RAIRO-Oper. Res. 55 (2021) 3293–3316
https://doi.org/10.1051/ro/2021154

RAIRO Operations Research
www.rairo-ro.org

ON THE DERIVATIVE-FREE QUASI-NEWTON-TYPE ALGORITHM FOR SEPARABLE SYSTEMS OF NONLINEAR EQUATIONS

Hassan Mohammad[1,2,*], Aliyu Muhammed Awwal[2,3,4], Auwal Bala Abubakar[1,2,5] and Ahmad Salihu Ben Musa[6]

Abstract. A derivative-free quasi-Newton-type algorithm in which its search direction is a product of a positive definite diagonal matrix and a residual vector is presented. The algorithm is simple to implement and has the ability to solve large-scale nonlinear systems of equations with separable functions. The diagonal matrix is simply obtained in a quasi-Newton manner at each iteration. Under some suitable conditions, the global and R-linear convergence result of the algorithm are presented. Numerical test on some benchmark separable nonlinear equations problems reveal the robustness and efficiency of the algorithm.

Mathematics Subject Classification. 65K05, 65H10, 90C30, 90C53.

Received December 27, 2020. Accepted October 9, 2021.

1. Introduction

Consider the problem of finding a solution of nonlinear system of equations

$$g(x) = 0, \tag{1.1}$$

where $g = (g_1,\ g_2, \dots g_n) : \mathbb{R}^n \to \mathbb{R}^n$ is a *separable function*. The separability here means each of the component g_i depends on only one or a few components of the vector x. This structure has been studied and regarded as *partial separability* by Griewank and Toint in [6–8].

Keywords. Separable nonlinear equations, derivative-free methods, quasi-Newton-type methods, convergence, numerical experiments.

[1] Department of Mathematical Sciences, Faculty of Physical Sciences, Bayero University, Kano, Nigeria.

[2] Numerical Optimization Research Group, Bayero University, Kano, Nigeria.

[3] Department of Mathematics, Faculty of Science, Gombe State University, Gombe, Nigeria.

[4] KMUTT Fixed Point Research Laboratory, Room SCL 802 Fixed Point Laboratory, Science Laboratory Building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thrung Khru, Bangkok 10140, Thailand.

[5] Department of Mathematics and Applied Mathematics, Sefako Makgatho Health Sciences University, Ga-Rankuwa, Pretoria, Medunsa 0204, South Africa.

[6] Department of Computer Science, Federal College of Agricultural Produce Technology, Kano, Nigeria.

*Corresponding author: hmuhd.mth@buk.edu.ng

Problem (1.1) may arise from an unconstrained optimization problem, for example, let $f(x) = \sum_{i=1}^{n} g_i(x)$. Then the nonlinear system of equations problem (1.1) is equivalent to the unconstrained optimization problem

$$\min f(x), \quad x \in \mathbb{R}^n. \tag{1.2}$$

For finding the solution of general nonlinear equations, quasi-Newton methods are famous and commonly used algorithms because of their derivative-free nature [17, 21]. However, among these methods, some are not suitable for large-scale problems due to matrix storage requirements. As such, methods that considered nonlinear equations with structured functions are given much attention. Nevertheless, several quasi-Newton-type alternatives are given over the last decade (see for example [14, 18, 19, 23, 25, 27]). The spectral gradient method initially introduced by Barzilai and Borwein has been successfully used as a derivative-free approach for solving large-scale nonlinear equations by La Cruz-Martínez-Raydan in [11, 13]. Specifically, La Cruz *et al.* [13] presented a derivative-free spectral residual method (`dfsane`) for solving large-scale nonlinear equations. The algorithm uses a scalar multiple of identity for estimating the Jacobian of the function g. Moreover, some algorithms that uses a diagonal matrix to approximate the Jacobian of the residual function g have been studied in the literature. For details, interested reader may refer to the following references [5, 10, 24, 26].

In this paper, we incorporate the diagonal Hessian approximation approach studied by Deng and Wan [2] and the spectral residual approach presented in [13] to propose, analyze and implement a derivative-free algorithm for separable problems, which can be seen as an improved version of the `dfsane` algorithm that used a positive definite diagonal matrix as the approximation of the Jacobian of the function g. A derivative-free line search is employed to analyze the convergence of the proposed algorithm.

The paper is organized as follows. Section 2 describes some preliminaries and the algorithm. Section 3 addresses the global convergence and rate of convergence results of the algorithm. Section 4 presents the numerical experiments, and conclusions are given in Section 5. Unless otherwise stated, throughout this paper we denote u_k^i to refer to the ith component of a vector u_k. Also, $\|\cdot\|$ stands for the Euclidean norm of vectors and the induced 2-norm of matrices.

2. Preliminaries and algorithm

In this section, we present the derivative-free quasi-Newton-type algorithm. We begin by briefly reviewing the conference paper by Deng and Wan [2].

Based on the idea of Shi and Sun in [22], Deng and Wan presented a spectral conjugate gradient method for solving unconstrained optimization problem (1.2), in which the spectral parameter is a specific diagonal matrix chosen such that it owns some quasi-Newton property. They considered a diagonal matrix $Q_k = \mathrm{diag}(q_k^1, q_k^2, \ldots, q_k^n)$, and solved the following constrained optimization problem

$$\min_{L_k \le q_k^i \le U_k} \frac{1}{2} \sum_{i=1}^{n} \left(q_k^i y_{k-1}^i - s_{k-1}^i\right)^2, \tag{2.1}$$

where L_k and U_k are given lower and upper bounds for q_k^i such that $0 < L_k \le q_k^i \le U_k$, and so Q_k is a safely positive definite matrix. The solution of the problem (2.1) is given by

$$q_k^i = \begin{cases} \frac{s_{k-1}^i}{y_{k-1}^i}, & \text{if } L_k \le \frac{s_{k-1}^i}{y_{k-1}^i} \le U_k \\ L_k, & \text{if } \frac{s_{k-1}^i}{y_{k-1}^i} < L_k \\ U_k, & \text{if } \frac{s_{k-1}^i}{y_{k-1}^i} > U_k \\ \frac{L_k+U_k}{2}, & \text{if } y_{k-1}^i = 0, \end{cases} \tag{2.2}$$

where $L_k = c_1\|g_k\|$, $U_k = c_1\|g_k\| + c_2$ and $c_1, c_2 > 0$. Unfortunately, the authors in [2] do not present numerical implementation of the method.

Next, to build our propose algorithm, we begin by assembling the diagonal matrix similar to the one proposed by Deng and Wan. The difference between the former and later is on the safeguard that ensure positive definiteness of the diagonal matrix. To construct the diagonal matrix of the proposed algorithm, we make use of the following Lemma (Lemma 1 in [20]).

Lemma 2.1. *Let $D = diag(d)$ be a diagonal matrix in $\mathbb{R}^{n\times n}$, and let u and v be vectors in $\mathbb{R}^n$. Then, the solution of the constrained linear least-squares problem with simple bounds*

$$\min_{d\in\mathbb{R}^n} \frac{1}{2}\|diag(d)v - u\|^2, \quad \text{subject to } -d \le 0,$$

is given by

$$d^i = \begin{cases} \frac{u^i}{v^i}, & \text{if } \frac{u^i}{v^i} > 0, \\ 0, & \text{if } \frac{u^i}{v^i} \le 0 \text{ or } v^i = 0. \end{cases} \quad i = 1, 2, \dots, n.$$

Based on the results of Lemma 2.1, the resulting diagonal matrix is positive semi-definite. However, to obtain a descent direction that will be used with a suitable line search technique, we define a positive definite diagonal matrix D_k $(k \ge 1)$ with entries

$$d_k^i = \begin{cases} \frac{y_{k-1}^i}{s_{k-1}^i}, & \text{if } \frac{y_{k-1}^i}{s_{k-1}^i} > 0, \\ 1, & \text{if } \frac{y_{k-1}^i}{s_{k-1}^i} \le 0 \text{ or } s_{k-1}^i = 0, \end{cases} \quad i = 1, 2, \dots, n. \tag{2.3}$$

where $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g(x_k) - g(x_{k-1})$. The search direction of the diagonal derivative-free method is obtained as a solution of the linear system:

$$D_k p_k + g(x_k) = 0, \tag{2.4}$$

where

$$D_k = \begin{cases} \text{diag}(d_k^1,\ d_k^2,\ ,\dots,\ d_k^n), & \text{if } k \ge 1 \\ I, & \text{if } k = 0 \end{cases} \tag{2.5}$$

is a diagonal matrix, whose entries are computed using equation (2.3).

Furthermore, we safeguard D_k for very small and very large values by means of a projection of its entries into a given scalar interval $[\underline{d}, \overline{d}]$ such that $0 < \underline{d} < 1$ and $\overline{d} \ge 1$. Hence, the i-th entry of the matrix D_k is

$$d_k^i = \begin{cases} \max\left\{\min\left\{\frac{y_{k-1}^i}{s_{k-1}^i},\ \overline{d}\right\},\ \underline{d}\right\}, & \text{if } s_{k-1}^i \ne 0 \\ 1, & \text{if } s_{k-1}^i = 0. \end{cases} \quad i = 1, 2, \dots, n. \tag{2.6}$$

It can be seen from equation (2.6), the sequence $\{d_k^i\}$ is uniformly bounded for each i and k. In fact, $0 < \underline{d} \le d_k^i \le \overline{d}$ $\forall i,\ \forall k$. Consequently, D_k is invertible for each $k \ge 0$.

In contrast to the diagonal matrix proposed by Deng and Wan [2], the safeguard procedure here is simple, as it set the nonpositive entries of the generated diagonal matrix to a nonnegative parameter $\underline{d}$, and the undefined entries to 1. Thus, at a certain iterate where some of the entries of the diagonal matrix becomes undefined, the entries are set to 1. Unlike Deng and Wan proposed diagonal matrix, where the undefined entries are set to the average of the lower and upper bounds L_k and U_k. The detail steps of the derivative-free quasi-Newton-type approach is given below.

Algorithm 1: Derivative-free quasi-Newton-type algorithm for separable nonlinear equations (`dfnwt`).

Input : Given $x_0 \in \mathbb{R}^n$, $\rho, \delta \in (0,1)$, $0 < \underline{d} < 1 \leq \overline{d}$, and a positive sequence $\{\varpi_k\}$ such that $\sum_{k=0}^{\infty} \varpi_k < \infty$. Set $k := 0$.

Step 1: Compute $g(x_k)$, **if** $\|g(x_k)\| = 0$, **then**
  stop.
end
Step 2: **if** $k = 0$, **then**
  set $p_k := -g(x_k)$;
else
  Compute $p_k := -D_k^{-1} g(x_k)$, where $D_k = \operatorname{diag}(d_k^1,\ d_k^2,\ \cdots,\ d_k^n)$,

$$d_k^i = \begin{cases} \max\left\{\min\left\{\frac{y_{k-1}^i}{s_{k-1}^i},\ \overline{d}\right\},\ \underline{d}\right\}, & \text{if } s_{k-1}^i \neq 0 \\ 1, & \text{if } s_{k-1}^i = 0, \end{cases} \quad i = 1, 2, \ldots, n.$$

  where $s_{k-1} = x_k - x_{k-1}$, and $y_{k-1} = g(x_k) - g(x_{k-1})$.
end
Step 3: Let $\alpha_k = \rho^j$, where j is the least non-negative integer satisfying

$$\|g(x_k + \rho^j p_k)\|^2 \leq (1 + \varpi_k)\|g(x_k)\|^2 + \delta(\rho^j)^2 \langle g(x_k),\ p_k\rangle, \tag{2.7}$$

  Compute $s_k = \alpha_k p_k$, $x_{k+1} = x_k + s_k$.
Step 4: Set $k = k + 1$ and go to Step 1.

Remark 2.2. Since the matrix D_k is diagonal, the product at Step 2 of Algorithm 1 when $k \neq 0$ is simply the product between the diagonal elements of D_k^{-1} and the corresponding components of $g(x_k)$, computed in $O(n)$ operations.

Remark 2.3. By the definition of the search direction in Step 2, it can be deduce easily that,

$$\frac{1}{\overline{d}}\|g(x_k)\| \leq \|p_k\| \leq \frac{1}{\underline{d}}\|g(x_k)\|. \tag{2.8}$$

Remark 2.4. The line search condition (2.7) has some similarity to the one used in [29]. The right hand side of the current line search in (2.7) has an additional term, a positive sequence that guaranty the well-definedness of the inequality. In fact, for sufficiently large k, the inequality (2.7) holds as the stepsize $\alpha_k \to 0^+$. Thus, α_k can be obtained by some backtracking approach such as Step 3 of Algorithm 1.

3. Convergence results

In this section, we prove the global and R-linear convergence of Algorithm 1. First we assume that $g(x_k) \neq 0$ for any $k \geq 0$ except at the solution.

Furthermore, we assume the following:

Assumption 3.1. (i.) *The function $g : \Theta \subseteq \mathbb{R}^n \to \mathbb{R}^n$ is continuously differentiable on Θ.*

(ii.) *The Jacobian J of g at x, denoted by $J(x)$, is bounded and uniformly nonsingular on Θ, i.e., there exist nonnegative scalars $\varepsilon_1, \varepsilon_2$ such that*

$$\varepsilon_1 \leq \|J(x)\| \leq \varepsilon_2, \quad \textit{for all} \quad x \in \Theta,$$
$$\frac{1}{\varepsilon_2} \leq \|J(x)^{-1}\| \leq \frac{1}{\varepsilon_1} \quad \textit{for all} \quad x \in \Theta.$$

(iii.) *The Jacobian J is Lipschitz continuous with Lipschitz constant γ on Θ. That is,*

$$\|J(x) - J(y)\| \leq \gamma\|x - y\|, \quad \textit{for all} \ \ x, y \in \Theta.$$

Assumption 3.1 implies that there is constants $M \geq m > 0$ such that

$$m\|x-y\| \leq \|g(x)-g(y)\| \leq M\|x-y\|, \quad \forall x,\ y \in \mathbb{R}^n. \tag{3.1}$$

Assumption 3.2. *The diagonal matrix D_k approximate the Jacobian matrix J of the function g at x_k along the direction p_k, therefore, D_k can be regarded as a good approximation of $J(x_k)$. That is,*

$$\|(D_k - J(x_k))p_k\| \leq r\|g(x_k)\|, \quad \textit{for all} \ \ k \geq 0, \tag{3.2}$$

where $r \in (0,\ 1)$ is a very small constant.

Lemma 3.3. *Let the sequence $\{x_k\}$ be generated by Algorithm 1, then for all $k \geq 0$*

(a)
$$\frac{1}{\overline{d}}\|g(x_k)\|^2 \leq -\langle g(x_k),\ p_k\rangle \leq \frac{1}{\underline{d}}\|g(x_k)\|^2, \tag{3.3}$$

(b)
$$\langle g(x_k),\ J(x_k)p_k\rangle \leq -(1-r)\|g(x_k)\|^2, \tag{3.4}$$

(c)
$$-\overline{d}\|p_k\|^2 \leq \langle g(x_k),\ p_k\rangle \leq -\underline{d}\|p_k\|^2. \tag{3.5}$$

Proof. (a) For $k = 0$,

$$\begin{aligned} -\langle g(x_0),\ p_0\rangle &= -\langle g(x_0),\ -D_0^{-1}g(x_0)\rangle \\ &= \|g(x_0)\|^2, \text{ since } \ D_0 = I. \end{aligned}$$

For $k \geq 1$,

$$\begin{aligned} -\langle g(x_k),\ p_k\rangle &= -\langle g(x_k),\ -D_k^{-1}g(x_k)\rangle \\ &= \left\langle g(x_k),\ \operatorname{diag}\left(\frac{1}{d_k^1}, \frac{1}{d_k^2}, \cdots, \frac{1}{d_k^n}\right) g(x_k)\right\rangle \\ &\leq \left\langle g(x_k),\ \operatorname{diag}\left(\frac{1}{\underline{d}}, \frac{1}{\underline{d}}, \cdots, \frac{1}{\underline{d}}\right) g(x_k)\right\rangle \\ &= \frac{1}{\underline{d}}\|g(x_k)\|^2. \end{aligned} \tag{3.6}$$

On the other hand,

$$\begin{aligned} -\langle g(x_k),\ p_k\rangle &= -\langle g(x_k),\ -D_k^{-1}g(x_k)\rangle \\ &= \left\langle g(x_k),\ \operatorname{diag}\left(\frac{1}{d_k^1}, \frac{1}{d_k^2}, \cdots, \frac{1}{d_k^n}\right) g(x_k)\right\rangle \\ &\geq \left\langle g(x_k),\ \operatorname{diag}\left(\frac{1}{\overline{d}}, \frac{1}{\overline{d}}, \cdots, \frac{1}{\overline{d}}\right) g(x_k)\right\rangle \\ &= \frac{1}{\overline{d}}\|g(x_k)\|^2. \end{aligned} \tag{3.7}$$

Combining (3.6) and (3.7), we obtain (3.3).

(b) For $k \geq 0$, equality (2.4) together with inequality (3.2) gives

$$\begin{aligned}\langle g(x_k),\ J(x_k)p_k\rangle &= \langle g(x_k),\ J(x_k)p_k - (D_kp_k + g(x_k))\rangle \\ &= \langle g(x_k),\ (J(x_k) - D_k)p_k\rangle - \|g(x_k)\|^2 \\ &\leq \|g(x_k)\|\|(J(x_k) - D_k)p_k\| - \|g(x_k)\|^2 \\ &\leq r\|g(x_k)\|^2 - \|g(x_k)\|^2 \\ &= -(1-r)\|g(x_k)\|^2.\end{aligned}$$

The proof of (c) follows directly from equation (2.4) and the definition of D_k in (2.5)–(2.6).

□

The following Lemma is from [3].

Lemma 3.4. *Let $\{a_k\}$ and $\{e_k\}$ be nonnegative sequences such that*

$$a_{k+1} \leq (1+e_k)a_k \quad \textit{and} \quad \sum_{k=0}^{\infty} e_k < \infty,$$

then the sequence $\{a_k\}$ has a limit in $\mathbb{R}$.

Lemma 3.5. *Let $\{x_k\}$ be the sequence generated by Algorithm 1, then we have*

(a) *$\{\|g(x_k)\|\}$ is convergent.*
(b) $\lim_{k\to\infty} -\alpha_k^2\langle g(x_k),\ p_k\rangle = 0$.

Proof. (a) Setting $a_k = \|g(x_k)\|^2$ and $e_k = \varpi_k$ in Lemma 3.4, we have

$$\|g(x_{k+1})\|^2 \leq (1+\varpi_k)\|g(x_k)\|^2.$$

Since $\|g(x_k)\|^2 \geq 0$ and $\sum_{k=0}^{\infty} \varpi_k < \infty$, it holds $\{\|g(x_k)\|\}$ is convergent.

(b) Using (2.7), we can get for any k

$$-\delta\alpha_k^2\langle g(x_k),\ p_k\rangle \leq \|g(x_k)\|^2 - \|g(x_{k+1})\|^2 + \varpi_k\|g(x_k)\|^2. \tag{3.8}$$

Summing both sides of (3.8) yields

$$\delta\sum_{k=0}^{\infty} -\alpha_k^2\langle g(x_k),\ p_k\rangle \leq \|g(x_0)\| + \sum_{k=0}^{\infty}\varpi_k\|g(x_k)\|^2.$$

Since $\{\|g(x_k)\|\}$ converges for all k, $\sum_{k=0}^{\infty}\varpi_k$ is convergent and δ is a positive constant, it follows that

$$\sum_{k=0}^{\infty} -\alpha_k^2\langle g(x_k),\ p_k\rangle < \infty.$$

Hence,

$$\lim_{k\to\infty} -\alpha_k^2\langle g(x_k),\ p_k\rangle = 0. \tag{3.9}$$

□

Lemma 3.6. *Suppose Assumptions 3.1 and 3.2 hold. Let* $\{x_k\}$ *be a sequence of iterates generated by Algorithm 1. Then*

$$\alpha_k \geq \min\left\{1, \frac{2\rho \underline{d}^2(1-r)}{\delta \bar{d}}\right\}, \tag{3.10}$$

for all sufficiently large k*.*

Proof. By the line search condition (2.7) if $\alpha_k \neq 1$, then $\frac{\alpha_k}{\rho}$ does not satisfy (2.7), that is

$$\left\| g\left(x_k + \frac{\alpha_k}{\rho} p_k\right)\right\|^2 > (1+\varpi_k)\|g(x_k)\|^2 + \delta \frac{\alpha_k^2}{\rho^2}\langle g(x_k),\ p_k\rangle.$$

This gives

$$-\delta\frac{\alpha_k^2}{\rho^2}\langle g(x_k),\ p_k\rangle > \|g(x_k)\|^2 - \left\| g\left(x_k + \frac{\alpha_k}{\rho} p_k\right)\right\|^2. \tag{3.11}$$

Using the right hand side of (3.11), inequalities (2.8), (3.4) and (3.5), we have

$$\begin{aligned} \|g(x_k)\|^2 - \left\| g\left(x_k + \frac{\alpha_k}{\rho} p_k\right)\right\|^2 &= -2\frac{\alpha_k}{\rho}\langle g(x_k),\ J(x_k)p_k\rangle + o\left(\frac{\alpha_k}{\rho}\|p_k\|\right) \\ &\geq 2\frac{\alpha_k}{\rho}(1-r)\|g(x_k)\|^2 + o\left(\frac{\alpha_k}{\rho}\|p_k\|\right) \\ &\geq 2\underline{d}^2\frac{\alpha_k}{\rho}(1-r)\|p_k\|^2 + o\left(\frac{\alpha_k}{\rho}\|p_k\|\right) \\ &\geq -2\underline{d}^2\frac{\alpha_k}{\rho\bar{d}}(1-r)\langle g(x_k),\ p_k\rangle, \end{aligned} \tag{3.12}$$

where $o : \mathbb{R}_+ \to \mathbb{R}$ is such that $\lim_{\xi\to 0}\frac{o(\xi)}{\xi} = 0$.

Combining (3.11) and (3.12), we have

$$\alpha_k > \frac{2\rho\underline{d}^2(1-r)}{\delta\bar{d}},$$

which means that (3.10) holds. □

Theorem 3.7. *Suppose Assumptions 3.1 and 3.2 hold. If the sequence* $\{x_k\}$ *is generated by Algorithm 1, then*

$$\lim_{k\to\infty}\|g(x_k)\| = 0. \tag{3.13}$$

Proof. By Lemma 3.6, there exists a nonnegative scalar say

$$\bar{\alpha} := \min\left\{1, \frac{2\underline{d}^2(1-r)}{\delta\bar{d}}\right\} \leq \alpha_k. \tag{3.14}$$

It follows from (3.3) and (3.14) that

$$-\alpha_k^2\langle g(x_k),\ p_k\rangle \geq \frac{\alpha_k^2}{\bar{d}}\|g(x_k)\|^2 \geq \frac{\bar{\alpha}^2}{\bar{d}}\|g(x_k)\|^2 \geq 0.$$

Therefore, using (3.9), we have

$$0 = \lim_{k\to\infty} -\alpha_k^2\langle g(x_k),\ p_k\rangle \geq \frac{\bar{\alpha}^2}{\bar{d}}\lim_{k\to\infty}\|g(x_k)\|^2 \geq 0.$$

This gives (3.13). □

We now present the R-liner convergence of Algorithm 1.

Theorem 3.8. *Suppose Assumption 3.1 holds. If the sequence $\{x_k\}$ generated by Algorithm 1 converges to x^*, then for sufficiently large k, there exist constants $C > 0$ and $\mu \in (0,1)$ such that*

$$\|x_k - x^*\| \leq C\mu^k. \tag{3.15}$$

Proof. From the line search condition (2.7), it follows that

$$\begin{aligned}\|g(x_{k+1})\|^2 &\leq (1+\varpi_k)\|g(x_k)\|^2 + \delta\alpha_k^2\langle g(x_k),\ p_k\rangle \\ &\leq (1+\varpi_k)\|g(x_k)\|^2 - \delta\alpha_k^2\frac{1}{\bar{d}}\|g(x_k)\|^2 \\ &\leq (1+\varpi_k)\|g(x_k)\|^2 - \delta\bar{\alpha}^2\frac{1}{\bar{d}}\|g(x_k)\|^2 \\ &= \left(1 - \delta\bar{\alpha}^2\frac{1}{\bar{d}} + \varpi_k\right)\|g(x_k)\|^2,\end{aligned}$$

where the second and third inequalities follow from (3.3) and (3.14) respectively. Since $\varpi_k \to 0$, without loss of generality, we assume that $\varpi_k \leq \delta\bar{\alpha}^2\frac{1}{2\bar{d}}$ for all k so that

$$\|g(x_{k+1})\| \leq \sqrt{\left(1 - \delta\bar{\alpha}^2\frac{1}{2\bar{d}}\right)}\|g(x_k)\|. \tag{3.16}$$

Inequality (3.16) and inductive process yields

$$\|g(x_k)\| \leq \mu^k\|g(x_0)\|, \tag{3.17}$$

where $\mu = \sqrt{\left(1 - \delta\bar{\alpha}^2\frac{1}{2\bar{d}}\right)} < 1$. Using (3.1) together with (3.17) we have

$$\|x_k - x^*\| \leq \mu^k\frac{\|g(x_0)\|}{m}.$$

Thus, (3.15) holds with $C = \frac{\|g(x_0)\|}{m}$. This means that Algorithm 1 converges R-linearly. □

4. Numerical experiments

In this section we report the results obtained with a preliminary MATLAB implementation of the proposed algorithm on the solution of some selected test problems. The set of the problems is made of ten almost separable nonlinear equations and can be found in the Appendix A. The detailed numerical results of this section can be found in Appendix B. Computations were carried out on an 8.00 GB RAM Intel Core i7 personal computer at 2.30 GHz. A failure is reported (denoted by *'F'*), if the number of iterations is greater than 1000. We used five different dimension with ten different initial points as follows:

– *dimensions*: $n = 1000,\ 5000,\ 10\,000,\ 50\,000,\ 100\,000$.
– *initial points*: $x_0^1 = (1,\dots,1)^T$, $x_0^2 = (0.1,\dots,0.1)^T$, $x_0^3 = \left(\frac{1}{2},\dots,\frac{1}{2^n}\right)^T$, $x_0^4 = \left(1-\frac{1}{n}, 1-\frac{2}{n},\dots,0\right)^T$, $x_0^5 = \left(0,\frac{1}{n},\frac{2}{n},\dots,\frac{n-1}{n}\right)^T$, $x_0^6 = \left(1,\frac{1}{2},\dots,\frac{1}{n}\right)^T$, $x_0^7 = \left(\frac{n-1}{n},\frac{n-2}{n},\dots,0\right)^T$, $x_0^8 = \left(\frac{1}{n},\frac{2}{n},\dots,1\right)^T$, $x_0^9 = (10,10,\dots,10)^T$ and $x_0^{10} = \text{rand}(n,1)$. Here, $\text{rand}(n,1)$ means the initial point is chosen randomly from the interval $(0,1)$.

TABLE 1. Winners with respect to #iter, #fval and time.

Method/Metric	dfnwt	dfsane	msgp
#iter	329	291	75
#fval	353	309	30
time	103	366	24

We compared Algorithm 1 (`dfnwt`) with similar algorithms in the literature, namely, `dfsane` algorithm by La Cruz *et al.* [13] and `msgp` algorithm by Yu *et al.* [28]. For all algorithms, we used the stopping criterion

$$\|g(x_k)\| \leq 10^{-6}.$$

We implemented `dfnwt` algorithm using the following parameters: $\rho = 0.5$, $\delta = 0.0001$, $\underline{d} = 10^{-10}$, $\overline{d} = 10^{10}$ and $\varpi_k = \dfrac{1}{e^{k^2}}$, $k > 0$. For `dfsane` and `msgp` algorithms, the parameters chosen are from references [13] and [28], respectively.

In Tables B.1–B.10 of Appendix B, we reported the number of iterations ($\#iter$), the number of function evaluations ($\#fval$), the CPU time in seconds (time) and the norm of the residual at the termination point (Fnorm), for all the ten tested problems. In Table B.1, `dfnwt` has the least $\#iter$ and $\#fval$ in all the problems. However, there was a tie between `dfnwt` and `dfsane` in Tables B.2–B.5, B.7 and B.8 except for the the initial point x_0^9 and some some few cases in Tables B.7 and B.8. In Table B.6, `dfsane` has the best performance in terms of $\#iter$ and $\#fval$ except for some few cases where `dfnwt` performs better. The algorithm `dfnwt` has recorded 17 failures in Table B.9, however, it outperforms `dfsane` and `msgp` algorithms in the remaining cases. Lastly, in Table B.10, unlike `dfsane` and `msgp`, `dfnwt` managed to solve almost all the problems. However, for the few cases where `msgp` solved a problem, it has the least $\#iter$ and $\#fval$. In addition, the summary of Tables B.1–B.10 is reported in Table 1.

To visualize the numerical behaviour of the algorithms, we plotted three figures using the popular Dolan and Moré [4] performance profile based on the $\#iter$, $\#fval$ and CPU time metrics. In Figure 1, we compare the performance of the `dfnwt` algorithm with the `dfsane` algorithm and the `msgp` algorithm with respect to $\#iter$ metric. Figure 1 shows that `dfnwt` performs better than `msgp` and `dfsane` having almost 70% success. In Figure 2, the performance of the three algorithms was tested based on $\#fval$ metric. The figure shows that `dfnwt` performs better than `msgp` and `dfsane` having over 70% success. Figure 3 shows that `dfsane` is faster than `dfnwt` and `msgp` for the fraction of $\tau \leq 4$. However, for $\tau > 4$, `dfnwt` is faster than `msgp` and `dfsane`. Based on the performed experiments, we observe that, the good performance of the `dfnwt` algorithm may be due to the diagonal approximation of the Jacobian matrix associated with the search direction. Similar argument applies to the `msgp` algorithm.

5. CONCLUSIONS

We have presented, analyzed, and implemented a derivative-free quasi-Newton-type algorithm for solving nonlinear systems of equations with separable functions (`dfnwt`). Different from the existing algorithms such as `dfsane` algorithm that approximate the Jacobian of g using a scalar multiple of identity at each iteration, the proposed `dfnwt` algorithm uses a diagonal matrix in a quasi-Newton manner for such approximations. Among the attractive feature of the presented algorithm is that it does not require gradient or approximation of the gradient for its implementation, this makes it more suitable for large-scale separable problems. Furthermore, the global and R-linear convergence of the sequence generated by `dfnwt` algorithm is obtained. Based on the numerical results presented, the proposed `dfnwt` compete with the well-known and efficient algorithm for solving nonlinear equations, that is, `dfsane`. This good efficiency of the `dfnwt` algorithm is due to the additional information obtained from the diagonal matrix used for the approximation of the Jacobian of the problems.

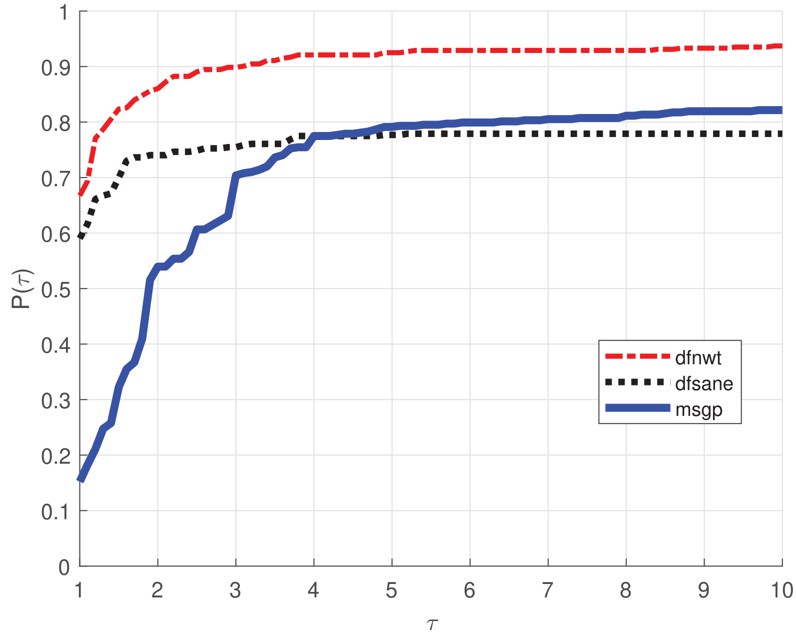

FIGURE 1. Dolan and Moré performance profile with respect to number of iterations.

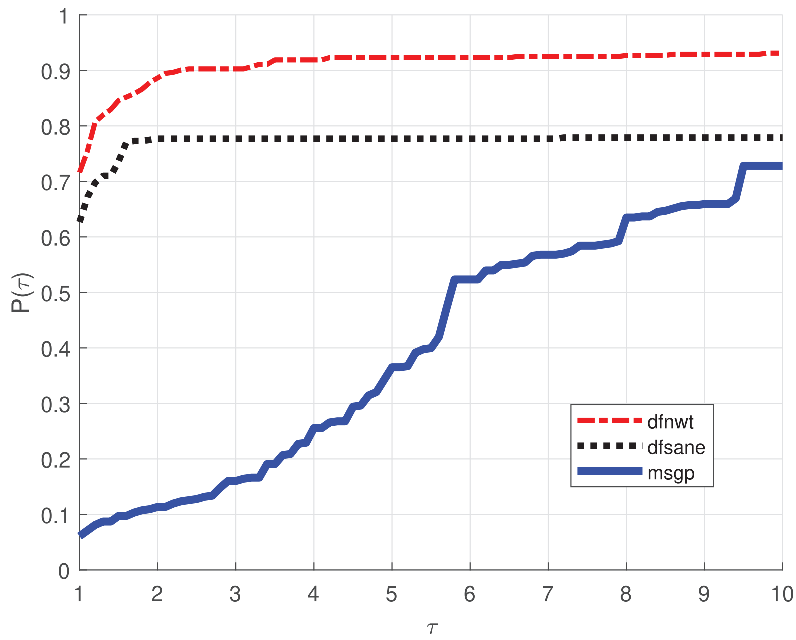

FIGURE 2. Dolan and Moré performance profile with respect to number of function evaluations.

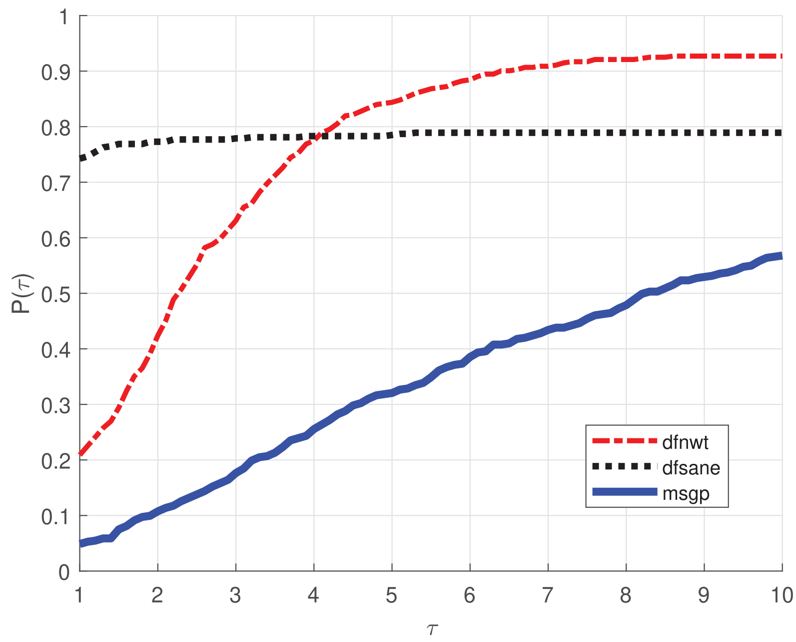

FIGURE 3. Dolan and Moré performance profile with respect to CPU time.

Investigation on the better approximation that exploits the structure of the problem and extensive numerical experiments that will unveil the effectiveness of the approach will be an interesting topic for future research.

APPENDIX A. LIST OF TEST PROBLEMS

We listed below the details of the test problems used in Section 4 where $g = (g_1, g_2, \ldots, g_n)^T$.

Problem 1: Modified exponential function [12]

$$g_1(x) = e^{x_1} - 1$$
$$g_i(x) = e^{x_i} + x_i - 1, \quad i = 2, 3 \ldots, n-1.$$

Problem 2: Logarithmic function [12]

$$g_i(x_i) = \log(x_i + 1) - \frac{x_i}{n}, \quad i = 1, 2, \ldots, n.$$

Problem 3: Strictly convex function I [12]

$$g_i(x) = e^{x_i} - 1, \quad i = 1, 2, \ldots, n.$$

Problem 4: Modified strictly convex function II [12]

$$g_i(x) = \left(\frac{i}{n+1}\right) e^{x_i} - 1, \quad i = 1, 2, \ldots, n.$$

Problem 5: Tridiagonal exponential function [1]

$$
\begin{aligned}
g_1(x) &= x_1 - e^{\cos(h(x_1+x_2))} \\
g_i(x) &= x_i - e^{\cos(h(x_{i-1}+x_i+x_{i+1}))}, \quad i = 2, \dots, n-1, \\
g_n(x) &= x_n - e^{\cos(h(x_{n-1}+x_n))}, \\
h &= \frac{1}{n+1}.
\end{aligned}
$$

Problem 6: Gradient of engval function [15]

$$
\begin{aligned}
g_1(x) &= x_1(x_1^2 + x_2^2) - 1 \\
g_i(x) &= x_i(x_{i-1}^2 + 2x_i^2 + x_{i+1}^2) - 1, \quad 2 \le i \le n-1 \\
g_n(x) &= x_n(x_{n-1}^2 + x_n^2).
\end{aligned}
$$

Problem 7: Chandrasekhar H-equation [9]

$$
g_i(x) = x_i - \left(1 - \frac{c}{2n} \sum_{j=1}^{n} \frac{\delta_i x_j}{\delta_i + \delta_j}\right)^{-1}, \quad c = 0.9, \quad \delta_i = \frac{i - 0.5}{n}, \quad i = 1, 2, \dots, n.
$$

Problem 8: Modified problem 3.34 in [16]

$$
\begin{aligned}
g_i(x) &= x_i - \frac{x_{i+1}^3}{100}, \quad 1 \le i \le n-1 \\
g_n(x) &= x_n - \frac{x_n^3}{100}.
\end{aligned}
$$

Problem 9: Trigonometric function [30]

$$
g_i(x) = 2\left(n + i(1 - \cos x_i) - \sin x_i - \sum_{j=1}^{n} \cos x_j\right)(2\sin x_i - \cos x_i), \text{ for } i = 1, 2, 3, \dots, n.
$$

Problem 10: Troesch problem [12]

$$
\begin{aligned}
g_1(x) &= 2x_1 + 10\frac{\sinh(10x_1)}{(n+1)^2} - x_2 \\
g_i(x) &= 2x_i + 10\frac{\sinh(10x_i)}{(n+1)^2} - x_{i-1} - x_{i+1}, \quad 2 \le i \le n-1 \\
g_n(x) &= 2x_n + 10\frac{\sinh(10x_n)}{(n+1)^2} - x_{n-1} - 1.
\end{aligned}
$$

Appendix B. Table of Numerical Experiments

Below is the details of the numerical experiments conducted in Section 4.

TABLE B.1. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 1 with given initial points and dimensions.

		dfnwt				dfsane				msgp			
Dimension	x_0	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
	x_0^1	7	7	0.0144	1.76E-08	11	11	0.0303	1.33E-08	8	25	0.0944	1.16E-07
	x_0^2	5	5	0.0072	4.87E-11	6	6	0.0133	1.63E-07	6	19	0.0284	9.53E-09
	x_0^3	5	5	0.0046	6.23E-08	8	8	0.0099	1.51E-07	9	28	0.1310	4.04E-08
	x_0^4	7	7	0.0082	1.38E-08	10	10	0.0079	8.47E-07	11	34	0.0160	1.75E-08
	x_0^5	7	7	0.0068	1.44E-09	7	7	0.0262	5.15E-11	10	31	0.0216	1.18E-07
	x_0^6	7	7	0.0070	1.43E-08	10	10	0.0075	1.04E-08	12	37	0.0152	8.39E-09
1000	x_0^7	7	7	0.0059	1.38E-08	10	10	0.0066	8.47E-07	11	34	0.0135	1.75E-08
	x_0^8	7	7	0.0049	1.47E-09	7	7	0.0045	5.03E-09	10	31	0.0130	1.22E-07
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.0368	2.17E-07
	x_0^{10}	7	7	0.0118	1.31E-09	10	10	0.0055	2.13E-08	10	31	0.0123	1.19E-07
	x_0^1	7	7	0.0332	2.71E-08	11	11	0.0132	6.70E-08	8	25	0.0385	6.68E-07
	x_0^2	5	5	0.0188	1.09E-10	6	6	0.0105	8.53E-08	6	19	0.0369	1.05E-08
	x_0^3	5	5	0.0153	6.23E-08	8	8	0.0138	1.51E-07	9	28	1.5317	4.04E-08
	x_0^4	7	7	0.1608	1.45E-08	10	10	0.0086	5.96E-07	11	34	0.0477	4.05E-07
	x_0^5	7	7	0.0141	3.24E-09	7	7	0.0174	1.16E-10	10	31	0.0569	2.68E-07
	x_0^6	7	7	0.0101	1.43E-08	10	10	0.0120	1.04E-08	12	37	0.0471	1.09E-08
5000	x_0^7	7	7	0.0129	1.45E-08	10	10	0.0187	5.96E-07	11	34	0.0353	4.05E-07
	x_0^8	7	7	0.0124	3.26E-09	7	7	0.0102	2.33E-10	10	31	0.0385	2.70E-07
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.0787	1.85E-07
	x_0^{10}	7	7	0.0244	3.14E-09	8	8	0.0114	5.41E-07	10	31	0.0629	2.90E-07
	x_0^1	7	7	0.0498	3.55E-08	11	11	0.0247	4.37E-08	8	25	0.0614	7.40E-07
	x_0^2	5	5	0.0381	1.54E-10	6	6	0.0108	5.09E-08	6	19	0.0379	1.06E-08
	x_0^3	5	5	0.0403	6.23E-08	8	8	0.0133	1.51E-07	9	28	2.6461	4.04E-08
	x_0^4	7	7	0.0606	1.49E-08	10	10	0.0124	5.05E-07	11	34	0.0767	8.47E-07
	x_0^5	7	7	0.0304	4.59E-09	7	7	0.0084	1.65E-10	10	31	0.0881	3.80E-07
	x_0^6	7	7	0.0221	1.43E-08	10	10	0.0103	1.04E-08	12	37	0.0773	1.13E-08
10 000	x_0^7	7	7	0.0315	1.49E-08	10	10	0.0140	5.05E-07	11	34	0.0667	8.47E-07
	x_0^8	7	7	0.0513	4.61E-09	7	7	0.0155	1.72E-10	10	31	0.0602	3.81E-07
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.1247	1.70E-07
	x_0^{10}	7	7	0.0746	4.62E-09	10	10	0.0264	4.76E-07	10	31	0.1081	3.66E-07
	x_0^1	7	7	0.1347	7.41E-08	11	11	0.0822	2.59E-08	8	25	0.2588	4.02E-07
	x_0^2	5	5	0.1261	3.44E-10	6	6	0.0392	2.23E-07	6	19	0.1970	1.26E-08
	x_0^3	5	5	0.0542	6.23E-08	8	8	0.0445	1.51E-07	9	28	20.1184	4.04E-08
	x_0^4	7	7	0.1977	1.76E-08	11	11	0.0493	8.95E-09	12	37	0.1819	9.92E-09
	x_0^5	7	7	0.2008	1.03E-08	7	7	0.0349	3.68E-10	10	31	0.3389	8.50E-07
	x_0^6	7	7	0.1411	1.43E-08	10	10	0.0903	1.04E-08	12	37	0.2629	1.16E-08
50 000	x_0^7	7	7	0.1466	1.76E-08	11	11	0.0644	8.95E-09	12	37	0.3564	9.92E-09
	x_0^8	7	7	0.2467	1.03E-08	7	7	0.0452	3.69E-10	10	31	0.3816	8.51E-07
	x_0^9	F	F	F	F	F	F	F	F	18	55	0.6111	2.52E-08
	x_0^{10}	7	7	0.2109	1.02E-08	8	8	0.0685	4.48E-07	10	31	0.2158	8.19E-07
	x_0^1	7	7	0.3991	1.04E-07	11	11	0.0844	3.33E-08	8	25	0.3365	3.34E-07
	x_0^2	5	5	0.1823	4.87E-10	6	6	0.0503	5.42E-07	6	19	0.4366	1.43E-08
	x_0^3	5	5	0.1929	6.23E-08	8	8	0.1260	1.51E-07	9	28	66.4908	4.04E-08
	x_0^4	7	7	0.4219	2.04E-08	11	11	0.1170	1.87E-08	11	34	0.7349	6.45E-07
	x_0^5	7	7	0.4051	1.45E-08	7	7	0.1085	5.21E-10	11	34	0.7134	5.98E-09
	x_0^6	7	7	0.3407	1.43E-08	10	10	0.0876	1.04E-08	12	37	0.6490	1.16E-08
100 000	x_0^7	7	7	0.3645	2.04E-08	11	11	0.0891	1.87E-08	11	34	0.7292	6.45E-07
	x_0^8	7	7	0.3735	1.45E-08	7	7	0.0885	5.21E-10	11	34	0.6609	5.99E-09
	x_0^9	F	F	F	F	F	F	F	F	18	55	1.2723	2.82E-08
	x_0^{10}	7	7	0.4249	1.46E-08	11	11	0.1426	3.83E-09	11	34	0.7735	5.87E-09

TABLE B.2. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 2 with given initial points and dimensions.

Dimension	x_0	dfnwt				dfsane				msgp			
		#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
1000	x_0^1	6	6	0.0097	2.58E-07	6	6	0.0054	2.58E-07	7	22	0.0172	7.57E-07
	x_0^2	4	4	0.0092	2.11E-09	4	4	0.0043	2.11E-09	5	16	0.0098	4.79E-08
	x_0^3	5	5	0.0132	5.77E-09	5	5	0.0062	2.07E-09	9	28	0.0774	4.19E-07
	x_0^4	6	6	0.0121	4.29E-08	6	6	0.0038	1.40E-09	11	34	0.0121	2.76E-08
	x_0^5	6	6	0.0086	4.29E-08	6	6	0.0042	1.40E-09	11	34	0.0205	2.76E-08
	x_0^6	6	6	0.0059	8.17E-09	5	5	0.0025	5.79E-07	11	34	0.0202	1.69E-08
	x_0^7	6	6	0.0050	4.29E-08	6	6	0.0027	1.40E-09	11	34	0.0152	2.76E-08
	x_0^8	6	6	0.0046	4.37E-08	6	6	0.0057	1.43E-09	11	34	0.0181	2.73E-08
	x_0^9	F	F	F	F	10	16	0.0144	7.19E-09	F	F	F	F
	x_0^{10}	6	6	0.0075	4.13E-08	6	6	0.0061	1.21E-09	11	34	0.0067	2.35E-08
5000	x_0^1	6	6	0.0143	5.60E-07	6	6	0.0098	5.60E-07	8	25	0.0366	1.71E-08
	x_0^2	4	4	0.0102	4.46E-09	4	4	0.0101	4.46E-09	5	16	0.0246	1.06E-07
	x_0^3	5	5	0.0083	5.62E-09	5	5	0.0080	2.02E-09	9	28	1.2953	4.30E-07
	x_0^4	6	6	0.0362	9.36E-08	6	6	0.0085	3.06E-09	11	34	0.0985	6.33E-08
	x_0^5	6	6	0.0373	9.36E-08	6	6	0.0151	3.06E-09	11	34	0.1229	6.33E-08
	x_0^6	6	6	0.0327	7.92E-09	5	5	0.0086	5.68E-07	11	34	0.0692	1.68E-08
	x_0^7	6	6	0.0284	9.36E-08	6	6	0.0307	3.06E-09	11	34	0.0748	6.33E-08
	x_0^8	6	6	0.0221	9.40E-08	6	6	0.0109	3.07E-09	11	34	0.0678	6.31E-08
	x_0^9	F	F	F	F	10	16	0.0236	2.09E-08	9	28	0.0658	9.66E-08
	x_0^{10}	6	6	0.0178	9.61E-08	6	6	0.0079	2.98E-09	11	34	0.0623	6.52E-08
10 000	x_0^1	6	6	0.0702	7.88E-07	6	6	0.0103	7.88E-07	8	25	0.0613	2.43E-08
	x_0^2	4	4	0.0288	6.27E-09	4	4	0.0170	6.27E-09	5	16	0.1003	1.50E-07
	x_0^3	5	5	0.0328	5.60E-09	5	5	0.0152	2.02E-09	9	28	2.7424	4.31E-07
	x_0^4	6	6	0.0397	1.32E-07	6	6	0.0131	4.32E-09	11	34	0.1765	8.97E-08
	x_0^5	6	6	0.0270	1.32E-07	6	6	0.0112	4.32E-09	11	34	0.1630	8.97E-08
	x_0^6	6	6	0.0333	7.88E-09	5	5	0.0159	5.67E-07	11	34	0.1345	1.68E-08
	x_0^7	6	6	0.0438	1.32E-07	6	6	0.0202	4.32E-09	11	34	0.1521	8.97E-08
	x_0^8	6	6	0.0557	1.32E-07	6	6	0.0228	4.32E-09	11	34	0.0949	8.96E-08
	x_0^9	F	F	F	F	10	16	0.0381	3.04E-08	9	28	0.1200	1.37E-07
	x_0^{10}	6	6	0.0449	1.31E-07	6	6	0.0187	4.16E-09	11	34	0.1252	9.24E-08
50 000	x_0^1	7	7	0.2191	1.08E-11	7	7	0.0473	1.08E-11	8	25	0.4062	5.44E-08
	x_0^2	4	4	0.0869	1.39E-08	4	4	0.0269	1.39E-08	5	16	0.2628	3.34E-07
	x_0^3	5	5	0.0856	5.59E-09	5	5	0.0302	2.01E-09	9	28	20.5273	4.32E-07
	x_0^4	6	6	0.1248	2.94E-07	6	6	0.0305	9.63E-09	11	34	0.5027	2.01E-07
	x_0^5	6	6	0.2103	2.94E-07	6	6	0.0291	9.63E-09	11	34	0.5402	2.01E-07
	x_0^6	6	6	0.1433	7.86E-09	5	5	0.0530	5.66E-07	11	34	0.4141	1.68E-08
	x_0^7	6	6	0.2749	2.94E-07	6	6	0.0964	9.63E-09	11	34	0.5028	2.01E-07
	x_0^8	6	6	0.1795	2.95E-07	6	6	0.0522	9.63E-09	11	34	0.4857	2.01E-07
	x_0^9	F	F	F	F	10	16	0.0929	6.97E-08	11	34	0.4943	4.42E-08
	x_0^{10}	6	6	0.1481	2.95E-07	6	6	0.0569	9.61E-09	11	34	0.5365	1.96E-07
100 000	x_0^1	7	7	0.2907	1.53E-11	7	7	0.0817	1.53E-11	8	25	0.8299	7.70E-08
	x_0^2	4	4	0.1811	1.97E-08	4	4	0.0448	1.97E-08	5	16	0.3797	4.73E-07
	x_0^3	5	5	0.1079	5.59E-09	5	5	0.0505	2.01E-09	9	28	73.4300	4.32E-07
	x_0^4	6	6	0.1762	4.16E-07	6	6	0.0578	1.36E-08	11	34	1.1085	2.85E-07
	x_0^5	6	6	0.1773	4.16E-07	6	6	0.1194	1.36E-08	11	34	1.1700	2.85E-07
	x_0^6	6	6	0.4746	7.86E-09	5	5	0.0935	5.65E-07	11	34	1.0038	1.68E-08
	x_0^7	6	6	0.5102	4.16E-07	6	6	0.1208	1.36E-08	11	34	0.7507	2.85E-07
	x_0^8	6	6	0.2775	4.16E-07	6	6	0.1150	1.36E-08	11	34	1.1527	2.84E-07
	x_0^9	F	F	F	F	10	16	0.2552	9.89E-08	11	34	1.4168	6.25E-08
	x_0^{10}	6	6	0.1824	4.20E-07	6	6	0.1125	1.39E-08	11	34	1.0825	2.87E-07

TABLE B.3. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 3 with given initial points and dimensions.

Dimension	x_0	dfnwt				dfsane				msgp			
		#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
1000	x_0^1	7	7	0.0082	4.51E-07	7	7	0.0056	4.51E-07	6	19	0.0112	6.51E-07
	x_0^2	4	4	0.0036	2.65E-09	4	4	0.0033	2.65E-09	5	16	0.0050	3.18E-07
	x_0^3	5	5	0.0036	6.22E-08	5	5	0.0041	4.37E-08	8	25	0.0927	1.78E-08
	x_0^4	7	7	0.0105	4.87E-08	7	7	0.0027	3.44E-10	13	40	0.0126	2.00E-08
	x_0^5	7	7	0.0116	4.87E-08	7	7	0.0035	3.44E-10	13	40	0.0137	2.00E-08
	x_0^6	7	7	0.0118	1.43E-08	7	7	0.0021	4.34E-09	9	28	0.0092	2.33E-07
	x_0^7	7	7	0.0071	4.87E-08	7	7	0.0021	3.44E-10	13	40	0.0166	2.00E-08
	x_0^8	7	7	0.0057	5.07E-08	7	7	0.0019	3.57E-10	13	40	0.0100	2.14E-08
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.0137	2.26E-08
	x_0^{10}	7	7	0.0105	4.66E-08	7	7	0.0016	3.64E-10	12	37	0.0203	5.01E-08
5000	x_0^1	8	8	0.0296	9.18E-12	8	8	0.0083	9.17E-12	7	22	0.0467	1.44E-08
	x_0^2	4	4	0.0164	5.92E-09	4	4	0.0041	5.92E-09	5	16	0.0186	7.10E-07
	x_0^3	5	5	0.0166	6.22E-08	5	5	0.0141	4.37E-08	8	25	1.1229	1.78E-08
	x_0^4	7	7	0.0275	1.11E-07	7	7	0.0130	7.80E-10	13	40	0.0418	4.61E-08
	x_0^5	7	7	0.0302	1.11E-07	7	7	0.0093	7.80E-10	13	40	0.0519	4.61E-08
	x_0^6	7	7	0.0140	1.43E-08	7	7	0.0092	4.33E-09	9	28	0.0269	2.40E-07
	x_0^7	7	7	0.0301	1.11E-07	7	7	0.0074	7.80E-10	13	40	0.0517	4.61E-08
	x_0^8	7	7	0.0320	1.12E-07	7	7	0.0069	7.86E-10	13	40	0.0652	4.67E-08
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.0786	5.05E-08
	x_0^{10}	7	7	0.0142	1.07E-07	7	7	0.0084	7.51E-10	13	40	0.0545	4.60E-08
10 000	x_0^1	8	8	0.0386	1.30E-11	8	8	0.0200	1.30E-11	7	22	0.0311	2.04E-08
	x_0^2	4	4	0.0225	8.38E-09	4	4	0.0075	8.38E-09	6	19	0.0614	9.95E-09
	x_0^3	5	5	0.0144	6.22E-08	5	5	0.0111	4.37E-08	8	25	2.3826	1.78E-08
	x_0^4	7	7	0.0627	1.57E-07	7	7	0.0159	1.11E-09	13	40	0.1486	6.54E-08
	x_0^5	7	7	0.0520	1.57E-07	7	7	0.0084	1.11E-09	13	40	0.0816	6.54E-08
	x_0^6	7	7	0.0199	1.43E-08	7	7	0.0119	4.33E-09	9	28	0.0475	2.40E-07
	x_0^7	7	7	0.0483	1.57E-07	7	7	0.0129	1.11E-09	13	40	0.0850	6.54E-08
	x_0^8	7	7	0.0311	1.57E-07	7	7	0.0161	1.11E-09	13	40	0.0953	6.58E-08
	x_0^9	F	F	F	F	F	F	F	F	15	46	0.1053	7.14E-08
	x_0^{10}	7	7	0.0425	1.54E-07	7	7	0.0149	1.04E-09	12	37	0.0663	1.07E-07
50 000	x_0^1	8	8	0.2026	2.90E-11	8	8	0.0429	2.90E-11	7	22	0.1691	4.56E-08
	x_0^2	4	4	0.0842	1.87E-08	4	4	0.0256	1.87E-08	6	19	0.1792	2.22E-08
	x_0^3	5	5	0.0571	6.22E-08	5	5	0.0446	4.37E-08	8	25	17.8536	1.78E-08
	x_0^4	7	7	0.2244	3.51E-07	7	7	0.0427	2.48E-09	13	40	0.3517	1.47E-07
	x_0^5	7	7	0.1762	3.51E-07	7	7	0.0295	2.48E-09	13	40	0.4788	1.47E-07
	x_0^6	7	7	0.0896	1.43E-08	7	7	0.0231	4.33E-09	9	28	0.1600	2.41E-07
	x_0^7	7	7	0.0905	3.51E-07	7	7	0.0420	2.48E-09	13	40	0.3265	1.47E-07
	x_0^8	7	7	0.0833	3.51E-07	7	7	0.0430	2.48E-09	13	40	0.2545	1.47E-07
	x_0^9	F	F	F	F	F	F	F	F	19	58	0.4303	4.58E-08
	x_0^{10}	7	7	0.2301	3.49E-07	7	7	0.0916	2.30E-09	13	40	0.2770	1.58E-07
100 000	x_0^1	8	8	0.4108	4.10E-11	8	8	0.0719	4.10E-11	7	22	0.3374	6.45E-08
	x_0^2	4	4	0.2424	2.65E-08	4	4	0.0496	2.65E-08	6	19	0.2872	3.14E-08
	x_0^3	5	5	0.1244	6.22E-08	5	5	0.0797	4.37E-08	8	25	59.9771	1.78E-08
	x_0^4	7	7	0.3881	4.97E-07	7	7	0.0662	3.50E-09	13	40	0.5892	2.07E-07
	x_0^5	7	7	0.3485	4.97E-07	7	7	0.0794	3.50E-09	13	40	0.6391	2.07E-07
	x_0^6	7	7	0.1863	1.43E-08	7	7	0.0733	4.33E-09	9	28	0.4211	2.41E-07
	x_0^7	7	7	0.3593	4.97E-07	7	7	0.0822	3.50E-09	13	40	0.7034	2.07E-07
	x_0^8	7	7	0.4207	4.97E-07	7	7	0.0636	3.50E-09	13	40	0.4962	2.07E-07
	x_0^9	F	F	F	F	F	F	F	F	19	58	1.1126	6.48E-08
	x_0^{10}	7	7	0.2652	4.98E-07	7	7	0.0650	3.46E-09	13	40	0.5084	1.99E-07

TABLE B.4. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 4 with given initial points and dimensions.

Dimension	x_0	dfnwt				dfsane				msgp			
		#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
	x_0^1	2	2	0.0058	5.17E-08	2	2	0.0058	5.18E-08	5	16	0.0118	2.60E-07
	x_0^2	2	2	0.0078	9.09E-08	2	2	0.0027	9.10E-08	5	16	0.0114	3.96E-07
	x_0^3	2	2	0.0045	9.45E-08	2	2	0.0033	9.46E-08	5	16	0.0115	4.11E-07
	x_0^4	2	2	0.0059	7.51E-08	2	2	0.0035	7.63E-08	5	16	0.0089	3.39E-07
	x_0^5	2	2	0.0144	7.51E-08	2	2	0.0043	7.63E-08	5	16	0.0122	3.39E-07
	x_0^6	2	2	0.0045	9.43E-08	2	2	0.0034	9.43E-08	5	16	0.0226	4.10E-07
1000	x_0^7	2	2	0.0053	7.51E-08	2	2	0.0031	7.63E-08	5	16	0.0112	3.39E-07
	x_0^8	2	2	0.0030	7.51E-08	2	2	0.0023	7.62E-08	5	16	0.0134	3.39E-07
	x_0^9	3	3	0.0034	2.57E-12	3	3	0.0022	2.82E-12	6	19	0.0135	1.11E-08
	x_0^{10}	2	2	0.0048	7.40E-08	2	2	0.0033	7.51E-08	5	16	0.0103	3.39E-07
	x_0^1	2	2	0.0160	1.86E-10	2	2	0.0068	1.86E-10	5	16	0.0273	5.84E-07
	x_0^2	2	2	0.0147	3.27E-10	2	2	0.0058	3.27E-10	5	16	0.0258	8.89E-07
	x_0^3	2	2	0.0123	3.40E-10	2	2	0.0097	3.40E-10	5	16	0.0342	9.23E-07
	x_0^4	2	2	0.0106	2.70E-10	2	2	0.0046	2.74E-10	5	16	0.0427	7.60E-07
	x_0^5	2	2	0.0149	2.70E-10	2	2	0.0092	2.74E-10	5	16	0.0336	7.60E-07
	x_0^6	2	2	0.0154	3.39E-10	2	2	0.0079	3.39E-10	5	16	0.0328	9.23E-07
5000	x_0^7	2	2	0.0129	2.70E-10	2	2	0.0069	2.74E-10	5	16	0.0479	7.60E-07
	x_0^8	2	2	0.0079	2.70E-10	2	2	0.0051	2.74E-10	5	16	0.0384	7.60E-07
	x_0^9	2	2	0.0177	1.14E-08	2	2	0.0082	1.14E-08	6	19	0.0347	2.45E-08
	x_0^{10}	2	2	0.0109	2.68E-10	2	2	0.0096	2.74E-10	5	16	0.0346	7.59E-07
	x_0^1	2	2	0.0115	1.64E-11	2	2	0.0152	1.64E-11	5	16	0.0779	8.26E-07
	x_0^2	2	2	0.0328	2.89E-11	2	2	0.0094	2.89E-11	6	19	0.0902	1.25E-08
	x_0^3	2	2	0.0236	3.01E-11	2	2	0.0107	3.01E-11	6	19	0.1176	1.29E-08
	x_0^4	2	2	0.0118	2.39E-11	2	2	0.0103	2.42E-11	6	19	0.0808	1.06E-08
	x_0^5	2	2	0.0254	2.39E-11	2	2	0.0098	2.42E-11	6	19	0.1006	1.06E-08
	x_0^6	2	2	0.0230	3.00E-11	2	2	0.0136	3.00E-11	6	19	0.0496	1.29E-08
10 000	x_0^7	2	2	0.0222	2.39E-11	2	2	0.0109	2.42E-11	6	19	0.0741	1.06E-08
	x_0^8	2	2	0.0237	2.39E-11	2	2	0.0114	2.42E-11	6	19	0.0697	1.06E-08
	x_0^9	2	2	0.0260	1.01E-09	2	2	0.0122	1.01E-09	6	19	0.0415	3.46E-08
	x_0^{10}	2	2	0.0188	2.36E-11	2	2	0.0126	2.43E-11	6	19	0.1114	1.06E-08
	x_0^1	2	2	0.0475	9.93E-14	2	2	0.0308	9.93E-14	6	19	0.3162	1.83E-08
	x_0^2	2	2	0.0895	9.93E-14	2	2	0.0387	9.93E-14	6	19	0.2823	2.79E-08
	x_0^3	2	2	0.0783	9.93E-14	2	2	0.0417	9.93E-14	6	19	0.3582	2.89E-08
	x_0^4	2	2	0.0897	9.93E-14	2	2	0.0372	9.93E-14	6	19	0.3796	2.38E-08
	x_0^5	2	2	0.0850	9.93E-14	2	2	0.0254	9.93E-14	6	19	0.3152	2.38E-08
	x_0^6	2	2	0.0633	9.93E-14	2	2	0.0298	9.93E-14	6	19	0.4268	2.89E-08
50 000	x_0^7	2	2	0.0886	9.93E-14	2	2	0.0381	9.93E-14	6	19	0.3497	2.38E-08
	x_0^8	2	2	0.0712	9.93E-14	2	2	0.0366	9.93E-14	6	19	0.2974	2.38E-08
	x_0^9	2	2	0.0689	3.57E-12	2	2	0.0312	3.57E-12	6	19	0.3753	7.75E-08
	x_0^{10}	2	2	0.0927	9.68E-14	2	2	0.0394	9.93E-14	6	19	0.2708	2.38E-08
	x_0^1	2	2	0.1756	0	2	2	0.0567	0	6	19	0.6949	2.59E-08
	x_0^2	2	2	0.1803	0	2	2	0.0532	0	6	19	0.7878	3.94E-08
	x_0^3	2	2	0.1982	0	2	2	0.0454	0	6	19	0.5761	4.09E-08
	x_0^4	2	2	0.2140	0	2	2	0.0335	0	6	19	0.4159	3.37E-08
	x_0^5	2	2	0.1873	0	2	2	0.0722	0	6	19	0.6970	3.37E-08
	x_0^6	2	2	0.2016	0	2	2	0.0655	0	6	19	0.6519	4.09E-08
100 000	x_0^7	2	2	0.1380	0	2	2	0.0815	0	6	19	0.7181	3.37E-08
	x_0^8	2	2	0.0927	0	2	2	0.0632	0	6	19	0.7006	3.37E-08
	x_0^9	2	2	0.1371	4.21E-13	2	2	0.0720	4.21E-13	7	22	0.7717	8.30E-08
	x_0^{10}	2	2	0.1736	0	2	2	0.0592	0	6	19	0.6921	3.37E-08

TABLE B.5. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 5 with given initial points and dimensions.

Dimension	x_0	dfnwt #iter	dfnwt #fval	dfnwt time	dfnwt Fnorm	dfsane #iter	dfsane #fval	dfsane time	dfsane Fnorm	msgp #iter	msgp #fval	msgp time	msgp Fnorm
1000	x_0^1	2	2	0.0058	5.17E-08	2	2	0.0058	5.18E-08	5	16	0.0118	2.60E-07
	x_0^2	2	2	0.0078	9.09E-08	2	2	0.0027	9.10E-08	5	16	0.0114	3.96E-07
	x_0^3	2	2	0.0045	9.45E-08	2	2	0.0033	9.46E-08	5	16	0.0115	4.11E-07
	x_0^4	2	2	0.0059	7.51E-08	2	2	0.0035	7.63E-08	5	16	0.0089	3.39E-07
	x_0^5	2	2	0.0144	7.51E-08	2	2	0.0043	7.63E-08	5	16	0.0122	3.39E-07
	x_0^6	2	2	0.0045	9.43E-08	2	2	0.0034	9.43E-08	5	16	0.0226	4.10E-07
	x_0^7	2	2	0.0053	7.51E-08	2	2	0.0031	7.63E-08	5	16	0.0112	3.39E-07
	x_0^8	2	2	0.0030	7.51E-08	2	2	0.0023	7.62E-08	5	16	0.0134	3.39E-07
	x_0^9	3	3	0.0034	2.57E-12	3	3	0.0022	2.82E-12	6	19	0.0135	1.11E-08
	x_0^{10}	2	2	0.0048	7.40E-08	2	2	0.0033	7.51E-08	5	16	0.0103	3.39E-07
5000	x_0^1	2	2	0.0160	1.86E-10	2	2	0.0068	1.86E-10	5	16	0.0273	5.84E-07
	x_0^2	2	2	0.0147	3.27E-10	2	2	0.0058	3.27E-10	5	16	0.0258	8.89E-07
	x_0^3	2	2	0.0123	3.40E-10	2	2	0.0097	3.40E-10	5	16	0.0342	9.23E-07
	x_0^4	2	2	0.0106	2.70E-10	2	2	0.0046	2.74E-10	5	16	0.0427	7.60E-07
	x_0^5	2	2	0.0149	2.70E-10	2	2	0.0092	2.74E-10	5	16	0.0336	7.60E-07
	x_0^6	2	2	0.0154	3.39E-10	2	2	0.0079	3.39E-10	5	16	0.0328	9.23E-07
	x_0^7	2	2	0.0129	2.70E-10	2	2	0.0069	2.74E-10	5	16	0.0479	7.60E-07
	x_0^8	2	2	0.0079	2.70E-10	2	2	0.0051	2.74E-10	5	16	0.0384	7.60E-07
	x_0^9	2	2	0.0177	1.14E-08	2	2	0.0082	1.14E-08	6	19	0.0347	2.45E-08
	x_0^{10}	2	2	0.0109	2.68E-10	2	2	0.0096	2.74E-10	5	16	0.0346	7.59E-07
10 000	x_0^1	2	2	0.0115	1.64E-11	2	2	0.0152	1.64E-11	5	16	0.0779	8.26E-07
	x_0^2	2	2	0.0328	2.89E-11	2	2	0.0094	2.89E-11	6	19	0.0902	1.25E-08
	x_0^3	2	2	0.0236	3.01E-11	2	2	0.0107	3.01E-11	6	19	0.1176	1.29E-08
	x_0^4	2	2	0.0118	2.39E-11	2	2	0.0103	2.42E-11	6	19	0.0808	1.06E-08
	x_0^5	2	2	0.0254	2.39E-11	2	2	0.0098	2.42E-11	6	19	0.1006	1.06E-08
	x_0^6	2	2	0.0230	3.00E-11	2	2	0.0136	3.00E-11	6	19	0.0496	1.29E-08
	x_0^7	2	2	0.0222	2.39E-11	2	2	0.0109	2.42E-11	6	19	0.0741	1.06E-08
	x_0^8	2	2	0.0237	2.39E-11	2	2	0.0114	2.42E-11	6	19	0.0697	1.06E-08
	x_0^9	2	2	0.0260	1.01E-09	2	2	0.0122	1.01E-09	6	19	0.0415	3.46E-08
	x_0^{10}	2	2	0.0188	2.36E-11	2	2	0.0126	2.43E-11	6	19	0.1114	1.06E-08
50 000	x_0^1	2	2	0.0475	9.93E-14	2	2	0.0308	9.93E-14	6	19	0.3162	1.83E-08
	x_0^2	2	2	0.0895	9.93E-14	2	2	0.0387	9.93E-14	6	19	0.2823	2.79E-08
	x_0^3	2	2	0.0783	9.93E-14	2	2	0.0417	9.93E-14	6	19	0.3582	2.89E-08
	x_0^4	2	2	0.0897	9.93E-14	2	2	0.0372	9.93E-14	6	19	0.3796	2.38E-08
	x_0^5	2	2	0.0850	9.93E-14	2	2	0.0254	9.93E-14	6	19	0.3152	2.38E-08
	x_0^6	2	2	0.0633	9.93E-14	2	2	0.0298	9.93E-14	6	19	0.4268	2.89E-08
	x_0^7	2	2	0.0886	9.93E-14	2	2	0.0381	9.93E-14	6	19	0.3497	2.38E-08
	x_0^8	2	2	0.0712	9.93E-14	2	2	0.0366	9.93E-14	6	19	0.2974	2.38E-08
	x_0^9	2	2	0.0689	3.57E-12	2	2	0.0312	3.57E-12	6	19	0.3753	7.75E-08
	x_0^{10}	2	2	0.0927	9.68E-14	2	2	0.0394	9.93E-14	6	19	0.2708	2.38E-08
100 000	x_0^1	2	2	0.1756	0	2	2	0.0567	0	6	19	0.6949	2.59E-08
	x_0^2	2	2	0.1803	0	2	2	0.0532	0	6	19	0.7878	3.94E-08
	x_0^3	2	2	0.1982	0	2	2	0.0454	0	6	19	0.5761	4.09E-08
	x_0^4	2	2	0.2140	0	2	2	0.0335	0	6	19	0.4159	3.37E-08
	x_0^5	2	2	0.1873	0	2	2	0.0722	0	6	19	0.6970	3.37E-08
	x_0^6	2	2	0.2016	0	2	2	0.0655	0	6	19	0.6519	4.09E-08
	x_0^7	2	2	0.1380	0	2	2	0.0815	0	6	19	0.7181	3.37E-08
	x_0^8	2	2	0.0927	0	2	2	0.0632	0	6	19	0.7006	3.37E-08
	x_0^9	2	2	0.1371	4.21E-13	2	2	0.0720	4.21E-13	7	22	0.7717	8.30E-08
	x_0^{10}	2	2	0.1736	0	2	2	0.0592	0	6	19	0.6921	3.37E-08

Table B.6. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 6 with given initial points and dimensions.

		dfnwt				dfsane				msgp			
Dimension	x_0	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
	x_0^1	40	72	0.0533	8.77E-07	31	33	0.0146	7.08E-07	92	277	0.3843	8.19E-07
	x_0^2	40	67	0.0484	8.53E-07	28	32	0.0090	8.72E-07	67	202	0.1304	8.92E-07
	x_0^3	35	64	0.0466	9.95E-07	57	123	0.0349	8.85E-07	51	154	0.1110	8.23E-07
	x_0^4	33	62	0.0293	7.65E-07	16	18	0.0076	1.56E-07	85	256	0.1858	5.73E-07
	x_0^5	35	66	0.0349	9.71E-07	34	36	0.0091	3.42E-07	120	361	0.2866	8.43E-07
	x_0^6	35	56	0.0247	7.43E-07	21	25	0.0055	5.03E-07	73	220	0.1455	6.18E-07
1000	x_0^7	33	62	0.0408	7.65E-07	16	18	0.0068	1.56E-07	78	235	0.1621	6.44E-07
	x_0^8	35	65	0.0554	9.05E-07	34	36	0.0108	3.41E-07	127	382	0.2510	9.76E-07
	x_0^9	41	60	0.0528	9.95E-07	35	41	0.0148	4.12E-07	118	355	0.5038	9.01E-07
	x_0^{10}	94	209	0.1801	5.68E-07	30	32	0.0075	6.20E-07	190	571	0.6026	9.74E-07
	x_0^1	40	69	0.1123	4.83E-07	31	33	0.0210	2.97E-07	86	259	1.0877	7.74E-07
	x_0^2	37	58	0.0677	7.05E-07	25	29	0.0354	7.49E-07	80	241	0.8746	8.51E-07
	x_0^3	34	59	0.0980	9.35E-07	18	29	0.0317	4.67E-07	61	184	1.7790	5.86E-07
	x_0^4	31	53	0.0669	4.43E-07	15	17	0.0178	7.61E-07	129	388	2.3796	8.90E-07
	x_0^5	37	70	0.1432	4.84E-07	28	30	0.0445	9.58E-07	136	409	1.9145	6.72E-07
	x_0^6	33	54	0.1486	5.96E-07	34	58	0.0457	5.78E-07	77	232	1.0393	8.28E-07
5000	x_0^7	31	53	0.2064	4.36E-07	15	17	0.0130	7.61E-07	122	367	1.9055	8.53E-07
	x_0^8	33	60	0.1476	7.02E-07	28	30	0.0182	8.49E-07	132	397	1.6132	9.56E-07
	x_0^9	41	60	0.1488	9.95E-07	35	41	0.0207	9.98E-07	105	316	3.8145	9.82E-07
	x_0^{10}	131	316	0.8585	7.63E-07	25	27	0.0278	9.51E-07	220	661	3.7618	9.11E-07
	x_0^1	41	71	0.2899	9.43E-07	29	31	0.0694	6.35E-07	107	322	3.1690	7.89E-07
	x_0^2	32	50	0.2106	8.24E-07	26	30	0.0605	6.56E-07	72	217	0.8598	6.79E-07
	x_0^3	30	50	0.1837	3.79E-07	18	29	0.0922	2.44E-07	50	151	3.4084	8.22E-07
	x_0^4	31	56	0.2773	4.55E-07	15	17	0.0369	5.32E-07	111	334	4.8736	7.53E-07
	x_0^5	31	54	0.2229	4.92E-07	29	31	0.0295	6.38E-07	138	415	3.7130	9.17E-07
	x_0^6	29	51	0.0876	8.08E-07	22	33	0.0590	8.30E-07	75	226	2.1067	6.79E-07
10 000	x_0^7	31	56	0.3922	4.55E-07	15	17	0.0352	5.32E-07	127	382	3.3177	7.95E-07
	x_0^8	30	56	0.2666	9.34E-07	29	31	0.0633	6.32E-07	122	367	1.8483	9.19E-07
	x_0^9	41	60	0.3137	9.95E-07	38	44	0.0887	3.11E-07	107	322	3.5247	9.91E-07
	x_0^{10}	147	361	1.6474	8.58E-07	30	32	0.0735	1.21E-07	380	1141	11.6596	6.62E-07
	x_0^1	39	68	1.1424	8.22E-07	31	33	0.3314	4.59E-07	94	283	84.4116	9.99E-07
	x_0^2	34	59	0.8731	9.09E-07	29	33	0.2046	5.68E-07	84	253	2.9527	6.51E-07
	x_0^3	29	51	0.8593	8.42E-07	30	65	0.3144	4.92E-07	57	172	48.0228	6.92E-07
	x_0^4	35	67	0.6959	4.17E-07	14	16	0.1137	9.98E-07	168	505	30.1035	8.46E-07
	x_0^5	37	64	1.2775	8.81E-07	32	34	0.2111	4.66E-07	161	484	27.2321	9.92E-07
	x_0^6	28	45	0.4870	9.49E-07	21	38	0.3314	7.15E-07	83	250	50.9847	8.51E-07
50 000	x_0^7	35	67	1.1905	3.27E-07	14	16	0.1393	9.98E-07	151	454	19.1251	7.54E-07
	x_0^8	34	68	1.2195	9.84E-07	32	34	0.3719	4.65E-07	188	565	28.1683	8.33E-07
	x_0^9	41	60	0.9033	9.95E-07	37	43	0.3274	8.67E-08	122	367	59.6917	7.39E-07
	x_0^{10}	233	602	5.9510	9.72E-07	23	25	0.1350	4.91E-07	542	1627	146.3774	9.65E-07
	x_0^1	38	63	2.0318	8.84E-07	31	33	0.3072	5.24E-08	87	262	131.6085	4.84E-07
	x_0^2	32	49	1.1158	8.74E-07	30	34	0.4896	6.62E-07	83	250	93.8113	5.67E-07
	x_0^3	27	44	0.8171	7.75E-07	31	69	0.9839	5.19E-07	44	133	76.9730	8.42E-07
	x_0^4	32	59	2.0740	6.66E-07	15	17	0.1879	2.46E-07	266	799	201.0899	7.64E-07
	x_0^5	34	64	1.9653	9.22E-07	33	35	0.4335	3.45E-07	317	952	110.9200	7.13E-07
	x_0^6	29	50	1.5946	9.39E-07	37	84	0.6884	5.85E-07	133	400	157.4404	9.84E-07
100 000	x_0^7	32	59	1.6270	6.66E-07	15	17	0.2700	2.46E-07	288	865	318.6357	6.43E-07
	x_0^8	33	56	1.1224	5.06E-07	33	35	0.4357	3.45E-07	264	793	119.2909	9.78E-07
	x_0^9	41	60	1.5228	9.95E-07	38	44	0.5275	7.19E-07	123	370	133.8615	9.13E-07
	x_0^{10}	269	708	15.3307	8.60E-07	24	26	0.3958	6.27E-07	962	2887	900.0506	9.47E-07

TABLE B.7. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 7 with given initial points and dimensions.

Dimension	x_0	dfnwt				dfsane				msgp			
		#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
1000	x_0^1	26	26	0.0389	1.59E-08	26	26	0.0154	1.59E-08	9	28	0.0566	8.99E-10
	x_0^2	21	21	0.0306	5.41E-08	21	21	0.0126	5.41E-08	13	40	0.0208	9.71E-09
	x_0^3	11	11	0.0157	2.44E-08	10	10	0.0066	1.85E-07	22	67	0.0344	6.38E-07
	x_0^4	27	27	0.0317	3.05E-08	24	24	0.0125	3.89E-08	33	100	0.0563	6.83E-07
	x_0^5	28	28	0.0408	9.89E-10	24	26	0.0114	3.92E-08	44	133	0.2208	9.78E-07
	x_0^6	12	12	0.0126	9.45E-07	14	14	0.0064	2.35E-08	29	88	0.0773	9.07E-07
	x_0^7	27	27	0.0261	3.05E-08	24	24	0.0114	3.89E-08	31	94	0.0503	8.13E-07
	x_0^8	25	25	0.0186	1.16E-08	24	26	0.0121	3.97E-08	74	223	0.3885	7.70E-07
	x_0^9	30	30	0.0441	7.96E-07	30	30	0.0093	7.96E-07	14	43	0.0321	9.88E-09
	x_0^{10}	22	22	0.0326	6.52E-07	24	24	0.0113	7.12E-07	32	97	0.1153	9.79E-07
5000	x_0^1	30	30	0.0748	6.63E-07	30	30	0.0572	6.63E-07	14	43	0.1062	8.82E-10
	x_0^2	26	26	0.0593	2.65E-08	26	26	0.0413	2.65E-08	10	31	0.0967	1.10E-09
	x_0^3	10	10	0.0648	8.16E-07	11	11	0.0188	2.69E-07	24	73	0.1847	5.44E-07
	x_0^4	30	30	0.1344	7.36E-08	29	29	0.0454	1.98E-08	16	49	0.3468	8.43E-07
	x_0^5	33	33	0.1106	3.90E-12	29	31	0.0354	1.99E-08	26	79	0.8413	9.03E-07
	x_0^6	14	14	0.0428	9.26E-08	15	15	0.0160	2.57E-07	21	64	0.4084	9.31E-07
	x_0^7	30	30	0.0619	7.36E-08	29	29	0.0348	1.98E-08	16	49	0.1925	8.43E-07
	x_0^8	29	29	0.1101	5.50E-07	29	31	0.3515	1.99E-08	18	55	0.2942	6.89E-07
	x_0^9	35	35	0.1438	3.08E-07	35	35	0.0260	3.08E-07	11	34	0.1426	1.23E-09
	x_0^{10}	26	26	0.0631	5.90E-07	29	29	0.0413	3.08E-07	22	67	0.1447	7.25E-07
10 000	x_0^1	32	32	0.1637	6.64E-07	32	32	0.0851	6.62E-07	15	46	0.1205	5.97E-07
	x_0^2	28	28	0.2257	3.70E-08	28	28	0.0747	3.70E-08	11	34	0.2267	8.92E-07
	x_0^3	11	11	0.1028	4.09E-07	12	12	0.0513	3.01E-08	19	58	0.2064	6.81E-07
	x_0^4	29	29	0.2824	8.92E-07	31	31	0.0483	2.86E-08	30	91	0.7539	7.04E-07
	x_0^5	32	32	0.1555	8.92E-07	31	33	0.0601	2.86E-08	24	73	0.9006	6.40E-07
	x_0^6	14	14	0.0606	7.62E-07	16	16	0.0196	5.30E-08	25	76	0.4245	7.66E-07
	x_0^7	29	29	0.1073	8.92E-07	31	31	0.0554	2.86E-08	26	79	0.7302	7.33E-07
	x_0^8	31	31	0.1736	5.61E-07	31	33	0.0836	2.86E-08	24	73	1.8388	9.30E-07
	x_0^9	37	37	0.3976	3.31E-07	37	37	0.0875	2.10E-07	12	37	0.1091	8.83E-07
	x_0^{10}	29	29	0.2560	7.01E-07	31	31	0.0833	3.21E-07	21	64	1.2359	6.93E-07
50 000	x_0^1	37	37	1.0737	1.98E-07	37	37	0.3802	8.31E-07	10	31	0.5562	8.15E-07
	x_0^2	32	32	1.0826	3.48E-07	32	32	0.2863	3.36E-07	13	40	0.7298	5.40E-07
	x_0^3	11	11	0.3151	6.62E-07	12	12	0.1084	7.29E-07	18	55	0.9958	4.99E-07
	x_0^4	33	33	0.7453	1.32E-07	35	35	0.4130	4.31E-07	39	118	9.5604	8.23E-07
	x_0^5	36	36	1.2797	1.84E-07	35	37	0.3255	2.87E-07	32	97	3.0909	8.94E-07
	x_0^6	16	16	0.4167	7.97E-08	17	17	0.1161	1.73E-07	15	46	1.0542	6.70E-07
	x_0^7	33	33	0.5856	1.32E-07	35	35	0.4178	4.31E-07	30	91	7.3606	7.11E-07
	x_0^8	36	36	0.4854	1.69E-07	35	37	0.4059	2.83E-07	26	79	3.5202	7.98E-07
	x_0^9	40	41	0.5997	3.48E-07	43	49	0.3682	3.14E-07	14	43	1.0170	8.14E-07
	x_0^{10}	34	34	0.4836	3.47E-07	35	35	0.2064	5.06E-07	89	268	25.9117	6.76E-07
100 000	x_0^1	39	39	1.1127	1.62E-07	40	40	0.5971	1.63E-07	11	34	1.4187	6.09E-07
	x_0^2	34	34	2.2214	2.85E-07	34	34	0.6097	4.93E-07	7	22	0.9081	5.90E-07
	x_0^3	12	12	0.8829	1.51E-07	13	13	0.3555	1.59E-07	13	40	1.7163	7.90E-07
	x_0^4	34	34	2.1487	7.34E-07	38	38	0.8273	1.85E-07	10	31	1.2102	8.40E-07
	x_0^5	38	38	2.5596	1.42E-07	37	39	0.8403	8.91E-07	10	31	1.2175	6.88E-07
	x_0^6	16	16	0.8450	3.15E-07	17	17	0.2553	5.37E-07	20	61	5.9072	8.52E-07
	x_0^7	34	34	1.1170	7.34E-07	38	38	0.4467	1.85E-07	10	31	1.2018	8.40E-07
	x_0^8	38	38	2.0177	1.39E-07	37	39	0.8196	5.45E-07	10	31	1.2149	6.88E-07
	x_0^9	41	43	2.4680	6.24E-07	41	47	0.6785	8.50E-07	8	25	1.0885	6.47E-08
	x_0^{10}	36	36	2.2039	2.00E-07	38	38	0.4295	8.77E-07	10	31	1.2719	7.93E-07

TABLE B.8. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 8 with given initial points and dimensions.

Dimension	x_0	dfnwt #iter	dfnwt #fval	dfnwt time	dfnwt Fnorm	dfsane #iter	dfsane #fval	dfsane time	dfsane Fnorm	msgp #iter	msgp #fval	msgp time	msgp Fnorm
1000	x_0^1	3	3	0.0231	3.19E-09	3	3	0.0124	3.19E-09	5	16	0.0869	1.53E-07
	x_0^2	2	2	0.0092	3.16E-08	2	2	0.0049	3.16E-08	3	10	0.1150	3.07E-08
	x_0^3	3	3	0.0083	3.81E-14	2	2	0.0039	3.09E-07	7	22	0.1342	2.18E-08
	x_0^4	3	3	0.0096	1.12E-08	3	3	0.0174	3.81E-10	12	37	0.0488	6.98E-08
	x_0^5	3	3	0.0096	6.61E-10	3	3	0.0050	3.84E-10	10	31	0.0377	1.35E-07
	x_0^6	3	3	0.0070	5.40E-09	3	3	0.0355	1.70E-14	8	25	0.0297	7.01E-07
	x_0^7	3	3	0.0093	1.12E-08	3	3	0.0054	3.81E-10	12	37	0.0247	6.98E-08
	x_0^8	3	3	0.0083	6.69E-10	3	3	0.0053	3.88E-10	11	34	0.0220	1.98E-08
	x_0^9	0	0	0.0037	0.00E+00	0	0	0.0052	0.00E+00	0	1	0.0073	0
	x_0^{10}	3	3	0.0084	2.71E-07	3	3	0.0055	8.22E-11	13	40	0.0611	2.00E-07
5000	x_0^1	3	3	0.0227	7.14E-09	3	3	0.0118	7.14E-09	5	16	0.0703	3.43E-07
	x_0^2	2	2	0.0363	7.07E-08	2	2	0.0109	7.07E-08	3	10	0.0319	6.86E-08
	x_0^3	3	3	0.0213	1.04E-11	2	2	0.0076	3.09E-07	7	22	1.2670	2.18E-08
	x_0^4	3	3	0.0570	1.14E-08	3	3	0.0131	8.64E-10	12	37	0.1550	5.00E-08
	x_0^5	3	3	0.0265	1.49E-09	3	3	0.0105	8.66E-10	11	34	0.1191	5.55E-08
	x_0^6	3	3	0.0502	5.34E-09	3	3	0.0121	1.69E-14	9	28	0.1539	4.39E-08
	x_0^7	3	3	0.0259	1.14E-08	3	3	0.0142	8.64E-10	12	37	0.1566	5.00E-08
	x_0^8	3	3	0.0363	1.49E-09	3	3	0.0121	8.68E-10	14	43	0.1882	2.47E-08
	x_0^9	0	0	0.0049	0.00E+00	0	0	0.0048	0.00E+00	0	1	0.0042	0
	x_0^{10}	4	4	0.0673	6.66E-07	3	3	0.0131	1.65E-10	17	52	0.1649	1.52E-08
10 000	x_0^1	3	3	0.0462	1.01E-08	3	3	0.0193	1.01E-08	5	16	0.0914	4.85E-07
	x_0^2	2	2	0.0514	1.00E-07	2	2	0.0140	1.00E-07	3	10	0.0408	9.71E-08
	x_0^3	3	3	0.0266	1.04E-11	2	2	0.0109	3.09E-07	7	22	2.2349	2.18E-08
	x_0^4	3	3	0.0475	1.15E-08	3	3	0.0228	1.22E-09	11	34	0.1849	1.56E-07
	x_0^5	3	3	0.0480	2.10E-09	3	3	0.0266	1.23E-09	11	34	0.1429	3.78E-08
	x_0^6	3	3	0.0370	5.34E-09	3	3	0.0178	1.69E-14	8	25	0.1062	2.83E-07
	x_0^7	3	3	0.0385	1.15E-08	3	3	0.0189	1.22E-09	11	34	0.2799	1.56E-07
	x_0^8	3	3	0.0336	2.11E-09	3	3	0.0226	1.23E-09	12	37	0.2799	1.41E-07
	x_0^9	0	0	0.0071	0.00E+00	0	0	0.0252	0.00E+00	0	1	0.0051	0
	x_0^{10}	4	4	0.0496	7.93E-07	3	3	0.0226	2.49E-10	21	64	0.4556	3.40E-08
50 000	x_0^1	3	3	0.1813	2.26E-08	3	3	0.1080	2.26E-08	6	19	0.6770	1.07E-08
	x_0^2	2	2	0.1506	2.24E-07	2	2	0.0714	2.24E-07	3	10	0.3081	2.17E-07
	x_0^3	3	3	0.0920	1.04E-11	2	2	0.0277	3.09E-07	7	22	18.6093	2.18E-08
	x_0^4	3	3	0.2179	1.23E-08	3	3	0.0886	2.74E-09	9	28	0.9879	1.77E-08
	x_0^5	3	3	0.1760	4.70E-09	3	3	0.0909	2.74E-09	12	37	1.1029	4.07E-07
	x_0^6	3	3	0.2484	5.33E-09	3	3	0.0776	1.69E-14	9	28	0.6215	2.54E-08
	x_0^7	3	3	0.1713	1.23E-08	3	3	0.0720	2.74E-09	9	28	0.5934	1.77E-08
	x_0^8	3	3	0.2158	4.71E-09	3	3	0.0599	2.74E-09	12	37	1.4156	2.86E-07
	x_0^9	0	0	0.0291	0.00E+00	0	0	0.0240	0.00E+00	0	1	0.0431	0
	x_0^{10}	5	5	0.2753	1.33E-15	3	3	0.0717	5.65E-10	24	73	2.6680	1.63E-07
100 000	x_0^1	3	3	0.4324	3.19E-08	3	3	0.1471	3.19E-08	6	19	1.2671	1.52E-08
	x_0^2	2	2	0.2843	3.16E-07	2	2	0.1322	3.16E-07	3	10	0.7252	3.07E-07
	x_0^3	3	3	0.1546	1.04E-11	2	2	0.0826	3.09E-07	7	22	65.3287	2.18E-08
	x_0^4	3	3	0.3276	1.31E-08	3	3	0.1850	3.88E-09	7	22	1.4817	9.87E-07
	x_0^5	3	3	0.4586	6.65E-09	3	3	0.1809	3.88E-09	12	37	2.3369	4.44E-07
	x_0^6	3	3	0.3022	5.33E-09	3	3	0.1835	1.69E-14	9	28	1.9256	1.53E-08
	x_0^7	3	3	0.4076	1.31E-08	3	3	0.1899	3.88E-09	7	22	1.1533	9.87E-07
	x_0^8	3	3	0.3245	6.65E-09	3	3	0.1790	3.88E-09	12	37	2.1227	4.47E-07
	x_0^9	0	0	0.0423	0.00E+00	0	0	0.0398	0.00E+00	0	1	0.0820	0
	x_0^{10}	5	5	0.6733	2.11E-18	3	3	0.1744	7.91E-10	20	61	4.9333	2.03E-07

TABLE B.9. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 9 with given initial points and dimensions.

		dfnwt				dfsane				msgp			
Dimension	x_0	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
1000	x_0^1	F	F	F	F	9	19	0.0711	7.69E-08	50	151	0.5026	9.73E-07
	x_0^2	F	F	F	F	21	30	0.0436	3.77E-07	67	202	0.4269	9.80E-07
	x_0^3	6	10	0.1611	3.92E-11	27	41	0.0285	1.31E-07	32	97	0.3032	5.63E-07
	x_0^4	6	10	0.0222	1.14E-07	8	16	0.0115	5.39E-07	35	106	0.1123	7.46E-07
	x_0^5	7	11	0.0375	4.57E-12	8	16	0.0230	4.30E-07	31	94	0.0933	5.80E-07
	x_0^6	6	10	0.0293	9.34E-10	12	22	0.0118	4.28E-08	48	145	0.3150	8.87E-07
	x_0^7	6	10	0.0291	1.14E-07	8	16	0.0138	5.39E-07	37	112	0.1361	7.37E-07
	x_0^8	7	11	0.0135	4.70E-12	8	16	0.0138	4.22E-07	33	100	0.0804	7.52E-07
	x_0^9	F	F	F	F	20	29	0.0235	7.35E-07	46	139	0.2161	7.70E-07
	x_0^{10}	7	11	0.0390	3.08E-12	9	17	0.0116	6.06E-08	38	115	0.0732	7.30E-07
5000	x_0^1	F	F	F	F	10	22	0.0625	7.30E-08	60	181	1.1448	6.36E-07
	x_0^2	F	F	F	F	9	19	0.0457	9.03E-07	73	220	1.4328	5.90E-07
	x_0^3	F	F	F	F	27	37	0.0764	9.75E-07	79	238	1.1495	8.51E-07
	x_0^4	6	10	0.0250	7.43E-08	9	19	0.0694	2.80E-07	65	196	1.7019	8.20E-07
	x_0^5	6	10	0.0506	4.53E-11	10	20	0.0729	2.75E-08	51	154	1.0233	8.90E-07
	x_0^6	F	F	F	F	24	39	0.0990	1.22E-07	112	337	1.1675	8.92E-07
	x_0^7	6	10	0.0195	7.43E-08	9	19	0.0353	2.80E-07	59	178	0.3586	7.84E-07
	x_0^8	6	10	0.0593	1.08E-10	9	19	0.0447	9.85E-07	43	130	0.4362	7.49E-07
	x_0^9	F	F	F	F	23	34	0.0804	1.46E-07	63	190	2.0268	7.53E-07
	x_0^{10}	6	10	0.0904	3.06E-09	10	20	0.0817	1.35E-08	50	151	1.0678	6.00E-07
10 000	x_0^1	37	88	0.8243	6.05E-07	10	22	0.0858	1.15E-08	60	181	1.1993	9.89E-07
	x_0^2	7	11	0.1119	1.27E-10	10	20	0.0582	9.55E-09	102	307	4.8826	9.01E-07
	x_0^3	F	F	F	F	23	33	0.1650	1.17E-07	113	340	3.3422	5.45E-07
	x_0^4	5	9	0.1246	9.26E-07	10	20	0.0896	1.09E-08	73	220	3.1656	6.60E-07
	x_0^5	7	11	0.1311	1.77E-09	9	19	0.0787	3.11E-07	69	208	3.2133	7.06E-07
	x_0^6	7	12	0.0808	1.37E-10	88	107	0.4210	6.62E-07	107	322	3.6785	8.99E-07
	x_0^7	5	9	0.0664	9.26E-07	10	20	0.0770	1.09E-08	58	175	2.1563	6.59E-07
	x_0^8	7	11	0.1071	1.75E-09	9	19	0.1426	3.15E-07	54	163	1.8405	9.31E-07
	x_0^9	F	F	F	F	24	35	0.3383	5.25E-07	98	295	4.9775	6.22E-07
	x_0^{10}	7	11	0.0529	6.64E-11	10	20	0.0679	2.14E-07	56	169	1.1605	9.42E-07
50 000	x_0^1	F	F	F	F	11	25	0.3264	6.24E-08	62	187	10.0833	7.58E-07
	x_0^2	7	12	0.1846	1.46E-09	12	26	0.8253	6.72E-07	51	154	9.2918	7.31E-07
	x_0^3	7	12	0.3246	6.38E-08	F	F	F	F	53	160	9.7014	8.61E-07
	x_0^4	7	12	0.4709	9.22E-10	10	22	0.1655	6.93E-07	56	169	6.7278	5.70E-07
	x_0^5	7	12	0.3799	1.37E-08	10	22	0.2751	8.16E-07	F	F	F	F
	x_0^6	F	F	F	F	31	44	1.2958	4.08E-07	F	F	F	F
	x_0^7	7	12	0.1788	9.22E-10	10	22	0.3327	6.93E-07	87	262	7.5469	7.97E-07
	x_0^8	7	12	0.4917	1.38E-08	10	22	0.6035	8.08E-07	58	175	12.0776	8.32E-07
	x_0^9	F	F	F	F	25	38	0.9191	8.68E-07	515	1546	201.1150	5.91E-07
	x_0^{10}	7	12	0.4765	3.41E-09	10	22	0.2823	4.48E-07	60	181	9.4425	5.33E-07
100 000	x_0^1	F	F	F	F	10	24	1.1012	3.35E-07	59	178	29.5893	5.30E-07
	x_0^2	7	12	0.3711	4.11E-09	10	22	0.8940	3.07E-07	57	172	22.1624	8.46E-07
	x_0^3	F	F	F	F	F	F	F	F	49	148	23.8793	5.04E-07
	x_0^4	7	12	0.9437	2.61E-09	10	22	0.4221	3.38E-07	58	175	25.5587	7.92E-07
	x_0^5	7	12	1.0413	2.45E-08	10	22	0.3702	9.44E-07	52	157	21.8180	5.14E-07
	x_0^6	F	F	F	F	17	31	0.6667	9.47E-07	50	151	23.8697	5.05E-07
	x_0^7	7	12	0.4419	2.61E-09	10	22	0.3764	3.38E-07	58	175	22.9366	7.86E-07
	x_0^8	7	12	0.9948	2.45E-08	10	22	1.0433	9.54E-07	52	157	22.3546	5.16E-07
	x_0^9	F	F	F	F	26	39	1.1405	4.99E-07	1001	3004	1333.5561	98731745
	x_0^{10}	7	12	0.9773	7.27E-09	11	23	0.8366	2.26E-08	59	178	20.6743	9.16E-07

TABLE B.10. Numerical results for `dfnwt`, `dfsane` and `msgp` for Problem 10 with given initial points and dimensions.

Dimension	x_0	dfnwt				dfsane				msgp			
		#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm	#iter	#fval	time	Fnorm
1000	x_0^1	141	222	0.2429	6.95E-07	F	F	F	F	81	244	0.3434	9.92E-07
	x_0^2	133	209	0.1519	9.33E-07	F	F	F	F	79	238	0.2446	9.83E-07
	x_0^3	159	261	0.1427	9.99E-07	F	F	F	F	47	142	0.2845	9.72E-07
	x_0^4	107	174	0.1232	8.65E-07	F	F	F	F	113	340	0.2719	9.60E-07
	x_0^5	87	143	0.1853	6.61E-07	F	F	F	F	77	232	0.1107	6.66E-07
	x_0^6	146	216	0.2215	9.74E-07	F	F	F	F	79	238	0.1403	9.26E-07
	x_0^7	111	191	0.2251	7.69E-07	F	F	F	F	92	277	0.2784	8.53E-07
	x_0^8	118	185	0.2335	9.00E-07	F	F	F	F	90	271	0.2394	9.00E-07
	x_0^9	F	F	F	F	F	F	F	F	F	F	F	F
	x_0^{10}	249	500	0.1952	8.82E-07	F	F	F	F	883	2650	4.4959	9.96E-07
5000	x_0^1	121	195	0.4829	9.72E-07	F	F	F	F	F	F	F	F
	x_0^2	156	250	0.5675	9.87E-07	F	F	F	F	81	244	0.6790	8.94E-07
	x_0^3	161	261	0.6929	7.20E-07	F	F	F	F	67	202	7.1707	9.81E-07
	x_0^4	113	187	0.5143	9.20E-07	F	F	F	F	F	F	F	F
	x_0^5	143	227	0.9067	9.37E-07	F	F	F	F	F	F	F	F
	x_0^6	138	221	0.5314	9.33E-07	F	F	F	F	81	244	0.8107	1.00E-06
	x_0^7	101	172	0.3722	7.60E-07	F	F	F	F	F	F	F	F
	x_0^8	109	176	0.3498	8.90E-07	F	F	F	F	F	F	F	F
	x_0^9	990	9850	23.4897	9.91E-07	F	F	F	F	F	F	F	F
	x_0^{10}	460	951	2.6919	8.48E-07	F	F	F	F	F	F	F	F
10 000	x_0^1	138	222	0.5810	9.38E-07	F	F	F	F	F	F	F	F
	x_0^2	127	214	0.9605	7.21E-07	F	F	F	F	86	259	1.6437	6.98E-07
	x_0^3	99	167	0.6457	9.66E-07	F	F	F	F	65	196	15.7560	9.66E-07
	x_0^4	135	214	1.6665	6.63E-07	F	F	F	F	F	F	F	F
	x_0^5	126	200	1.5207	9.93E-07	F	F	F	F	F	F	F	F
	x_0^6	153	249	1.1652	9.36E-07	F	F	F	F	87	262	2.6951	9.99E-07
	x_0^7	172	278	1.9711	8.48E-07	F	F	F	F	F	F	F	F
	x_0^8	132	211	1.2663	8.70E-07	F	F	F	F	F	F	F	F
	x_0^9	961	9498	23.2381	8.65E-07	F	F	F	F	F	F	F	F
	x_0^{10}	670	1534	5.9113	7.96E-07	F	F	F	F	F	F	F	F
50 000	x_0^1	124	198	3.1203	9.92E-07	F	F	F	F	F	F	F	F
	x_0^2	119	194	2.6387	6.32E-07	F	F	F	F	198	595	8.5793	9.11E-07
	x_0^3	104	173	2.0316	9.36E-07	F	F	F	F	58	175	126.7026	8.39E-07
	x_0^4	127	205	3.0142	6.67E-07	F	F	F	F	F	F	F	F
	x_0^5	129	193	4.3183	9.49E-07	F	F	F	F	F	F	F	F
	x_0^6	139	233	6.2279	8.72E-07	F	F	F	F	70	211	4.1423	9.76E-07
	x_0^7	94	143	3.5436	7.93E-07	F	F	F	F	F	F	F	F
	x_0^8	125	188	4.0646	8.06E-07	F	F	F	F	F	F	F	F
	x_0^9	245	1090	12.9963	8.50E-07	F	F	F	F	F	F	F	F
	x_0^{10}	F	F	F	F	F	F	F	F	F	F	F	F
100 000	x_0^1	150	244	5.8961	8.89E-07	F	F	F	F	F	F	F	F
	x_0^2	133	213	5.2076	9.94E-07	F	F	F	F	F	F	F	F
	x_0^3	101	168	3.8083	8.27E-07	F	F	F	F	F	F	F	F
	x_0^4	112	185	8.3720	8.91E-07	F	F	F	F	F	F	F	F
	x_0^5	123	179	7.3988	9.91E-07	F	F	F	F	F	F	F	F
	x_0^6	132	220	8.7011	7.23E-07	F	F	F	F	F	F	F	F
	x_0^7	131	212	8.6776	7.57E-07	F	F	F	F	F	F	F	F
	x_0^8	151	234	9.5741	8.54E-07	F	F	F	F	F	F	F	F
	x_0^9	392	2982	68.1927	9.25E-07	F	F	F	F	F	F	F	F
	x_0^{10}	F	F	F	F	F	F	F	F	F	F	F	F

Acknowledgements. We would like to thank Professor Marcos Raydan for providing us with the MATLAB version of the `dfsane` code for numerical comparison. We gratefully acknowledge the educative suggestions and useful comments of the reviewers. The third author acknowledge with thanks, the Department of Mathematics and Applied Mathematics at the Sefako Makgatho Health Sciences University, Pretoria, South Africa.

References

[1] Y. Bing and G. Lin, An Efficient Implementation of Merrill's Method for Sparse or Partially Separable Systems of Nonlinear Equations. *SIAM J. Optim.* **1** (1991) 206–221.

[2] S. Deng and Z. Wan, A diagonal quasi-newton spectral conjugate gradient algorithm for nonconvex unconstrained optimization problems, In: Proceedings of the 5th International Conference on Optimization and Control with Applications. Curtin University (2012) 305–310.

[3] J.E. Dennis and J.J. Moré, A characterization of superlinear convergence and its application to quasi-Newton methods. *Math. Comput.* **28** (1974) 549–560.

[4] E.D. Dolan and J.J. Moré, Benchmarking optimization software with performance profiles. *Math. Program.* **91** (2002) 201–213.

[5] M.A. Gomes-Ruggiero, J.M. Martínez and A.C. Moretti, Comparing algorithms for solving sparse nonlinear systems of equations. *SIAM J. Sci. Statist. Comput.* **13** (1992) 459–483.

[6] A. Griewank and Ph.L. Toint, Local convergence analysis for partitioned quasi-newton updates. *Numer. Math.* **39** (1982) 429–448.

[7] A. Griewank and Ph.L. Toint, On the unconstrained optimization of partially separable functions, In: Nonlinear Optimization 1981. Academic press (1982) 301–312.

[8] A. Griewank and Ph.L. Toint, Partitioned variable metric updates for large structured optimization problems. *Numer. Math.* **39** (1982) 119–137.

[9] C.T. Kelly, A comparison of iteration schemes for chandrasekhar h-equations in multigroup neutron transport. *J. Math. Phys.* **21** (1980) 408–409

[10] N. Krejić, Z. Lužanin and I. Radeka, Newton-like method for nonlinear banded block diagonal system. *Appl. Math. Comput.* **189** (2007) 1705–1711.

[11] W. La Cruz and M. Raydan, Nonmonotone spectral methods for large-scale nonlinear systems. *Optim. Methods Softw.* **18** (2003) 583–599.

[12] W. La Cruz, J.M. Martínez and M. Raydan, Spectral residual method without gradient information for solving large-scale nonlinear systems: theory and experiments. *Citeseer*, Technical Report RT-04-08(https://www.ime.unicamp.br/~martinez/lmrreport.pdf) (2004).

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

[14] W.J. Leong, M.A. Hassan and M.W. Yusuf, A matrix-free quasi-Newton method for solving large-scale nonlinear systems. *Comput. Math. with Appl.* **62** (2011) 2354–2363.

[15] D. Li and M. Fukushima, A globally and superlinearly convergent Gauss–Newton-based BFGS method for symmetric nonlinear equations. *SIAM J. Numer. Anal.* **37** (1999) 152–172.

[16] L. Lukšan, C. Matonoha and J. Vlcek, Problems for nonlinear least squares and nonlinear equations. Technical report (2018).

[17] J.M. Martınez, Practical quasi-Newton methods for solving nonlinear systems. *J. Comput. Appl. Math.* **124** (2000) 97–121.

[18] H. Mohammad, Barzilai-Borwein-like method for solving large-scale nonlinear systems of equations. *J. Nigerian Math. Soc.* **36** (2017) 71–83.

[19] H. Mohammad and M.Y. Waziri, On Broyden-like update via some quadratures for solving nonlinear systems of equations. *Turk. J. Math.* **39** (2015) 335–345.

[20] H. Mohammad and S.A. Santos, A structured diagonal Hessian approximation method with evaluation complexity analysis for nonlinear least squares. *Comput. Appl. Math.* (2018).

[21] J. Nocedal and S.J. Wright, Numerical Optimization. Springer Science (2006).

[22] Z.J. Shi and G. Sun, A diagonal-sparse quasi-Newton method for unconstrained optimization problem. *J. Syst. Sci. Math. Sci.* **26** (2006) 101–112.

[23] Z. Wan, Y. Chen, S. Huang and D. Feng, A modified nonmonotone BFGS algorithm for solving smooth nonlinear equations. *Optim. Lett.* **8** (2014) 1845–1860.

[24] M.Y. Waziri and Z. Abdul Majid, An improved diagonal Jacobian approximation via a new quasi-cauchy condition for solving large-scale systems of nonlinear equations. *J. Appl. Math.* **2013** (2013).

[25] M.Y. Waziri, W.J. Leong, M.A. Hassan and M. Monsi, Jacobian computation-free Newton's method for systems of non-linear equations. *J. Numer. Math. Stoch.* **2** (2010) 54–63.

[26] M.Y. Waziri, W.J. Leong, M. Mamat and A.U. Moyi, Two-step derivative-free diagonally Newton's method for large-scale nonlinear equations. *World Appl. Sci. J.* **21** (2013) 86–94.

[27] S. Wu and H. Wang, A modified Newton-like method for nonlinear equations. *Comput. Appl. Math.* **39** (2020) 1–18.

[28] G. Yu, S. Niu and J. Ma, Multivariate spectral gradient projection method for nonlinear monotone equations with convex constraints. *J. Ind. Manag. Optim.* **9** (2013) 117–129.

[29] G. Yuan and X. Lu, A new backtracking inexact BFGS method for symmetric nonlinear equations. *Comput. Math. Appl.* **55** (2008) 116–129.

[30] L. Zhang, A derivative-free conjugate residual method using secant condition for general large-scale nonlinear equations. *Numer. Algorithms* **83** (2020) 1277–1293.