CompAMa

Bull. Comput. Appl. Math., Vol.4, No.2, 2016

www.compama.co.usb.ve

ISSN 2244-8659

Derivative-free method for bound constrained nonlinear monotone equations and its application in solving steady state reaction-diffusion problems

Octavio Batta [*], *William La Cruz* [†], *Gilberto Noguera* [‡]

CompAMa Vol.4, No.2, pp.71-93, 2016 - Accepted Octuber 9, 2016

In Honor of the 60th Birthday of Professor Marcos Raydan

Abstract

We present a derivative-free algorithm for solving bound constrained systems of nonlinear monotone equations. The algorithm generates feasible iterates using in a systematic way the residual as search direction and a suitable step-length closely related to the Barzilai-Borwein choice. A convergence analysis is described. We also present one application in solving problems related with the study of reaction-diffusion processes that can be described by nonlinear partial differential equations of elliptic type. Numerical experiences are included to highlight the efficacy of proposed algorithm.

Keywords: Nonlinear equations, Derivative-free algorithm, Monotone mapping, Reaction-diffusion problems.

[*] Área de Matemática, Facultad de Arquitectura y Urbanismo, Universidad Central de Venezuela, Venezuela (octavio.batta@ucv.ve).

[†] Departamento de Electrónica, Computación y Control, Facultad de Ingeniería, Universidad Central de Venezuela, Caracas 1053, Venezuela (william.lacruz@ucv.ve).

[‡] Área de Matemática, Universidad Nacional Abierta, Caracas, Venezuela (gilberto.noguero@ucv.ve).

1 Introduction

The problem we face is a system of nonlinear monotone equations with bound constrained whose mathematical formulation reads as follows:

$$\text{Find } \mathbf{x} \in \Omega \text{ such that } \; F(\mathbf{x}) = 0, \tag{1}$$

where $F : \mathcal{O} \to \mathbb{R}^n$ is monotone and Lipschitz continuous on an open set $\mathcal{O} \subset \mathbb{R}^n$ containing to nonempty n-dimensional box

$$\Omega = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\}.$$

The vectors $\mathbf{l} = (l_1, \ldots, l_n)^T \in (\mathbb{R} \cup -\infty)^n$ and $\mathbf{u} = (u_1, \ldots, u_n)^T \in (\mathbb{R} \cup \infty)^n$ are respectively the lower and upper bounds of the variables so that Ω has nonempty interior. We are interested in the large-scale case for which the Jacobian matrix of F is either not available or requires a prohibitive amount of storage.

Throughout the paper, the space $\mathbb{R}^n$ is equipped with the Euclidean norm $\|\cdot\|$ and the inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T\mathbf{y}$, for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. A mapping F is called *monotone* on $\mathcal{O} \subset \mathbb{R}^n$ if, for all $\mathbf{x}, \mathbf{y} \in \mathcal{O}$, $\langle F(\mathbf{x}) - F(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq 0$, and *strictly monotone* on $\mathcal{O}$ if this inequality is strict whenever $\mathbf{x} \neq \mathbf{y}$. Likewise, F is called *Lipschitz continuous* on $\mathcal{O} \subset \mathbb{R}^n$, if there exists a constant $L > 0$ such that, for all $\mathbf{x}, \mathbf{y} \in \mathcal{O}$, $\|F(\mathbf{x}) - F(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$.

The systems of nonlinear monotone equations arise in various applications, such as differential equations, as well as economics, engineering, management science, probability theory and other applied sciences. The study of iterative methods for monotone equations have been developed in the last few years [1–8].

We introduce and analyze a residual method for solving (1), which is an extension of the methods SANE and DF-SANE for nonlinear systems of equations [9,10]. These methods systematically used the residual $\mathbf{d} = \mp F(\mathbf{x})$ as search direction, combined with a suitable step-length and a nonmonotone line search globalization strategy. In particular, the proposed algorithm uses a variation of the line search used by DF-SANE. The main objective of this variation is to guarantee the convergence without require of the differentiability of F. The proposed method generated feasible points using the projection onto a sphere, which is contained in the interior of the box Ω. The projected point never is explicitly built, but rather allows to choose the initial step-length at each iteration.

Here we also show the application of our method in solving problems related with the study of reaction-diffusion processes that can be described by nonlinear partial differential equations of elliptic type, which appear in many problems of practical interest (see, e.g., [11–13]). For this particular type of problems, we prove that the proposed method converges to the solution thereof.

The remainder of this paper is organized as follows. In Section 2 our method is presented and its convergence is established. Section 3 shows the application of proposed method in solving of the system of quasilinear equations obtained by the discretization of the nonlinear partial differential equations of elliptic type, that describe the reaction-diffusion processes. In Section 4 the numerical results for a set of test problems is described, and in Section 5 are given some final comments.

Finally, we explain some terminology and fix the notation used throughout the paper. The ℓ_1 norm of $\mathbf{v} \in \mathbb{R}^n$ is defined by $\|\mathbf{v}\|_1 = \sum_i |v_i|$. The open ball of radius $\delta > 0$ about a point $\mathbf{x} \in \mathbb{R}^n$ will be denoted $B(\mathbf{x}, \delta) = \{\mathbf{z} \in \mathbb{R}^n : \|\mathbf{z} - \mathbf{x}\| < \delta\}$ and $\overline{B}(\mathbf{x}, \delta) = \{\mathbf{z} \in \mathbb{R}^n : \|\mathbf{z} - \mathbf{x}\| \leq \delta\}$ is the closed ball. For a given box $\Omega = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\}$, we denoted with $\mathrm{int}(\Omega)$ the strict interior of Ω.

2 Description of the method and its convergence

The goal of this section is to describe the derivative-free residual algorithm for solving (1), namely, `rabc_monotone` (*Residual Algorithm to Bound Constrained Nonlinear Monotone Equations*).

Given a feasible point $\mathbf{x}_k = (x_1^k, \ldots, x_n^k)^T \in \Omega$, `rabc_monotone` chooses the search direction $\mathbf{d}_k = -\alpha_k F(\mathbf{x}_k)$, and generates the iterate $\mathbf{x}_{k+1}$ within Ω by the formula $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k$, for $k \geq 0$, where the scalar α_k is the *spectral coefficient*.

In order to guarantee of global convergence, `rabc_monotone` compute the step-length $\lambda_k = \lambda \in (0, 1]$ by using a finite backtracking process so that the following inequality is satisfies:

$$\|F(\mathbf{x}_k + \lambda \mathbf{d}_k)\|^2 \leq \|F(\mathbf{x}_k)\|^2 + \eta_k - \gamma \lambda^2 \|F(\mathbf{x}_k)\|^2, \tag{2}$$

where η_k and γ are parameters to be define.

Now, to ensure that the iterate $\mathbf{x}_{k+1}$ is a feasible point (see Proposition 3), the initial value of the step-length λ, that is used in the bactracking process, is computed as follows. If $\mathbf{x}_k + \mathbf{d}_k$ is a feasible point, then $\lambda = 1$. But, if $\mathbf{x}_k + \mathbf{d}_k$ is an infeasible point, then $\lambda = r_k/\|\mathbf{d}_k\| < 1$, where

$$r_k = \upsilon \min \left\{ \min_i \{|u_i - x_i^k|\}, \min_i \{|x_i^k - l_i|\} \right\}$$

with $\upsilon \in (0, 1)$. It is clear that the closed ball $\overline{B}(\mathbf{x}_k, r_k)$ belong to Ω and also the point $\mathbf{x}_k + (r_k/\|\mathbf{d}_k\|)\mathbf{d}_k$ is the projection of $\mathbf{x}_k + \mathbf{d}_k$ on $\overline{B}(\mathbf{x}_k, r_k)$. In Fig. 1 is shown, with a graphic example, the position of the point $\mathbf{x}_k + \mathbf{d}_k$ with respect to the ball $\overline{B}(\mathbf{x}_k, r_k)$. In this figure the point $\mathbf{x}_k + (r_k/\|\mathbf{d}_k\|)\mathbf{d}_k$ is also shown when $\mathbf{x}_k + \mathbf{d}_k$ is an infeasible point.

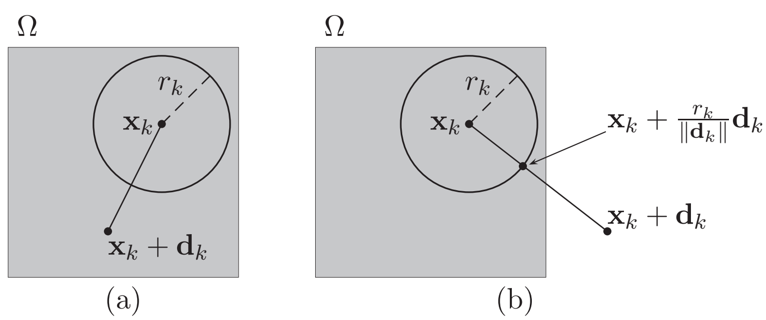

Fig. 1: Position of the point $\mathbf{x}_k + \mathbf{d}_k$ with respect to the ball $\overline{B}(\mathbf{x}_k, r_k)$: (a) $\mathbf{x}_k + \mathbf{d}_k$ is a feasible point; (b) $\mathbf{x}_k + \mathbf{d}_k$ is an infeasible point.

On the other hand, the use of the inequality (2) is called *linear search* (see [14] for details), which is similar to the proposition presented in [10] with some variations and it will allow us to establish the convergence of the algorithm without to assume that F is differentiable. As in [10], the linear search also requires some given parameters: $\gamma, \sigma \in (0, 1)$, and a fixed positive sequence $\{\eta_k\}$ such that

$$\sum_{k=0}^{\infty} \eta_k \leq \eta < \infty. \tag{3}$$

The detailed description of `rabc_monotone` is shown formally in Algorithm 2.

Algorithm 2: (`rabc_monotone` method)

Require: $\mathbf{x}_0 \in \text{int}(\Omega)$, $\gamma, \sigma, \upsilon \in (0,1)$, $0 < \alpha_{max} < \infty$, $0 < \alpha_0 < \alpha_{max}$, and a positive sequence $\{\eta_k\}$ such that $\sum_{k=0}^{\infty} \eta_k < \infty$;

$k \leftarrow 0$;

while $\|F(\mathbf{x}_k)\| > 0$ **do**

  Define $\mathbf{d}_k = -\alpha_k F(\mathbf{x}_k)$;

  if $\mathbf{x}_k + \mathbf{d}_k \notin \text{int}(\Omega)$ **then**

    $r_k = \upsilon \min\left\{\min_i\{|u_i - x_i^k|\}, \min_i\{|x_i^k - l_i|\}\right\}$;

    Set $\lambda \leftarrow r_k / \|\mathbf{d}_k\|$;

  else

    Set $\lambda \leftarrow 1$;

  end if

  while $\|F(\mathbf{x}_k + \lambda \mathbf{d}_k)\|^2 > \|F(\mathbf{x}_k)\|^2 + \eta_k - \gamma\lambda^2\|F(\mathbf{x}_k)\|^2$ **do**

    Set $\lambda \leftarrow \sigma\lambda$; `// backtracking process`

  end while

  Define $\lambda_k = \lambda$;

  Define $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k$, $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$, and $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$;

  Define $\alpha_{k+1} = \min\{\alpha_{max}, \langle \mathbf{s}_k, \mathbf{s}_k\rangle / \langle \mathbf{s}_k, \mathbf{y}_k\rangle\}$;

  Set $k \leftarrow k+1$;

end while

Some of the remarks that can be drawn from Algorithm 2 are as follows:

Remark 1. *(i) The* `rabc_monotone` *algorithm is well defined. Indeed, by the continuity of F and $\eta_k > 0$, the inequality of the liner search is satisfied after a finite number of reductions of λ.*

(ii) The sequence of residuals $\{F(\mathbf{x}_k)\}$ generated by Algorithm 2 satisfies:

$$\|F(\mathbf{x}_{k+1})\|^2 \leq \|F(\mathbf{x}_k)\|^2 + \eta_k - \gamma\lambda_k^2\|F(\mathbf{x}_k)\|^2, \quad \textit{for all } k \geq 0. \tag{4}$$

(iii) In general, taking $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$, and $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$, the scalar α_{k+1} is compute by the formula

$$\alpha_{k+1} = \frac{\langle \mathbf{s}_k, \mathbf{s}_k\rangle}{\langle \mathbf{s}_k, \mathbf{y}_k\rangle},$$

which is closely related to the Barzilai-Borwein choice of steplength [15]. When $\langle \mathbf{s}_k, \mathbf{s}_k\rangle / \langle \mathbf{s}_k, \mathbf{y}_k\rangle$ is not within the interval $(0, \alpha_{min}]$, α_{k+1} is the *projection of* $\langle \mathbf{s}_k, \mathbf{s}_k\rangle / \langle \mathbf{s}_k, \mathbf{y}_k\rangle$ *on* $(0, \alpha_{max}]$. *Therefore, the scalar* α_k *is chosen so that*

$$\alpha_k \in (0, \alpha_{max}], \quad \textit{for all } k \geq 0. \tag{5}$$

We are going to present some convergence properties of Algorithm 2. Proposition 2 states that the sequence $\{\mathbf{x}_k\}$ generated by Algorithm 2 is contained in a closed set.

Proposition 2. *Assume that* F *is continuous on* $\mathbb{R}^n$. *The sequence* $\{\mathbf{x}_k\}$ *generated by Algorithm 2 is contained in the closed set*

$$\Psi_0 = \left\{ \mathbf{x} \in \mathbb{R}^n : 0 \leq \|F(\mathbf{x})\|^2 \leq \|F(\mathbf{x}_0)\|^2 + \eta \right\}.$$

Proof. Using (4) and an inductive process we get:

$$\|F(\mathbf{x}_{j+k})\|^2 \leq \|F(\mathbf{x}_j)\|^2 + \sum_{i=j}^{j+k-1} \eta_i, \quad \text{for } j \geq 0 \text{ and } k \geq 1. \tag{6}$$

By (3) and (6), $\|F(\mathbf{x}_k)\|^2 \leq \|F(\mathbf{x}_0)\|^2 + \eta$, for $k \geq 1$; that is, the sequence $\{\mathbf{x}_k\}$ is contained in the set Ψ_0. $\square$

Proposition 3 establishes that the sequence $\{\mathbf{x}_k\}$ generated by Algorithm 2 is contained in $\mathrm{int}(\Omega)$, that is to say, all iterates $\mathbf{x}_k$ are feasible points.

Proposition 3. *Algorithm 2 generates the sequences* $\{\mathbf{x}_k\}$ *and* $\{\lambda_k\}$ *such that the sequence* $\{\mathbf{x}_k\}$ *is contained in* $\mathrm{int}(\Omega)$ *and the sequence* $\{\lambda_k\}$ *is uniformly bounded in the interval* $(0, 1]$. *Particularly, the sequence* $\{\mathbf{x}_k\}$ *is contained in* $\Psi_0 \cap \mathrm{int}(\Omega)$.

Proof. By definition of Algorithm 2 we have that $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k$ and $\upsilon \in (0, 1)$. Thus, if $\mathbf{x}_k + \mathbf{d}_k \notin \mathrm{int}(\Omega)$, then

$$r_k = \upsilon \min \left\{ \min_i \{|u_i - x_i^k|\}, \min_i \{|x_i^k - l_i|\} \right\} \leq \|\mathbf{d}_k\|.$$

In this case, since the point $\mathbf{x}_k + (r_k/\|\mathbf{d}_k\|)\mathbf{d}_k$ is the projection of $\mathbf{x}_k + \mathbf{d}_k$ on the closed ball $\overline{B}(\mathbf{x}_k, r_k) \subset \mathrm{int}(\Omega)$ and $\lambda_k \leq r_k/\|\mathbf{d}_k\| \leq 1$, then $\mathbf{x}_{k+1} \in \mathrm{int}(\Omega)$. Now, when $\mathbf{x}_k + \mathbf{d}_k \in \mathrm{int}(\Omega)$, we obtain that $\lambda_k \leq 1$ and $\mathbf{x}_{k+1} \in \mathrm{int}(\Omega)$. Therefore, the iterates $\mathbf{x}_k$ are feasible points and the step length λ_k is uniformly bounded in the interval $(0, 1]$, for all $k \geq 0$. Finally, as $\{\mathbf{x}_k\} \subset \mathrm{int}(\Omega)$ then by Proposition 2 $\{\mathbf{x}_k\} \subset \Psi_0 \cap \mathrm{int}(\Omega)$. $\square$

Proposition 4 correspond to Lemma 3.3 from [16]. We have included here for the sake of completeness.

Proposition 4. *Let $\{\mu_k\}$ and $\{\xi_k\}$ positive sequences such that $\mu_{k+1} \leq (1+\xi_k)\mu_k + \xi_k$ and $\sum_{k=0}^{\infty} \xi_k < \infty$. Then the sequence $\{\mu_k\}$ converges.*

To continue with the convergence analysis, we need make the following assumption.

Assumption 1.

i) The mapping $F : \mathbb{R}^n \to \mathbb{R}^n$ is monotone and Lipschitz continuous on an open, convex set $\mathcal{O} \subset \mathbb{R}^n$ containing to Ω; in addition, F is strictly monotone on $\mathrm{int}(\Omega)$.

ii) The solution set $S \subset \mathbb{R}^n$ of (1) is not empty and $S \cap \mathrm{int}(\Omega) \neq \emptyset$.

Proposition 5. *Assumption 1 holds. Let $\{\mathbf{x}_k\}$ be the sequence generated by Algorithm 2. Then, the following statements hold.*

1) The sequence $\{\|F(\mathbf{x}_k)\|\}$ converges.

2) There exists a constant $\alpha_{min} > 0$ such that

$$\alpha_k \geq \alpha_{min}, \quad \textit{for } k \geq 0. \tag{7}$$

3) The sequence $\{\lambda_k \|F(\mathbf{x}_k)\|\}$ converges and

$$\lim_{k\to\infty} \lambda_k \|F(\mathbf{x}_k)\| = 0. \tag{8}$$

Proof. We first prove 1). Set $\mu_k = \|F(\mathbf{x}_k)\|^2$ and $\xi_k = \eta_k$. Since $\|F(\mathbf{x}_k)\|^2 \geq 0$ and $(1+\eta_k) \geq 1$, by (4) we can write

$$\mu_{k+1} \leq \mu_k + \xi_k \leq (1+\xi_k)\mu_k + \xi_k.$$

Now, by Proposition 4 the sequence $\{\|F(\mathbf{x}_k)\|^2\}$ converges; thus, the sequence $\{\|F(\mathbf{x}_k)\|\}$ converges.

Now we prove 2). As $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$, $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$, and F is strictly monotone, we have that

$$\frac{\langle \mathbf{s}_k, \mathbf{y}_k \rangle}{\langle \mathbf{s}_k, \mathbf{s}_k \rangle} = \frac{\langle \mathbf{x}_{k+1} - \mathbf{x}_k, F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k) \rangle}{\|\mathbf{s}_k\|^2} > 0.$$

Since $\alpha_{k+1} = \langle \mathbf{s}_k, \mathbf{s}_k \rangle / \langle \mathbf{s}_k, \mathbf{y}_k \rangle$ or $\alpha_{k+1} = \alpha_{max}$, then $\alpha_k > 0$, for $k \geq 0$. Define $\alpha_{min} = \min\{L^{-1}, \alpha_0\}$, where L is the Lipschitz constant for F. Using the Cauchy-Schwartz inequality and the Lipschitz condition, we can write

$$\frac{\langle \mathbf{s}_k, \mathbf{y}_k \rangle}{\langle \mathbf{s}_k, \mathbf{s}_k \rangle} \leq \frac{\|\mathbf{s}_k\| \, \|F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)\|}{\|\mathbf{s}_k\|^2} \leq L.$$

This inequality combined with the definition of α_k implies that (7) holds.

Finally we prove 3). By (4) and (7) we have, for $k \geq 0$,

$$\lambda_k^2 \|F(\mathbf{x}_k)\|^2 \leq \frac{\eta_k}{\gamma} + \frac{\|F(\mathbf{x}_k)\|^2 - \|F(\mathbf{x}_{k+1})\|^2}{\gamma}. \tag{9}$$

Since η_k satisfies (3), adding all terms in both sides of (9) it follows that

$$\begin{aligned} \sum_{k=0}^{\infty} \lambda_k^2 \|F(\mathbf{x}_k)\|^2 &\leq \frac{\left(\sum_{k=0}^{\infty} \eta_k + \sum_{k=0}^{\infty} \left(\|F(\mathbf{x}_k)\|^2 - \|F(\mathbf{x}_{k+1})\|^2 \right) \right)}{\gamma \alpha_{min}^2} \\ &\leq \frac{\eta + \|F(\mathbf{x}_0)\|^2}{\gamma} < \infty. \end{aligned}$$

As $\lambda_k \|F(\mathbf{x}_k)\| \geq 0$, the about inequality implies that (8) holds. □

Proposition 6. *Assumption 1 holds. For* $\mathbf{x} \in \mathcal{O}$*, define the limit*

$$\Lambda_{\mathbf{x}} = \lim_{t \to 0^+} \frac{\|F(\mathbf{x} - tF(\mathbf{x}))\|^2 - \|F(\mathbf{x})\|^2}{t}. \tag{10}$$

Then, $-\infty < \Lambda_{\mathbf{x}} < 0$ *for every* $\mathbf{x} \in \mathcal{O}$ *whenever* $F(\mathbf{x}) \neq 0$.

Proof. Let us prove that $|\Lambda_{\mathbf{x}}| < \infty$, for each $\mathbf{x} \in \mathcal{O}$. Let $M > 0$ be such that $\|F(\mathbf{z})\| \leq M$ for all $\mathbf{z} \in \mathcal{O}$. Using the Cauchy-Schwarz inequality and the Lipschitz condition, we can write

$$\begin{aligned} &\left| \|F(\mathbf{x} - tF(\mathbf{x}))\|^2 - \|F(\mathbf{x})\|^2 \right| \\ &\qquad = |\langle F(\mathbf{x} - tF(\mathbf{x})) + F(\mathbf{x}) \, , \, F(\mathbf{x} - tF(\mathbf{x})) - F(\mathbf{x}) \rangle| \\ &\qquad \leq \|F(\mathbf{x} - tF(\mathbf{x})) + F(\mathbf{x})\| \, \|F(\mathbf{x} - tF(\mathbf{x})) - F(\mathbf{x})\| \\ &\qquad\qquad \leq \left(\|F(\mathbf{x} - tF(\mathbf{x}))\| + \|F(\mathbf{x})\| \right) tL \|F(\mathbf{x})\|. \end{aligned}$$

Now, by the continuity of F there is $T > 0$ such that $\|F(\mathbf{x} - tF(\mathbf{x}))\| \leq M$, for $t \in (0, T]$. So, we obtain

$$\frac{|\|F(\mathbf{x} - tF(\mathbf{x}))\|^2 - \|F(\mathbf{x})\|^2|}{t} \leq 2M^2L, \quad \text{for all } t \in (0, T].$$

This implies that $|\Lambda_{\mathbf{x}}| < \infty$, for each $x \in \mathcal{O}$. Now, using the Lipschitz condition we have, for $\mathbf{x} \in \mathcal{O}$ and $t > 0$ sufficiently small,

$$\begin{aligned} L^2t^2\|F(\mathbf{x})\|^2 &\geq \|F(\mathbf{x} - tF(\mathbf{x})) - F(\mathbf{x})\|^2 \\ &= \|F(\mathbf{x} - tF(\mathbf{x}))\|^2 - 2\langle F(\mathbf{x} - tF(\mathbf{x})), F(\mathbf{x})\rangle + \|F(\mathbf{x})\|^2. \end{aligned} \tag{11}$$

As F is strictly monotone, we can write, for $t > 0$ and $\mathbf{x} \in \mathcal{O}$ whenever $F(\mathbf{x}) \neq 0$,

$$\langle F(\mathbf{x} - tF(\mathbf{x})) - F(\mathbf{x}), -tF(\mathbf{x})\rangle = -t\left(\langle F(\mathbf{x} - tF(\mathbf{x})), F(\mathbf{x})\rangle - \|F(\mathbf{x})\|^2\right) > 0,$$

so,

$$\langle F(\mathbf{x} - tF(\mathbf{x})), F(\mathbf{x})\rangle < \|F(\mathbf{x})\|^2. \tag{12}$$

By (11) and (12) we get

$$L^2t\|F(\mathbf{x})\|^2 > \frac{\|F(\mathbf{x} - tF(\mathbf{x}))\|^2 - \|F(\mathbf{x})\|^2}{t}.$$

This implies that $-\infty < \Lambda_{\mathbf{x}} < 0$, for every $\mathbf{x} \in \mathcal{O}$ whenever $F(\mathbf{x}) \neq 0$. □

Theorem 1 is our convergence theorem. This theorem says that if F is a bounded strictly monotone mapping on the box Ω that satisfies the Lipschitz condition, then all the limits points of the sequence $\{\mathbf{x}_k\}$ generated by Algorithm 2 are solutions of the system of nonlinear equations (1).

Theorem 1. *Assumption 1 holds. Let $\{\mathbf{x}_k\}$ be the sequence generated by Algorithm 2. Then,* $\lim_{k\to\infty} \|F(\mathbf{x}_k)\| = 0$.

Proof. Let $\mathbf{x}_*$ be a limit point of $\{\mathbf{x}_k\}$. Without loss of generality we can assume that the sequence $\{\mathbf{x}_k\}$ converges to $\mathbf{x}_*$. Since (8) holds, then $\liminf_{k\to\infty} \lambda_k = 0$ or $F(\mathbf{x}_*) = 0$. If $F(\mathbf{x}_*) = 0$, the conclusion of the theorem is immediate. Next we show that if $\liminf_{k\to\infty} \lambda_k = 0$, then $F(\mathbf{x}_*) = 0$. Without loss of generality we can assume that $\lim_{k\to\infty} \lambda_k = 0$. Now, by the line search rule, if $\lambda_k \neq 1$, then $\widehat{\lambda}_k = \sigma^{-1}\lambda_k$ does not satisfy the line search rule. This means

$$\begin{aligned} \|F(\mathbf{x}_k - \widehat{\lambda}_k\alpha_k F(\mathbf{x}_k))\|^2 &> \|F(\mathbf{x}_k)\|^2 + \eta_k - \gamma\widehat{\lambda}_k^2\|F(\mathbf{x}_k)\|^2 \\ &> \|F(\mathbf{x}_k)\|^2 - \gamma\widehat{\lambda}_k^2\|F(\mathbf{x}_k)\|^2. \end{aligned}$$

As $\alpha_{min} < \alpha_k \leq \alpha_{max}$ and $\|F(\mathbf{x}_k)\|^2 \leq \|F(\mathbf{x}_0)\|^2 + \eta$, for $k \geq 0$, then using the about inequality we get

$$-\left(\frac{\|F(\mathbf{x}_k - \widehat{\lambda}_k\alpha_k F(\mathbf{x}_k))\|^2 - \|F(\mathbf{x}_k)\|^2}{\widehat{\lambda}_k\alpha_k}\right) < \lambda_k\, c, \tag{13}$$

where $c = \sigma^{-1}\gamma\alpha_{min}^{-1}(\|F(\mathbf{x}_0)\|^2 + \eta) > 0$. Taking limits on both sides of (13) and using Proposition 6, we obtain

$$0 = \lim_{k\to\infty} \lambda_k \geq -c^{-1}\,\Lambda_{\mathbf{x}_*} > 0 \qquad \Rightarrow \qquad \Lambda_{\mathbf{x}_*} = 0.$$

But $\Lambda_{\mathbf{x}_*} = 0$ if, and only if $F(\mathbf{x}_*) = 0$. Therefore, by the continuity of F, we have that $\lim_{k\to\infty} \|F(\mathbf{x}_k)\| = 0$. □

3 Application in solving steady state reaction-diffusion problems

This section presents the application of proposed method in the numerical solution of the system of nonlinear equations which is obtained in the discretization of steady state reaction-diffusion problems, using the finite difference method. In general, this system has the form

$$F(\mathbf{v}) \equiv A(\mathbf{v})\mathbf{v} + G(\mathbf{v}) - s = 0, \tag{14}$$

where, for each $\mathbf{v} \in \mathbb{R}^n$, $A(\mathbf{v})$ is a real matrix of order n, $G : \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable mapping, and $s \in \mathbb{R}^n$ is a constant vector.

The steady state reaction-diffusion problems are defined as follows. In the rectangular domain $R \subset \mathbb{R}^2$ with boundary ∂R, we consider the following reaction-diffusion equation subject to the Dirichlet boundary conditions:

$$\begin{cases} \operatorname{div}(\delta(x,y,\varphi)\nabla\varphi(x,y)) + g(x,y,\varphi) = s(x,y), \quad \text{for } (x,y) \in R, \\ \varphi(x,y) = 0, \quad \text{for } (x,y) \in \partial R, \end{cases} \tag{15}$$

where the functions $\delta(x,y,\varphi)$, $g(x,y,\varphi)$ and $s(x,y)$ are assumed to satisfy the following *Smoothness Conditions* for $(x,y) \in R \cup \partial R$, and φ in a neighbourhood of a solution of (15). We assume that this model problem has a solution.

Smoothness Conditions

(I) The functions $\delta(x,y,\varphi)$ and $g(x,y,\varphi)$ are piecewise continuous in x and y, and continuous in φ; the source term $s(x,y)$ is piecewise continuous in x, y.

(II) There exist two positive constants δ_{min} and δ_{max} such that $0 < \delta_{min} \leq \delta(x,y,\varphi) \leq \delta_{max}$.

(III) For fixed x and y, the function $\delta(x,y,\varphi)$ is locally Lipschitz continuous at φ (uniformly in x and y), with constant $\Lambda > 0$.

(IV) For fixed x and y, the function $g(x,y,\varphi)$ is strongly monotone at φ (strictly in x and y), with constant $C > 0$, that is $|g(x,y,\varphi) - g(x,y,\omega)| > C|\varphi - \omega|^2$. Also, $g(x,y,\varphi)$ is continuously differentiable at φ.

Under the above conditions the system (14) has special properties. Assume that R is the unit square. Let N grid points (x_i, y_i) in R, where $x_i = ih$ and $y_i = jh$ with $i = 1,\ldots,N$, $j = 1,\ldots,N$, and $h = 1/(N+1)$. At each grid point, the value of $\varphi(x_i, y_j)$ is approximated by φ_{ij}, that is $\varphi(x_i, y_j) \approx \varphi_{ij}$. Now, using the five point discretization formulas and the natural ordering by horizontal lines of the mesh points, the equation (15) is transformed into a system of nonlinear n algebraic equations (14) (see [17] for details), where $n = N^2$ and the vector $\mathbf{v} = (v_1,\ldots,v_n)^T \in \mathbb{R}^n$ is given by

$$\mathbf{v} = (\varphi_{11}, \varphi_{21}, \ldots, \varphi_{N1}, \varphi_{12}, \varphi_{22}, \ldots, \varphi_{N2}, \ldots, \varphi_{1N}, \varphi_{2N}, \ldots, \varphi_{NN})^T.$$

Taking $l = (j-1)N + i$, the elements $a_{lp}(\mathbf{v})$ of the lth row of $A(\mathbf{v})$, for

$l = 1, \ldots, n$, are given by

$$
\begin{aligned}
a_{l,l-1}(\mathbf{v}) &= -\frac{1}{h^2}\delta_{ij}(v_{ij}), \quad a_{l,l+1}(\mathbf{v}) = -\frac{1}{h^2}\delta_{i+1,j}(v_{i+1,j}), \\
a_{l,l-N}(\mathbf{v}) &= -\frac{1}{h^2}\delta_{ij}(v_{ij}), \quad a_{l,l+N}(\mathbf{v}) = -\frac{1}{h^2}\delta_{i,j+1}(v_{i,j+1}), \\
a_{ll}(\mathbf{v}) &= \frac{1}{h^2}(2\delta_{ij}(v_{ij}) + \delta_{i+1,j}(v_{i+1,j}) + \delta_{i,j+1}(v_{i,j+1})), \\
a_{lp}(\mathbf{v}) &= 0, \quad \text{for } p \neq l-N, l-1, l, l+1, l+N, \quad l = 1, \ldots, n,
\end{aligned}
$$

where $\delta_{mp}(v_{ij}) = \delta(x_m, y_p, v_{ij})$. The lth component $G_k(\mathbf{v})$ of the mapping $G(\mathbf{v}) = (G_1(\mathbf{v}), \ldots, G_n(\mathbf{v}))^T$ is a function of the coordinates (x, y) of the lth grid point and of the approximation of $\varphi(x, y)$ in this point.

The system (14) has at least one solution and all the solutions belong to a well defined closed ball in $\mathbb{R}^n$

$$\overline{B}_h(0, \varepsilon) = \{\mathbf{v} \in \mathbb{R}^n : \|\mathbf{v}\|_h \leq \varepsilon\},$$

where ε is independent of h and of $\|s\|_h$ (see [18]). Here, $\|\mathbf{v}\|_h$ is defined as

$$\|\mathbf{v}\|_h \equiv \left(\sum_{i=1}^{N}\sum_{j=1}^{N} h^2\varphi_{ij}^2\right)^{1/2} = \left(\sum_{l=1}^{n} h^2 v_l^2\right)^{1/2} = h\|\mathbf{v}\|.$$

Moreover, Galligani [17] proves that the mapping F given by (14) and subject to Smoothness Conditions (I), (II), (III), and (IV), satisfies:

$$\langle F(\mathbf{v}) - F(\mathbf{w}), \mathbf{v} - \mathbf{w}\rangle \geq \left(C - \frac{\Lambda^2\beta^2}{2\delta_{min}}\right)\|\mathbf{v} - \mathbf{w}\|_h^2, \quad \text{for } \mathbf{v}, \mathbf{w} \in \overline{B}_h(0, \varepsilon),$$

where β is a positive constant (see [17] for details). Now, since $\|\mathbf{v} - \mathbf{w}\|_h^2 = h^2\|\mathbf{v} - \mathbf{w}\|^2$, the above inequality implies that, for $\mathbf{v}, \mathbf{w} \in \overline{B}_h(0, \varepsilon)$,

$$\langle F(\mathbf{v}) - F(\mathbf{w}), \mathbf{v} - \mathbf{w}\rangle \geq \left(C - \frac{\Lambda^2\beta^2}{2\delta_{min}}\right)h^2\|\mathbf{v} - \mathbf{w}\|^2. \tag{16}$$

Therefore, when $C - \frac{\Lambda^2\beta^2}{2\delta_{min}} > 0$ or $\frac{\Lambda^2\beta^2}{2C\delta_{min}} < 1$, the inequality (16) shows that the mapping F given by (14) is strongly monotone on $\overline{B}_h(0, \varepsilon)$.

The following theorem says, under special conditions, that if Algorithm 2 is applied to the system (14), it generates a sequence $\{\mathbf{v}_k\}$ such that all its limits points are solutions of (14).

Theorem 2. *Let F be the mapping given by* (14)*. Let Ω be the n-dimensional box defined as*

$$\Omega = \left\{ \mathbf{v} \in \mathbb{R}^n : -\frac{\varepsilon}{h} \leq \mathbf{v} \leq \frac{\varepsilon}{h} \right\}. \tag{17}$$

Assume that Smoothness Conditions (I), (II), (III), and (IV) hold so that the inequality (16) *is satisfies for all $\mathbf{v}, \mathbf{w} \in \Omega$, with $C - \frac{\Lambda^2\beta^2}{2\delta_{min}} > 0$. Then, $\lim_{k\to\infty} \|F(\mathbf{v}_k)\| = 0$, where $\{\mathbf{v}_k\}$ is the sequence generated by Algorithm 2 when it is applied to the system $F(\mathbf{v}) = 0$ with bound constrained given by* (17)*.*

Proof. Since $\overline{B}_h(0,\varepsilon) \subset \Omega$, then all the solutions of the system $F(\mathbf{v}) = 0$ belong to Ω. Also, it is clear that F is bounded, strictly monotone and Lipschitz continuous on Ω. Therefore, by Theorem 1 the sequence $\{\|F(\mathbf{v}_k)\|\}$ converges to 0. $\square$

4 Numerical results

This section reports detailed results of ours numerical experiments to show the efficiency of the method `rabc_monotone`. We present some numerical results on two set of test problems. The first set of problems are systems of nonlinear monotone equations with box constraints commonly used in the literature. The second set of problems considers the numerical solving of the system of nonlinear equations that is obtained in the discretization of a steady state reaction-diffusion problem using the finite difference method.

We compare the performance of `rabc_monotone` with a recent derivative-free projection method proposed by Liu and Li [19], which is denoted as PCG and can be viewed as an extension of CG-DESCENT method [20]. In all experiments described here we use Matlab for the implementation of the algorithms. All the runs were carried out on an Intel Core i5 at 2.4 GHz with 8 GB of RAM.

For the implementation of PCG method, we used the same parameter $\xi = 1$, $\rho = 0.55$ and $\sigma = 0.0001$ as in [19]. PCG method needs the operator projection $P_\Omega : \mathbb{R}^n \to \Omega$ which is defined as $P_\Omega(\mathbf{x}) = \mathbf{y}$, where

$$y_i = \begin{cases} x_i, & x_i \in [l_i, u_i], \\ l_i, & x_i < l_i, \\ u_i, & x_i > u_i, \end{cases} \qquad \text{for } i = 1, 2, \ldots, n.$$

We implement to `rabc_monotone` with the following parameters: $\gamma = 10^{-4}$, $\sigma = 0.5$, $\upsilon = 0.9$, $\alpha_{max} = 10^{30}$, $\alpha_0 = 1$, and $\eta_k = (0.99999)^k(1000 + \|F(x_0)\|^2)$, for all $k \geq 0$.

In all the algorithms we stop the process when

$$\|F(\mathbf{x}_k)\| \leq \varepsilon_a, \tag{18}$$

where $\varepsilon_a \in (0, 1)$. We claim that the method fails, and use the symbol "*", when the number of iterations is greater than 100000.

We begin the numerical experiments studying the performance of the methods PCG and `rabc_monotone` in to solution of (1), when

$$F(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_n(\mathbf{x}))^T$$

is a monotone mapping that maps $\mathbb{R}^n$ into itself. In these problems, we take a box given by $\Omega = \{\mathbf{x} \in \mathbb{R}^n : x_i \geq -1,\ i = 1, \ldots, n\}$. We use the following test problems:

Problem 1. [21]. The elements of $F(\mathbf{x})$ are given by:

$$f_i(\mathbf{x}) = e^{x_i} - 1, \quad i = 1, \ldots, n.$$

Problem 2 [8]. The elements of $F(\mathbf{x})$ are given by:

$$f_i(\mathbf{x}) = 2x_i - \sin(x_i), \quad i = 1, \ldots, n.$$

Problem 3 [8]. The elements of $F(\mathbf{x})$ are given by:

$$f_i(\mathbf{x}) = 2x_i - \sin|x_i|, \quad i = 1, \ldots, n.$$

Problem 4 [5]. The elements of $F(\mathbf{x})$ are given by:

$$f_i(\mathbf{x}) = x_i - \sin|x_i - 1|, \quad i = 1, \ldots, n.$$

Problem 5 [4,5]. The elements of $F(\mathbf{x})$ are given by:

$$\begin{aligned}
f_1(x) &= x_1 - e^{\left(\cos\left(\frac{x_1+x_2}{n+1}\right)\right)},\\
f_i(x) &= x_i - e^{\left(\cos\left(\frac{x_{i-1}+x_i+x_{i+1}}{n+1}\right)\right)}, \quad i = 2, \ldots, n-1,\\
f_n(x) &= x_n - e^{\left(\cos\left(\frac{x_{n-1}+x_n}{n+1}\right)\right)}.
\end{aligned}$$

Problem 6 [9]. The elements of $F(\mathbf{x})$ are given by:

$$f_i(\mathbf{x}) = \frac{i}{10}(e^{x_i} - 1), \quad i = 1, \dots, n.$$

Problem 7 [6]. The function $F(\mathbf{x})$ is given by:

$$F(\mathbf{x}) = A\mathbf{x} + \psi(\mathbf{x}),$$

where $\psi(\mathbf{x}) = (e^{(x_i - 1)} - 1, \dots, e^{(x_n - n)} - 1)^T$ and

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

Problem 8 [14]. The function $F(\mathbf{x})$ is given by:

$$F(\mathbf{x}) = A\mathbf{x} + \psi(\mathbf{x}),$$

where A is the bidimensional finite differences Laplacian $n_0^2 \times n_0^2$ matrix, say $A = \text{gallery('poisson'}, n_0)$, $\psi(x) = (h^2(x_1^3 - 10), \dots, h^2(x_n^3 - 10))^T$, $h = 1/(n_0 + 1)$, and $n = n_0^2$.

We test Problems 1-7 with the number of variables $n = 5000, 10000, 50000, 100000$, and Problem 8 with $n = 100, 400, 1600, 3600$. As in [19], we choose the same starting points to test the robustness of the considered algorithms: $\mathbf{x}_0^{(1)} = (1, \dots, 1)^T$, $\mathbf{x}_0^{(2)} = (2, \dots, 2)^T$, $\mathbf{x}_0^{(3)} = (8, \dots, 8)^T$, $x_0^{(4)} = (10, \dots, 10)^T$. In addition, we test each problem with 10 stochastically independent random initial vectors with uniform distribution on the hypercube $[0, 1]^{n \times n}$, generated by the Matlab code `rand(n,1)`. Tables 1-4 show the obtained results when (18) holds for $\varepsilon_a = 10^{-6}$. We report the number of problem (P), dimension (n), initial iterate (x_0), the number of iterations (*It*), the number of function evaluations (*feval*), the CPU time in seconds (*time*), and the norm $\|F(\mathbf{x}_{It})\|$, where $\mathbf{x}_{It}$ is the solution found by the method. This notation is also used in the remaining tables. In Tab. 4 we reported the average number of iterations, the average number of function evaluations, the average CPU time in seconds (*time*), and the average norm $\|F(\mathbf{x}_{It})\|$. These results suggest that the performances of the proposed algorithm is competitive with PCG method. Furthermore, we appreciate that `rabc_monotone` is a robust method because it found a solution to all problems.

Tab. 1: Results for Problems 1-3

			PCG	rabc_monotone
P	$\mathbf{x}_0$	n	$It/feval/time/\|F(\mathbf{x}_{It})\|$	$It/feval/time/\|F(\mathbf{x}_{It})\|$
1	$\mathbf{x}_0^{(1)}$	5000	8/16/0.04/4.9e-07	8/9/0.01/9.2e-12
		10000	8/16/0.03/7.0e-07	8/9/0.02/1.3e-11
		50000	9/18/0.10/1.7e-07	8/9/0.04/2.9e-11
		100000	9/18/0.23/2.4e-07	8/9/0.08/4.1e-11
	$\mathbf{x}_0^{(2)}$	5000	8/17/0.01/5.6e-07	9/10/0.02/5.0e-08
		10000	8/17/0.03/7.9e-07	9/10/0.01/7.9e-08
		50000	9/19/0.13/1.9e-07	9/10/0.07/2.0e-07
		100000	9/19/0.20/2.7e-07	9/10/0.13/3.0e-07
	$\mathbf{x}_0^{(3)}$	5000	9/28/0.05/6.6e-07	18/19/0.00/1.9e-09
		10000	9/28/0.04/9.3e-07	18/19/0.05/3.9e-09
		50000	10/30/0.13/2.2e-07	18/19/0.11/1.4e-08
		100000	10/30/0.28/3.2e-07	18/19/0.22/2.2e-08
	$\mathbf{x}_0^{(4)}$	5000	9/31/0.00/3.4e-07	21/22/0.05/3.4e-10
		10000	9/31/0.04/4.8e-07	21/22/0.05/7.8e-10
		50000	10/33/0.14/1.2e-07	21/22/0.14/3.2e-09
		100000	10/33/0.28/1.6e-07	21/22/0.24/5.3e-09
2	$\mathbf{x}_0^{(1)}$	5000	9/18/0.03/4.2e-07	4/5/0.00/3.5e-07
		10000	9/18/0.03/6.0e-07	4/5/0.01/5.0e-07
		50000	10/20/0.10/1.4e-07	5/6/0.03/1.4e-15
		100000	10/20/0.25/2.0e-07	5/6/0.05/2.0e-15
	$\mathbf{x}_0^{(2)}$	5000	9/18/0.03/3.5e-07	7/8/0.01/1.0e-13
		10000	9/18/0.02/4.9e-07	7/8/0.02/1.6e-13
		50000	10/20/0.11/1.2e-07	7/8/0.05/4.4e-13
		100000	10/20/0.24/1.7e-07	7/8/0.08/6.4e-13
	$\mathbf{x}_0^{(3)}$	5000	11/26/0.03/4.5e-07	7/8/0.01/5.0e-12
		10000	11/26/0.04/6.4e-07	7/8/0.01/2.2e-11
		50000	12/28/0.14/1.5e-07	7/8/0.05/1.9e-10
		100000	12/28/0.29/2.2e-07	7/8/0.08/3.6e-10
	$\mathbf{x}_0^{(4)}$	5000	11/26/0.02/4.5e-07	7/8/0.01/5.1e-10
		10000	11/26/0.02/6.3e-07	7/8/0.03/1.5e-09
		50000	12/28/0.12/1.5e-07	7/8/0.05/7.9e-09
		100000	12/28/0.30/2.1e-07	7/8/0.08/1.4e-08
3	$\mathbf{x}_0^{(1)}$	5000	9/18/0.02/4.2e-07	6/7/0.01/4.4e-12
		10000	9/18/0.03/6.0e-07	6/7/0.00/6.2e-12
		50000	10/20/0.11/1.4e-07	6/7/0.03/1.4e-11
		100000	10/20/0.23/2.0e-07	6/7/0.06/2.0e-11
	$\mathbf{x}_0^{(2)}$	5000	9/18/0.02/3.5e-07	7/8/0.00/1.0e-13
		10000	9/18/0.02/4.9e-07	7/8/0.03/1.6e-13
		50000	10/20/0.10/1.2e-07	7/8/0.05/4.4e-13
		100000	10/20/0.24/1.7e-07	7/8/0.08/6.4e-13
	$\mathbf{x}_0^{(3)}$	5000	11/26/0.02/4.5e-07	7/8/0.01/5.0e-12
		10000	11/26/0.02/6.4e-07	7/8/0.03/2.2e-11
		50000	12/28/0.13/1.5e-07	7/8/0.04/1.9e-10
		100000	12/28/0.28/2.2e-07	7/8/0.08/3.6e-10
	$\mathbf{x}_0^{(4)}$	5000	11/26/0.03/4.5e-07	7/8/0.01/5.1e-10
		10000	11/26/0.04/6.3e-07	7/8/0.00/1.5e-09
		50000	12/28/0.14/1.5e-07	7/8/0.04/7.9e-09
		100000	12/28/0.28/2.1e-07	7/8/0.08/1.4e-08

We conclude the numerical experiences considering the system (14). More specifically, we will consider reaction diffusion problems in which the functions $\delta(x, y, \varphi)$ and $g(x, y, \varphi)$ only depend of φ. This type of problems describe many reaction diffusion processes in realistic applications (see e.g.,

Tab. 2: Results for Problems 4-6

P	$\mathbf{x}_0$	n	PCG $It/feval/time/\|F(\mathbf{x}_{It})\|$	rabc_monotone $It/feval/time/\|F(\mathbf{x}_{It})\|$
4	$\mathbf{x}_0^{(1)}$	5000	8/24/0.05/5.9e-07	5/6/0.00/6.1e-07
		10000	8/24/0.03/8.3e-07	5/6/0.05/8.7e-07
		50000	9/27/0.09/1.5e-07	6/7/0.03/3.1e-12
		100000	9/27/0.25/2.1e-07	6/7/0.06/4.4e-12
	$\mathbf{x}_0^{(2)}$	5000	8/22/0.05/8.5e-07	6/8/0.00/2.5e-07
		10000	9/25/0.05/9.8e-08	6/8/0.00/3.5e-07
		50000	9/25/0.09/2.2e-07	6/8/0.03/7.9e-07
		100000	9/25/0.27/3.1e-07	7/9/0.09/9.8e-13
	$\mathbf{x}_0^{(3)}$	5000	8/22/0.00/4.1e-07	6/7/0.03/6.8e-11
		10000	8/22/0.05/5.8e-07	6/7/0.00/9.6e-11
		50000	9/25/0.11/1.0e-07	6/7/0.06/2.1e-10
		100000	9/25/0.23/1.5e-07	6/7/0.03/3.0e-10
	$\mathbf{x}_0^{(4)}$	5000	11/31/0.03/2.7e-07	5/6/0.05/5.7e-08
		10000	11/31/0.03/3.9e-07	5/6/0.00/8.0e-08
		50000	11/31/0.13/8.7e-07	5/6/0.00/1.8e-07
		100000	12/34/0.34/1.0e-07	5/6/0.05/2.5e-07
5	$\mathbf{x}_0^{(1)}$	5000	11/23/0.03/2.3e-07	2/3/0.00/1.9e-10
		10000	10/20/0.03/1.5e-07	2/3/0.00/1.6e-11
		50000	10/20/0.17/3.8e-07	2/3/0.06/9.9e-14
		100000	10/20/0.41/5.4e-07	2/3/0.11/0.0e+00
	$\mathbf{x}_0^{(2)}$	5000	9/18/0.00/4.4e-07	2/3/0.05/4.1e-11
		10000	9/18/0.03/6.6e-07	2/3/0.05/3.6e-12
		50000	10/20/0.17/1.6e-07	2/3/0.00/6.3e-16
		100000	10/20/0.41/2.2e-07	2/3/0.09/0.0e+00
	$\mathbf{x}_0^{(3)}$	5000	12/25/0.00/9.1e-08	2/3/0.00/5.1e-09
		10000	10/20/0.05/3.5e-07	2/3/0.05/4.5e-10
		50000	11/22/0.17/1.2e-07	2/3/0.00/1.7e-12
		100000	11/22/0.48/1.8e-07	2/3/0.08/1.4e-13
	$\mathbf{x}_0^{(4)}$	5000	12/25/0.05/1.3e-07	2/3/0.00/1.1e-08
		10000	10/20/0.06/4.2e-07	2/3/0.00/1.0e-09
		50000	11/22/0.22/1.7e-07	2/3/0.06/3.6e-12
		100000	11/22/0.56/2.4e-07	2/3/0.05/4.2e-13
6	$\mathbf{x}_0^{(1)}$	5000	*/*/*/*	851/852/0.50/6.9e-07
		10000	*/*/*/*	1002/1003/0.84/9.3e-07
		50000	*/*/*/*	2866/2867/11.09/2.7e-07
		100000	*/*/*/*	3349/3351/26.56/8.8e-07
	$\mathbf{x}_0^{(2)}$	5000	*/*/*/*	900/901/0.52/9.8e-07
		10000	*/*/*/*	1126/1127/0.98/7.5e-07
		50000	*/*/*/*	2076/2077/7.78/9.5e-07
		100000	*/*/*/*	4396/4399/34.05/2.6e-07
	$\mathbf{x}_0^{(3)}$	5000	*/*/*/*	739/740/0.39/9.3e-07
		10000	*/*/*/*	1382/1383/1.13/9.7e-07
		50000	*/*/*/*	2395/2396/8.97/9.6e-07
		100000	*/*/*/*	3425/3426/27.52/4.9e-07
	$\mathbf{x}_0^{(4)}$	5000	*/*/*/*	717/718/0.41/2.9e-07
		10000	*/*/*/*	1571/1572/1.41/5.9e-07
		50000	*/*/*/*	2289/2290/8.59/9.9e-07
		100000	*/*/*/*	3508/3509/27.89/9.8e-07

[11–13, 22]). The source function $s(x,y)$ is chosen in order to satisfy a prespecified exact solution $\mathbf{v}_*$ of the nonlinear system (14). The solution is chosen as $\varphi_*(x,y) = \sin(\pi x)\sin(\pi y)$. For this type of problems the equation (15) becomes as follows, for $(x,y) \in R$:

$$\begin{cases} -\delta(\varphi)\nabla^2\varphi(x,y) - \delta'(\varphi)\left[\left(\dfrac{\partial\varphi}{\partial x}\right)^2 + \left(\dfrac{\partial\varphi}{\partial y}\right)^2\right] + g(\varphi) = s(x,y), \\ \varphi(x,y) = 0, \quad (x,y) \in \partial R. \end{cases} \tag{19}$$

Tab. 3: Results for Problems 7 y 8

			PCG	rabc_monotone
P	$\mathbf{x}_0$	n	$It/feval/time/\lVert F(\mathbf{x}_{It})\rVert$	$It/feval/time/\lVert F(\mathbf{x}_{It})\rVert$
7	$\mathbf{x}_0^{(1)}$	5000	59/274/1.64/9.8e-07	17/18/0.16/1.0e-06
		10000	59/274/2.84/9.8e-07	17/18/0.20/1.0e-06
		50000	59/274/15.64/9.8e-07	17/18/1.09/1.0e-06
		100000	59/274/24.45/9.8e-07	17/18/1.52/1.0e-06
	$\mathbf{x}_0^{(2)}$	5000	92/495/2.95/8.1e-07	25/26/0.16/1.3e-07
		10000	94/514/5.25/7.7e-07	25/26/0.27/6.6e-07
		50000	92/506/29.03/8.9e-07	25/26/1.48/9.3e-07
		100000	103/562/50.31/8.2e-07	25/26/2.38/8.0e-07
	$\mathbf{x}_0^{(3)}$	5000	120/633/3.95/9.4e-07	27/28/0.23/7.3e-07
		10000	110/576/6.09/8.2e-07	28/29/0.30/8.3e-07
		50000	97/542/31.38/8.3e-07	30/31/1.92/9.6e-07
		100000	132/834/74.63/7.1e-07	30/31/2.70/2.2e-07
	$\mathbf{x}_0^{(4)}$	5000	124/675/4.95/7.5e-07	31/32/0.28/4.6e-07
		10000	127/707/8.48/8.4e-07	31/32/0.42/3.2e-07
		50000	271/2295/134.08/7.3e-07	33/34/2.17/6.8e-07
		100000	1304/13488/1213.23/9.7e-07	34/35/3.13/2.1e-07
8	$\mathbf{x}_0^{(1)}$	100	1006/5251/3.47/9.9e-07	62/63/0.05/6.8e-07
		400	2956/15800/13.06/9.9e-07	102/104/0.11/5.5e-07
		1600	8612/47337/68.94/1.0e-06	272/276/0.47/9.4e-07
		3600	18635/102463/562.50/1.0e-06	399/408/2.09/9.1e-07
	$\mathbf{x}_0^{(2)}$	100	1061/5550/3.58/1.0e-06	68/69/0.05/9.4e-07
		400	3186/17024/14.03/1.0e-06	159/163/0.14/2.5e-07
		1600	9298/51110/73.92/1.0e-06	258/264/0.36/3.9e-07
		3600	20157/110834/611.83/1.0e-06	397/406/2.00/4.1e-07
	$\mathbf{x}_0^{(3)}$	100	1413/7320/4.95/9.8e-07	68/69/0.05/4.8e-07
		400	3247/17428/14.78/9.9e-07	147/149/0.19/9.5e-07
		1600	9888/54359/78.75/1.0e-06	338/340/0.50/9.3e-07
		3600	21473/118069/650.52/1.0e-06	576/580/3.06/1.5e-07
	$\mathbf{x}_0^{(4)}$	100	1380/7161/5.55/9.8e-07	61/62/0.09/8.1e-07
		400	3241/17405/15.89/9.9e-07	147/148/0.13/7.9e-07
		1600	9918/54527/79.88/1.0e-06	278/280/0.41/1.0e-06
		3600	21531/118394/648.28/1.0e-06	464/466/2.42/7.9e-07

Now, discretizing the equation (19) with the method of finite differences, taking R the unit square and N grid points $(x_i, y_j) \in R$ with $x_i = ih$, $y_j = jh$ and $h = 1/(N+1)$, the following system of $n = N^2$ nonlinear equations is obtained:

$$F(\mathbf{v}) \equiv \frac{1}{h^2} d(\mathbf{v}) \circ (A\mathbf{v}) + \frac{1}{4h^2} d'(\mathbf{v}) \circ [(B\mathbf{v}) \wedge 2 + (C\mathbf{v}) \wedge 2] + G(\mathbf{v}) - s = 0, \quad (20)$$

where: $A \in \mathbb{R}^{n \times n}$ is the matrix obtained from the discretization of the Laplacean; $B \in \mathbb{R}^{n \times n}$ is the matrix defined by

$$B = \begin{pmatrix} D & & & \\ & D & & \\ & & \ddots & \\ & & & D \end{pmatrix}, \quad D = \begin{pmatrix} 0 & -1 & & & \\ 1 & 0 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & -1 \\ & & & 1 & 0 \end{pmatrix} \in \mathbb{R}^{N \times N};$$

$C \in \mathbb{R}^{n \times n}$ whose elements c_{ij} are all zero except $c_{i,i+1} = -1$ and $c_{i+N,i} = 1$, for $i = 1, 2, \ldots, N^2 - N$; $d(\mathbf{v}) = (\delta(v_1), \ldots, \delta(v_n))^T$; $d'(\mathbf{v}) = (\delta'(v_1), \ldots, \delta'(v_n))^T$;

Tab. 4: Results for Problems 1-8 with the randomly generated initial points

P	n	PCG $It/feval/time/\|F(\mathbf{x}_{It})\|$	rabc_monotone $It/feval/time/\|F(\mathbf{x}_{It})\|$
1	5000	15/36/0.04/5.8e-07	7/8/0.00/8.0e-10
	10000	17/41/0.05/2.5e-07	7/8/0.02/1.0e-09
	50000	16/38/0.17/1.2e-07	7/8/0.04/2.6e-09
	100000	16/38/0.37/8.1e-08	7/8/0.08/3.5e-09
2	5000	15/34/0.02/3.3e-07	4/5/0.01/1.3e-08
	10000	15/34/0.04/4.6e-07	4/5/0.01/1.7e-08
	50000	18/41/0.15/6.0e-08	4/5/0.03/3.8e-08
	100000	18/41/0.40/8.4e-08	4/5/0.04/5.3e-08
3	5000	16/59/0.04/3.9e-07	6/7/0.00/4.1e-13
	10000	16/59/0.05/6.7e-07	6/7/0.00/6.0e-13
	50000	17/63/0.20/5.1e-07	6/7/0.03/1.3e-12
	100000	17/63/0.48/9.4e-07	6/7/0.03/1.8e-12
4	5000	13/43/0.02/5.3e-07	5/6/0.01/5.1e-09
	10000	13/43/0.03/7.1e-07	5/6/0.01/6.2e-09
	50000	14/46/0.17/1.2e-07	5/6/0.02/1.7e-08
	100000	14/46/0.38/1.7e-07	5/6/0.06/2.3e-08
5	5000	11/23/0.04/3.0e-07	2/3/0.00/2.7e-10
	10000	11/23/0.05/7.7e-07	2/3/0.00/2.4e-11
	50000	10/20/0.19/4.8e-07	2/3/0.03/9.9e-14
	100000	10/20/0.42/7.0e-07	2/3/0.06/0.0e+00
6	5000	*/*/*/*	1001/1002/0.56/5.3e-07
	10000	*/*/*/*	942/943/0.86/9.6e-07
	50000	*/*/*/*	2202/2203/8.59/1.0e-06
	100000	*/*/*/*	3099/3100/22.63/8.9e-07
7	5000	97/452/2.53/7.6e-07	23/25/0.16/7.4e-07
	10000	99/462/3.66/9.0e-07	23/25/0.20/9.9e-07
	50000	103/481/15.08/8.9e-07	26/28/0.89/3.2e-07
	100000	105/490/29.66/9.8e-07	26/28/1.69/1.7e-07
8	100	507/2776/1.92/9.9e-07	54/55/0.05/4.7e-07
	400	1842/10129/9.33/9.9e-07	110/111/0.11/8.4e-07
	1600	4940/27169/39.47/1.0e-06	186/188/0.31/5.7e-07
	3600	10309/56699/327.00/1.0e-06	390/392/2.27/6.8e-07

$G(\mathbf{v}) = (g(v_1), \ldots, g(v_n))^T$; $s \in \mathbb{R}^n$ is a constant vector whose components correspond to the evaluation of the source function $s(x, y)$ at each grid point; the symbol "$\circ$" denotes the *Hadamard product*, which in the case of two vectors $\mathbf{v}, \mathbf{w} \in \mathbb{R}^n$, $\mathbf{v} \circ \mathbf{w}$ is another vector whose components are $(\mathbf{v} \circ \mathbf{w})_i = v_i w_i$, $i = 1, 2, \ldots, n$; and the symbol "$\wedge 2$" denotes an operation in $\mathbb{R}^n$ given by $(\mathbf{v} \wedge 2)_i = v_i^2$, $i = 1, 2, \ldots, n$.

We test PCG and `rabc_monotone` with the system (20) with $\delta(\varphi) = 0.02 + 0.5\varphi^2$ and $g(\varphi) = \dfrac{\varphi^2}{1+\varphi^2}$, the n-dimensional box $\Omega = \{\mathbf{v} \in \mathbb{R}^n : 0 \leq \mathbf{v} \leq 100\}$, and the initial iterate $\mathbf{v}_0 = (1, 1, \ldots, 1)^T$. Tab. 5 shows the obtained results when $N = 10$, 20, 40, 60 and (18) holds for $\varepsilon_a = 10^{-8}$. In this table we also report the error $err = \|\mathbf{v}_* - \mathbf{v}_{It}\|/\|\mathbf{v}_*\|$. These results suggest that the proposed algorithm has better performance than PCG method.

Tab. 5: Results for the system (20) with $\delta(\varphi) = 0.02+0.5\varphi^2$ and $g(\varphi) = \frac{\varphi^2}{1+\varphi^2}$

		PCG	rabc_monotone
N	$\|F(\mathbf{v}_0)\|$	*It/feval/time/err*	*It/feval/time/err*
10	6.4e+02	7658/91935/8.98/1.8e-10	92/93/0.03/3.2e-12
20	3.2e+03	6244/88787/9.66/8.7e-11	224/225/0.05/5.6e-11
40	1.7e+04	19575/327581/50.42/4.5e-11	640/642/0.16/3.6e-13
60	4.5e+04	372039/6699519/6536.88/3.0e-11	1052/1055/1.16/2.8e-11

5 Final remarks

The `rabc_monotone` method, proposed in this paper, is a derivative-free method for solving nonlinear monotone equations with bound constrained. Due to its simplicity for building the search direction, `rabc_monotone` is very easy to implement, memory requirements are minimal and, so, its use for solving large-scale nonlinear monotone equations with bound constrained is attractive.

Our preliminary numerical results indicate that `rabc_monotone` efficiently solved all test problems, as long as, the problem is well-conditioned. Essentially, the behavior of `rabc_monotone` is similar to the behavior of the algorithms SANE and DF-SANE, which can be considered damped quasi-Newton methods for solving nonlinear systems (see [10] and references therein), which use a multiple of the identity matrix for approximating the Jacobian matrix of F (when F is differentiable). This simple approximation of the Jacobian can be the cause that the algorithms require a greater number of iterations and function evaluations. The slow convergence of `rabc_monotone` for ill-conditioned problems may be enhanced by incorporating a suitable preconditioning technique.

As to application in solving steady state reaction-diffusion problems, our preliminary numerical results shows that `rabc_monotone` efficiently find a solution of the problem. However, the computational effort of the algorithm tends to increase drastically when the problem is ill-conditioned. Therefore, the design of preconditioning strategies and studying the behavior of `rabc_monotone` in the simulation of reaction diffusion processes in realistic applications, are topics to develop in future researches.

Acknowledgements:

This work has been supported by CDCH-UCV, project PG-08-8628-2013/2.

We would like to thank the referees by carefully reading the manuscript and for several suggestions that improved the presentation of this work.

References

[1] Q.N. Li and D.H. Li. A class of derivative-free methods for large-scale nonlinear monotone equations. *IMA Journal of Numerical Analysis*, 31(4):1625–1635, 2011. doi: 10.1093/imanum/drq015.

[2] C.W. Wang, Y.J. Wang, and C.L. Xu. A projection method for a system of nonlinear monotone equations with convex constraints. *Mathematical Methods of Operations Research*, 66(1):33–46, 2007. doi: 10.1007/s00186-006-0140-y.

[3] Y.H. Xiao and H. Zhu. A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing. *Journal of Mathematical Analysis and Applications*, 405(1):310–319, 2013. doi: 10.1016/j.jmaa.2013.04.017.

[4] Q.R. Yan, X.Z. Peng, and D.H. Li. A globally convergent derivative-free method for solving large scale nonlinear monotone equations. *Journal of Computational and Applied Mathematics*, 234(3):649–657, 2010. doi: 10.1016/j.cam.2010.01.001.

[5] Z.S. Yu, J. Lin, J. Sun, Y.X. XIao, L.Y. Liu, and Z.H. Li. Spectral gradient projection method for monotone nonlinear equations with convex constraints. *Applied Numerical Mathematics*, 59(10):2416–2423, 2009. doi: 10.1016/j.apnum.2009.04.004.

[6] G. Yu, S. Niu, J. Ma, and Y. Song. An adaptive prediction-correction method for solving large-scale nonlinear systems of monotone equations with applications. *Abstract and Applied Analysis*, 2013:1–13, 2013. doi: 10.1155/2013/619123.

[7] W. Zhou and D.H. Li. A globally convergent bfgs method for nonlinear monotone equations without any merit functions. *Math. Comput.*, 77(264):2231–2240, 2008. http://www.jstor.org/stable/40234609.

[8] W. Zhou and D.H. Li. Limited memory bfgs method for nonlinear monotone equations. *J. Comput. Math.*, 25(1):89–96, 2007. http://www.jstor.org/stable/43693347.

[9] W. La Cruz and M. Raydan. Nonmonotone spectral methods for large-scale nonlinear systems. *Optimization Methods & Software*, 18(5):583–599, 2003. doi: 10.1080/10556780310001610493.

[10] W. La Cruz, J.M. Martínez, , and M. Raydan. Spectral residual method without gradient information for solving large-scale nonlinear systems of equations. *Mathematics of Computation*, 75(255):1429–1448, 2006. doi: 10.1090/S0025-5718-06-01840-0.

[11] D. Jones, M. Plank, and Sleeman B. *Differential Equations and Mathematical Biology*. CRC Press, second edition, 2010. ISBN 978-1420083576.

[12] Britton N. *Reaction-Diffusion Equations and Their Applications to Biology*. Academic Press, 1986. ISBN 978-0121351403.

[13] J.D. Murray. *Mathematical Biology, Vol. I, II*. Springer&Verlag, Berlin, 2003. ISBN 0-387-95223-3 and 0-387-95228-4.

[14] J. Ortega and W. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, 1970. ISBN 978-0-12-528550-6.

[15] Barzilai J. and Borwein J. Two point step gradient methods. *IMA Journal of Numerical Analysis*, 8:141–148, 1988. doi: 10.1093/imanum/8.1.141.

[16] J.E. Dennis and J.J. Moré. A characterization of superlinear convergence and its applications to quasi-newton methods. *Math. Comput.*, 28:549–560, 1974. http://www.ams.org/journals/mcom/1974-28-126/S0025-5718-1974-0343581-1/S0025-5718-1974-0343581-1.pdf.

[17] Galligani E. Lagged diffusivity fixed point iteration for solving steady state reaction diffusion problems. *International Journal of Computer Mathematics*, 89(8):998–1016, 2012. doi: 10.1080/00207160.2012.671478.

[18] G.H. Meyer. *The numerical solution of quasilinear elliptic equations, In: Numerical Solution of Systems of Nonlinear Algebraic Equations (G. Byrne, C.A. Hall eds.)*. Academic Press, New York, 1973. ISBN 978-0-12-148950-2.

[19] J.k. Liu and S.J. Li. A projection method for convex constrained monotone nonlinear equations with applications. *Computers and Mathematics with Applications*, 70(10):2442–2453, 2015. doi: 10.1016/j.camwa.2015.09.014.

[20] W.W. Hager and H. Zhang. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J. Optim.*, 16(1):170–192, 2005. doi: 10.1137/030601880.

[21] C. Wang and Y. Wang. A superlinearly convergent projection method for constrained systems of nonlinear equations. *Journal of global optimization*, 44(2):283–296, 2009. doi: 10.1007/s10898-008-9324-8.

[22] G. Buffoni, Griffa A., Z. Li, and P. de Mottoni. Spatially distributed communities: the resource–consumer system. *Journal of Mathematical Biology*, 33(7):723–743, 1995. doi: 10.1007/BF00184646.