinitely many successful iterations in BBATR.

*Proof.* Assume that there are only finitely many successful iterations. Let us denote $x_{k_p}$ as the last successful iterate, and $\Delta_{k_p} > 0$ as the corresponding trust-region radius. In all subsequent iterations k, we have $x_k = x_{k_p}$, whereas the corresponding trust-region radius gets reduced and converges to zero; in particular, we therefore have $\Delta_k \leq \Delta_{k_p}$, so that all approximate solutions $d_k$ of the corresponding trust-region sub problem (3) satisfy $\|d_k\| \leq \Delta_{k_p}$ for all $k \geq k_p$.

Since $x_k$ is not the optimum point of Eq. (2), that there exists a constant $\varepsilon > 0$ such that $\|F_k\| \geq \varepsilon$. Then, for all $k \geq k_p$ sufficiently large, we have from (A2) and (A3)

$$\begin{aligned} m_k(x_k)-m_k(x_k+d_k) &\geq \beta\|g_k\|\min\left\{\Delta_{k_p}, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta M_0\|F_k\|\min\left\{\Delta_{k_p}, \frac{M_0\varepsilon}{M_1^2}\right\} \\ &\geq \beta M_0 \varepsilon \Delta_{k_p}. \end{aligned} \tag{14}$$

By Lemma 1 and Eq. (14), for all $k \in N_0$ sufficiently large, we get

$$\begin{aligned} \left|\frac{f_k - f(x_k+d_k)}{m_k(x_k)-m_k(x_k+d_k)} - 1\right| &= \left|\frac{f(x_k+d_k)-m_k(x_k+d_k)}{m_k(x_k)-m_k(x_k+d_k)}\right| \\ &\leq \frac{O\left(\|d_k\|^2\right)}{\beta M_0\varepsilon\Delta_{k_p}} \leq \frac{O\left(\left(\Delta_{k_p}\right)^2\right)}{\beta M_0\varepsilon\Delta_{k_p}} \to 0, \end{aligned}$$

yielding to

$$r_k = \frac{f_k - f(x_k+d_k)}{m_k(x_k)-m_k(x_k+d_k)} \geq \mu,$$

this means that there exists eventually an another successful iteration, a contradiction to our assumption.

## 3. Convergence Theory

We now investigate the global convergence and q-quadratic rate of results of the proposed algorithm. The following lemma helps us to establish the global convergence.

**Lemma 3.** Supposes that Assumptions (A2) and (A3) hold and the sequence $\{x_k\}_{k\geq 0}$ is generated by BBATR. Let $d_k$ be a solution of the sub problem (3). Then we have

$$m_k(x_k) - m_k(x_k + d_k) \geq L_k \|F_k\|^2, \tag{15}$$

where $L_k := \beta M_0 \min\left\{c^{p_k}\theta_{\min}, \frac{M_0}{M_1^2}\right\}$.

*Proof***.** Assumptions (A2) and (A3) and Eq. (10) along with Eq. (12) lead to

$$\begin{aligned} m_k(x_k) - m_k(x_k + d_k) &\geq \beta\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta\|g_k\| \min\left\{c^{p_k} \max\{\tilde{\theta}_k R_k, \Delta_{k-1}\}, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta\|g_k\| \min\left\{c^{p_k}\tilde{\theta}_k R_k, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta\|g_k\| \min\left\{c^{p_k}\theta_{\min} R_k, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta\|g_k\| \min\left\{c^{p_k}\theta_{\min} F_{l(k)}, \frac{\|g_k\|}{\|J_k^T J_k\|}\right\} \\ &\geq \beta M_0\|F_k\| \min\left\{c^{p_k}\theta_{\min}\|F_k\|, \frac{M_0\|F_k\|}{M_1^2}\right\} \\ &\geq \beta M_0\|F_k\|^2 \min\left\{c^{p_k}\theta_{\min}, \frac{M_0}{M_1^2}\right\} \\ &= L_k\|F_k\|^2 \end{aligned}$$

where $L_k := \beta M_0 \min\left\{c^{p_k}\theta_{\min}, \frac{M_0}{M_1^2}\right\}$. Therefore, the proof is complete.

The main global convergence result of BBATR can be established according to Assumptions (A1)-(A4) by the following theorem.

**Theorem 1.** Supposes that Assumptions (A1)-(A4) hold. Then BBATR either stops at a stationary point of $f(x)$ or generates an infinite sequence $\{x_k\}_{k\geq 0}$ such that

$$\lim_{k\to\infty} \|F_k\| = 0. \tag{16}$$

*Proof*. By contradiction, let $\varepsilon > 0$ be a constant and $K$ be an infinite subset of $N$ such that

$$\|F_k\| > \varepsilon, \quad \forall k \in K. \tag{17}$$

This fact along with Eq. (15) implies

$$f_k - f(x_k + d_k) \geq \mu[m_k(x_k) - m_k(x_k + d_k)] \geq \mu\|F_k\|^2 L_k \geq \mu\varepsilon^2 L_k.$$

By taking a limit of both sides in this expression, $k \to \infty$ we get $L_k \to 0$, leading $p_k \to \infty$. This fact clearly is a contradiction with Lemma 3. Therefore, the hypothesis (17) is not true and the proof is complete.

Note that the q-quadratic convergence rate of the sequence generated by BBATR, under some standard assumptions, can be established similar to Theorem 3.8 in [8].

## 4. Numerical Experiments

In this section, we compare our algorithm (BBATR) with the following four algorithms:

TTR: The traditional trust-region algorithm from Conn et al. in [6].

ATRZ: The adaptive trust-region algorithm from Zhang et al. in [23].

ATRF: The adaptive trust-region algorithm of Fan and Pan in [11].

ATRE: The adaptive trust-region algorithm of Esmaeili and Kimiaei in [8].

It is suggested that all algorithms be tested on a set of nonlinear systems of equations with the dimension from 2 to 1000. In addition, problems 1-28, problems 29-55, and problems 56-62 are chosen from [15], [13], and [16], respectively. Table 1 provides the name and dimension of each test problem. All codes are written in MATLAB 2015 programming environment with double precision format in the same subroutine. We utilize the test $\|F_k\| \leq 10^{-6}\sqrt{n}$ as the main termination criterion, in which $n$ is dimension. In other case, we count the corresponding test run as a failure if the total number of iterates exceeded 1000. The trust-region subproblems of the proposed algorithms are solved by Steihaug-Toint procedure (see [6]) which terminates at $x_k + d$ if $\|\nabla m_k(x_k + d)\| \leq \min\left\{\frac{1}{10}, \|\nabla m_k(x_k + d)\|^{\frac{1}{2}}\right\}\|\nabla m_k(x_k + d)\|.$

The Jacobian matrix $J_k$ evaluated with finite-differences formula, as follows:

$$[J_k]_{.j} \approx \frac{1}{h_j}\left(F(x_k + h_j e_j) - F_k\right),$$

where $[J_k]_{.j}$ denotes the *j*-th column of $J_k$, $e_j$ is the *j*-th vector of the canonic basis and

$$h_j := \begin{cases} \sqrt{\varepsilon_m} & \text{if } x_{k_j} = 0, \\ \sqrt{\varepsilon_m}\,\mathrm{sign}(x_k)\max\left\{\left|x_{k_j}\right|, \frac{\|x_k\|_1}{n}\right\} & \text{else,} \end{cases}$$

where $\varepsilon_m$ denotes the machine epsilon provided by the Matlab function eps.

**Table 1.** List of test functions.

| **Problem name** | **Dim** | **Problem name** | **Dim** |
|---|---|---|---|
| **Five-Diagonal System** | 5 | Chandrasekhar's h-equation | 500 |
| **Flow in a channel** | 10 | Singular | 500 |
| **Convection-diffusion** | 16 | Logarithmic | 500 |
| **Swirling flow** | 20 | Variable band 1 | 500 |
| **Extended powell badly scaled** | 100 | Variable band 2 | 500 |
| **Thorech** | 100 | Function 15 | 500 |
| **Tridiagnal system** | 100 | Strictly convex 1 | 500 |
| **Seven-Diagonal System** | 140 | Strictly convex 2 | 100 |
| **Tridiagnal exponential** | 200 | Function 18 | 90 |
| **Brent** | 200 | Zero Jacobian | 500 |
| **Bratu** | 200 | Geometric programming | 10 |
| **Poisson 2** | 200 | Function 21 | 501 |
| **Nonlinear biharmonic** | 200 | Linear function-full rank 1 | 500 |
| **Driven cavity** | 200 | Linear function-full rank 2 | 500 |
| **Countercurrent reactors 1** | 400 | Penalty | 10 |
| **Countercurrent reactors 2** | 400 | Brown almost linear | 500 |
| **Trigonometric** | 400 | Variable dimensioned | 500 |
| **Trigonometric exponential system 1** | 400 | Function 27 | 500 |
| **Singular Broyden** | 400 | Tridimensioal valley | 501 |
| **Structured Jacobian** | 400 | Complementary | 500 |
| **Extended Powell Singular** | 400 | Hanbook | 500 |
| **Extended Cragg and Levy** | 400 | Extended Freudentein and Roth | 500 |
| **Broyden tridiagonal** | 400 | Broyden banded | 200 |
| **Generalized Broyden banded** | 400 | Geometric | 20 |
| **Extended Wood** | 400 | Rosenbrock | 2 |
| **Discrete boundary value** | 400 | Powell singular | 4 |
| **Poisson** | 400 | Powell badley scaled | 2 |
| **Porous medium** | 400 | Helical valley | 3 |
| **Exponential 1** | 500 | Watson | 31 |
| **Exponential 2** | 1000 | Chebyquad | 2 |
| **Extended Rosenbrock** | 500 | Discrete integral equation | 100 |

The initial radius ($\Delta_0 = 1$) is chosen for all algorithms similar to [19]. ATRZ, ATRP and BBATR takes advantages of the parameters $\mu = 10^{-6}$, $c = 0.5$, $\theta_{\min} = 10^{-10}$ and $\theta_{\max} = 10^{10}$. On the other hand, TTR employs the parameters $\mu_1 = 0.1$, $\mu_2 = 0.9$, $c_1 = 0.25$ and $c_2 = 0.3$ and updates its radius like [6] by the following formula:

$$\Delta_{k+1} = \begin{cases} c_1 \|d_k\|, & \text{if } r_k < \mu_1, \\ \Delta_k, & \text{if } \mu_1 \le r_k \le \mu_2, \\ c_2 \Delta_k, & \text{if } r_k \ge \mu_2, \end{cases}$$

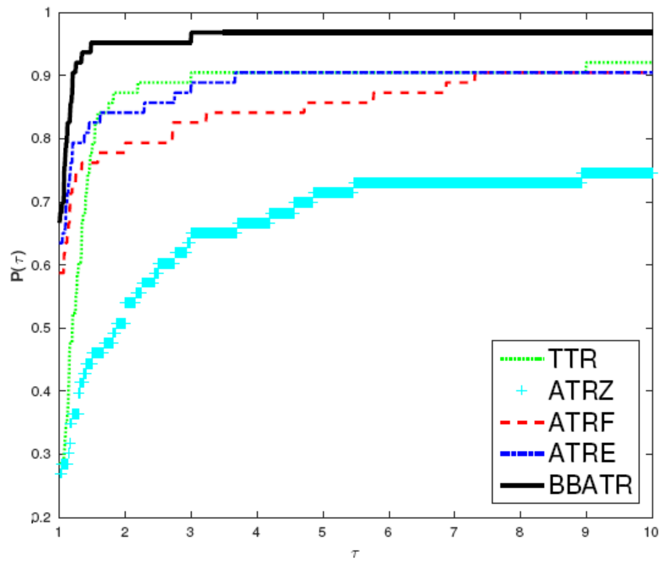


**Figure 1.** Iterates performance profile for the presented algorithms.

The performance profiles for all of the algorithms are given for the number of (successful) iterations $(N_i)$, number of function evaluations (which is equal to the number of total iterations plus one) $(N_f)$ and CPU-times $(C_t)$ by Figures 1-3, respectively. In the figures, $P$ designates the percentage of problems, which are solved within a factor $\tau$ of the best solver, cf. [7].

In Figure 1, it can be seen that BBATR is the best solver, in terms of number of iterations, on 68% of the problems while ATRE is the best in more or less 64% of the cases. Figure 2 shows that BBATR is the faster on approximately 54% of the test problems. Figure 3 shows that BBATR is significant about 25% in terms of CPU time. These results show that the proposed algorithm is an efficient and robust approach for solving systems of nonlinear equations.

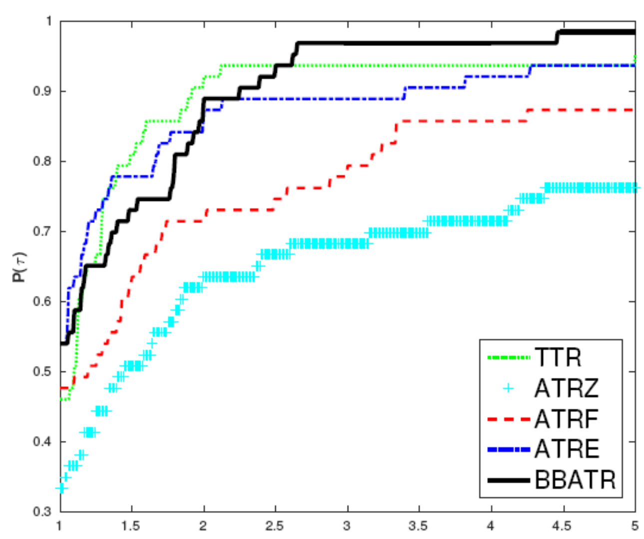


**Figure 2.** Function evaluations performance profile for the presented algorithms.

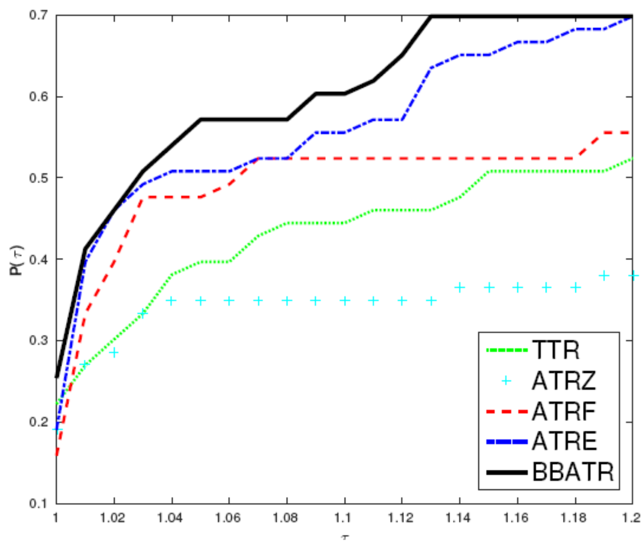


**Figure 3.** CPU time performance profile for the presented algorithm.

## 5. Concluding Remarks

This paper introduced a new adaptive trust-region strategy for system of nonlinear equations. Having taken Barzilai and Borwein strategy to produce an efficient algorithm, the updating rule for the trust-region radius leads to produce the smaller (bigger) of trust-region radius close to (far away from) the optimizer. The global and q-quadratic convergence rate properties of BBATR are established. Numerical results on a set of nonlinear systems indicate that BBATR is the best solver for solving nonlinear systems.

## 6. Funding

The second author acknowledges the financial support of the Doctoral Program 'Vienna Graduate School on Computational Optimization' funded by Austrian Science Foundation under Project No W1260-N35.

## References

[1] Ahookhosh, M., & Amini, K. (2010). A nonmonotone trust region method with adaptive radius for unconstrained optimization problems. *Computers & mathematics with applications*, *60*(3), 411-422.

[2] Ahookhosh, M., Amini, K., & Peyghami, M. R. (2012). A nonmonotone trust-region line search method for large-scale unconstrained optimization. *Applied mathematical modelling*, *36*(1), 478-487.

[3] Barzilai, J., & Borwein, J. M. (1988). Two-point step size gradient methods. *IMA journal of numerical analysis*, *8*(1), 141-148.

[4] Bouaricha, A., & Schnabel, R. B. (1998). Tensor methods for large sparse systems of nonlinear equations. *Mathematical programming*, *82*(3), 377-400.

[5] Broyden, C. G. (1971). The convergence of an algorithm for solving sparse nonlinear systems. *Mathematics of computation*, *25*(114), 285-294.

[6] Conn, A. R., Gould, N. I., & Toint, P. L. (2000). *Trust region methods*. Society for Industrial and Applied Mathematics.

[7] Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming*, *91*(2), 201-213.

[8] Esmaeili, H., & Kimiaei, M. (2014). A new adaptive trust-region method for system of nonlinear equations. *Applied mathematical modelling*, *38*(11), 3003-3015.

[9] Fan, J. (2006). Convergence rate of the trust region method for nonlinear equations under local error bound condition. *Computational optimization and applications*, *34*(2), 215-227.

[10] Fan, J., & Pan, J. (2011). An improved trust region algorithm for nonlinear equations. *Computational optimization and applications*, *48*(1), 59-70.

[11] Fan, J., & Pan, J. (2010). A modified trust region algorithm for nonlinear equations with new updating rule of trust region radius. *International journal of computer mathematics*, *87*(14), 3186-3195.

[12] Grippo, L., & Sciandrone, M. (2007). Nonmonotone derivative-free methods for nonlinear equations. *Computational optimization and applications*, *37*(3), 297-328.

[13] La Cruz, W., Martínez, J., & Raydan, M. (2006). Spectral residual method without gradient information for solving large-scale nonlinear systems of equations. *Mathematics of computation*, *75*(255), 1429-1448.

[14] Li, D. H., & Fukushima, M. (2000). A derivative-free line search and global convergence of Broyden-like method for nonlinear equations. *Optimization methods and software*, *13*(3), 181-201.

[15] Lukšan, L., & Vlček, J. (1999). Sparse and partially separable test problems for unconstrained and equality constrained optimization. *Technical report*, *767.*

[16] Moré, J. J., Garbow, B. S., & Hillstrom, K. E. (1981). Testing unconstrained optimization software. *ACM transactions on mathematical software (TOMS)*, *7*(1), 17-41.

[17] Nesterov, Y. (2007). Modified Gauss–Newton scheme with worst case guarantees for global performance. *Optimisation methods and software*, *22*(3), 469-483.

[18] Bonnans, J. F., Gilbert, J. C., Lemaréchal, C., & Sagastizábal, C. A. (2006). *Numerical optimization: Theoretical and practical aspects*. Springer Science & Business Media.

[19] Toint, P. L. (1986). Numerical solution of large sets of algebraic nonlinear equations. *Mathematics of computation*, *46*(173), 175-189.

[20] Yuan, G., Lu, S., & Wei, Z. (2011). A new trust-region method with line search for solving symmetric nonlinear equations. *International journal of computer mathematics*, *88*(10), 2109-2123.

[21] Yuan, G., Wei, Z., & Lu, X. (2011). A BFGS trust-region method for nonlinear equations. *Computing*, *92*(4), 317-333.

[22] Yuan, Y. X. (1994). *Trust region algorithms for nonlinear equations*. Hong Kong Baptist University, Department of Mathematics.

[23] Zhang, J. L., & Wang, Y. (2003). A new trust region method for nonlinear equations. *Mathematical methods of operations research*, *58*(2), 283-298.