A global hybrid derivative-free method for large-scale systems of nonlinear equations

R. G. Begiato*, A. L. Custódio†, M. A. Gomes-Ruggiero‡,

April 16, 2018

Abstract

This work concerns the numerical solution of large-scale systems of nonlinear equations, when derivatives are not available for use, but assuming that all functions defining the problem are continuously differentiable. A hybrid approach is taken, based on a derivative-free iterative method, organized in three phases.

The first phase is defined by derivative-free versions of a fixed-point method that employs spectral parameters to define the steplength along the residual direction. The second phase consists on a matrix-free inexact Newton method that employs the GMRES to solve the linear system that computes the search direction. This second phase will only take place if the first one fails to find a better point after a predefined number of reductions in the step size. A third phase, based on directional direct search, should act whenever too many line searches have excessively decreased the steplenght along the inexact-Newton direction. In all stages, the criterion to accept a new point considers a nonmonotone decrease condition upon a merit function.

Convergence results are established for two scenarios: for the proposed three phases method and when the last phase is disabled. The numerical performance is assessed through experiments in a set of problems collected from the literature. Both the theoretical and the experimental analysis support the feasibility of the proposed hybrid strategy.

Keywords: Nonlinear equations; derivative-free methods; spectral residual; inexact Newton; matrix-free strategy; nonmonotone line search; directional direct search.

*Departamento Acadêmico de Matemática, UTFPR, 80230-901, Curitiba, PR, Brazil (begiato@utfpr.edu.br)

†Department of Mathematics, FCT-UNL-CMA, Campus de Caparica, 2829-516 Caparica, Portugal (alcustodio@fct.unl.pt). Support for this author was provided by Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) under the project UID/MAT/00297/2013 (CMA).

‡Departamento de Matemática Aplicada, IMECC, UNICAMP, 13083-970, Campinas, SP, Brazil (marcia@ime.unicamp.br). Partially supported by PRONEX-Optimization; CNPq (Grants 306220/2009-1 and 304032/2010-7); FAPESP (Grants 2013/05475-7 and 2013/07375-0) and CAPES.

1 Introduction

In this work, we propose two methods for solving the system of nonlinear equations

$$\begin{aligned} &F(x) = 0, \\ &F : \mathbb{R}^n \to \mathbb{R}^n, \end{aligned} \tag{1}$$

where F is a continuously differentiable function, but derivatives are not available for use, neither could be approximated by numerical techniques. Thus, the algorithms developed in this work are derivative-free, although differentiability assumptions are considered when deriving theoretical results.

To solve problem (1), we use iterative methods, where each iterate takes the form

$$x_{k+1} = x_k + \lambda_k d_k, \tag{2}$$

with d_k a search direction and $\lambda_k \in (0, 1]$ a step length.

When derivatives are available for use, Newton method is a classical approach for solving (1). In this case, the search direction d_k is computed as the solution of the linear system:

$$J(x_k)d = -F(x_k), \tag{3}$$

where $J(x_k)$ denotes the Jacobian matrix of F at x_k.

Nevertheless, in the presence of large problems, computing an exact solution of (3) could be unpractical, which motivates the use of inexact Newton methods [7]. In this case, the search direction should satisfy:

$$\|J(x_k)d_k + F(x_k)\| \leq \eta_k \|F(x_k)\|, \tag{4}$$

where $\eta_k \in [0, 1)$ is called a forcing term. Krylov methods [14] are a classical approach to compute d_k satisfying (4), allowing derivative-free versions [13, 14].

Fixed-point iterations have also been considered for solving (1)([6, 5, 12]). In this case, a multiple of the residual vector is used as search direction, $d_k = \alpha_k F(x_k)$, $\alpha_k \in \mathbb{R}$, avoiding derivative calculations and the use of matrices.

Under suitable assumptions, theoretical results can be derived establishing local convergence of the previous methods [14]. For global convergence, a merit function needs to be considered, traducing the solution of the nonlinear system into a minimization problem, and a globalization procedure needs to be adopted.

Merit functions are usually defined as $f : \mathbb{R}^n \to \mathbb{R}$, with $f(x) = \|F(x)\|_2$ or $f(x) = \|F(x)\|_2^2$. Regarding the globalization procedure, typical approaches lie on a line search with an Armijo type condition [1] to accept new points. However, in some cases, this requirement can lead to a large number of function evaluations. More flexible criteria can be adopted, overcoming this difficulty without jeopardizing the global convergence properties. Grippo, Lampariello and Lucidi [10] proved the convergence of inexact Newton methods under a nonmonotone acceptance criterion. In Li and Fukushima [16] and La Cruz, Martínez and Raydan [5], nonmonotone acceptance criteria which do not require derivatives were proposed.

In derivative-free optimization, directional direct search is a popular class of methods, mainly due to the simplicity of the corresponding algorithmic structure and to the theoretical properties that can be established regarding its convergence [4, 15]. Each iteration of a directional direct search method is organized in a search and a poll step. The search step is optional and not required for establishing the convergence of the method, which is a direct consequence of polling. In the poll step, a set of directions with good geometrical properties is considered, typically a positive basis, and the objective function is evaluated in this set of directions, scaled by a step length parameter.

The polling procedure is responsible for the low convergence rate of directional direct search, with a bad performance when directly applied to the solution of large nonlinear systems. Nevertheless, some authors [11, 13] used a direct search step to improve the robustness of their base strategy, in case of failure. This approach will also be adopted in the current work.

We propose a hybrid three-step approach to solve large-scale systems of nonlinear equations:

1. `step 1`: use of a fixed point method, where the computation of the step length for the residual direction is based in a *spectral approach* [2, 6];

2. `step 2`: use of an inexact Newton method, where a matrix-free version of GMRES [18] is used to solve the inner linear system;

3. `step 3`: use of directional direct search, with a specialized set of poll directions.

The three steps are applied sequentially, but only in case of failure of the previous ones. So, direct search will be applied only when the two initial steps failed to find a new iterate. A two-phase version of the previous approach, recommended for larger problems, is also presented, where the direct search step is omitted.

In any case, we will consider $\|\cdot\| \equiv \|\cdot\|_2$ in the definition of the merit function and use a nonmonotone globalization strategy based on La Cruz, Martínez and Raydan [5].

The paper is organized as follows. Section 2 provides some background on spectral residual methods and on derivative-free globalization strategies. The three-phase hybrid nonmonotone derivative-free algorithm is proposed in Section 3 and its convergence is analyzed. In Section 4, we start by recalling the structure and the basic properties of a FDGMRES iteration, motivating the hybrid two-phase algorithm. Convergence properties for the method are also established. Numerical experiments on some test problems are reported in Section 5, comparing the hybrid approaches with pure methods. Finally, Section 6 is dedicated to some concluding remarks.

2 Derivative-free methods and line search techniques

The Spectral Residual Method (SANE), introduced by La Cruz and Raydan [6], uses as search directions

$$d_k = (1/\alpha_k)F(x_k) \text{ and } d_k = -(1/\alpha_k)F(x_k), \tag{5}$$

with α_k a *spectral scaling parameter*.

This is a fixed point method where the parameter α_k is computed by adjusting the Barzilai-Borwein [2] procedure for computing step sizes to the solution of systems of nonlinear equations. The vector $d_k = -(1/\alpha_k)F(x_k)$ is not necessarily a descent direction for the merit function, what justifies the systematic evaluation of both directions (5).

Being a quasi-Newton method, the computation of the spectral parameter requires that a Jacobian approximation $B_k = \alpha_k I$ satisfies the secant equation $B_k(x_k - x_{k-1}) = F(x_k) - F(x_{k-1})$. Thus:

$$\alpha_k = \frac{s_k^\top y_k}{s_k^\top s_k} = \frac{(x_k - x_{k-1})^\top (F(x_k) - F(x_{k-1}))}{(x_k - x_{k-1})^\top (x_k - x_{k-1})}. \tag{6}$$

In practical applications, scaling gradients or residual directions using spectral parameters conducted to good algorithmic performances, although the corresponding theoretical justification is not completely understood. Thus, flexible globalization strategies are required, allowing the algorithms to accept the use of the spectral parameter as a step size, without any attempt of backtracking. Rigid strategies to accept a new point can conduct to a performance similar to the one of the steepest descent method.

In [6] a nonmonotone line search was considered to guarantee global convergence of the method. The following condition:

$$f(x_k + \lambda_k d_k) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \gamma \lambda_k \nabla f(x_k)^\top d_k, \tag{7}$$

where $\gamma \in (0,1)$ and $M \in \mathbb{N}$, proposed by Grippo, Lampariello and Lucidi [10], was used as acceptance criterion for a new point, generating a sequence $\{x_k\}$ such that $\{f(x_k)\}$ is not necessarily decreasing.

La Cruz, Martínez and Raydan [5] proposed a derivative-free version of SANE. The new algorithm, named DFSANE, preserves the use of the residual direction and the spectral stepsize, but introduces a new acceptance criterion for new points, which does not use derivatives. This criteria blends (7) with the strategy proposed by Li and Fukushima [16].

In [16], a new point is accepted if

$$\|F(x_k + \lambda_k d_k)\| \leq (1 + \zeta_k)\|F(x_k)\| - \gamma \lambda_k^2 \|d_k\|^2, \tag{8}$$

with $\zeta_k > 0$ for all k, $\sum_k \zeta_k = \zeta < \infty$ and $\gamma \in (0,1)$. The approach followed by La Cruz, Martínez and Raydan [5] accepts new points that satisfy

$$f(x_{k+1}) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \zeta_k - \gamma \lambda_k^2 f(x_k) \tag{9}$$

where $f : \mathbb{R}^n \to \mathbb{R}_0^+$ is a merit function, $M \in \mathbb{N}$, $\gamma \in (0,1)$, $\zeta_k > 0$ for all $k \in \mathbb{N}$ and $\sum_k \zeta_k = \zeta < \infty$.

Algorithm 2.1 details an iteration of DFSANE.

Algorithm 2.1 *DFSANE*

Input parameters: x_k, $0 < \tau_{\min} < \tau_{\max} < 1$, $NBL_{\max}, M \in \mathbb{N}$, $\gamma \in (0,1)$, $\zeta_k > 0$ *and* $0 < \alpha_{\min} < \alpha_{\max}$.

1. *Choose* α_k *such that* $\alpha_k \in [\alpha_{\min}, \alpha_{\max}]$. *Set* $d = -(1/\alpha_k)F(x_k)$, $\lambda_+ = \lambda_- = 1$ *and* $NBL = 0$.

2. *If* $NBL = NBL_{\max}$, *set* $flag = 0$ *and terminate.*

3. *If* $f(x_k + \lambda_+ d) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \zeta_k - \gamma \lambda_+^2 f(x_k)$, *define* $d_k = d$, $\lambda_k = \lambda_+$, $flag = 1$ *and terminate.*

4. *if* $f(x_k - \lambda_- d) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \zeta_k - \gamma \lambda_-^2 f(x_k)$, *define* $d_k = -d$, $\lambda_k = \lambda_-$, $flag = 1$ *and terminate.*

5. *Choose* $\lambda_+ \in [\tau_{\min}\lambda_+, \tau_{\max}\lambda_+]$ *and* $\lambda_- \in [\tau_{\min}\lambda_-, \tau_{\max}\lambda_-]$, *set* $NBL = NBL + 1$ *and go to Step 2.*

Remark 1 The logical variable $flag$ does not play any active role in this algorithmic description. It is defined only for easiness of presentation, when combining this algorithm with others in the proposed hybrid approach. □

Convergence results were established by assuming continuity of the partial derivatives of $F : \mathbb{R}^n \to \mathbb{R}^n$ and considering $f : \mathbb{R}^n \to \mathbb{R}_0^+$ with $f(x) = \|F(x)\|$ or $f(x) = \|F(x)\|^2$ as merit functions [5].

Grippo and Sciandrone [13] proposed two different approaches to address the solution of a system of nonlinear equations. The first is an inexact Newton method, combining a nonmonotone watchdog phase [3] with a nonmonotone linesearch.

To compute d_k satisfying (4), a *matrix free* version of the classical Newton-GMRES [14] method was employed. In the classical GMRES method, the Jacobian matrix is required to solve the linear system (3). However, this matrix is only used in matrix-vector products. In the derivative-free case, these products are approximated by:

$$J(x_k)w \approx \frac{F(x_k + \sigma w) - F(x_k)}{\sigma}, \quad \sigma \in \mathbb{R} \setminus \{0\},\ w \in \mathbb{R}^n. \tag{10}$$

The new matrix free method is named FDGMRES and the corresponding inexact Newton method is known as Newton-FDGMRES [14].

In case of failure of the inexact Newton method, a second approach [13] adds a coordinate search phase to the first algorithm. This change allowed to weaken the assumptions required for establishing convergence and, in practical tests, improved the numerical robustness of the method. The coordinate search step consists in evaluating the objective function in the set $\{x_k \pm \lambda_k e_i : i = 1, 2, \ldots, n\}$, where e_i denotes the i-th column of the identity matrix.

As in [5], a nonmonotone line search is used as globalization strategy, but with different conditions associated to each type of steps. In the watchdog step the acceptance condition is

$$f(x_{k+1}) \leq \rho \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}),$$

whereas in the line search step it is considered

$$f(x_k + \lambda_k d_k) \leq (1 - \gamma \lambda_k) \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}),$$

with ρ, $\gamma \in (0, 1)$. Finally, in the coordinate search step it is used the condition

$$f(x_k + \lambda_k d_k) \leq (1 - \gamma \lambda_k^2) \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}). \tag{11}$$

3 A three-phase hybrid algorithm for systems of nonlinear equations

The practical numerical behavior of the spectral residual method indicates that it rarely succeeds in finding a better point when a high number of reductions in the stepsize needs to be performed. Therefore, we propose to impose a maximum limit to this number of reductions, replacing the spectral direction with a new search direction when this limit is reached. In the case, as in [13], we will use the inexact Newton direction. To accept new points, we consider the nonmonotone condition proposed by La Cruz, Martínez and Raydan (9), since it is more flexible than the one proposed by Grippo and Sciandrone [13]. The inexact Newton procedure is described in Algorithm 3.1.

Algorithm 3.1 *Inexact Newton I*

Input parameters: $x_k \in \mathbb{R}^n$, $0 < \lambda_{\min} \leq 1$, $\gamma \in (0, 1)$, $M \in \mathbb{N}$, $\zeta_k > 0$ *and* $\eta_k \in [0, 1)$.

1. *Try to compute* d_k *that satisfies (4). In case of failure, set* $d_k = 0$, $flag = 0$ *and terminate.*
2. *Set* $\lambda = 1$.
3. *If* $f(x_k + \lambda d_k) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \zeta_k - \gamma \lambda^2 f(x_k)$, *set* $\lambda_k = \lambda$, $flag = 1$ *and terminate.*

4. Reduce λ.

5. If $\lambda < \lambda_{\min}$, *set* $\lambda_k = 0$, $flag = 0$ *and terminate. Else, go to 3.*

To improve the robustness, the algorithm is enhanced with a directional direct search procedure. Instead of using coordinate search, as in [12] and in [13], we developed a new orthogonal set of poll directions containing

$$\{-(1/\alpha_k)F(x_k), (1/\alpha_k)F(x_k)\}.$$

Due to the high computational cost in terms of functions evaluations, this directional direct search phase should only be used in case of failure of the two previous steps. In practice, this means that the line search stepsize in the inexact Newton method is less than a predetermined (and sufficiently small) value $\lambda_{\min}$.

The computation of the new set of orthogonal poll directions resources to the scaled Householder transformation

$$H = \|q\|^2 (I - 2vv^\top), \tag{12}$$

where $q \in \mathbb{R}^n \setminus \{0\}$, I is the identity matrix and $v = \|q\|^{-1} q$. An appropriate definition of the vector q guarantees that one of the spectral directions (either $d_- = -(1/\alpha_k)F(x_k)$ or $d_+ = (1/\alpha_k)F(x_k)$) is a column of H. The union of the columns of H and $-H$ defines a positive basis for $\mathbb{R}^n$. This condition is necessary to establish the convergence of the algorithm.

One of the drawbacks of directional direct search is its inefficiency when solving large problems. As mentioned before, the main focus of this work is the solution of smooth large-scale nonlinear systems (1), when derivative information is not available for use. Thus, we propose a new globalization strategy to improve the practical performance of the algorithm. In this new strategy, which we call *multi-directional search*, the poll directions included in the positive basis are tested in pairs of symmetric directions. In case of failure in finding a better point, the stepsize is reduced, and the algorithm switches to a new pair of symmetric directions belonging to the positive basis. Instead, in classical directional direct search all poll directions should be tested before being able to reduce the stepsize parameter (see [4]).

The direct search procedure developed in this work is described in Algorithm 3.2. Steps 1 to 7 correspond to the computation of the orthogonal matrix H, where one of its columns corresponds to one of the spectral directions. Steps 8 to 12 correspond to the polling procedure, performed in pairs of symmetric poll directions. Similarly to [13], the condition

$$f(x_k + \lambda d_k) \leq (1 - \gamma\lambda^2) \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}),$$

is used for accepting new points.

Algorithm 3.2 *Multi-directional Search*

Input parameters: $x_k \in \mathbb{R}^n$, $\alpha_k > 0$, $\beta, \mu, \gamma \in (0,1)$, $a \in (0,1]$, $M \in \mathbb{N}$ *and* $0 < \xi_{\min} < \xi_{\max} < 1$.

1. *Choose* $i \in \{1, 2, \ldots, n\}$ *such that* $F_i(x_k) \neq 0$.
2. *If* $-(1/\alpha_k)F_i(x_k) > 0$, *set* $d = -(1/\alpha_k)F(x_k)$. *Else, set* $d = (1/\alpha_k)F(x_k)$.
3. *If* $d_i = \|d\|$, *set* $H = d_i I$ *and go to 8.*
4. *Set* $q_i = \sqrt{\frac{1}{2}(\|d\| - d_i)}$.
5. *For* $j = 1, 2, \ldots, n$ *with* $j \neq i$, *set* $q_j = \frac{-d_j}{2q_i}$.
6. *Compute* $v = \|q\|^{-1} q$.
7. *Compute* $H = [h_1\, h_2 \ldots h_n] = \|q\|^2 (I - 2vv^\top)$.
8. *Set* $r = mod(i, n) + 1$, $t = 0$ *and* $W_k = \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j})$.
9. $\lambda = a$
10. *(Line Search) While* $f(x_k + \lambda h_r) > (1 - \gamma\lambda^2)W_k$ *and* $f(x_k - \lambda h_r) > (1 - \gamma\lambda^2)W_k$, *do:*
    (a) *If* $\lambda \geq \mu a$ *choose* $\xi \in [\xi_{\min}, \xi_{\max}]$, *set* $\lambda = \xi\lambda$, $r_{marc} = r$ *and go to 10c.*
    (b) *Else:*
        i. *if* $r \neq r_{marc}$, *go to 10c ;*
        ii. *if* $r = r_{marc}$, *set* $\lambda = 0$, $\delta = \lambda$, $\mu = \beta\mu$, $t = t + 1$ *and go to 9.*
    (c) *Set* $r = mod(r, n) + 1$.
11. *If* $f(x_k + \lambda h_r) \leq (1 - \gamma\lambda^2)W_k$, *set* $d_k = h_r$. *Else, set* $d_k = -h_r$.
12. *Set* $\lambda_k = \lambda$. *Terminate.*

Remark 1 Variable δ and counter t are only defined for clarifying the theoretical proofs regarding the algorithmic convergence. In a practical implementation, both are unnecessary.

Remark 2 Matrix H do not need to be explicitly computed, neither stored. In each pairwise testing of poll directions, if $d_i = \|d\|$ then it is sufficient to set $h_r = d_i e_r$. Otherwise, the poll directions can be computed twice a time, whenever they are required.

Algorithm 3.3 describes the three-phase hybrid method, which we named **H3P**.

Algorithm 3.3 ***H3P***

Input parameters: $x_0 \in \mathbb{R}^n$, $0 < \lambda_{\min} \leq 1$, $NBL_{\max}, M \in \mathbb{N}$, $\beta, \gamma, \mu \in (0, 1)$, $0 < \tau_{\min} < \tau_{\max} < 1$, $0 < \xi_{\min} < \xi_{\max} < 1$, $0 < \alpha_{\min} < \alpha_{\max}$, $\{\zeta_k\}$, $\{\eta_k\}$ *and* $a \in (0, 1]$.

1. *Set* $k = 0$.

2. *Compute* $d_k, \lambda_k, flag$ *using Algorithm 2.1 (DFSANE).*

3. *If* $flag = 0$*, compute* $d_k, \lambda_k, flag$ *using Algorithm 3.1 (Inexact Newton I).*

4. *If* $flag = 0$*, compute* d_k, λ_k *using Algorithm 3.2 (Multi-directional Search).*

5. *Set* $x_{k+1} = x_k + \lambda_k d_k$ *and* $k = k + 1$.

6. *If* $F(x_{k+1}) = 0$*, terminate. Else, go to Step 2.*

3.1 Convergence Analysis

In this subsection, we analyze the convergence of Algorithm 3.3. Similarly to the approach of [13], we establish the convergence of a subsequence generated by the algorithm to a critical point of the merit function or the existence of a critical point of the merit function generated by the algorithm. As before, we will consider $f(x) = \left\|F(x)\right\|^2$ as merit function, we define $W_k = \max_{0\leq j\leq \min\{k,M-1\}} f(x_{k-j})$ and the sequence $\{\nu(k)\}$ such that $f(x_{\nu(k)}) = W_k$.

Lemma 3.1 is an auxiliary result for the desired convergence result.

Lemma 3.1 *Consider* $f : \mathbb{R}^n \to \mathbb{R}^+_0$, $M \in \mathbb{N}$, $W_k = \max_{0\leq j\leq \min\{k,M-1\}} f(x_{k-j})$, $\varrho \in \mathbb{R}$ *with* $\varrho > 0$ *and* $\{\zeta_k\}$ *a sequence in* $\mathbb{R}^+$ *such that* $\sum_{i=0}^{\infty} \zeta_i = \zeta < \infty$. *Then*

$$\lim_{k\to\infty} [\varrho f(x_k) - \zeta_k] = 0 \Rightarrow \lim_{k\to\infty} W_k = 0. \tag{13}$$

Proof Since

$$\lim_{k\to\infty} [\varrho f(x_k) - \zeta_k] = 0, \tag{14}$$

and by assumption $\lim_{k\to\infty} -\zeta_k = 0$, then $\lim_{k\to\infty} \varrho f(x_k) = 0$. Consequently,

$$\lim_{k\to\infty} f(x_k) = 0. \tag{15}$$

This result allows us to state that

$$\lim_{k\to\infty} f(x_k) = 0 \Rightarrow \lim_{k\to\infty} W_k = 0. \tag{16}$$

Indeed, by (15), for all $\varepsilon > 0$, there exists $\bar{k} \in \mathbb{N}$ such that, for all $k > \bar{k}$, $f(x_k) < \varepsilon$. Thus, for all $k > \bar{k} + M$ we have $W_k = f(x_{\nu(k)})$ with $\nu(k) > \bar{k}$. Therefore, $W_k < \varepsilon$, which allows to establish the result. ■

Proposition 3.1 is motivated by similar results in [11, 12, 13]. It provides necessary conditions to state Proposition 3.2 which, in turn, is used to establish the convergence result for the three-phase hybrid algorithm.

Proposition 3.1 *Let* $\{x_k\} \subset \mathbb{R}^n$ *be a sequence such that*

$$f(x_{k+1}) \leq W_k + \zeta_k, \tag{17}$$

with $\zeta_k > 0$ *for all* k *and* $\sum_{i=0}^{\infty} \zeta_i = \zeta < \infty$.

1. *Then for all k, $x_k \in \bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$.*

2. *Moreover, assuming the existence of $\bar{k} \in \mathbb{N}$ such that*

$$f(x_{k+1}) \leq W_k \text{ for all } k \geq \bar{k}, \tag{18}$$

the sequence $\{W_k\}_{k>\bar{k}}$ is monotonically non increasing.

Proof Since $\min\{k+1, M-1\} \leq \min\{k, M-1\} + 1$, by definition of $\nu(k)$, we have:

$$\begin{array}{rcl} W_{k+1} = f(x_{\nu(k+1)}) = & \max_{0\leq j\leq \min\{k+1,M-1\}} f(x_{k+1-j}) \\ \leq & \max_{0\leq j\leq \min\{k,M-1\}+1} f(x_{k+1-j}) \\ = & \max\{\max_{1\leq j\leq \min\{k,M-1\}+1} f(x_{k+1-j}), f(x_{k+1})\} \\ = & \max\{f(x_{\nu(k)}), f(x_{k+1})\}. \end{array} \tag{19}$$

By hypothesis, $f(x_{k+1}) \leq W_k + \zeta_k$ and $f(x_{\nu(k)}) = W_k \leq W_k + \zeta_k$. So, we conclude that $W_{k+1} \leq W_k + \zeta_k$ for all k, and an inductive argument allow us to state:

$$W_{k+1} = f(x_{\nu(k+1)}) \leq f(x_{\nu(0)}) + \sum_{i=0}^{k} \zeta_i \leq f(x_0) + \zeta. \tag{20}$$

Since $f(x_{k+1}) \leq W_{k+1}$, we have $x_k \in \bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$ for all k, and part 1. is established.

Part 2. is a direct consequence of the assumption regarding the existence of $\bar{k}$ sufficiently large such that $f(x_{k+1}) \leq f(x_{\nu(k)})$ for all $k \geq \bar{k}$ and of inequality (19). ∎

We are now in conditions of establishing Proposition 3.2.

Proposition 3.2 *Let $f : \mathbb{R}^n \to \mathbb{R}_0^+$ be a function and $\{x_k\} \subset \mathbb{R}^n$ a sequence such that:*

$$f(x_{k+1}) \leq W_k + \zeta_k - \varpi f(x_k), \tag{21}$$

where $\varpi > 0$, $\zeta_k > 0$ for all k and $\sum_{i=0}^{\infty} \zeta_i = \zeta < \infty$. Moreover, assume that for each $\varrho > 0$ there exists $\bar{k} \in \mathbb{N}$ such that for all $k > \bar{k}$ inequality

$$\zeta_k - \varrho f(x_k) < 0, \quad \textit{holds.} \tag{22}$$

Then $x_k \in \bar{\mathcal{L}}_0$ for all k and, furthermore,

$$\lim_{k\to\infty} f(x_k) = \lim_{k\to\infty} W_k = 0. \tag{23}$$

Proof Condition (21) implies that inequality (17) is satisfied for all k. Thus, Proposition 3.1 ensures that $x_k \in \bar{\mathcal{L}}_0$ for all k. Since condition (22) holds for $k > \bar{k}$, using again Proposition 3.1 we have that the non negative sequence $\{W_k\}_{k>\bar{k}}$ is monotonically non increasing. Thus, there exists a limit $W_* \geq 0$ for this sequence, when $k \to \infty$.

Reasoning by contradiction, we assume that $W_* \neq 0$. Thus, there exists $k_1 \in \mathbb{N}$ such that, for all $k > k_1$, we have $W_k > t_1 > 0$. By Lemma 3.1, there exists $k_2 \in \mathbb{N}$ such that $|-\zeta_k + \varpi f(x_k)| > t_2 > 0$ for all $k > k_2$.

Without loss of generality, we can assume that $k_2 \geq \bar{k}$. By hypothesis, we have that

$$f(x_{k+1}) \leq W_k - t_2,\ k > k_2 \geq \bar{k}. \tag{24}$$

Taking $k \geq k_2 + M + 1$, we have $\nu(k) - 1 > k - M - 1 \geq k_2$. Consequently, using definition of $\nu(k)$ and equation (24), it follows:

$$f(x_{\nu(k)}) \leq f(x_{\nu(\nu(k)-1)}) - t_2. \tag{25}$$

Taking limits in both sides and being that $f(x_{\nu(k)}) \to W_*$, we have $t_2 \leq 0$, which contradicts our assumption. ■

Proposition 3.3 is an adaptation of Proposition 3.1 of [13]. It allows to ensure that Step 10 of our multi-directional search is well defined, meaning that it can not iterate indefinitely.

Proposition 3.3 *Let $\mu \in (0,1)$ and $a \in (0,1]$ be fixed. Step 10 (Line Search) of Algorithm 3.2 determines, in a finite number of iterations, a direction d_k and a scalar $\lambda \in [0,a]$ such that:*

$$f(x_k + \lambda d_k) \leq (1 - \gamma\lambda^2) W_k. \tag{26}$$

Additionally, at least one of the following conditions holds:

1. *$\lambda = 0$ and*

$$\left\|F(x_k + \delta h_r)\right\|^2 > (1-\gamma\delta^2) W_k \geq (1-\gamma\delta^2)\left\|F(x_k)\right\|^2, \tag{27}$$

   for all $r = 1, 2, \ldots, 2n$ with $\delta < \mu a$ and h_r computed in Steps 1 to 7 of Algorithm 3.2;

2. *$\lambda \geq \xi_{\min}\mu a$.*

Proof Since λ is initially set equal to a and reduced, in each iteration of Step 10, by a factor $\xi \leq \xi_{\max} < 1$, Step 10 of Algorithm 3.2 ends on Step 10(b)ii with $\lambda = 0$ (thus trivially satisfying condition (26)) or it finds a nonzero value λ such that condition (26) holds.

If Step 10 of Algorithm 3.2 ends on Step 10(b)ii we have $\lambda = 0$ and condition (27) is satisfied. Else, either the initial stepsize is accepted (in this case $\lambda = a$) or a new stepsize λ is computed such that $\lambda/\xi \geq \mu a$. In both cases, $\lambda \geq \xi_{\min}\mu a$. ■

We are now in conditions of stating that Algorithm 3.2 is well defined, thus not continuously cycling between Step 9 and Step 10.

Proposition 3.4 *Let x_k be one input parameter of Algorithm 3.2 such that $\nabla f(x_k) \neq 0$ and f is continuously differentiable on a neighborhood of x_k. Then Algorithm 3.2 will succeed in computing a new iterate x_{k+1}, meaning that it can not indefinitely cycle between Steps 9 and 10.*

Proof By contradiction, let k be an iteration for which Algorithm 3.2 is not able to compute x_{k+1}, meaning that Step 10 ends unsuccessfully. Proposition 3.3 ensures the existence of values $\delta(t)$ such that:

$$\left\|F(x_k + \delta(t)h_r)\right\|^2 > (1 - \gamma\delta(t)^2)\left\|F(x_k)\right\|^2, \forall r \in \{1, \ldots, 2n\}, \tag{28}$$

with $0 < \delta(t) < \mu(t)a$, where $\mu(t) = \beta^t\mu_0$ and t is a counter, increased each time that the previous condition holds for a given $\mu(t)$. When $t \to \infty$, we have $\mu(t) \to 0$ and, hence, $\delta(t) \to 0$. Therefore, for a sufficiently large value t, the point $x_k + \delta(t)h_r$ is in a neighborhood of x_k such that f is continuously differentiable.

By the mean value theorem, we have:

$$f(x_k + \delta(t)h_r) = f(x_k) + \delta(t)\nabla f(x_k + b(t)\delta(t)h_r)^\top h_r \tag{29}$$

for some $b(t) \in (0, 1)$. Using inequality (28), we have:

$$f(x_k) + \delta(t)\nabla f(x_k + b(t)\delta(t)h_r)^\top h_r > f(x_k) - \gamma\delta(t)^2 f(x_k). \tag{30}$$

Thus, for sufficiently large values of t, we have:

$$\nabla f(x_k)^\top h_r \geq 0,\ r = 1, 2, \ldots, 2n. \tag{31}$$

Since $\{h_1, \ldots, h_{2n}\}$ is a positive basis, there exist values $\beta_r \geq 0$, $r = 1, \ldots, 2n$ such that:

$$-\nabla f(x_k) = \sum_{r=1}^{2n} \beta_r h_r. \tag{32}$$

Therefore,

$$-\left\|\nabla f(x_k)\right\|^2 = \sum_{r=1}^{2n} \beta_r \nabla f(x_k)^\top h_r \geq 0, \tag{33}$$

which implies $\nabla f(x_k) = 0$, thus arriving to a contradiction. ∎

We will now present Theorem 1 that establishes the convergence of Algorithm 3.3. As in [13], we consider that x_* is a critical point of the merit function $f(x) = \|F(x)\|^2$ if it satisfies: (i) $\|F(x_*)\|^2 = 0$ or (ii) $\|F(x_*)\|^2 > 0$ and $\nabla f(x_*) = 2J(x_*)^\top F(x_*) = 0$.

Theorem 1 *Let $\{x_k\}$ be the sequence of iterates generated by Algorithm 3.3 (H3P). Consider the sequence $\{\zeta_k\}$, required for accepting new points, defined as $\zeta_k > 0$ for all k and $\sum_{k=0}^{\infty}\zeta_k = \zeta < \infty$. Assume that the level set $\bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$ is bounded and let f be continuously differentiable in $\bar{\mathcal{L}}_0$. For each $\varrho > 0$, assume that there exists $\bar{k} \in \mathbb{N}$ such that for all $k > \bar{k}$*

$$\zeta_k - \varrho f(x_k) < 0. \tag{34}$$

Then, Algorithm 3.3 (H3P) ends at some point x_k that is a critical point of the merit function, or it is well defined and generates a sequence $\{x_k\}$ such that every limit point of it is a critical point of the merit function.

Proof Suppose that $\nabla f(x_k) \neq 0, \forall k$. Let us start by stating that Algorithm 3.3 (H3P) is well defined. Each point x_k generated by Algorithm 3.3 satisfies:

$$f(x_{k+1}) \leq W_k + \zeta_k. \tag{35}$$

Proposition 3.1 guarantees that, for all k, $x_k \in \bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$. Therefore, Proposition 3.4 ensures that Algorithm 3.2 can not cycle. Thus, Algorithm 3.3 is well defined.

We will now establish that every limit point of the sequence of iterates generated by Algorithm 3.3 is a critical point of the merit function. Reasoning by contradiction, suppose that there is an infinite subset $K \subseteq \mathbb{N}$ such that the subsequence $\{x_k\}_{k\in K}$ converges to a point $\bar{x}$ and $\bar{x}$ is not a critical point of the merit function.

For sufficiently large values of $k \in K$, we have

$$\big\|F(x_k)\big\| > \varepsilon, \tag{36}$$

for some $\varepsilon > 0$ and $\nabla f(\bar{x}) \neq 0$. Proposition 3.1 guarantees that, for all k, $x_k \in \bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$. Therefore, $\{W_k\}_{k\in K}$ is bounded and admits a limit point $W^* > 0$ (if $W^* = 0$ we would have a contradiction with (36)). Let $\{W_k\}_{k\in K'}$ and $K' \subseteq K$ be the corresponding sequence.

We will now prove that it is possible to obtain a value $\varpi > 0$ such that

$$\big\|F(x_{k+1})\big\|^2 \leq W_k + \zeta_k - \varpi\big\|F(x_k)\big\|^2, \forall k \in K'. \tag{37}$$

Since Algorithm 3.3 is well defined, for all $k \in K'$, a point x_{k+1} is generated such that one of the following conditions holds:

(a) x_{k+1} is computed at Step 2 by DFSANE method. In this case, condition (9) is satisfied with stepsize $\lambda_k \geq \tau_{\min}^{NBL_{\max}-1}$ and we have

$$f(x_{k+1}) \leq W_k + \zeta_k - \gamma(\tau_{\min}^{NBL_{\max}-1})^2 f(x_k); \tag{38}$$

(b) x_{k+1} is computed at Step 3 by the Inexact Newton method. In this case, there exists a value $\lambda_k \geq \lambda_{\min}$ that satisfies condition (9), thus

$$f(x_{k+1}) \leq W_k + \zeta_k - \gamma\lambda_{\min}^2 f(x_k); \tag{39}$$

(c) x_{k+1} is computed at Step 4 by the Multi-directional Search method.

Therefore, all that remains to prove is that inequality (37) also holds in case (c). Let $t_k \geq 0$ be the number of consecutive calls (starting from x_k) of Step 10 of Algorithm 3.2 before generating x_{k+1}. Two cases can occur:

(c_1) there is some value $\bar{t} > 0$ such that $t_k \leq \bar{t}$ for all $k \in K'$;

(c_2) there is some infinite subsequence $\{x_{k+1}\}_{k\in K''}$, with $K'' \subseteq K'$ such that every point x_{k+1} is obtained on Step 4 of Algorithm 3.3 and

$$\lim_{k\to\infty,\, k\in K''} t_k = \infty. \tag{40}$$

We will show that $\{t_k\}_{k\in K'}$ is bounded, meaning that (c_2) is not possible.

Again, reasoning by contradiction, suppose that (c_2) occurs. Condition (40) implies that, for $k \in K''$ sufficiently large, $\lambda = 0$ and Proposition 3.3 holds, for each h_r, a value δ_k, such that

$$\left\|F(x_k + \delta_k h_r)\right\|^2 > (1 - \gamma\delta_k^2)\left\|F(x_k)\right\|^2. \tag{41}$$

Using similar arguments to the ones of the well-definiteness proof (see equations (29), (30), (31), (32)) we have:

$$f(x_k + \delta_k h_r) = f(x_k) + \delta_k \nabla f(x_k + b_k\delta_k h_r)^\top h_r \tag{42}$$

for some value $b_k \in (0, 1)$. Thus:

$$f(x_k) + \delta_k \nabla f(x_k + b_k\delta_k h_r)^\top h_r > f(x_k) - \gamma\delta_k^2 f(x_k). \tag{43}$$

Now, considering μ_0 as the initialization of μ at each iteration k, we have $0 < \delta_k < \beta^{t_k}\mu_0 a$. Considering the limit $k \to \infty$, $k \in K''$, we have $\delta_k \to 0$ and, by inequality (43):

$$\nabla f(\bar{x})^\top h_r \geq 0,\ r = 1, 2, \ldots, 2n. \tag{44}$$

Since $\{h_1, \ldots, h_{2n}\}$ is a positive basis, a similar reasoning to the one of (32) and (33) implies that $\nabla f(\bar{x}) = 0$, contradicting the assumption of $\bar{x}$ not being a critical point. Thus, the subsequence $\{t_k\}_{k\in K'}$ must be bounded and case (c_2) does not occur.

Now, if x_k is obtained from the Multi-directional Search, according to the instructions of Algorithm 3.2, we have:

$$f(x_{k+1}) \leq (1 - \gamma\bar{\lambda}^2)W_k, \tag{45}$$

where $\bar{\lambda} \geq \beta^{\bar{t}}\mu_0 a$. Therefore,

$$f(x_{k+1}) \leq W_k + \zeta_k - \gamma\bar{\lambda}^2 f(x_k). \tag{46}$$

Taking $\varpi = \min\{\tau_{\min}^{NBL_{\max}-1}, \lambda_{\min}, \bar{\lambda}\}$, (38), (39) and (46), imply that condition (37) is satisfied for all $k \in K'$. So, from (37) and (34), the hypothesis of Proposition 3.2 are satisfied for all $k \geq \bar{k}$. Hence,

$$\lim_{k\to\infty,\, k\in K'} f(x_k) = \lim_{k\to\infty,\, k\in K'} W_k = 0, \tag{47}$$

contradicting $W^* > 0$ and completing the proof. ∎

4 A two-phase hybrid algorithm for systems of nonlinear equations

In the last section, we have proposed and stated the convergence of a three-phase hybrid algorithm (H3P). However, as has been already pointed out, the

Multi-directional Search step is too expensive and should be used with the single purpose of improving the algorithmic robustness. A single iteration of this step will decrease the algorithmic performance in terms of efficiency, in particular when applied to large-scale problems, as the number of function evaluations increases with the problem dimension.

Thus, it would be interesting to establish that only the two initial steps (DFSANE and Inexact Newton) are sufficient to ensure the convergence of the method, developing a two-phase hybrid algorithm (H2P). With this purpose, a few modifications are required in the line search of Algorithm 3.1. The new structure is proposed in Algorithm 4.1.

Algorithm 4.1 *Inexact Newton II*

Input parameters: $x_k \in \mathbb{R}^n$, $\mu, \gamma, \sigma, \theta_1, \theta_2, \theta_3 \in (0,1)$, $a \in (0,1]$, $\eta_k \in (0,1)$, $M \in \mathbb{N}$, $\zeta_k > 0$, *and* $0 < \xi_{\min} < \xi_{\max} < 1$.

1. *Set* $t = 0$ *and* $\eta = \eta_k$.
2. *Compute* d_k *satisfying (4), using* σ *in the matrix-vector products (10). Set* $\lambda = 1$.
3. *If* $f(x_k + \lambda d_k) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}) + \zeta_k - \gamma \lambda^2 f(x_k)$, *go to Step 8.*
4. *Set* $\alpha = a, t = t + 1$ *and* $i = 0$.
5. *While* $f(x_k + \alpha d_k) > W_k + \zeta_k - \gamma \alpha^2 f(x_k)$, *do:*
   (a) *If* $\alpha < \mu a$, *then* $\lambda = 0$, $\delta = \alpha$ *and go to Step 7.*
   (b) *Choose* $\xi \in [\xi_{\min}, \xi_{\max}]$ *and set* $\alpha = \alpha\, \xi$, $i = i + 1$.
6. *Set* $\lambda = \alpha$ *and go to Step 8.*
7. *Set* $\sigma = \theta_1 \sigma$, $\eta = \theta_2 \eta$, $\eta_k = \eta$, $\mu = \theta_3 \mu$ *and go to Step 2.*
8. *Set* $\tilde{\sigma}_k = \sigma$, $\tilde{\eta}_k = \eta$ *and* $\lambda_k = \lambda$.

Remark 3 In a practical implementation of the algorithm, variable δ, counter t and sequences $\{\tilde{\sigma}_k\}, \{\tilde{\eta}_k\}$ are not required. They are only defined to facilitate the presentation of the convergence analysis.

To prove that Algorithm 4.1 is well defined, first we need to show that a direction satisfying (4) can be computed at Step 2. Moreover, we need to ensure that the line search ends with a nonzero value for λ, meaning that Algorithm 4.1 does not cycle between Step 2 and Step 7. Both results depend on the following assumption:

Hypothesis 1 *Function* J *is Lipschitz continuous on a convex set* $\Omega \subseteq \mathbb{R}^n$, *with Lipschitz constant* $L_J > 0$. *Function* J *is also nonsingular on* Ω *and satisfies* $\left\|J^{-1}(y)\right\| \leq m_J$ *for all* $y \in \Omega$, *with* $m_J > 0$.

Proposition 4.1, stated in [13], follows from Proposition 6.2.1 in [14]. It ensures that it is possible to compute a direction satisfying the inexact Newton condition (4) using the FDGMRES method proposed in [14], which is a GMRES matrix free method.

Proposition 4.1 *Let $x_k \in \mathbb{R}^n$ be a point such that $F(x_k) \neq 0$. Assume that F satisfies Hypothesis 1 for a convex set Ω_k, such that $x_k \in \Omega_k$ with $L_J = L_k$ and $m_J = c_k$. Let*

$$\hat{\sigma}_k = \frac{1}{2n^{1/2}L_k c_k} \tag{48}$$

and

$$C_k = 4n^{1/2}L_k c_k. \tag{49}$$

Then, for each $\sigma \in (0, \hat{\sigma}_k]$ and for each $\eta_k \in (0,1)$, procedure FDGMRES determines a direction d_k satisfying

$$\big\|J(x_k)d_k + F(x_k)\big\| \leq (\eta_k + C_k\sigma)\big\|F(x_k)\big\|. \tag{50}$$

By adjusting Lema 8.2.1 in [14], Grippo and Sciandrone [13] established Proposition 4.2, which guarantees that Hypothesis 1 is sufficient to ensure that Algorithm 4.1 does not cycle between Step 2 and Step 7.

Proposition 4.2 *Let $x \in \mathbb{R}^n$ be a point such that $F(x) \neq 0$ and satisfies Hypothesis 1 for some set $\Omega = \{y \in \mathbb{R}^n \mid \|x - y\| \leq r\}$, with $r > 0$. Let $d \in \mathbb{R}^n$ be a vector satisfying the Inexact Newton Condition:*

$$\big\|J(x)d + F(x)\big\| \leq \eta\big\|F(x)\big\| \tag{51}$$

with $0 \leq \eta \leq \bar{\eta} < (1-\gamma)$ and $\gamma \in (0,1)$. Then, we have

$$\big\|F(x + \lambda d)\big\| \leq (1 - \gamma\lambda)\big\|F(x)\big\| \tag{52}$$

with $\lambda \in [0, \bar{\lambda}(x)]$, where

$$\bar{\lambda}(x) = \min\left(\frac{r}{m_J(1+\bar{\eta})\big\|F(x)\big\|}, \frac{2(1-\gamma-\bar{\eta})}{(1+\bar{\eta})^2 m_J^2 L_J\big\|F(x)\big\|}\right). \tag{53}$$

Proposition 4.2 holds a result similar to condition (9), used for accepting new points. In fact, if we consider the merit function $f(x) = \big\|F(x)\big\|^2$ and λ in the interval $[0,1]$, we have:

$$\begin{aligned}\big\|F(x_k + \lambda d)\big\|^2 &\leq (1-\lambda\gamma)^2\big\|F(x_k)\big\|^2 \leq (1-\lambda\gamma)\big\|F(x_k)\big\|^2 \\ &\leq \big\|F(x_k)\big\|^2 - \lambda^2\gamma\big\|F(x_k)\big\|^2 < W_k + \zeta_k - \gamma\lambda^2\big\|F(x_k)\big\|^2.\end{aligned} \tag{54}$$

In the line search procedure of Algorithm 4.1, between Step 2 and Step 7, we considered the acceptance condition (9) proposed in [5]. This condition is still nonmonotone, but differs from the one used in the Multi-directional Search step. Proposition 3.3 will be adapted to the new acceptance condition, ensuring that Step 5 is finite. We omit the proof since it is similar to the one of Proposition 3.3.

Proposition 4.3 *Let $\mu \in (0,1)$ and $a \in (0,1]$ be fixed. Step 5 (Line Search) of Algorithm 4.1 determines, in a finite number of iterations, a scalar $\lambda \in [0,a]$ such that:*

$$f(x_k + \lambda d_k) \leq W_k + \zeta_k - \gamma\lambda^2 f(x_k). \tag{55}$$

Additionally, at least one of the following conditions holds:

1. $\lambda = 0$ *and*

$$\left\|F(x_k+\delta d_k)\right\|^2 > W_k+\zeta_k-\gamma\delta^2\left\|F(x_k)\right\|^2 \geq (1-\gamma\delta^2)\left\|F(x_k)\right\|^2+\zeta_k, \tag{56}$$

   with $\delta < \mu a$; or

2. $\lambda \geq \xi_{\min}\mu a$.

Algorithm 4.2 **H2P** corresponds to the two-phase hybrid algorithm.

Algorithm 4.2 ***H2P***

Input parameters: $x_0 \in \mathbb{R}^n$, $NBL_{\max}, M \in \mathbb{N}$, $\gamma, \mu, \sigma, \theta_1, \theta_2, \theta_3 \in (0,1)$, $0 < \tau_{\min} < \tau_{\max} < 1$, $0 < \xi_{\min} < \xi_{\max} < 1$, $0 < \alpha_{\min} < \alpha_{\max}$, $\{\zeta_k\}$, $\{\eta_k\}$, *and* $a \in (0,1]$.

1. *Set* $k = 0$.
2. *Compute* $d_k, \lambda_k, flag$ *using Algorithm 2.1 (DFSANE).*
3. *If* $flag = 0$, *compute* d_k, λ_k *using Algorithm 4.1 (Inexact Newton II).*
4. *Set* $x_{k+1} = x_k + \lambda_k d_k$ *and* $k = k + 1$.
5. *If* $F(x_{k+1}) = 0$, *terminate. Else go to Step 2.*

4.1 Convergence Analysis

The following theorem establishes that Algorithm 4.2 (H2P) is well defined and states the corresponding convergence.

Theorem 2 *Let $\{x_k\}$ be the sequence of iterates generated by Algorithm 4.2 (H2P). Consider the sequence $\{\zeta_k\}$, required for accepting new points, defined as $\zeta_k > 0$ for all k and $\sum_{k=0}^{\infty}\zeta_k = \zeta < \infty$ and define $\bar{\mathcal{L}}_0 = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0) + \zeta\}$. Suppose that there exists $r > 0$ such that, for all $x \in \bar{\mathcal{L}}_0$, the closed ball $\bar{B}(x,r)$ is contained on an open convex set Ω where Hypothesis 1 is satisfied. For each $\varrho > 0$, assume that there exists $\bar{k} \in \mathbb{N}$ such that for all $k > \bar{k}$*

$$\zeta_k - \varrho f(x_k) < 0. \tag{57}$$

Then, Algorithm 4.2 (H2P) ends at some point x_k satisfying $F(x_k) = 0$, or it is well defined and generates a sequence $\{x_k\}$ such that

$$\lim_{k\to\infty} F(x_k) = 0. \tag{58}$$

Proof Consider $F(x_k) \neq 0, \forall k$. Let us start by showing that Algorithm 4.2 is well defined, meaning that at each iteration Algorithm 4.2 will generate a stepsize parameter $\lambda_k > 0$.

Reasoning by contradiction, suppose not. In Algorithm 4.1, the line search occurs between Step 2 and Step 7, where the values σ, η and μ are reduced. At iteration k, let $\{t_l\}$ be the sequence used to count at Step 4 the current number of calls of the line search. We have $t_l \to \infty$ and sequences $\{\sigma_l\}$, $\{\eta_l\}$ and $\{\mu_l\}$ converge to zero. Otherwise, Algorithm 4.2 would have computed $\lambda_k > 0$.

Proposition 3.1 guarantees that $x_k \in \bar{\mathcal{L}}_0$ for all $k \in \mathbb{N}$. Consider L_J and m_J, the constants of Hypothesis 1 associated with the convex set Ω. Define $\hat{\sigma} = \frac{1}{2n^{1/2}L_J m_J}$ and $C = 4n^{1/2}L_J m_J$. For l sufficiently large $\eta_l \in (0,1)$ and $0 < \sigma_l \leq \hat{\sigma}$. Thus, Proposition 4.1 guarantees that procedure FDGMRES computes a direction d_l such that

$$\big\|J(x_k)d_l + F(x_k)\big\| \leq (\eta_l + C\sigma_l)\big\|F(x_k)\big\| \leq \bar{\eta}\big\|F(x_k)\big\| < (1-\gamma)\big\|F(x_k)\big\|, \tag{59}$$

with $\eta_l + C\sigma_l \leq \bar{\eta} < (1-\gamma)$.

By applying Proposition 4.2 we can conclude that

$$\|F(x_k + \lambda d_l)\| \leq (1 - \gamma\lambda)\|F(x_k)\|,$$

for $\lambda \in [0, \bar{\lambda}(x_k)]$ and $\bar{\lambda}(x_k)$ defined as in (53) with $F(x) = F(x_k)$.

We note that $a \in (0,1]$ and $\xi \in [\xi_{\min}, \xi_{\max}]$, resulting $\xi_{\min}^i a \leq \alpha_i \leq \xi_{\max}^i$. Therefore, it is possible choose i_* that satisfies

$$i_* \geq \max\left\{0, \frac{\log(\bar{\lambda}(x_k))}{\log(\xi_{\max})}\right\} \tag{60}$$

ensuring $0 < \alpha(i_*) \leq \bar{\lambda}(x_k)$. Since $\mu_l \to 0$, for l sufficiently large we have $\mu_l a \leq \xi_{\min}^{i_*} a \leq \alpha(i_*)$. Thus,

$$\|F(x_k + \alpha(i_*)d_l)\| \leq (1 - \gamma\alpha(i_*))\|F(x_k)\|$$

and, since $\alpha(i_*) \in (0,1]$, the acceptance condition

$$f(x_k + \alpha(i_*)d_l) \leq W_k + \zeta_k - \gamma\alpha(i_*)^2 f(x_k)$$

will be satisfied. Iteration k of Algorithm 4.1 ends, with a positive value for λ_k, ensuring that Algorithm 3.3 is well defined.

We will now prove that the sequence $\{\lambda_k\}$ is lower bounded by a constant $\varpi > 0$. Once more, reasoning by contradiction, let us suppose that there is $K \subseteq \mathbb{N}$ such that

$$\lim_{k \in K,\, k \to \infty} \lambda_k = 0. \tag{61}$$

In Algorithm 2.1 DFSANE, for all k we have $\lambda_k \geq \tau_{\min}^{NBL_{\max}-1} > 0$. Thus, for $k \in K$ sufficiently large, λ_k is generated by the Inexact Newton Algorithm 4.1. The line search procedure of Algorithm 4.1 (between Step 2 and Step 7) then implies that $\tilde{\sigma}_k \to 0$ and $\tilde{\eta}_k \to 0$, when $k \in K, k \to \infty$.

Again, let L_J and m_J be the constants of Hypothesis 1 associated with the convex set Ω. Define $\hat{\sigma} = \frac{1}{2n^{1/2}L_J m_J}$ and $C = 4n^{1/2}L_J m_J$. For $k \in K$ sufficiently large $\tilde{\eta}_k \in (0,1)$ and $0 < \tilde{\sigma}_k \leq \hat{\sigma}$. Thus, Proposition 4.1 guarantees that procedure FDGMRES computes a direction d_k such that

$$\big\|J(x_k)d_k + F(x_k)\big\| \leq (\tilde{\eta}_k + C\tilde{\sigma}_k)\big\|F(x_k)\big\| \leq \bar{\eta}\big\|F(x_k)\big\| < (1-\gamma)\big\|F(x_k)\big\| \tag{62}$$

with $\tilde{\eta}_k + C\tilde{\sigma}_k \leq \bar{\eta} < (1-\gamma)$.

Proposition 4.2 then establishes that

$$\|F(x_k + \lambda d_k)\| \leq (1 - \gamma\lambda)\|F(x_k)\|,$$

for $\lambda \in [0, \bar{\lambda}(x_k)]$ and $\bar{\lambda}(x_k)$ defined as in (53), with $F(x) = F(x_k)$. For $\lambda \in [0,1]$, the previous condition implies

$$f(x_k + \lambda d_k) \leq W_k + \zeta_k - \gamma\lambda^2 f(x_k).$$

Since $x_k \in \bar{\mathcal{L}}_0$, we have $f(x_k) \leq f(x_0) + \zeta$ and $\big\|F(x_k)\big\|^2 \leq \big\|F(x_0)\big\|^2 + \zeta$. By setting $b = \sqrt{2\max\{\big\|F(x_0)\big\|^2, \zeta\}}$ we can conclude that $\big\|F(x_k)\big\| \leq b$ for all $k \in K$. It is now possible to define a lower bound for $\bar{\lambda}(x_k)$, considering

$$\omega = \min\left(\frac{r}{m_J(1+\bar{\eta})b}, \frac{2(1-\gamma-\bar{\eta})}{(1+\bar{\eta})^2 m_J^2 L_J b}\right) \leq \bar{\lambda}(x_k), \text{ for all } k \in K. \tag{63}$$

Since $\lambda_k \to 0$ for $k \in K$, for $k \in K$ sufficiently large $\lambda_k < a \leq 1$. In this case, λ_k was computed in the line search procedure of Step 5, and satisfies $\lambda_k \geq \xi_{\min}\bar{\lambda}(x_k)$, since $\lambda_k \neq 0$.

Thus, for all $k \in K$ we have

$$\lambda_k \geq \min\{a, \xi_{\min}\bar{\lambda}(x_k)\} \geq \min\{a, \xi_{\min}\omega\}, \tag{64}$$

which contradicts limit (61). Thus, $\{\lambda_k\}$ is lower bounded by some $\varpi > 0$.

Consequently, for all k, we have

$$\big\|F(x_k + \lambda_k d_k)\big\|^2 \leq W_k + \zeta_k - \varpi\big\|F(x_k)\big\|^2. \tag{65}$$

For k sufficiently large, condition (57) holds. Thus, the assumptions of Proposition 3.2 are satisfied for $f(x_k) = \big\|F(x_k)\big\|^2$, resulting

$$\lim_{k\to\infty} \big\|F(x_k)\big\| = 0. \tag{66}$$ ∎

Remark 4 Before ending this section, we would like to point out that condition (57) can be easily satisfied, for example, by defining $\zeta_k = \frac{\min\{f(x_0), f(x_k)\}}{(k+1)^{1.1}}$. Note that, in this case, we also have $\zeta_k > 0$ for all k and $\sum_{k=0}^{\infty}\zeta_k = \zeta < \infty$.

5 Numerical experiments

This numerical section has two main goals: to evaluate the contribution of hybrid versions to improve the practical performance of algorithms and to access the contribution of direct search to increase the robustness of the results. For achieving the first goal, we compared implementations of the Hybrid Algorithm 4.2 (`H2P`), the original version of `DFSANE` [5] and a derivative-free version of an inexact Newton method, based on Algorithm NM1 proposed in [13] (without considering the watchdog procedure). For accessing the value of direct search, an implementation of algorithm `H3P` was compared with algorithm `H2P`. All the codes were implemented in Matlab 7.0 and tested on a Intel(R) Core I3-2100 personal computer, with 3.10 GHz and 4 Gb RAM.

In the hybrid algorithms, whenever possible, we consider the original settings proposed in [5] and [13], for DFSANE and the inexact Newton method, respectively. Thus, the stepsize in DFSANE is reduced using a quadratic interpolation procedure, considering the identity matrix as an approximation to the Jacobian. As example, for computing a new value λ_+ on Step 5 of Algorithm 2.1, define

$$\begin{array}{l} \varphi : [0, \lambda_+] \to \mathbb{R} \\ \varphi(\lambda) = f(x_k - \lambda(1/\alpha_k)F(x_k)). \end{array} \tag{67}$$

The minimizer λ_{new} of (67) is computed and used to define the new stepsize λ_+, considering the safeguards

$$\lambda_+ = \begin{cases} \tau_{\min}\lambda_+, & \text{if } \lambda_{new} < \tau_{\min}\lambda_+ \\ \tau_{\max}\lambda_+, & \text{if } \lambda_{new} > \tau_{\max}\lambda_+ \\ \lambda_{new}, & \text{otherwise.} \end{cases} \tag{68}$$

The safeguards were set to $\tau_{\min} = 0.1$ and $\tau_{\max} = 0.5$.

Furthermore, we considered $\alpha_0 = 1$, $\gamma = 10^{-4}$, $M = 7$ and, to define α_k, we use equation (6), if $\alpha_k \in [10^{-10}, 10^{10}]$. Otherwise, a new α_k is computed, depending on $F(x_k)$, as:

$$\alpha_k = \begin{cases} 1, & \text{if } \left\|F(x_k)\right\| > 1, \\ \left\|F(x_k)\right\|, & \text{if } 10^{-5} \leq \left\|F(x_k)\right\| \leq 1, \\ 10^{-5}, & \text{if } \left\|F(x_k)\right\| < 10^{-5}. \end{cases} \tag{69}$$

Regarding the settings of [5], the only modification is the definition of $\zeta_k = \frac{\min\{f(x_0), f(x_k)\}}{(k+1)^{1.1}}$, to satisfy the Hypothesis of Theorem 2.

In the inexact Newton step, following [14], we use the GMRES(m). This strategy restarts the GMRES method after one cycle of m iterations, where m is a predefined integer. At the end of each cycle, the last direction d_m (if not satisfactory) is used as initialization for the new cycle of m iterations. This strategy intends to reduce the memory requirements and the computational cost, associated with the increase in the number of iterations of GMRES, which is responsible for its inefficiency when used for solving larger problems. By adopting GMRES(m), we do not guarantee the conditions required to establish

the theoretical properties of the method. Nevertheless, due to its good practical performance, some authors also adopt this procedure [13].

We set the maximum number of GMRES iterations $m = 30$ and the maximum number of GMRES cycles $nc_{\max} = 30$. Differently from Algorithm NM1 [13], where a sequence of constant forcing terms is used, we adopt the sequence proposed by Eisenstat and Walker [9] both in our Hybrid Algorithm and in the implementation of the inexact Newton method (based on `NM1`) :

$$\eta_k = \gamma \left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|} \right)^{\alpha}, \tag{70}$$

with $\gamma = 1$ and $\alpha = 0.5(1+\sqrt{5})$ (in this last case we use a safeguard $[10^{-6}, 10^{-2}]$).

As in [5], we consider the stopping criterion:

$$\frac{\|F(x_k)\|}{\sqrt{n}} \leq \varepsilon_a + \frac{\varepsilon_r \|F(x_0)\|}{\sqrt{n}}, \tag{71}$$

where $\varepsilon_a = 10^{-5}$ and $\varepsilon_r = 10^{-4}$.

Moreover, all algorithms will end with a failure ($flag = 0$) in the following conditions:

FII If the number of inner GMRES(m) iterations equals or exceeds $nc_{\max} = 30$ cycles of $m = 30$ iterations;

FST If the stepsize is equal or lower than 10^{-12};

FFE If the total number of function evaluations equals or exceeds 10000;

FOU *Overflow* or *underflow* cases.

If condition (71) is satisfied, then the algorithms end with a success (denoted by **S**, when reporting the numerical results).

Two versions of the Hybrid Algorithm 4.2 (H2P) were implemented. In the first, named as `H2P1`, the line search procedure is not performed in the DFSANE method ($NBL_{\max} = 0$). In the second version, named as `H2P6`, the stepsize accepts a maximum of five reductions ($NBL_{\max} = 5$). DFSANE implementation is named as `DFSANE` and the derivative-free implementation of the inexact Newton method is named as `NI`.

All algorithms were tested in two sets of large scale problems, one collected from La Cruz and Raydan [6] (problems 1-20) and another collected from Section 4 of Lukšan and Vlček [17] (problems 1-21).

Problems collected from [6], were solved for dimensions $n = 100$, 500, 1000, 2000 and 5000, except problems 4, 7 and 18, for which we have considered $n = 99$, 498, 999, 1998 and 4998.

For each problem, 10 initializations were uniformly randomly generated in a neighborhood of the initial points proposed in [6]. Let $x_0 = (x_1, x_2, \ldots, x_n)^\top$ be

the proposed initial point for a given problem. The i^{th}-component of the new initialization is uniformly randomly generated in the interval

$$[x_i + \min\{-5, -5|x_i|\}, x_i + \max\{5, 5|x_i|\}].$$

Additionally, 10 other initializations were randomly generated for each problem, considering a normal distribution. The i^{th}-component was generated using x_i as mean and $\max\{5, 5|x_i|\}$ as standard deviation.

The numerical results in this first set of problems are reported in Table 1 and in the performance profiles [8] of Figure 1. The two left plots are more adequate to analyze efficiency, since $\tau \in [1, 2]$. The right graphs are more suitable to evaluate robustness, since $\tau \in [1, 10]$.

	FII	**FST** ($\lambda < 10^{-12}$)	**FFE** (max.10000)	**FOU**	**S**
H2P1	54.5%	0%	0%	6.4%	39.1%
DFSANE	—	1.5%	44.7%	3.1%	50.8%
H2P6	39.9%	0.4%	3.1%	3.4%	53.2%
NI	57.2%	0%	0%	3.6%	39.2%

Table 1: Performance on the first test set - Problems 1-20 in [6], considering random initializations.

DFSANE method is characterized by carrying out many low cost iterations, when compared with inexact Newton methods. Additionally, in the class of problems suited for derivative-free optimization, function evaluation is mainly responsible for the computational cost. So, we considered that the number of iterations is not an adequate indicator of efficiency, using the number of function evaluations as the main performance measure. Computational time was used as a secondary criterion.

Problems 1-21 from [17] were initially tested for the initialization suggested and a dimension $n = 5000$, except Problem 5 for which $n = 4999$. Since some of these initial points were close to the problem solution, we decided to generate a new set of initializations, increasing the distance from the problem solution. In this new test we have only included problems for which at least one of the solvers tested had succeeded with the initialization proposed in [17]. As result, problems 1, 2, 5 and 10 were excluded.

Twelve new initializations were generated for each selected problem. With this purpose, we defined the vector $d_t = x_0 - \bar{x}$, where x_0 is the initial point reported in [17] and $\bar{x}$ is an approximation to the problem solution computed in the initial numerical test. Additionally, we generated a random direction d_{rand} such that the cosine of angle between d_t and d_{rand} is lower than 0.95. A new initialization is defined as $x_0^{new} = x_0 + \omega d_t + \upsilon d_{rand}$, where the parameter ω takes the values 1, 20 and 200 and υ is set equal to 0, 1, 20 and 200, in a total of 12 combinations.

Table 2 and the performance profiles of Figure 2 correspond to the results in this new set of problems.

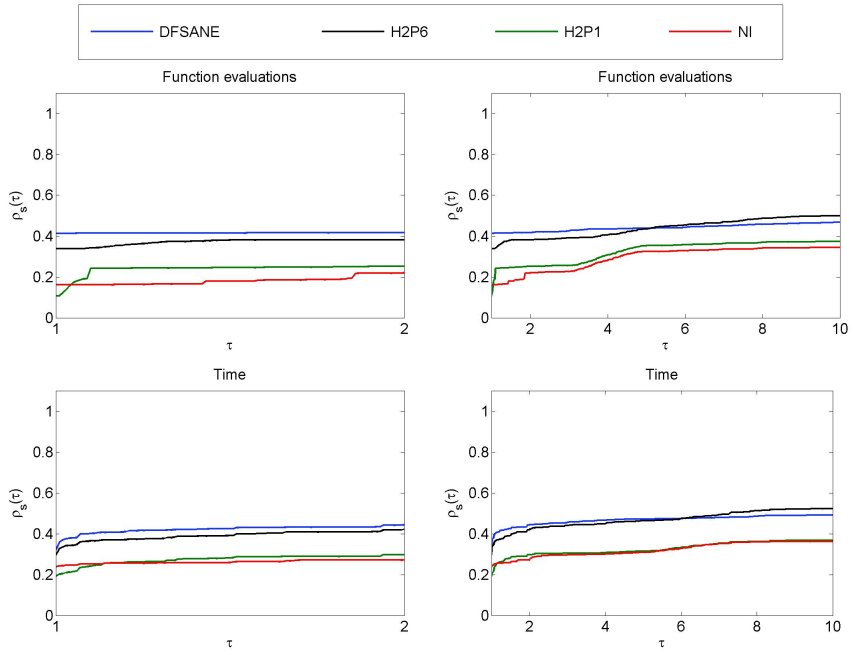

Figure 1: Performance profiles - Problems 1-20 in [6], considering random initializations.

In the two test sets, the hybrid algorithm, particularly the version `H2P6`, presented a good performance both in terms of efficiency and robustness. In the problems collected from [6], algorithm `H2P6` was the most robust and the second more efficient. For this test set, `DFSANE` presented the best performance in terms of efficiency. In this test set, algorithms which tend to often use spectral directions presented a superior performance, which may indicate that some characteristics of the problems favor algorithms enhanced with this type of directions.

In the second test set, where problems were selected from [17], algorithm `H2P6` was the most robust and the most efficient. Contrary to the previous results, `DFSANE` presents the worst performance. Although this fact indicates some unsuitability of spectral methods to solve this test set of problems, the good performance of `H2P6` reinforces the advantage of hybrid algorithms.

Due to the variability on the type of directions considered in each step, it was expected that the use of a hybrid strategy brought advantages in what respects to robustness, when compared with pure methods. However, the additional good performance obtained in terms of efficiency validates the benefit of using hybrid algorithms for solving large scale nonlinear systems of equations.

Aiming to test the efficacy of adding a direct search step in the hybrid algorithm to increase robustness, Algorithm 3.2 was implemented as an addi-

	FII	**FST** ($\lambda < 10^{-12}$)	**FFE** (max.10000)	**FOU**	**S**
H2P1	35.01%	1.11%	2.22%	4.44%	57.22%
DFSANE	0.00%	1.67%	41.67%	3.33%	53.33%
H2P6	29.44%	1.11%	3.33%	4.44%	61.67%
NI	34.44%	1.11%	1.11%	3.33%	60.01%

Table 2: Performance on the second test set - Selected problems from [17], considering random initializations.

tional step in algorithms DFSANE, H2P1 and H2P6, giving rise, respectively, to algorithms DFSANE-BD, H3P1 and H3P6 (the last two, following the algorithmic description of Algorithm 3.3).

With the purpose of increasing robustness, a new algorithmic variant was defined, which allows extrapolation steps in the direct search step for successful directions. If the direct search step is successful in one direction, the step size is consecutively doubled and the same direction is tested with the new step size parameter until failure. In this case, the new iterate corresponds to the last point successfully tested. This variant was implemented in DFSANE-BD(EXT), H3P1(EXT) and H3P6(EXT), based on DFSANE-BD, H3P1 and H3P6, respectively.

To accept new points in the direct search step, according to the algorithmic description of Step 10 of Algorithm 3.2, we did not use the condition (9). Instead, we adopted condition (11), proposed by Grippo and Sciandrone in [13]. In this case, we set $\gamma = 10^{-4}$ and $M = 7$. In multi-directional search, we considered $\xi = 0.95$, $a = 1$, and defined $\mu = 10^{-10}$. Algorithm 3.3 ends in Step 10 of Algorithm 3.2 if multi-directional search returns $\lambda = 0$. This way, the implementation enables Step 10b only once, so it is unnecessary to define a value for β.

From problems 1-21 in [17], considering $n = 100$, we selected the ones for which the line search failed in FDGMRES in at least one of algorithms DFSANE, H2P1 or H2P6, due to the large number of inner iterations. Problems 1, 2, 5, 6, 10, 19 and 20 are part of the new test set.

Algorithm H3P1 benefits the most with the addition of the direct search step. The two new versions of this algorithm were able to solve problem 19, and problem 20 was solved when the extrapolation strategy was adopted. In the remaining problems, although convergence was not achieved, there was a reduction in the residual value $\|F(x_k)\|$ computed for the best point found. For all other algorithms, there were not significative performance improvements.

Table 3 reports a resume of the numerical results obtained in this test set. Column **P** corresponds to problem number, column **Final** records the output flag, column **Iter** indicates the number of iterations, and column **EvalF** indicates the total number of function evaluations.

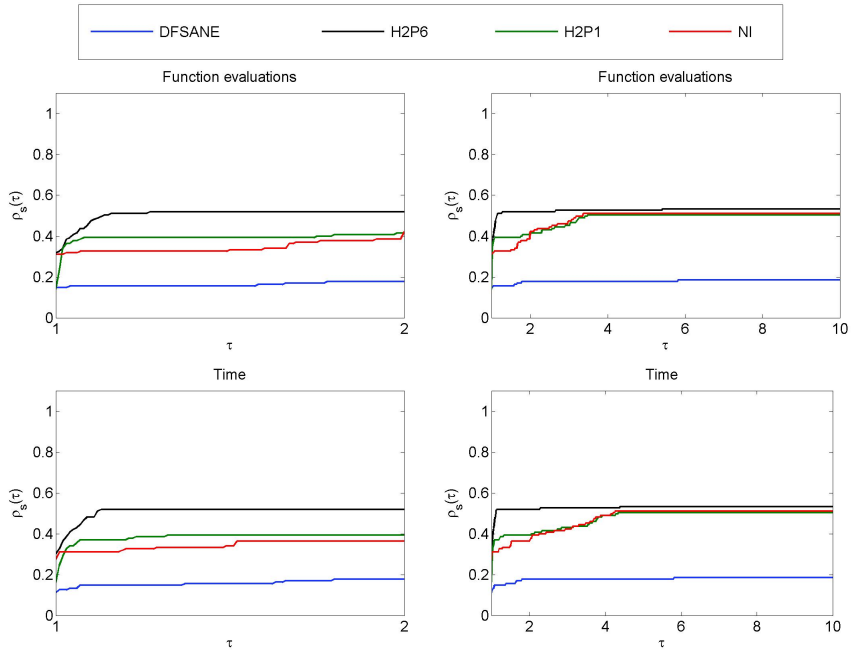

Figure 2: Performance profiles - Selected problems from [17], considering random initializations.

6 Final remarks

In this work, we proposed a hybrid approach to address the resolution of large scale systems of nonlinear equations, in the situation where derivatives are not available for use. Two algorithms were developed and analyzed: the two-steps Algorithm 4.2 (H2P), combining the Spectral Residual Method ([6, 5]) and the inexact Newton method; and the three-steps Algorithm 3.3 (H3P), that considers an additional direct search step, in case of failure of the two previous steps.

Theorem 2 is the main theoretical contribution of this work, ensuring that the sequence of iterates generated by Algorithm 4.2 (H2P) determines at least one accumulation point that is a solution of the nonlinear system. The second theoretical contribution of this work is Theorem 1, which guarantees the convergence of Algorithm 3.3.

The result stated in Theorem 1 is similar to Proposition 6.1 in [13]. Although, it introduces an innovation, replacing the globalization strategy used in [13] by the multi-directional search of Algorithm 3.2. This multi-directional search is advantageous when solving large scale problems, in the sense that it changes direction while reducing the stepsize, instead of keeping the same direction until reaching the minimum step size allowed, as is the case of [13]. With

	H2P1			H3P1			H3P1(EXT)		
P	**Final**	**Iter**	**EvalF**	**Final**	**Iter**	**EvalF**	**Final**	**Iter**	**EvalF**
1	FII	4	110	FFE	193	10001	FFE	465	10001
2	FII	1	48	FFE	176	10001	FST	188	7326
5	FII	0	34	FFE	49	10001	FOU	2	213
6	S	6	41	S	6	41	S	6	41
10	FII	1	35	FFE	190	10001	FFE	583	10001
19	FII	2	36	S	188	4084	S	188	4115
20	FII	4	34	FFE	89	10001	S	24	591

Table 3: Performance of the Multi-directional Search step.

this strategy, we want to avoid the persistency in bad directions, which will necessary increase the number of function evaluations.

Additionally, Algorithm 3.3 proposes a new orthogonal poll set in which both directions used in the spectral step are included. Nevertheless, convergence is not affected if any other positive spanning set is considered.

We highlight that, in both algorithms H2P and H3P, theoretical convergence only depends on the last step. In this sense, the initial steps (one in H2P and two in H3P) are designed only to facilitate the practical understanding of the algorithms. The theoretical results extend to methods in which the initial steps do not exist or differ from those proposed in this work. For example, the results presented for algorithm H2P can be extended to pure inexact Newton methods, considering as acceptance criteria (9), which is more flexible than the one used in [10].

From the numerical point of view, according to our test sets, Algorithm 4.2 is competitive when compared with a pure version of an inexact Newton method and DFSANE. Algorithm 3.3 was implemented in order to increase the robustness of Algorithm 4.2 and succeeded.

Acknowledgments

We are thankful to Sandra Augusta Santos, from University of Campinas, Brazil, for suggestions and valuable discussions on the subject, which greatly helped us to improve the paper.

References

[1] L. Armijo, *Minimization of functions having Lipschitz continuous first partial derivatives*, Pacific Journal of Mathematics, 16, pp. 1-3, 1966.

[2] J. Barzilai & J. M. Borwein, *Two-point step size gradient methods*, IMA J. Numer. Anal., 8, pp. 141-148, 1988.

[3] R. M. Chamberlain, M. J. D. Powell, C. Lemarechal & H. C. Pedersen, The watchdog technique for forcing convergence in algorithms for constrained optimization, In: Buckley A.G., Goffin J.L. (eds), *Algorithms for Constrained Minimization of Smooth Nonlinear Functions*, Mathematical Programming Studies, 16, pp. 1-17, Springer Berlin Heidelberg, Berlin, Heidelberg, 1982.

[4] A. R. Conn, K. Scheinberg & L. N. Vicente, *Introduction to Derivative-Free Optimization*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2009.

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

[6] W. La Cruz & M. Raydan, *Nonmonotone spectral methods for large-scale nonlinear systems*, Optimization Methods and Software, 18, pp. 583-599, 2003.

[7] R. S. Dembo, S. C. Eisenstat & T. Steihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19, pp. 400-408, 1982.

[8] E. D. Dolan & J. J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming, 91, pp. 201-213, 2002.

[9] S. C. Eisenstat & H. F. Walker, *Choosing the forcing terms in an inexact-Newton method*, SIAM Journal on Scientific Computing, 17, pp. 16-32, 1996.

[10] L. Grippo, F. Lampariello & S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23, pp. 707-716, 1986.

[11] L. Grippo & M. Sciandrone, *Nonmonotone globalization techniques for the Barzilai-Borwein gradient method*, Computational Optimization and Applications, 23, pp. 143-169, 2002.

[12] L. Grippo & M. Sciandrone, *Nonmonotone derivative-free methods for nonlinear equations*, Computational Optimization and Applications, 37, pp. 297-328, 2007.

[13] L. Grippo & M. Sciandrone, *Nonmonotone globalization of the finite-difference Newton-GMRES method for nonlinear equations*, Optimization Methods and Software, 25, pp. 971-999, 2010.

[14] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.

[15] T. G. Kolda, R. M. Lewis & V. Torczon, *Optimization by direct search: new perspectives on some classical and modern methods*, SIAM Review, 45, pp. 385-482, 2003.

[16] D.H. Li & M. Fukushima, *A derivative-free line search and global convergence of Broyden-like method for nonlinear equations*, Optim. Methods and Software, 13, pp. 181-201, 2000.

[17] L. Lukšan, J. Vlček, *Sparse and partially separable test problems for unconstrained and equality constrained optimization*, Report V-767, Prague, ICS AS CR, 1998.

[18] Y. Saad, & M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7, pp. 856-869, 1986.