High-order Krylov-Newton and fast Krylov-Secant Methods for Systems of Non-linear Partial Differential Equations

Hector Klie
Marcelo Rame
Mary Wheeler

CRPC-TR96661
October 1996

Center for Research on Parallel Computation
Rice University
6100 South Main Street
CRPC - MS 41
Houston, TX 77005

HIGHER-ORDER KRYLOV-NEWTON AND FAST KRYLOV-SECANT METHODS FOR SYSTEMS ON NONLINEAR PARTIAL DIFFERENTIAL EQUATIONS

HÉCTOR KLÍE *, MARCELO RAMÉ † AND MARY F. WHEELER ‡

Abstract.

Keywords: secant methods, Krylov subspace methods, nonlinear equations, Newton's method, Broyden's method

AMS(MOS) subject classification: 3504, 35Q35, 35M10

1. Introduction.

The solution of the nonlinear system of equations

$$F(u) = 0, \tag{1}$$

where $F : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$, is cornerstone in many scientific and engineering applications. In not rare cases, the number of variables involved in this problem surpasses the computing capabilities today. Therefore, it is necessary not only to come up with strategies to exploit the mathematical and physical structure of the problem but also to create algorithms that resuse as much as possible the inherent information produced towrd the solution of the problem.

Among several methods, Newton's method and Broyden's method have been two of the main choices to solve (1) [12, 28, 34, 35]. The former is very popular due to its robustness and well known q-quadratic local convergence. The latter is an alternative to the former when the computation of the Jacobian matrix is highly expensive or infeasible to obtain. Broyden's method is an iterative procedure based on Jacobian approximations (through rank-one updates) that obey a secant condition. In general, secant methods (those based on a secant condition) have play an important role in linear and nonlinear programming.

Traditionally, Broyden's method has been considered impractical as a linear solver and consequently, almost forgotten throughout the iterative algorithms literature. Eirola and Nevanlinna [19] revitalized the interest on secant methods for solving iteratively nonsymmetric systems with an algorithm that provides variable approximation to the linear system matrix via rank-one updates which incidentally, it is competitive with the GMRES Krylov iterative solver. This procedure is better known as the EN algorithm and has been subject of theoretical study and implementation enhancements by several authors [18, 19, 43, 45, **?**, 47]. These recent developments have shed light on new connections between secant methods and other well established iterative methods.

Yang in her doctoral thesis [47] provides an interpretation of the EN algorithm for solving nonlinear systems of equations which converge twice as fast as Broyden's method (this result also holds in the linear case). In our particular context, we are

* Department of Computational and Applied Mathematics, Rice University, Houston Texas 77251, USA; E-Mail: klie@rice.edu. Support of this author has been provided by Intevep S.A., Los Teques, Edo. Miranda, Venezuela.

† Department of Computational and Applied Mathematics, Rice University, Houston Texas 77251, USA; E-Mail: marcelo@rice.edu.

‡ Texas Institute for Computational and Applied Mathematics, University of Texas, Austin, Texas 78712, USA; E-Mail: mfw@ticam.utexas.edu

interested in providing and efficient implementation of an inexact version to the nonlinear EN algorithm (NEN) for large scale settings. The inexactness arises as consequence of solving the linear Jacobian equation by an iterative procedure (such as GMRES) to a specified tolerance. Hence, our development falls into the theory of Dembo, Eisenstat and Steihaug [11] which, was later extended in [21, 41, 17, 16] for secant methods.

We propose to update the Arnoldi decomposition on which GMRES is based in order to perform two minimal residual approximation solutions per GMRES call. In this way, we are able to come up with an improved nonlinear step at each nonlinear cycle. These updates lead to implicit Krylov-Broyden updates (i.e., Broyden updates restricted to the underlying Krylov subspace) of the current Jacobian approximation. We name this new approach as the nonlinear Krylov-Eirola-Nevanlinna (KEN) algorithm. The idea can be easily taylored to the inexact Newton method in the form of a higher order procedure: the HOKN algorithm.

Approaches that seek to combine both secant and inexact nonlinear methods has been matter of interest to some researchers [4, 32, 33, 29]. A more recent approach based on the combination of limited memory BFGS and truncated Newton methods is reported by Byrd, Nocedal and Zhu [8] in the context of unconstrained optimization. The Krylov-Broyden update to be described in this paper has been also instrumentary in generating hybrid Krylov-secant methods for solving systems of nonlinear equations. The idea is to replace GMRES calls by cheaper Richardson iterations in the computation of descent directions for $\|F\|$ [**?**]. However, the question of producing faster local methods is addressed here for first time.

The paper is organized as follows. The discussion sets out with Broyden's method and the nonlinear EN algorithm and the subject inexactness. This encompasses Section 2. In Section 3 we suggest a way to perform rank-one updates of the Hessenberg matrix resulting from the Arnoldi factorization and motivate the philosophy behind Krylov-Broyden updates. In Section 4, we describe how the previous development allow us to reuse the Krylov information and devise the KEN and HOKN algorithms. Numerical experiments are in order in Section 5. In Section 6 we give conclusions and further direction of work.

2. Secant methods.

Our goal in this section is to introduce Broyden's method and the nonlinear EN algorithm. The evolutionary path leading to the current nonlinear EN algorithm requires that some of the developments in the linear case be covered first. However, this should serve as further motivation of the ideas in this chapter. We emphasize the essence of the EN algorithm, which incidentally presents a close affinity to higher-order methods derived from Newton's method and already known for around thirty years. The key result is that inexactness can be introduced into these rank-one methods without loosing much of their local rapid convergence.

ASSUMPTION 2.1. *(Standard assumptions).* Consider a nonlinear function $F : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$ for which we seek to solve (1).

- The equation above has a solution at u^*,
- $F' : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^{n \times n} \in L_\gamma(\Omega)$,
- $F'(u^*)$ is invertible.

2.1. Broyden's method. Given $u \approx u^*$ and $M \approx J(u)$, we can find an approximate new Newton step, u^+, by

$$u^+ = u - M^{-1}F(u). \tag{2}$$

Broyden's method computes a new M^+ by means of the following rank-one update

$$M^+ = M + \frac{[F^+(u) - F(u) - Ms]\, g^t}{g^t s}, \tag{3}$$

which, whenever the approximated Jacobian equation is solved exactly, i.e. $Ms = -F(u)$, it reduces to

$$M^+ = M + \frac{F^+(u)\, g^t}{g^t s},$$

for $g^t s \neq 0$. The vector g can be chosen in several ways. For instance, when $g \equiv s$, we obtain the "good Broyden's update" and when $g \equiv M^t [F^+(u) - F(u)]$, we have the "bad Broyden's update" [12] [1].

Applying the Sherman-Morrison-Woodbury formula (**??**) we obtain the corresponding inverse form of (3)

$$(M^+)^{-1} = M^{-1} + \frac{(s - M^{-1}y)\, f^t}{f^t y}, \tag{4}$$

where $y = F^+(u) - F(u)$, $f^t = g^t M^{-1}$ and provided that $f^t y \neq 0$.

In particular, if $F(u) = Ax - b = 0$ is a linear function, then it is not hard to see that (2) represents the instance of a stationary iterative method with preconditioner M. In such case, we have the following formula to update M^{-1} at the ith iteration

$$M_{i+1}^{-1} = M_i^{-1} + \frac{\left(p_i - M_i^{-1}q_i\right) f_i^t}{f_i^t q_i}, \tag{5}$$

with $q_i = r_{i+1} - r_i$, $f_i^t q_i \neq 0$. Here, r_i, denotes the ith residual of the linear iteration. As in the nonlinear case, there are several possible choices for f_i. Yang cites a comprehensive list of choices for which we refer the interested reader to [47]. In summary, she suggests the "good Broyden's update" $f_i = M_i^t p_i$ as the best option. Deuflhard, Freund and Walter [18] incorporate a line-search strategy to refine the proper step length for updating intermediate residuals and solutions. This feature was absent in Broyden's former algorithm [6], making the method to terminate within at most $2n$ steps [24]. However, Broyden's update with projected updates can converge within at most n steps. (See [25, 47] for a detailed discussion on this.) Deuflhard, Freund and Walter found the best choice for this step length is

$$\alpha_i = \frac{f_i^t r_i}{f_i^t q_i},$$

which turns out to give a competitive procedure with GMRES in terms of convergence and floating point operations. Broyden's method for the linear case looks as follows

ALGORITHM 2.1. (Linear Broyden iterative solver)

[1] Throughout this paper, we restrict the attention to the "good" versions of Broyden's update and since it has been observed to be the most effective in practice and it does not introduce a loss of generality to our discussion.

1. Give an initial guess x_0 and inverse preconditioner M_0^{-1}.
2. Compute $r_0 = b - Ax_0$.
3. For $i = 0, 1, \ldots$ until convergence do
   3.1 $p_i = M_i^{-1} r_i$.
   3.2 $q_i = Ap_i$.
   3.3 $M_{i+1}^{-1} = M_i^{-1} + \frac{(p_i - M_i^{-1} q_i) f_i^t}{f_i^t q_i}$. Provided that $f_i^t q_i \neq 0$.
   3.4 $\alpha_i = \frac{f_i^t r_i}{f_i^t q_i}$. Provided that $f_i^t q_i \neq 0$.
   3.5 $r_{i+1} = r_i - \alpha_i q_i$.
   3.6 $x_{i+1} = x_i + \alpha_i p_i$.

Note that except for the update in step 3.3 and defining $f_i = q_i$, for all $i = 0, 1, \ldots$, this algorithm is a general form of a descent method for linear systems. Eisenstat, Elman and Schultz [20] use this presentation to derive the generalized conjugate residual method (GCR) and other three closely related methods.

Given a Jacobian approximation A^k at the kth nonlinear iteration, the nonlinear EN algorithm generates an intermediate descent direction by solving

$$A^{(k)} s^{(k)} = -F^{(k)} \tag{6}$$

and constructing a new secant update

Broyden's method for the nonlinear case relies on equations (2) and (3) above. Assuming that A is the Jacobian approximation at the current nonlinear iteration, one of the major virtue of the method consist of finding the minimal solution in Frobenius norm (i.e., $\|A^+ - A\|_F$) over all matrices satisfying the secant equation

$$A^+ s = y = F^+ - F. \tag{7}$$

Broyden's algorithm can be depicted as follows:

ALGORITHM 2.2. (Nonlinear Broyden)

1. Give an initial guess $u^{(0)}$ and Jacobian approximation M_0.
2. For $k = 0, 1, \ldots$ until convergence do
   2.1 Solve $M^{(k)} s^{(k)} = -F^{(k)}$.
   2.2 Update solution $u^{(k+1)} = u^{(k)} + s^{(k)}$.
   2.3 $q^{(k)} = F^{(k+1)} - F^{(k)}$.
   2.4 $M^{(k+1)} = M^{(k)} + \frac{\left(q^{(k)} - M^{(k)} s^{(k)}\right)\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}}$.

It has be shown (see e.g., [12, 28]) that Broyden's method iterates converge q-superlinearly to $F^* = F(u^*) = 0$ under standard assumptions and given that $\lim_{k\to\infty} u^{(k)} = u^*$, $u^{(k)} \neq u^*$ if and only if

$$\lim_{k\to\infty} \frac{\left\| \left(M^{(k)} - J^*\right) s^{(k)} \right\|}{\left\| s^{(k)} \right\|} = 0. \tag{8}$$

Condition (8) is better known as the *Dennis-Moré characterization* and it is cornerstone in proving local q-superlinear convergence for general secant updates in optimization (see e.g., [12, 13]).

2.2. The nonlinear EN algorithm. provides a new direction based on generating new directions based on a approximation M_{i+1} rather than on M_i at a given ith step. More precisely, the EN algorithm looks one step forward to generarte compared

to Broyden's method. Hence, the computational complexity of the EN algorithm approximately doubles both Broyden's and the GMRES algorithm [18, 47]. However, the it is about twice faster than the other two. Careful implementations in terms of memory management and computation (through restarts, truncation and implicit updates) give apparently slight advantage to the EN algorithm [47].

ALGORITHM 2.3. (Nonlinear EN)

1. Give an initial guess $u^{(0)}$ and Jacobian approximation M_0.
2. For $k = 0, 1, \ldots$ until convergence do
   - 2.1 Solve $M^{(k)} s^{(k)} = -F^{(k)}$.
   - 2.2 $q^{(k)} = F^{(k+1)} - F^{(k)}$.
   - 2.3 $M^{(k+1)} = M^{(k)} + \frac{\left(q^{(k)} - M^{(k)} s^{(k)}\right)\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}}$.
   - 2.4 Solve $M^{(k+1)} \widetilde{s}^{(k)} = -F^{(k)}$.
   - 2.5 Update solution $u^{(k+1)} = u^{(k)} + \widetilde{s}^{(k)}$.

Notice that the direction computed by the nonlinear EN algorithm is a linear combination of the direction delivered by Broyden's method and an extra direction coming from step 2.4. In fact, it can be shown after some algebraic manipulation that

$$
\begin{aligned}
u^{(k+1)} &= u^{(k)} + \widetilde{s}^{(k)} \\
&= u^{(k)} + s^{(k)} + \theta^{(k)} \widehat{s}^{(k)},
\end{aligned}
\tag{9}
$$

where

$$
s^{(k)} = -\left(M^{(k)}\right)^{-1} F^{(k)},
$$

$$
\widehat{s}^{(k)} = -\left(M^{(k)}\right)^{-1} F\left(u^{(k)} + s^{(k)}\right),
$$

and

$$
\theta^{(k)} = \frac{1}{1 - \frac{\left(s^{(k)}\right)^t \widehat{s}^{(k)}}{\left(s^{(k)}\right)^t s^{(k)}}},
$$

provided that $\left(s^{(k)}\right)^t s^{(k)} \neq 0$. Furthermore,

$$
u^{(k+1)} = u^{(k)} - \left(M^{(k)}\right)^{-1} \left[F^{(k)} + \theta^{(k)} F\left(u^{(k)} - \left(M^{(k)}\right)^{-1} F^{(k)}\right)\right],
\tag{10}
$$

for $k = 0, 1, \ldots$

The last expression clearly exhibits that the updated solution is formed by combining a Broyden's step and the damped step of a chord method. The chord step is defined by fixing the Jacobian (its approximation in this case) for some iterations. Incidentally, Kelley presents an updated analysis of this method in [28].

Since the angle between the two directions $s^{(k)}$ and $\widehat{s}^{(k)}$ is defined by

$$
\cos\varphi = \frac{\left(s^{(k)}\right)^t \widehat{s}^{(k)}}{\left\|s^{(k)}\right\| \left\|\widehat{s}^{(k)}\right\|},
$$

the damping parameter $\theta^{(k)}$ can be reformulated as

$$\theta^{(k)} = \frac{1}{1 - \frac{\left\|\widehat{s}^{(k)}\right\|}{\left\|s^{(k)}\right\|} \cos\varphi}.$$

This clearly shows that for mutually orthogonal directions $s^{(k)}$ and $\widehat{s}^{(k)}$, a full chord step is performed. On the other hand, if both entities are identical in direction and magnitude, then the chord step contribution vanishes.

If $M^{(k)} = J^{(k)}$ and $\theta^{(k)} = 1$ for $k = 0, 1, \ldots$ then (10) becomes

$$u^{(k+1)} = u^{(k)} - \left(J^{(k)}\right)^{-1} \left[F^{(k)} + F\left(u^{(k)} - \left(J^{(k)}\right)^{-1} F^{(k)}\right)\right], \tag{11}$$

for $k = 0, 1, \ldots$

This recurrence represents a higher-order modification of Newton's method. Iterates generated by (11) converge q-superlinearly with q-order 3 [36]. These methods were studied by Shamanskii [40] and Traub [42]. They pointed out that even higher-order methods can be built out of a longer sequence of chord steps alternated with regular Newton steps. In a more recent treatment, Kelley names those methods after Shamanskii and compares the particular case (11) numerically against Newton's method [28]. Here, we rather adopt the term *composite Newton's method* for referring to the recurrence (11).

Along the lines of Gay's local convergence analysis for Broyden's method, Yang was able to show that the nonlinear EN algorithm converges n-step q-quadratically for n-dimensional problems [47]. Therefore, as in the linear case, the nonlinear EN method converges twice as fast as Broyden's method.

Hence, the nonlinear EN algorithm converges q-quadratically as Newton's method in the one-dimensional case. Note that the method reduces to a forward finite difference method in 1-D [47] which is sometimes referred to as Steffensen's method [36]. In such case, the above equations give rise to the following recurrence

$$\begin{aligned} s^{(k)} &= -\frac{F^{(k)}}{a^{(k)}}, \\ a^{(k+1)} &= \frac{F\left(u^{(k)} + s^{(k)}\right) - F^{(k)}}{s^{(k)}}, \\ u^{(k+1)} &= u^{(k)} - \frac{F^{(k)}}{a^{(k+1)}}. \end{aligned} \tag{12}$$

The first equality provides a systematic way to adjust the step length within the forward finite difference scheme as the iteration progresses. The steeper the slope $a^{(k)}$ the shorter the step $s^{(k)}$ and, vice versa. Moreover, current derivatives are estimated in terms of the previous derivative rather than two consecutive function values as it occurs with the secant method. It has been proven that the secant method for one-dimensional problem converges 2-step q-quadratically [24]. In terms of complexity, we can easily determine that the EN-algorithm requires one extra function evaluation and two extra floating point operations compared to Broyden's method.

A key point can be made. Broyden's method is to Newton's method what the nonlinear EN method is to composite Newton's method. Hence, it is possible (in fact,

not rare in practice) that the nonlinear EN method produces faster converging iterates than those of Newton's method, specially, when $M^{(0)}$ and $u^{(0)}$ are sufficiently good.

For small and moderate problem sizes, the composite Newton's method and the nonlinear EN method can be efficiently implemented using the LU decomposition of the Jacobian (or its approximation). Note that the underlying LU factorization can be reused to solve two linear systems with different right hand sides. This implies significant savings in pivoting operations whereas the total number of functions evaluations and rank-one updates are reduced due to the higher-order convergence induced by both methods. Kelley observes that this alternation of chord steps and Newton's steps are potentially attractive for large scale problems where the cost of building the Jacobian is computationally expensive compare to function evaluations [28]. The reader can infer that in the setting of large algebraic systems arising from transient problems (i.e., implicit formulation of parabolic equations) it is not unusual to have nearby initial Newton iterates to the root. Here, chord steps may be a plausible and an effective option. In particular, in simulations approaching the steady state (see e.g., [23]).

However, in large scale implementations where iterative methods are virtually a must choice, the efficiency line described by the composite Newton's method and the nonlinear EN method seems to appear as a blur. The problem is that most iterative methods (including almost all known Krylov subspace methods) do not preserve a reusable form in the advent of linear system changes. In other words, the iterative method starts from scratch every time a new Jacobian (or approximation to it) arises. In general, this makes a possible inexact step of the nonlinear EN algorithm as computationally expensive as two steps of an inexact Broyden's method.

Fortunately, as we saw in §§2.2 the GMRES algorithm preserves Krylov information delivered by its intrinsic Arnoldi factorization. However, until now, this information has been restricted to build preconditioners in subsequent utilizations of GMRES within the inexact Newton's method. We show that chord steps can be still performed upon the current underlying Krylov basis. In this way, we are able to preserve much of the integrity of an inexact nonlinear EN algorithm and recover the efficiency that it promises compared to Newton's and Broyden's method.

2.3. Inexactness in secant methods. The issue of inexactness in quasi-Newton methods has been examined in [21, 41]. Reference [21] is of particular interest since in there it shows the local q-superlinear rate of convergence is still attained for the inexact Broyden's method. In fact, those results are a generalization of the work previously developed by Dembo, Eisenstat and Steihaug in [11].

Since the same conditions stated in [21] can be also imposed upon the inexact nonlinear EN algorithm, it is straightforward to show that it produces q-superlinearly convergent iterates. These conditions are given by

$$\lim_{k\to\infty} \frac{\left\| M^{(k)} s^{(k)} + F^{(k)} \right\|}{\left\| F^{(k)} \right\|} = 0,$$

and

$$\lim_{k\to\infty} \frac{\left\| M^{(k)} s^{(k)} + F^{(k)} - F^{(k+1)} \right\|}{\left\| s^{(k)} \right\|} = 0.$$

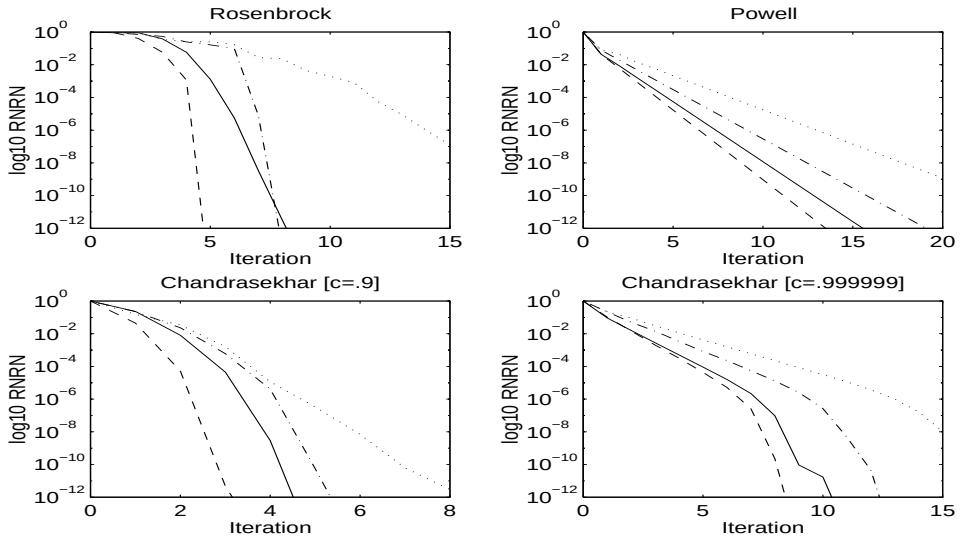

FIG. 1. *Convergenge comparison of Newton's method (dash-dotted line), Broyden's method (dotted line), the composite Newton's method (dashed line) and the nonlinear EN algorithm (solid line) in their inexact versions.*

Clearly, the first condition follows if the forcing terms converge to zero as $k \to \infty$. The second one suggests that the residual should look like the value of the function at the new point with a discrepancy size converging faster to zero than the size of the direction produced for $k \to \infty$. Eisensat and Steihaug show that whenever both of these conditions hold and $u^{(k)} \to u^*$, it follows that the sequence $\left\{u^{(k)}\right\}$ converges in a q-superlinear way.

Rather than going over the lengthy details of this proof, we consider it more illuminating to present the convergence results for the cases exposed in Example **??** with GMRES solving the Jacobian equations.

OJO: CORREGIR The following example corroborates the previous observation. The cases shown there will be frequently brought up as the ideas are developed throughout the present and next chapter. We momentarily look at convergence in terms of nonlinear iterations and leave the discussion on computational cost (i.e., in terms of floating point operations) to Chapter 5.

EXAMPLE 2.1. *XXXXXXXXXXXXXXXXXXXXXXXXX We consider the extended versions of the Rosenbrock function and Powell function described in Appendix B of [12] with initial guesses* $u^{(0)} = (0,1,0,1,\ldots,0,1)^t$ *and* $u^{(0)} = (0,-1,0,1,\ldots,0,-1,0,1)^t$, *respectively. We also consider two variants of a more physical sound problem which arises in radiative heat transfer applications and modeled by the so-called Chandrasekhar H-equation (see [9, 28]):*

$$F(u) = H(u) - \frac{1}{1 - \frac{c}{2}\int_0^1 \frac{uH(\xi)}{u+\xi} d\xi} = 0,$$

with $u \in [0,1]$.

There are two solutions known for a $c \in (0,1)$ *and, as this value approaches one, the problem becomes harder to solve. Here, we closely follow the specifications given in [28]; that is,* $H(u) \equiv u$, $u^{(0)} = (0,0,0,\ldots,0,0)^t$ *and the composite midpoint rule to discretize the integral. The two variants of the H-equation are determined by setting*

$c = .9$ and $c = .999999$. For all four different cases we specify 100 unknown variables. Figure ?? shows the relative nonlinear residual norms (NRNR) against the number of nonlinear iterations for Broyden's method (dotted line), Newton's method (dash-dotted line), the composite Newton's method (dashed line) and the EN method (solid line). For the first and last method the initial Jacobian approximation $M^{(0)} = J^{(0)}$ was defined. The backtracking line-search method was utilized in all methods.

In the case of the Rosenbrock function, both Broyden's method and the EN method were unable to generate a descent direction for $\|F\|$ at the first few steps of the process. In such case it was required to reevaluate the Jacobian by the finite difference approximation. However, in all cases we can see that the nonlinear EN method takes roughly half number the iterations employed by Broyden's method. This reduction surpasses in 50% the reduction in iterations showed by the composite Newton's method over Newton's method. The nonlinear EN method appears to converge faster than Newton's method except in the Rosenbrock case at relative small nonlinear residual norms. In the remaining cases, the nonlinear EN method appears converging superlinearly with a q-order between 2 and 3. Again, this trend breaks down in the Rosenbrock case, where also Broyden's method has serious difficulties and seems to have a q-order close to unity. XXXXXXXXXXXXXXXXXXXXXXXXX

Figure 1 presents the convergence history when GMRES was used as inexact solver of the Jacobian equation. We follow the backtracking line-search strategy and the forcing term selection discussed in Chapter ??. The GMRES restart parameter was chosen to be 30, $\eta_{\max} = .1$ and no preconditioning was specified. As Figure 1 shows there is no apparent change in the convergence of the composite Newton's method and Newton's method. The secant methods instead, show a slight increase in the number of iterations but without altering the convergence margin that both have between each other. Rarely enough, the inexactness and reevaluation of the Jacobian were more beneficial to the nonlinear EN algorithm in achieving better convergence rates than Newton's method itself for the Rosenbrock function. Table 1 and Table 2 complement these results by illustrating the number of GMRES iterations and values of η along the iterations for the particular case of the Chandrasekhar H-equation with $c = .999999$.

3. Exploiting Krylov basis information. The previous discussion motivates us to take advantage of the Krylov information associated with $J^{(k)}$ or its approximation in a different way. Rather than building preconditioners, we restrict the generation of successive descent directions for $\|F\|$ to the current Krylov basis. This implies to perform rank-one updates in the Hessenberg matrix resulting from the Arnoldi factorization (**??**) and implicitly reproduce an approximation of Broyden's update of the Jacobian matrix. Hence, the main objective here is to minimize the direct manipulation of the Jacobian matrix and the use of GMRES as much as possible in the process of converging to the root of F. Note that in contrast to Martínez's approach, we do not perform Jacobian evaluations and secant updates at the same time.

Consider A a an approximation to the current Jacobian matrix J. We are interested in looking at a minimum change to A consistent with $A^+ s = F^+ - F$ restricted to the underlying Krylov subspace. A basis for this subspace arises as result of using an iterative linear solver such as GMRES for solving the approximated Jacobian system with A.

We quote however, that the present development is not only valid for the GMRES algorithm. The *Full Orthogonalization Method (FOM)* also known as the Arnoldi

TABLE 1
Comparison of Broyden's method and Newton's method for solving the the Chandrasekhar H-equation with $c = .999999$.

	Broyden			Newton		
k	RNR	η_k	LI	RNR	η_k	LI
1	2.24e-01	1.00e-01	2	2.24e-01	1.00e-01	2
2	8.45e-02	1.00e-01	1	4.98e-02	1.00e-01	1
3	3.47e-02	1.00e-01	1	1.44e-02	1.00e-01	2
4	1.15e-02	1.00e-01	2	3.40e-03	1.00e-01	2
5	3.96e-03	1.00e-01	2	8.25e-04	1.00e-01	2
6	1.51e-03	1.00e-01	2	2.28e-04	1.00e-01	2
7	6.25e-04	1.00e-01	2	5.49e-05	1.00e-01	3
8	2.21e-04	1.00e-01	2	1.29e-05	1.00e-01	3
9	1.10e-04	1.00e-01	3	2.53e-06	1.00e-01	3
10	3.13e-05	1.00e-01	3	2.65e-07	1.00e-01	3
11	9.05e-06	1.00e-01	3	4.74e-09	1.65e-02	3
12	3.25e-06	1.00e-01	3	4.54e-11	2.72e-03	4
13	9.98e-07	1.00e-01	3	1.29e-15		
14	1.55e-07	1.00e-01	3			
15	5.12e-09	2.88e-02	3			
16	2.53e-10	2.73e-02	3			
17	4.39e-12					

TABLE 2
Comparison of the nonlinear EN and the composite Newton's method for solving the the Chandrasekhar H-equation with $c = .999999$.

	Nonlinear EN			Comp. Newton		
k	RNR	η_k	LI	RNR	η_k	LI
1	9.16e-02	1.00e-01	2	1.04e-01	1.00e-01	1
2	1.60e-02	8.80e-02	2	1.46e-02	1.00e-01	2
3	2.66e-03	1.00e-01	2	2.03e-03	1.00e-01	2
4	4.08e-04	1.00e-01	2	3.14e-04	1.00e-01	2
5	9.69e-05	1.00e-01	3	4.36e-05	1.00e-01	3
6	3.17e-06	1.00e-01	3	5.20e-06	1.00e-01	3
7	6.68e-07	2.66e-02	3	2.50e-07	1.00e-01	3
8	2.82e-08	4.01e-02	3	2.23e-10	1.93e-02	4
9	6.24e-10	1.45e-02	4	1.05e-15		
10	2.23e-11	3.57e-02	3			
11	2.90e-14					

iterative method [39] can be employed for the purposes underlined here. It is important to remark, however, that the GMRES algorithm is still more robust an efficient than this approach [2].

3.1. Updating the Arnoldi factorization. In Section § 2.2 we discussed the role that the Arnoldi process plays in GMRES. It is basically the vehicle to express the minimal residual approximation (**??**) in a more manageable way. The Arnoldi factorization provides valuable information that should not be discarded at all every time a GMRES solution starts over. We now show how to reflect secant updates on the Jacobian matrix without altering the current Krylov basis. For the sake of simplicity, let us omit the sources of inexactness induced by the use of GMRES whose relative residuals are ought to converge at a predefined tolerance (i.e., to a prescribed forcing term value).

Consider the solution to the following approximated Jacobian equation at the kth nonlinear iteration

$$A^{(k)}s^{(k)} = -F^{(k)}, \tag{13}$$

with m steps of the GMRES algorithm. This linear solution can be regarded as embedded in an inexact Broyden's method. Let $s_m^{(k)} = s_0^{(k)} + V^{(k)}y^{(k)}$ be the solution obtained. The associated Krylov subspace for this problem is given by $\mathcal{K}_m^{(k)}\left(A^{(k)}, r_0^{(k)}\right)$. Now, we wish to use the information gathered during the solution of (13) to provide an approximation to the system

$$A^{(k+1)}s^{(k+1)} = -F^{(k+1)}, \tag{14}$$

with corresponding Krylov basis $\mathcal{K}_m^{(k)}(A^{(k+1)}, r_0^{(k+1)})$. Clearly, in general we can not guarantee that $\mathcal{K}_m^{(k+1)}(A^{(k+1)}, r_0^{(k+1)}) = \mathcal{K}_m^{(k)}(A^{(k)}, r_0^{(k)})$. However, rank-one updates onto the corresponding Arnoldi factorization of (13) can be done without destroying the Krylov basis. That is,

$$\left(A^{(k)} + V^{(k)}zw^t\left(V^{(k)}\right)^t\right)V^{(k)} = V^{(k)}\left(H_m^{(k)} + zw^t\right) + h_{m+1,m}^{(k)}v_{m+1}^{(k)}e_m^t, \tag{15}$$

or equivalently,

$$\left(V^{(k)}\right)^t\left[A^{(k)} + V^{(k)}zw^t\left(V^{(k)}\right)^t\right]V^{(k)} = H_m^{(k)} + zw^t, \tag{16}$$

for any vectors $z, w \in \mathbb{R}^m$. Expression (16) suggests a clearer way to update $H_m^{(k)}$ rather than $A^{(k)}$. Note that the current Jacobian approximation appears to be updated by a rank-one matrix whose range lies on $\mathcal{K}_m^{(k)}\left(A^{(k)}, r_0^{(k)}\right)$.

Before proceeding, it would be convenient to express the secant equation

$$A^{(k+1)}s^{(k)} = F^{(k+1)} - F^{(k)}, \tag{17}$$

in terms of a solution lying strictly on the Krylov subspace. Otherwise, this would introduce an implicit secant equation in terms of $A^{(k+1)}$. To remove the shift from the origin, we reformulate (13) as

$$A^{(k)}s^{(k)} = -F^{(k)} - A^{(k)}s_0^{(k)} = r_0^{(k)}, \tag{18}$$

and redefine the final solution as $s_m^{(k)} = V^{(k)} y^{(k)}$, that is, as if the initial guess were zero. Obviously, the associated Krylov basis is the same depicted above. Therefore, the secant equation (17) becomes

$$A^{(k+1)} s^{(k)} = F^{(k+1)} + r_0^{(k)}, \tag{19}$$

for $s^{(k)} = V^{(k)} y^{(k)}$. Multiplying both sides by $\left(V^{(k)}\right)^t$ it readily follows that $H_m^{(k+1)}$ should satisfy the following secant equation

$$H_m^{(k+1)} y^{(k)} = \left(V^{(k)}\right)^t F^{(k+1)} + \beta e_1, \tag{20}$$

where $\beta = \left\| r_0^{(k)} \right\|$. Hence, the Krylov subspace projected version of the secant equation (17) can be written as

$$H_m^{(k+1)} = H_m^{(k)} + \frac{\left(\left(V^{(k)}\right)^t F^{(k+1)} + \beta e_1 - H_m^{(k)} y^{(k)}\right)\left(y^{(k)}\right)^t}{\left(y^{(k)}\right)^t y^{(k)}}. \tag{21}$$

REMARK 3.1. *The form (15) has been previously used in the context of* partial pole assignment *problems in control theory. The idea is to replace a few eigenvalues conforming the spectrum of a matrix A by another set of eigenvalues representing more stable modes within the system. This technique is applied once the Arnoldi process have delivered* $\mathcal{K}_m(A, v)$ *as a small invariant subspace under A for a given vector v. Further details and pointers to this problem can be seen in [38].*

The following theorem states that update (21) yields a modified version of Broyden's update for $A^{(k)}$:

THEOREM 3.1. *Let (21) be the rank-one update of* $H_m^{(k)}$*, then the corresponding update of* $A^{(k)}$ *according to (15) is given by*

$$\begin{aligned} A^{(k+1)} &= A^{(k)} + \frac{\left[P^{(k)} F^{(k+1)} + r_0^{(k)} - \left(P^{(k)} A^{(k)} P^{(k)}\right) s^{(k)}\right]\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}} \\ &= A^{(k)} + \frac{\left[F^{(k+1)} - F^{(k)} - A^{(k)} s^{(k)}\right]\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}} + \\ &\quad - \frac{\left[\left(I - P^{(k)}\right)\left(F^{(k+1)} + A^{(k)} s^{(k)}\right) - A^{(k)} s_0^{(k)}\right]\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}}, \end{aligned} \tag{22}$$

where $P^{(k)} = V^{(k)}\left(V^{(k)}\right)^t$.

Proof. For notational convenience, let us drop the superscripts k and replace the superscripts $k+1$ by the symbol $+$. Thus, in view of (15) choose

$$z = V^t F^+ + \beta e_1 - H_m y = V^t F^+ + \beta e_1 - H_m V^t s,$$

and

$$w^t = \frac{y^t}{y^t y} = \frac{s^t V}{y^t V^t V y} = \frac{s^t V}{s^t s}.$$

Therefore,

$$A^+ = A + Vzw^tV^t = A + \frac{\left(VV^tF^+ + r_0 - VH_mV^ts\right)s^t}{s^ts},$$

since $V\beta e_1 = \beta v_1 = r_0$ and, $s^tVV^t = y^tV^tVV^t = y^tV^t = (Vy)^t = s^t$. Using the Arnoldi factorization we substitute $H_m = V^tAV$ into the above expression.

Thus

$$A^+ = A + \frac{\left(VV^tF^+ + r_0 - VV^tAVV^ts\right)s^t}{s^ts}, \tag{23}$$

which can be split up in the desired form (22). Notice that $P^{(k)}s^{(k)} = s^{(k)}$, since $s^{(k)} \in \mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)$. ∎

We refer to the update (22) as the *Krylov-Broyden update*. Note that the operator $P^{(k)}$ is an orthogonal projector onto the Krylov subspace $\mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)$. That is,

- $\left(P^{(k)}\right)^2 = P^{(k)}$ (Idempotency).
- $\left(P^{(k)}\right)^t = P^{(k)}$ (Symmetry).
- Range$\left(P^{(k)}\right) = \mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)$.

The update of $H_m^{(k)}$ reflects an update of $A^{(k)}$ on a lower dimensional space. The larger the value of m the closer both updates (22) and (24) are to Broyden's update. The following observation provides us with further insights.

REMARK 3.2. *If* $s_0^{(k)} = 0$ *then*

$$\left(V^{(k)}\right)^t A_B^{(k+1)} = \left(V^{(k)}\right)^t A^{(k+1)};$$

furthermore,

$$\left(V^{(k)}\right)^t A_B^{(k+1)} V^{(k)} = H_m^{(k+1)},$$

where $A_B^{(k+1)}$ *is the Jacobian operator resulting from Broyden's update. This stems from the fact that the third term of (22) is orthogonal to* $\mathcal{K}_m^{(k)}\left(A^{(k)}, r_0^{(k)}\right)$.

A little algebra leads to the following alternative form of (22)

$$\begin{aligned} A^{(k+1)} = A^{(k)} &+ \frac{\left(F^{(k+1)} - F^{(k)} - A^{(k)}s^{(k)}\right)\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}} \\ &- \frac{\left[\left(I - P^{(k)}\right)F^{(k+1)} + A^{(k)}s_0^{(k)} - h_{m+1,m}^{(k)} v_{m+1}\left(v_m^t s^{(k)}\right)\right]\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}}, \end{aligned} \tag{24}$$

Assuming $s_0^{(k)} = 0$, the above expression tells us that the departure of (24) from Broyden's update does not only depend on the acute angle between $F^{(k+1)}$ and the underlying Krylov subspace but also on how nearly the columns of $V^{(k)}$ span an invariant subspace of $A^{(k)}$.

Clearly, $H_m^{(k+1)}$ is not necessarily an upper Hessenberg matrix. However, we quote that expression (21) can be efficiently performed by updating a given QR form of $H_m^{(k)}$ (see e.g., [12, 27]). This form is not readily available, instead most standard implementations of GMRES progressively compute a QR factorization of $\overline{H}_m^{(k)}$ as every new column enters the Arnoldi process (recall discussion in §§ **??**). Fortunately, there are efficient ways to perform the QR factorization of $H_m^{(k)}$ by just deleting the last row of $\overline{H}_m^{(k)}$ already factorized in QR form. This requires $\mathcal{O}(m^2)$ floating point operations (see [27, pp. 596-597]). An even more efficient way to obtain this factorization consists of keeping an immediate copy of the QR factorization of $\overline{H}_m^{(k)}$ before applying all previous Givens rotations to the new entering column. In other words, if

$$\overline{H}_{m-1}^{(k)} = Q_{m-1} R_{m-1} = Q_{m-1} \begin{pmatrix} \widetilde{R}_{m-1} \\ 0 \end{pmatrix}$$

is the QR factorization of the augmented Hessenberg at the $(m-1)$th GMRES step and $r^t = \begin{pmatrix} \widetilde{r}^t & \gamma_1 & \gamma_2 \end{pmatrix}$, with $r \in \mathbb{R}^{m-1}$ is the entering column, then the QR factorization of $H_m^{(k)}$ at the mth GMRES step is given by

$$H_m^{(k)} = \begin{pmatrix} Q_{m-1} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \widetilde{R}_{m-1} & \widetilde{r}^t \\ 0 & \gamma_1 \end{pmatrix}. \tag{25}$$

In both cases, it is necessary to use $\mathcal{O}(m^2)$ memory locations for storing the factor Q to keep update (21) within a cost of $\mathcal{O}(m^2)$ floating point operations.

3.2. On the Krylov-Broyden update. Expression (22) is the solution to the problem

$$\min_{B \in \mathbb{R}^{n \times n}} \left\| B - A^{(k)} \right\|_F \quad \text{subject to } \left(P^{(k)} B P^{(k)} \right) s^{(k)} = P^{(k)} F^{(k+1)} + r_0^{(k)}.$$

In fact,

$$\begin{aligned} \left\| A^{(k+1)} - A^{(k)} \right\|_F &= \left\| \frac{\left[\left(P^{(k)} B P^{(k)} \right) s^{(k)} - \left(P^{(k)} A^{(k)} P^{(k)} \right) s^{(k)} \right] \left(s^{(k)} \right)^t}{\left(s^{(k)} \right)^t s^{(k)}} \right\|_F \\ &\leq \left\| P^{(k)} \left(B - A^{(k)} \right) P^{(k)} \right\|_F \left\| \frac{s^{(k)} \left(s^{(k)} \right)^t}{\left(s^{(k)} \right)^t s^{(k)}} \right\| \\ &\leq \left\| B - A^{(k)} \right\|_F, \end{aligned} \tag{26}$$

due to the consistency property of the Frobenius norm and to the fact that the l_2-norm of an orthogonal projector is bounded above by 1. Uniqueness of the solution follows from the convexity of the functional $\left\| B - A^{(k)} \right\|_F$ over all B satisfying $\left(P^{(k)} B P^{(k)} \right) s^{(k)} = P^{(k)} F^{(k+1)} + r_0^{(k)}$.

On the other hand, it similarly follows that expression (21) is the solution to the problem

$$\min_{G \in \mathbb{R}^{m \times m}} \left\| G - H_m^{(k)} \right\|_F \quad \text{subject to } Gy = \left(V^{(k)} \right)^t F^{(k)} + \beta e_1.$$

Theorem 3.1 establishes the equivalence between these two minimization problems. However, other view of update (22) can be stated as follows. Consider $\mathcal{Q}$, the set of matrix quotients of $y = F^{(k+1)} - F^{(k)}$ by $s = u^{(k+1)} - u^{(k)}$ defined by

$$\mathcal{Q} = \{B \in \mathbb{R}^{n\times n} \mid Bs = y\}, \tag{27}$$

and $\mathcal{X}$, the set of matrices generating the same Krylov subspace $\mathcal{K}_m \equiv \mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)$. That is,

$$\mathcal{X} = \{B \in \mathbb{R}^{n\times n} \mid \mathcal{K}_m(B, r_0^{(k)}) = \mathcal{K}_m\}. \tag{28}$$

The resulting matrix $A^{(k+1)}$ in (22) can be thought as the nearest to $A^{(k)}$ of all matrices in $\mathcal{X}$ to $\mathcal{Q}$. Furthermore, if the intersection of these two sets is not empty, then $A^{(k+1)} \in \mathcal{X} \cap \mathcal{Q}$. This observation is key in the construction of least–change secant updates consistent with operators satisfying the standard secant condition and other property prescribed by a given affine subspace (e.g., sparsity pattern, positive definiteness) in $\mathbb{R}^{n\times n}$ (see [14, 15]).

The vectors z and w in (15) are arbitrary in $\mathbb{R}^m$. However, since by assumption, the solution to (13) lies on the Krylov subspace we could pick $V^{(k)}w = Vy^{(k)}/(y^{(k)})^t y^{(k)}) = s^{(k)}/((s^{(k)})^t s^{(k)})$. On the other hand, finding $\tilde{z} = V^{(k)}z$ consistent with the secant equation (19) and having $A^{(k)}$ in $\mathcal{X}$ amounts to solving the following minimization problem

$$\min_{z\in\mathbb{R}^m} \|V^{(k)}z - (y - A^{(k)}s)\|. \tag{29}$$

Since the solution of (29) is given by $z_* = (V^{(k)})^t(y - A^{(k)}s)$, then the update implying the nearest $A^{(k+1)}$ to to $A^{(k)}$ of all matrices in $\mathcal{X}$ to $\mathcal{Q}$ is given by (22). This interpretation is nothing more than a particular case of the general result established by Dennis and Schnabel in [14].

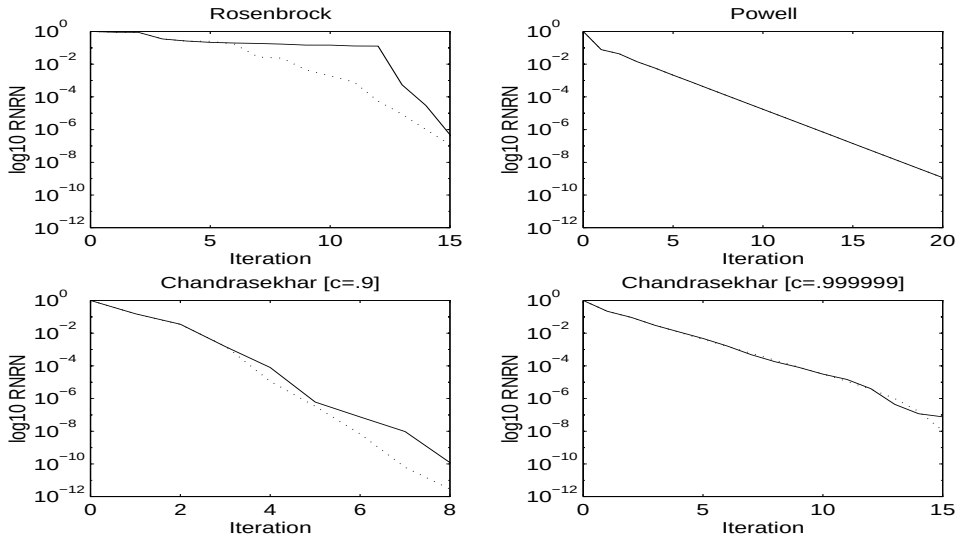

FIG. 2. *Convergence comparison between Broyden's method (dotted line) and the Krylov-Broyden method (solid line).*

Exhaustive experimentation reveals that the last term on the right side of the equality of (24) is "harmless" in the sense that this update produces almost the

same convergence behavior as Broyden's method. In fact, theoretical tools already developed for convergence of Broyden's method in its exact and inexact implementations can be extended to show q-superlinear convergence of update (22) with a few adaptations. The bounded deterioration property for update (24) or (22) holds as a consequence of the bounded deterioration of (21). In the same way the Dennis-Moré characterization can be verified.

EXAMPLE 3.1. *In this example, it is compared the convergence behavior of Broyden's method (dotted line) and of the Krylov-Broyden method (solid line) for the same four cases presented previously. Among all of them, the more noticeable differences are detected in the case of the Rosenbrock function and the Chandrasekhar equation for* $c = .9$. *In the former one, although the Krylov-Broyden method gets stuck within a given region, at some point it starts delivering more rapidly converging iterates and eventually surpasses (not shown) the performance of Broyden's method at relative nonlinear residual norms approaching* 1.0×10^{-10}. *In the easy case of the Chandrasekhar equation, the crossing between both methods performance does not happen but again the Krylov-Broyden method produces faster iterates than Broyden's method does at some points. In the more difficult version of the problem, both curves look alike but with several crossing points. The case of the extended Powell function illustrates a case were the Krylov-Broyden method and Broyden's method perform identically. In this situation, an invariant subspace were generated by the columns of* $V^{(k)}$ *after four GMRES iterations (i.e., a happy breakdown was reached). As it can be observed, nothing can be asserted about which approach may be better. Nevertheless, broader experimentation indicates that there is no major difference between both approaches in general.*

Note that the last term of (15) is preserved after several secant updates of $H_m^{(k)}$ implying that the error size in approximating the eigenvalues of corresponding Jacobian operators remains constant. Therefore, the smaller the term $\left\| h_{m+1,m}^{(k)} v_{m+1} \right\|$ (i.e., the closer the columns of $V^{(k)}$ span an invariant subspace for $A^{(k)}$) the better not only the approximation to the current Jacobian but also the approximation to the eigenvalues of subsequent implicit Jacobians with this approach. This is a key observation and its usefulness shall become more evident in Chapter **??**.

4. Nonlinear Krylov-EN methods. In this section we present two algorithms that make use of the Krylov information generated via GMRES as a device to generate acceptable directions for decreasing relative nonlinear residuals. The first algorithm is an extension of the nonlinear EN algorithm for the inexact case and it is based on only one GMRES solution per iteration. The second algorithm is a high order version of Newton's method and amounts to solving several consecutive residual minimization problems in $\mathbb{R}^{m \times m}$ (with $m \ll n$) until whether the Krylov basis produced by GMRES is exhausted and unable to generate a descent direction for $\|F\|$ or a maximum prespecified user value is exceeded.

4.1. The nonlinear KEN algorithm. We are now in a position to describe an inexact nonlinear version of the EN algorithm that exploits the information left behind by the GMRES method. Hence, we introduce the nonlinear Krylov-Eirola-Nevanlinna (KEN) algorithm as follows.

ALGORITHM 4.1. (Nonlinear Krylov-EN)

1. Give an initial guess $u^{(0)}$ and Jacobian approximation $A^{(0)}$.
2. For $k = 0, 1, \ldots$ until convergence do

2.1 $\left[s^{(k)}, y^{(k)}, H_m^{(k)}, V_m^{(k)}, h_{m+1,m}^{(k)}, \beta \equiv \left\|r_0^{(k)}\right\|\right]$ =GMRES$\left(A^{(k)}, -F^{(k)}, s^{(k)}\right)$.

2.2 $q^{(k)} = \left(V_m^{(k)}\right)^t F^{(k+1)} + \beta e_1$.

2.3 $H_m^{(k+1)} = H_m^{(k)} + \dfrac{\left(q^{(k)} - H_m^{(k)} y^{(k)}\right)\left(y^{(k)}\right)^t}{\left(y^{(k)}\right)^t y^{(k)}}$.

2.4 Solve

$$(30) \qquad \min_{y \in \mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)} \left\|\beta e_1 + \overline{H}_m^{(k+1)} y\right\|, \text{ with } \overline{H}_m^{(k+1)} = \begin{pmatrix} H_m^{(k+1)} \\ h_{m+1,m}^{(k)} e_m^t \end{pmatrix}.$$

Denote its solution by $\widetilde{y}^{(k)}$.

2.5 $\widetilde{s}^{(k)} = V_m^{(k)} \widetilde{y}^{(k)}$.

2.6 Perform

$$A^{(k+1)} = A^{(k)} + P^{(k)} \frac{\left[F^{(k+1)} + r_0^{(k)} - A^{(k)} s^{(k)}\right]\left(s^{(k)}\right)^t}{\left(s^{(k)}\right)^t s^{(k)}},$$

with $P^{(k)} = V^{(k)}\left(V^{(k)}\right)^t$.

2.7 $u^{(k+1)} = u^{(k)} + \widetilde{s}^{(k)}$.

Some comments are in order.

- The Jacobian could be updated by limited memory formulations (subject to be addressed in the next chapter). Note also that, for efficiency purposes, the explicit formulation of $P^{(k)}$ is not required to that end.
- In order to carry out step 2.4 efficiently, we suggest to retrieve the form (25) from GMRES and work upon the Hessenberg QR factorization. Note that the rest of the values returned by GMRES are readily available as part of its machinery and consequently, no extra storage is required.
- We have not included criteria to handle situations where the update in Step 2.6 gives rise to a ill-conditioned system or the update does not generate a descent direction for $\|F\|$. Basically, these situations lead to reset the current Jacobian approximation and restart the process with a new Jacobian approximation (usually obtained by finite differences). Discussion on this topic for the particular context of Broyden's method can be found in [12].

4.2. A higher-order Krylov-Newton algorithm. The higher-order version of the nonlinear KEN algorithm can be attained by performing rank-one updates of the Hessenberg matrix as long as possible before making the next GMRES call. The extend of these updates is determined by the capability the the residual minimization problem in producing descent directions. In this opportunity, we abandon simultaneous Jacobian and Hessenberg updates and check instead if a sufficient amount of decrease of $\|F\|$ is delivered by verifying the condition (**??**).

This presentation allows us to illustrate further uses of the Krylov-Broyden update such as in the context of Newton's method. We stress that further updates to the Hessenberg matrix may result in relative less overhead for a higher order version of the inexact Newton method than a possible higher nonlinear KEN algorithm. The point is that the latter one requires simultaneous updates of $H_m^{(k)}$ and $A^{(k)}$ which may readily increase the total number of updates. Of course, this may be a desirable

situation in terms of rapid convergence updates but it may turn out to be expensive in terms of computer memory use.

The algorithm can be outlined as follows.

ALGORITHM 4.2. (Higher-Order Krylov-Newton)

1. Give an initial guess $u^{(0)}$ and define l_{max}.
2. For $k = 0, 1, \ldots$ until convergence do
   - 2.1 $\left[s^{(k)}, y^{(k)}, H_m^{(k)}, V_m^{(k)}, h_{m+1,m}^{(k)}, \beta \equiv \left\|r_0^{(k)}\right\|\right] = \text{GMRES}\left(J^{(k)}, -F^{(k)}, s^{(k)}\right)$.
   - 2.2 $l = 0$.
   - 2.3 Repeat
     - 2.3.1 $q^{(k+l)} = \left(V_m^{(k)}\right)^t F^{(k+l+1)} + \beta e_1$.
     - 2.3.2 $H_m^{(k+l+1)} = H_m^{(k+l)} + \frac{\left(q^{(k+l)} - H_m^{(k+l)} y^{(k+l)}\right)\left(y^{(k+l)}\right)^t}{\left(y^{(k+l)}\right)^t y^{(k+l)}}$.
     - 2.3.3 Solve

$$\min_{y \in \mathcal{K}_m\left(A^{(k)}, r_0^{(k)}\right)} \left\|\beta e_1 + \overline{H}_m^{(k+1+l)} y\right\|, \text{ with } \overline{H}_m^{(k+l+1)} = \begin{pmatrix} H_m^{(k+l+1)} \\ h_{m+1,m}^{(k)} e_m^t \end{pmatrix}. \quad (31)$$

     Denote its solution by $y^{(k+l+1)}$.
     - 2.3.4 $s^{(k+l+1)} = V_m^{(k)} y^{(k+l+1)}$.
     - 2.3.5 $l = l + 1$.
   - 2.4 Until ($l = l_{max}$) OR $s^{(k+l)}$ is not a decreasing step for $\|F^{(k+l)}\|$. % Note, $\|F^{(k+l)}\| \equiv \|F(u^{(k)} + s^{(k+l)})\|$, for $l = 0, 1, \ldots$
   - 2.5 if $s^{(k+l)}$ is a decreasing step for $\|F^{(k+l)}\|$ then
     - 2.5.1 $u^{(k+1)} = u^{(k)} + s^{(k+l_{max})}$.
   - 2.6 else
     - 2.6.1 $u^{(k+1)} = u^{(k)} + s^{(k+l-1)}$.
3. EndFor

This algorithm can be devised as a variant of the composite Newton's method that seek chord directions belonging to the underlying Krylov subspace. A possible higher-order version of the KEN algorithm can be easily stated from the above presentation by just including the Krylov-Broyden update of $A^{(k)}$ within the repeat loop 2.3. This version should be appealing in situations where Broyden's method or the nonlinear EN are effective compared to Newton's method.

To verify that $s^{(k+l)}$ represents a sufficient decrease for $\|F^{(k+l)}\|$ implies one extra evaluation of F. However, this computation can be reused by a line-search backtracking method following the end of the repeat loop. In general, the failure of this sufficient decrease can be corrected by shortening the step afterwards or by accepting the previous acceptable step as suggested in step 2.6.1.

The following example illustrates the performance of the last two algorithms presented.

Before concluding, a coupled of points need to be addressed. First, how does one perform line-search globalization strategies and forcing term selection criteria in this context of Krylov-Broyden updates? We have implicitly commented on their use throughout the examples without much detailing on their practical implementation. Secondly, what are the effects, if any, on both algorithms due to the use of preconditioners for the Jacobian or its approximations? The reader may have already

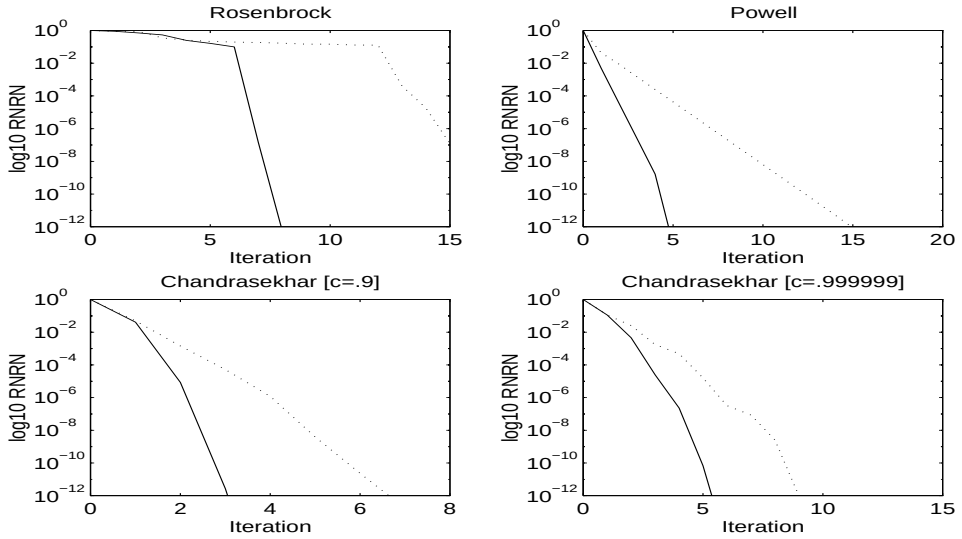

FIG. 3. *Convergence comparison between the nonlinear KEN algorithm (dotted line) and the HOKN algorithm (solid line).*

TABLE 3
Number of successful Hessenberg updates (NHU) and GMRES iterations (LI) in the HOKN algorithm.

	Rosenbrock		Powell		Chand.(c=.9)		Chand.(c=.999999)	
k	NHU	LI	NHU	LI	NHU	LI	NHU	LI
1	0	10	10	4	3	2	3	2
2	0	12	10	4	3	3	4	3
3	1	12	10	4	2	4	10	4
4	1	10	10	4			6	4
5	1	8	10	4			2	4
6	1	6	10	4				
7	2	4						
8	2	4						

suspected some implications due to preconditioning since it must be somehow consistent with successive Krylov-Broyden updates of the preconditioned Jacobian which are implicitly carried out by Hessenberg updates. Both questions are to be discussed in the Chapter **??** in conjunction with three new algorithms based upon the same Krylov-Broyden update philosophy.

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

4.3. Preconditioning. So far, our algorithms have been described under the assumption that we have not employed a preconditioner in GMRES. It is not difficult to realize that, if a preconditioner is used, the update (21) rather than reflecting a secant update of the Jacobian, reflects a secant update of the Jacobian times the inverse of its preconditioner (i.e., assuming right preconditioning). In other words, given $M^{(k)}$ as the preconditioner, (22) would become

$$\left(AM^{-1}\right)^{(k+1)} = A^{(k)}\left(M^{-1}\right)^{(k)} + P^{(k)}\frac{\left(F^{(k+1)} + r_0^{(k)} - A^{(k)}s^{(k)}\right)\left(\widetilde{s}^{(k)}\right)^t}{\left(\widetilde{s}^{(k)}\right)^t\widetilde{s}^{(k)}}, \tag{32}$$

where

$$\widetilde{s}^{(k)} = M^{(k)}s^{(k)} = V^{(k)}y^{(k)}.$$

This means that the spectrum information on which the Richardson relaxation parameters are based corresponds to the form above. Therefore, in order to apply effectively Algorithm **??** we should ensure that the Jacobian operator $J^{(k+1)}$ together with its preconditioner are somehow consistent with the associated Richardson relaxation parameters.

There are three possible ways to overcome this problem. Firstly, we may perform update (32) and carry out the matrix vector products within the Richardson iteration in terms of $(AM^{-1})^{(k+1)}$. This certainly makes the relaxation parameters consistent with the preconditioned Richardson iteration. This approach is equivalent to solving the following preconditioned Jacobian system by Richardson iteration

$$\left(AM^{-1}\right)^{(k+1)}s^{(k+1)} = -F^{(k+1)}. \tag{33}$$

Clearly, in order to obtain a meaningful nonlinear step we need to remove the preconditioning effect embedded in the operator $(AM^{-1})^{(k+1)}$. Unfortunately, there is no explicit form of the preconditioner leading to the right unpreconditioned solution. One possible approximation to the problem is to perform the Krylov-Broyden update $(M^{-1})^{(k)}$ by means of the Sherman-Morrison-Woodbury formula, that is, compute

$$\left(M^{-1}\right)^{(k+1)} = \left(M^{-1}\right)^{(k)} + \frac{\left[s^{(k)} - \left(M^{-1}\right)^{(k)}q^{(k)}\right]\left(s^{(k)}\right)^t\left(M^{-1}\right)^{(k)}}{\left(s^{(k)}\right)^t\left(M^{-1}\right)^{(k)}q^{(k)}}, \tag{34}$$

where

$$q^{(k)} = P^{(k)}\left(F^{(k+1)} + r_0^{(k)}\right),$$

and apply it to the preconditioned solution delivered at convergence by the Richardson iteration. This implies the solution of the linear system

$$\left[\left(AM^{-1}\right)^{(k+1)} M^{(k+1)}\right] s^{(k+1)} = -F^{(k+1)}. \tag{35}$$

Note that the solution of this linear system does not necessarily represent that obtained from a Krylov-Broyden update. Moreover, the operator $\left(AM^{-1}\right)^{(k+1)} M^{(k+1)}$ may introduce a significant overhead in the implementation of a globalization strategy and in the computation of the future updates where the Jacobian (or an approximation of it) is required. Consequently, its manipulation may cause misleading situations where even rapidly convergent Richardson iterants for solving (35) lead to poorly nonlinear steps (i.e., insufficient in producing a descent direction for $\|F\|$).

The following theorem provides an upper bound for this approximation with respect to the Krylov-Broyden update of the Jacobian. For notational simplicity (as in the proof of Theorem 3.1) we drop the superscript on k and adopt the conventional $+$ sign to indicate the operators updated by the Krylov-Broyden update.

THEOREM 4.1. *Let the Krylov-Broyden update of AM^{-1} be given by (32). Also, let the Krylov-Broyden update of both A and M be given by the formula (22), then*

$$\begin{aligned}\left\|\left(AM^{-1}\right)^{+} M^{+} - A^{+}\right\| \leq & \frac{\left\|I - AM^{-1}\right\|}{\|s\|}\left(\|q - As\| + \kappa(M)\,\|A\|\,\|s\|\right) \\ & + \frac{\|q - As\|}{\|s\|}\left(1 + \|q\|\right)\kappa(M).\end{aligned} \tag{36}$$

where $q = F^{+} + r_0$, $\kappa(M) = \|M\|\left\|M^{-1}\right\|$ and, provided that $\|s\| \neq 0$.

Proof. A simple algebraic manipulation yields the following expression

$$\begin{aligned}\left(AM^{-1}\right)^{+} M^{+} = & A + AM^{-1}P\frac{(q - Ms)\,s^t}{s^t s} + P\frac{(q - As)\,(Ms)^t M}{(Ms)^t Ms} \\ & + \frac{(Ms)^t y}{(Ms)^t Ms}\cdot P\frac{(q - As)\,s^t}{s^t s} - P\frac{(q - As)\,s^t}{s^t s}.\end{aligned}$$

In this development we have used the fact that $(Ms)^t P = (Ms)^t$ (i.e., Ms belongs to the Krylov subspace) in order to split the product of the two rank-one terms in the last two terms appearing at the right hand side of the above expression. Hence,

$$\begin{aligned}\left(AM^{-1}\right)^{+} M^{+} - A^{+} = & \left(AM^{-1} - I\right) P\frac{(q - As)\,s^t}{s^t s} \\ & + P(q - As)\left[\frac{(Ms)^t y}{(Ms)^t Ms}\cdot\frac{s^t}{s^t s} + \frac{(Ms)^t M}{(Ms)^t Ms}\right] \\ & + AM^{-1}P\left(AM^{-1} - I\right) Ms\frac{s^t}{s^t s}.\end{aligned}$$

Taking norm on both sides and noting that $\|P\| \leq 1$, we obtain

$$\begin{aligned}\left\|\left(AM^{-1}\right)^{+} M^{+} - A^{+}\right\| \leq & \left\|I - AM^{-1}\right\|\frac{\|q - As\|}{\|s\|} + \|q - As\|\left(\frac{\|y\|}{\|s\|} + \frac{\|M\|}{\|Ms\|}\right) \\ & + \left\|I - AM^{-1}\right\|\left\|AM^{-1}\right\|\frac{\|Ms\|}{\|s\|}\end{aligned}$$

Thus

$$\begin{aligned}\left\| \left(AM^{-1}\right)^{+} M^{+} - A^{+} \right\| \leq & \frac{\left\| I - AM^{-1} \right\|}{\|s\|} \left(\|q - As\| + \left\| AM^{-1} \right\| \|Ms\| \right) \\ & + \|q - As\| \frac{\|Ms\| \|y\| + \|M\| \|s\|}{\|Ms\| \|s\|} \\ \leq & \frac{\left\| I - AM^{-1} \right\|}{\|s\|} \left(\|q - As\| + \kappa(M) \|A\| \|s\| \right) \\ & + \frac{\|q - As\|}{\|s\|} (1 + \|q\|) \kappa(M).\end{aligned}$$

∎

REMARK 4.1. *Note that the first term at the right hand side of (36) vanishes if $M = A$, leaving us with the following upper bound*

$$\left\| I^{+} A^{+} - A^{+} \right\| \leq \frac{\|q - As\|}{\|s\|} (1 + \|q\|) \kappa(M).$$

Moreover, a sharp relative error bound can be easily obtained by working directly with Krylov-Broyden update of the identity matrix,

$$\frac{\left\| I^{+} A^{+} - A^{+} \right\|}{\left\| A^{+} \right\|} \leq \frac{\|(q - As)\|}{\|s\|}.$$

This bound arises since I^{+} is a rank-one perturbation of the identity matrix which by itself, perturbs the Krylov-Broyden update of A. In view of this, we conclude that there is no way to recover the unpreconditioned Krylov-Broyden update exactly.

Although the upper bound (36) is not sharp, it suggests that well conditioned Jacobian operators with effective preconditioning may help this approach approximate closely the nonlinear step delivered by the Krylov-Broyden update.

The second approach consists of updating $J^{(k)}$ and $\left(M^{-1}\right)^{(k)}$ separately according to (22) and in that fashion, use them to carry out the Richardson iteration. In other words, solve

$$\left[A^{(k+1)} \left(M^{-1} \right)^{(k+1)} \right] \left(M^{(k+1)} s^{(k+1)} \right) = F^{(k+1)}. \tag{37}$$

Now, there is no guarantee that the relaxation parameters are necessarily adequate to the linear system problem. However, in contrast to the previous approach, we are reproducing and solving a linear system arising from a true Krylov-Broyden update. Additionally, since the Jacobian is available, clearer and more efficient implementations of a globalization method and future Krylov-Broyden updates can be carried out. In the HKS-B and HKS-EN algorithms, recoveries from Richardson iteration failures are handled by GMRES without strict recomputation of all current operators, whereas failures in generating a descent direction for $\|F\|$ causes the penalty of reevaluating the Jacobian plus the cost invested in a useless convergent Richardson iteration (as it may occur with the first approach).

Note that the discrepancy between $\left(AM^{-1}\right)^{(k+1)}$ and $A^{(k+1)} \left(M^{-1}\right)^{(k+1)}$ is larger than $\left(AM^{-1}\right)^{(k+1)} M^{(k+1)}$ and $A^{(k+1)}$. In fact, using the notation in Theorem 4.1, it

follows that

$$\left\| \left(AM^{-1}\right)^{+} - A^{+}(M^{-1})^{+} \right\| \leq \left\| \left(AM^{-1}\right)^{+} M^{+} - A^{+} \right\| \left\| \left(M^{-1}\right)^{+} \right\|. \tag{38}$$

Hence, this upper bound magnifies by $\left\| (M^{-1})^{+} \right\|$ the upper bound (36). Unfortunately, this adds an extra penalty factor to the reliability of the Richardson relaxation parameters. Nevertheless, good preconditioning should hopefully attenuate this negative effect as the Theorem 4.1 itself claims.

In summary, both approaches differ in the following sense. While the first approach may affect the nonlinear convergence the second one may hurt the linear convergence of Richardson. Generally speaking, both problems result from the fact that rank-one updates do not distribute with respect matrix products.

The other evident inconvenient is that there is an important cost associated to updating two operators (i.e., $(AM^{-1})^{(k)}$ and $(M^{-1})^{(k)}$ or $A^{(k)}$ and $(M^{-1})^{(k)}$) at every nonlinear step. It is worth to remark, however, that the explicit form of any of these operators is not required. We only need the action of the preconditioner onto a given residual. Hence, for a relatively moderate incremental cost (given by one inner product and an AXPY) we can carry out the action of both the updated Jacobian and its updated preconditioner as a function of the oldest one (of course, allowing a linear growth in memory utilization as directions are to be stored). In the next section we discuss more in depth this particular topic.

The third approach is basically a simplification of the last two approaches. Update (32) is determined by the term $A^{(k)} \left(M^{-1}\right)^{(k)}$ plus a rank-one matrix whose column basis is given by a multiple of the vector $P^{(k)} \left(F^{(k+1)} + r_0^{(k)} - A^{(k)} s^{(k)}\right) \in K\left(A^{(k)} \left(M^{-1}\right)^{(k)}, r_0^{(k)}\right)$. This vector is independent of the preconditioner, so one may skip the update of the preconditioner for the sake of approximating $\left(AM^{-1}\right)^{(k+1)}$. In other words, to implement the HKS-B and the HKS-EN algorithms, we propose to update the Jacobian via the Krylov-Broyden update (32) and retrieve the unpreconditioned solution without changing the preconditioner operator at all. This implies that the preconditioner does not have to be rebuilt and updated at every nonlinear step. Although this may represent substantial savings, we stress that the quality of the preconditioner may deteriorate as the nonlinear procedure advances. The point is that the preconditioner does not evolve in agreement to the undergoing Krylov-Broyden updates of each Jacobian matrix.

In view of this approach and the particular case of the nonlinear KEN and the HOKN algorithm, the incorporation of preconditioning forces us to approximate $\left(AM^{-1}\right)^{(k+1)} M^{(k)}$. It is clear there that the solution of the minimal residual approximation problems (30) and (31) is referred to the linear system (33), with the exception that the value of the nonlinear function is taken at exception that the value of the nonlinear function is taken at the kth step. Although this may imply the adoption of some of the potential difficulties already discussed in the case of that approach, fortunately this does not occur here. In order to perform globalization strategies we do not need the explicit form of the Jacobian matrix, or there is no purpose in either updating the Jacobian by a Krylov-Broyden update. There is even a much stronger reason: the failure of the Krylov-Broyden step (i.e., delivered by the least squares solution of the preconditioned problems (30) and (31)) compromise much less possible unwasted computation than the HKS-B and the HKS-EN algorithms. The key

observation stems from the fact that the KEN and HOKN algorithms does not imply recomputation of the Jacobian and its preconditioner if the computation providing the higher-order step fails.

Thus, fixing the preconditioner is a suitable approach in the nonlinear KEN and HOKN algorithms. Obviously, any attempt to update the preconditioner introduces a relative high overhead to a computation that does not require direct manipulation with the Jacobian matrix. Furthermore, the following theorem shows that the best approach is to keep the preconditioner fixed in all Krylov-secant algorithms.

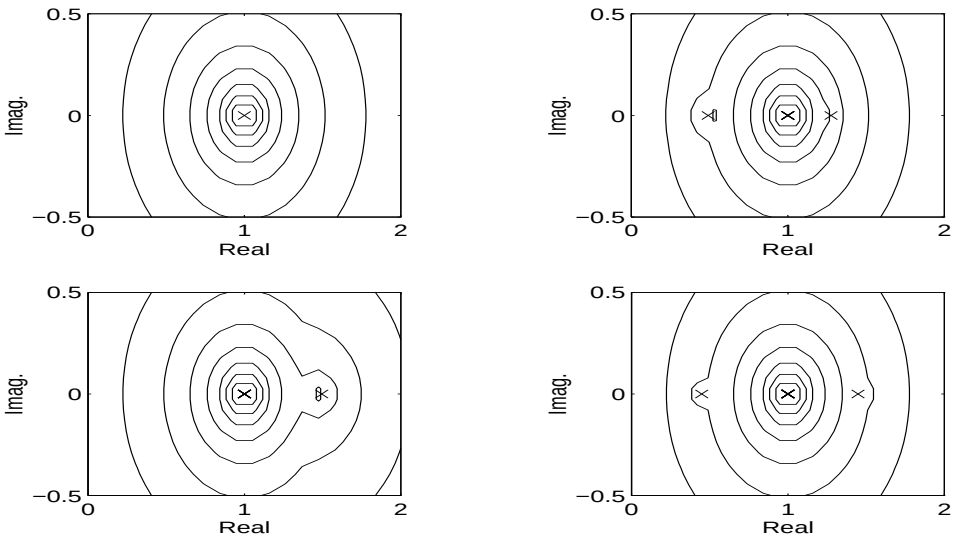

FIG. 4. *Pseudospectra of preconditioned Jacobian matrices for the extended Rosenbrock function. Upper left corner: AM^{-1}; upper right corner: $A^{+}M^{-1}$; lower left corner $A^{+}(M^{-1})^{+}$ and, lower right corner: $(AM^{-1})^{+}$.*

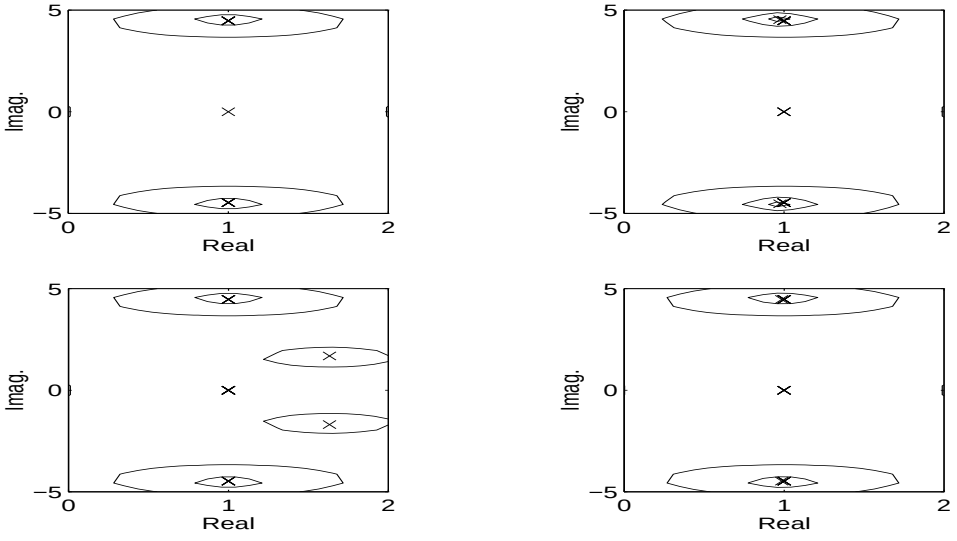

FIG. 5. *Pseudospectra of preconditioned Jacobian matrices for the Powell singular function. Upper left corner: AM^{-1}; upper right corner: $A^{+}M^{-1}$; lower left corner $A^{+}(M^{-1})^{+}$ and, lower right corner: $(AM^{-1})^{+}$.*

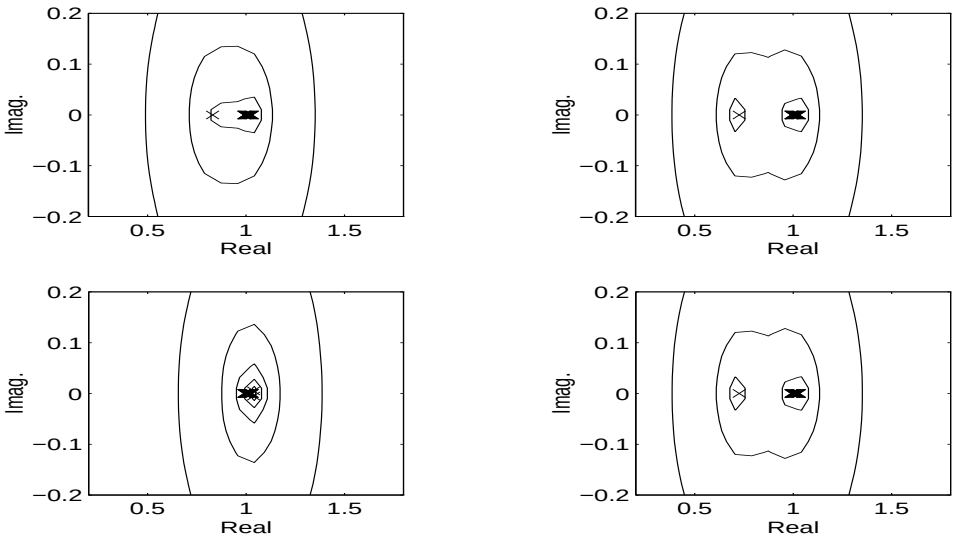

FIG. 6. *Pseudospectra of preconditioned Jacobian matrices for the easy case of the Chandrasekhar H-equation. Upper left corner:* AM^{-1}*; upper right corner:* $A^{+}M^{-1}$*; lower left corner* $A^{+}(M^{-1})^{+}$ *and, lower right corner:* $(AM^{-1})^{+}$*.*

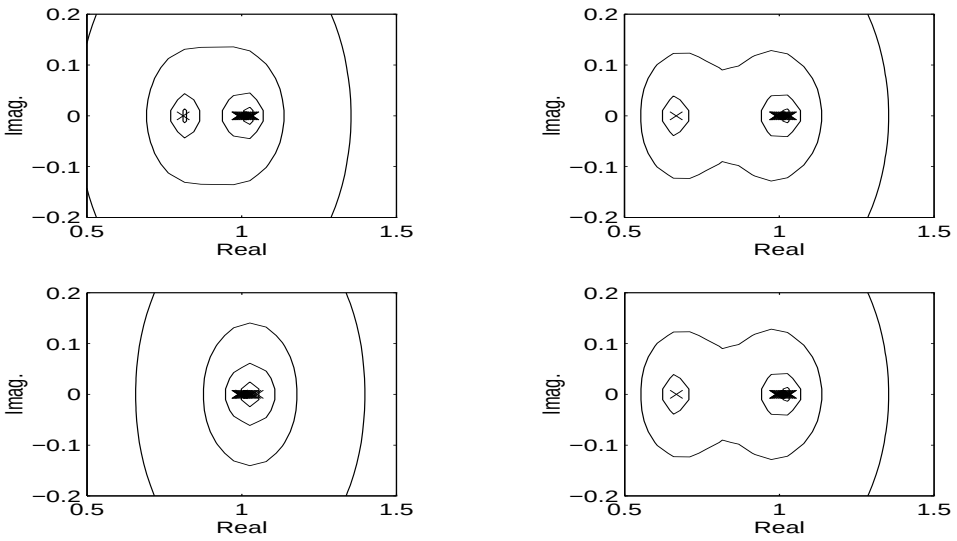

FIG. 7. *Pseudospectra of preconditioned Jacobian matrices for the hard case of the Chandrasekhar H-equation. Upper left corner:* AM^{-1}*; upper right corner:* $A^{+}M^{-1}$*; lower left corner* $A^{+}(M^{-1})^{+}$ *and, lower right corner:* $(AM^{-1})^{+}$*.*

THEOREM 4.2. *Let the Krylov-Broyden update of AM^{-1} be given by (32). Also, let M be a preconditioner for A and, let the Krylov-Broyden update of A be given by the formula (22), then*

$$\left\| \left(AM^{-1}\right)^{+} M - A^{+} \right\| \leq \frac{\|q - As\|}{\|s\|} \kappa(M). \tag{39}$$

where $q = F^{+} + r_0$, $\kappa(M) = \|M\| \left\|M^{-1}\right\|$ and, provided that $\|s\| \neq 0$.

Proof. It easily follows that

$$\left(AM^{-1}\right)^{+} M - A^{+} = P \frac{(q - As)}{(Ms)^t Ms} (Ms)^t M. \tag{40}$$

Taking the l_2-norm on both sides, it results

$$\begin{aligned} \left\| \left(AM^{-1}\right)^{+} M - A^{+} \right\| &\leq \|P\| \, \|q - As\| \, \frac{\|M\|}{\|Ms\|} \frac{\left\|M^{-1}\right\|}{\|M^{-1}\|} \\ &\leq \frac{\|q - As\|}{\|s\|} \kappa(M). \end{aligned}$$

∎

The result above applies directly to the nonlinear KEN and the HOKN algorithms. To characterize the difference between the new preconditioned Jacobian matrix and that implicitly associated to the Richardson relaxation parameters in the HKS-B and HKS-EN algorithms, we again have that

$$\begin{aligned} \left\| \left(AM^{-1}\right)^{+} - A^{+} M^{-1} \right\| &\leq \left\| \left(AM^{-1}\right)^{+} M - A^{+} \right\| \left\| M^{-1} \right\| \\ &\leq \frac{\|q - As\|}{\|s\|} \left\| M^{-1} \right\| \kappa(M). \end{aligned}$$

The entire discussion boils down to realizing that rather than maintaining (or enhancing) the quality of the preconditioner, we should better let the preconditioner change in close correspondence to update (22). Therefore, the performance of preconditioned Krylov-secant methods is dictated by how close the combined updated Jacobian and its preconditioner are to $\left(AM^{-1}\right)^{(k+1)}$ and how this itself, is close to the identity matrix. Obviously, maintaining this consistency (or resemblance) among these operators does not prevent the use of the best preconditioning strategy each time the GMRES iterative solver is required.

The following example illustrates very clearly all the above discussion.

EXAMPLE 4.1. *Figures 7-4 show pseudospectra plots of the extended Rosenbrock function, the extended Powell function and the two cases of the Chandrasekhar H-equation. In every Figure, subplots for AM^{-1}, $A^{+}M^{-1}$, $A^{+}\left(M^{-1}\right)^{+}$ and $\left(AM^{-1}\right)^{+}$ are presented. All plots are generated in terms of the first and second nonlinear iteration. A tridiagonal preconditioner was employed in all cases. The reader can realize the close pseudospectra similarity shown by the operators $A^{+}M^{-1}$ and $\left(AM^{-1}\right)^{+}$ in all problem cases, which confirms the result established in Theorem 4.2. The Rosenbrock function case perfectly illustrates how the Krylov-Broyden updates may cause certain quality deterioration of the preconditioned new Jacobian. In this particular, $A^{+}\left(M^{-1}\right)^{+}$ presents a slightly better condition number than both $A^{+}M^{-1}$ and*

$(AM^{-1})^+$. *However, this situation does not always hold as the Powell singular function subplots indicate. Note, that one conjugate eigenvalue pair of* $A^+ (M^{-1})^+$ *would be out of the convex hull (i.e., the GMRES lemniscate) associated to* $(AM^{-1})^+$. *This may negatively affect Richardson's rate of convergence as it was discussed in* §§4.1.3. *The Chandrasekhar H-equation shows that one may obtain a better conditioned matrix* $A^+ (M^{-1})^+$ *than* AM^{-1}. *This trend is emphasized from the easiest to the most difficult case of this nonlinear integral equation.*

OJO: CORREJIR Since the procedure to introduce preconditioning in the HKS algorithms has been established, it is now convenient to show its performance in practice.

XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

5. Computational experiments. This chapter encompasses numerical experimentation of both Krylov-secant methods and preconditioners for coupled linear systems. The first two sections of the chapter are devoted to analyze the performance of each one separately. The last section introduces ideas from these two approaches in a parallel black-oil reservoir simulator described by Wheeler and Smith in [46] and later improved by Dawson *et al.* in [10].

In this section we present numerical experiments to illustrate the effectiveness of the algorithms presented here, i.e., the nonlinear KEN and higher order Krylov-Newton (HOKN).

The discussion begins reviewing the example cases shown throughout Chapter **??** and Chapter **??**: the extended Rosenbrock's function, the extended singular Powell's function and two levels of difficulty of the Chandrasekhar H-equation.

Two additional problems were also chosen for these tests. The first of them is a nonlinear steady-state equation known as the *(modified) Bratu problem.* This is a model for the steady-state temperature distribution in reacting systems in two space dimensions and is included here because it has been used repeatedly as a test bed for inexact Newton methods [3, 22, 37].

The second example involves a simplification of *Richards' equation*, which is used to model groundwater transport in the unsaturated zone. This time-dependent model in two space dimensions serves as a window to observe the Krylov-secant algorithms in action for underground simulation applications. We believe that this (or a similar) algorithm should benefit reservoir simulators in use by the petroleum and environmental industries. This should prepare the ground for the forthcoming experimentation on a parallel two-phase reservoir simulator in § 3.

All Krylov-secant methods are compared to Newton's method, the composite Newton's method, Broyden's method and the nonlinear Eirola-Nevanlinna algorithm throughout this section. All of them are presented in their inexact versions. More specifically, the Jacobian or Newton equation is solved by GMRES each time in conjunction with the use of the forcing term criteria and the line-search backtracking method described in Chapter **??**. All numerical experiments were run in this section on `Matlab` v4.2c on a Sun workstation SPARC 10.

5.1. Preliminary examples. XXXXXXXXXXXXXXXXXXXXXXXXXXXX In this particular example, we use the line-search backtracking method with the same parameter specifications of Example 2.1. Figure 3 shows the relative nonlinear residual history of the nonlinear KEN algorithm (dotted line) and the higher-order imple-

mentation of Krylov-Newton (solid line). Table 3 supports part of the convergence behavior of both approaches. In all cases, GMRES was able to converge within a prespecified restart value of $m = 20$, a zero initial guess vector and no preconditioning. For the higher-order version of the Krylov-Newton algorithm, we set $l_{max} = 10$. As was observed before, the Rosenbrock function represents the hardest case. Hence, the algorithms do not show important improvements compared to their Newton's method and Broyden's method counterparts. The plateau portion exhibited by the KEN algorithm at the first iterations obeys to the difficulties encountered by the Krylov-Broyden update for the same case (see Figure 1). Not surprisingly, Table 3 confirms the lack of success of the Krylov-Broyden update for the higher-order version of the Krylov-Newton algorithm (the zeros denote the occurrence of backtracking steps at the first to two nonlinear cycles). The Powell function introduces an opposite situation. The solution to the minimal residual problem resulting from every Hessenberg updates was always able to generate a decreasing step for $\|F\|$. Consequently, the nonlinear KEN algorithm reproduces almost exactly the behavior of the nonlinear EN algorithm and the higher-order Krylov-Newton (HOKN) algorithm dramatically outperforms the composite Newton's method (with only one GMRES call per nonlinear cycle). It is important to remark that GMRES generates an invariant Krylov subspace under the Jacobian after 4 iterations to the level of double precision roundoff errors (i.e., the residual term in (15) was of order 1.0×10^{-16}). Note, however, that this does not necessarily imply that the value of the function at the new point belongs to that invariant subspace as it seems to be the case here. An intermediate behavior is shown by the Chandrasekhar equation, with a more favorable tendency as the difficulty of the problem increases, though. In the easy case, the higher-order Krylov-Newton method is competitive with composite Newton's method. The nonlinear KEN algorithm outperforms Broyden's method but it is slightly worse than the nonlinear EN algorithm. The difficult case delivers similar conclusions to the Powell function case. The reader can verify that each convergence history is qualitative reflecting to that observed in 1. As a final comment, Table 3 clearly illustrates that the larger the dimension of the Krylov subspace does not mean a longer chain of decreasing directions for $\|F\|$ in step 2.3 of Algorithm 4.2. XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

We present performance of the examples shown before in terms of floating point operations. Figures 8, 9 and **??** show the computational work in millions of accumulated floating point operations employed to decrease relative nonlinear residual norms for each one of the methods discussed in this thesis.

According to their order of appearance, all methods in this subsection have been classified as Newton-like (those that evaluate the Jacobian at every step), secant-type (those that provide and approximation to the Jacobian), and the last set of three algorithms based upon the hybrid Krylov-secant idea. In the next subsections, however, we rather categorized the nine methods in Newton type of methods (Newton's method, the composite Newton's method, the HOKN algorithm and the HKS-N algorithm) and secant type of methods (Broyden's method, the NEN algorithm, the nonlinear KEN algorithm, the HKS-B algorithm and the HKS-EN algorithm).

Comparison of the set of Newton-like methods is given in Figure 8. The extended Rosenbrock function is definitely the most difficult case. It requires several backtracking steps before entering to the region of rapid convergence. In this case, the clear winner is the composite Newton's method which is incidentally the one that converges in the fewest number of nonlinear iterations. The Newton's method and the HOKN al-

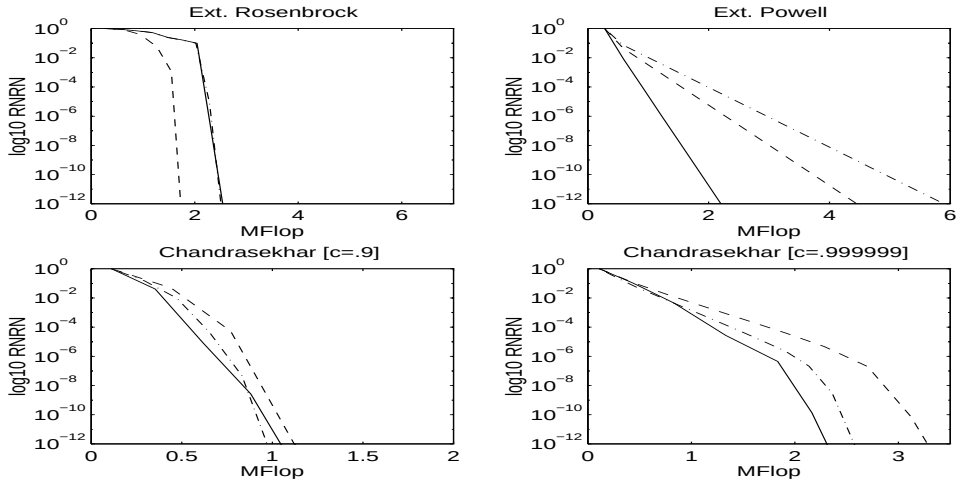

FIG. 8. *Performance in millions of floating point operations of Newton's method (dash-dotted line), composite Newton's method (dashed line) and the HOKN algorithm (solid line) for solving the extended Rosenbrock's function, the extended Powell's function and two levels of difficulty of the Chandrasekhar H-equation.*

gorithm spend about the same effort due to the poor Krylov-Broyden steps performed by the latter. The reader can confirm the same trend on Figures **??** and 3.

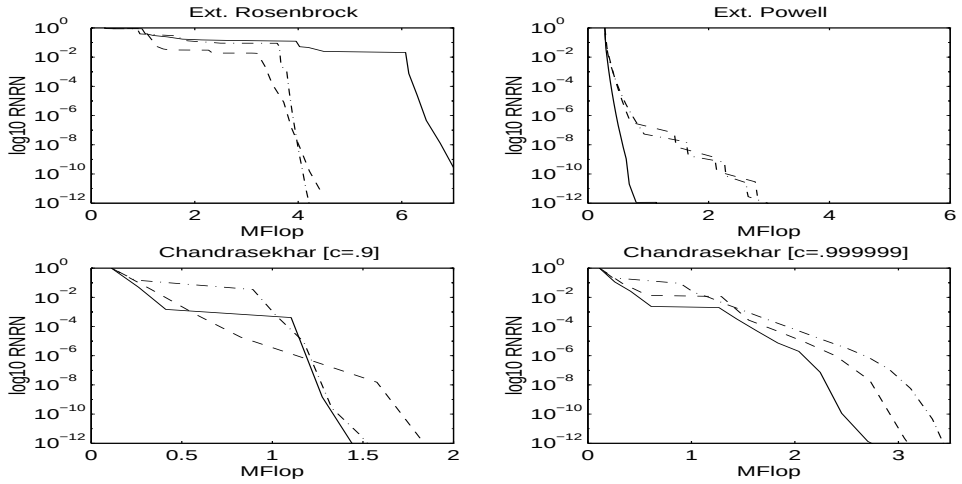

FIG. 9. *Performance in millions of floating point operations of Broyden's method (dash-dotted line), the nonlinear Eirola-Nevanlinna algorithm (dashed line) and the nonlinear KEN algorithm (solid line) for solving the extended Rosenbrock's function, the extended Powell's function and two levels of difficulty of the Chandrasekhar H-equation.*

The extended Powell equation case reveals the great potential of the HOKN algorithm. In this case, the four consecutive Krylov-Broyden steps drive nonlinear residual norms much faster to zero than even the composite Newton's method, theoretically a q-cubic local convergent method.

Note that the composite Newton's method is better than Newton's method although it requires two GMRES solution per nonlinear iteration. The Chandrasekhar H-equation reflects the same trend seen before in terms of the nonlinear iteration count. In this particular case, increasing the nonlinearity of the problem favors the

HOKN algorithm. This underlines some additional robustness of the algorithm for certain harder situations. Not accidentally, this favorable circumstance comes to light also in Figures 9 and ?? when a Krylov-Broyden step is performed.

Figure 9 shows performance for the secant-like group of methods. In general, these methods perform poorly in handling the extended Rosenbrock function compared with Newton type of approaches. This explains in part the wasted effort displayed by the Krylov-Broyden steps in the nonlinear KEN algorithm. The NEN algorithm is slightly superior than Broyden's method for small nonlinear tolerances. Once again, the Krylov-Broyden step is very effective in dealing with the extended Powell function and therefore, the nonlinear KEN algorithm outperforms the other ones, all of which converge with a similar computational cost.

In the Chandrasekhar H-equation the convergence behavior of these methods is not clear for the easier case (i.e., $c = .9$). The plateau portion of Broyden's method and the nonlinear KEN algorithm is due to the difficulty in solving the associated Jacobian linear systems. However, both methods perform better than the NEN algorithm at some relatively small nonlinear tolerances. The case with $c = .9999999$, suggests the nonlinear KEN algorithm as the best choice. In this case, every linear system obtained in every method implies about the same amount of work. The additional saving obtained in the nonlinear KEN algorithm at small tolerances corresponds to a better Krylov-Newton step towards the nonlinear solution.

The last set of methods, i.e., those alternatively using Richardson iteration, are depicted in Figure ??. The failure of Broyden and Krylov-Broyden steps to handle the extended Rosenbrock function produces no clear distinction among the three algorithms. However, the use of a cheaper Richardson iteration explains the slight saving in million of floating point operations in comparison to a Newton's method primarily based on GMRES. An opposite phenomenon was observed in the case of the HKS-N algorithm for the extended Powell function. In such case, Richardson fails to converge after every GMRES solution causing the slight increment in computational cost with respect Newton's method.

The success of the Richardson iteration at the first steps of HKS-B and HKS-EN algorithms introduces additional savings with respect the corresponding counterparts Broyden's method and the NEN algorithm. However, the KEN algorithm is still the most efficient among all. For the Chandrasekhar H-equation, this group of HKS methods performed modestly well. The reader can observe that the HKS-N algorithm is hardly more efficient than Newton's method in the easy case. Additionally, the HKS-EN is competitive with Broyden's method, specially in the hardest case. However, the performance of the HKS-B is disappointing due to an excessive number of iterations in solving the linear systems with both GMRES and the Richardson iteration.

5.2. The modified Bratu problem. The modified Bratu problem is given by [2]

$$\nabla^2 u + \alpha \frac{\partial u}{\partial x} + \lambda e^u = f \quad \text{in} \quad \Omega,$$

$$u = 0 \quad \text{on} \quad \partial\Omega.$$

[2] The actual Bratu (or Gelfand) problem has $\alpha = 0$.

This problem plays an important role in combustion modeling and semiconductor design and processing and represents a simplified model for nonlinear diffusion phenomena. In the absence of the convection term, this operator is monotone with respect u and hence it always has a solution for $\lambda < 0$. When $\lambda > 0$, there is a threshold value λ_* for which the equation has no solution for $\lambda > \lambda_*$ and at least one solution for $\lambda \leq \lambda_*$. For more details, we refer the reader to [26, 31] and pointers therein.

We solve this problem in the unit square with homogeneous Dirichlet boundary conditions. See, e.g., Glowinski, Keller and Reinhart proposed problem in [31] for a detailed description. In this work, the problem is discretized by a block-centered finite-difference scheme and no upwinding was used for the convective coefficient. The linear system generated by the Newton step becomes harder as α and λ grow. In this particular situation, we consider $\lambda = 97$ and $\alpha = 128$ as suggested in [37]. A block Jacobi (with 8 blocks of approximately equal size) preconditioner was used for the Richardson iteration, except where indicated in the tables. A Newton tolerance of 1×10^{-12} was considered for these experiments and the linear solution tolerances were computed by means of equations (**??**) and (**??**).

TABLE 4

Total number of linear iterations (LI) and nonlinear iterations (NI) for all methods discussed in this thesis applied to the modified Bratu problem. The quantities in parentheses indicate the number of Richardson iterations employed by the HKS algorithms.

	10		20		30		40		50	
Method	LI	NI	LI	NI	LI	NI	LI	NI	LI	NI
Newton	34	4	46	4	70	4	98	4	125	4
Comp. Newton	43	3	60	3	92	3	131	3	173	3
HOKN	38	4	40	3	61	3	83	3	108	3
Broyden	67	9	74	8	113	8	156	8	203	8
NEN	72	5	84	4	127	4	178	4	234	4
KEN	70	9	102	8	135	7	186	7	244	7

Table 4 shows the comparison of all nonlinear methods utilized in these tests for six different problem sizes N. These are indicated on the first row of this table, i.e., evenly spaced meshes with 10, 20, 30, 40 and 50 grid blocks, respectively, in each of the coordinate directions. A tridiagonal preconditioner was used to accelerate the GMRES convergence. Several interesting points can be made on these results.

The problem size affects in a higher degree the linear iterations than the nonlinear iterations. All HKS methods represent a reduction of about half the number of GMRES iterations employed by Newton's method, Broyden's method and NEN algorithm counterparts. Moreover, adding up the number of Richardson iteration (shown in parenthesis) we can still appreciate savings in the overall number of linear iterations employed by the HKS algorithms. Conversely, we can observe a small increment in the number of nonlinear iterations for these algorithms, though. The bottom line here is that the Krylov-Broyden updates governing these Krylov-secant algorithms reproduce well the convergence properties of Broyden's method. This last observation is important because the Hessenberg matrix update, i.e., an operator of much smaller dimension than the Jacobian matrix, behaves like Broyden's update of the Jacobian. Therefore, the HKS methods promise to approximate the convergence quality of Broyden's method with the added savings in floating point operations stemming from the

fact that updates are performed on a matrix of considerably lower order.

On the other hand, we can observe the savings in accumulated GMRES iterations by the HOKN algorithm compared with the composite Newton's method. In addition, as it has been observed before, the HOKN has the potential to generate better steps than the composite Newton's method. Although, both basically spend the same number of nonlinear iterations, it is important to remark that the norm of the final nonlinear residual in the HOKN is several orders of magnitude smaller than the one delivered by the composite Newton's method (see e.g., Figure 10). The difficulty of the linear systems in the HOKN was slightly higher than Newton's method (in terms of the number of linear iterations employed by GMRES). On this matter it is important to remark that the overall number of GMRES iterations may result deceiving in some situations. In other words, more accumulated GMRES iterations does not necessarily imply more computational work since the number of floating point operations grows quadratically with the number of iterations taken in a particular GMRES solution. Usually, this number of linear iterations is higher as the nonlinear solution is approached due to the tightening of linear tolerances (i.e., decrease of η_k) prescribed by the Eisenstat and Walker criteria [22]. This fact shall be important to take into account for the convergence analysis of the HOKN and the nonlinear KEN in terms of floating point operations below. Finally, we remark that the nonlinear residuals norms delivered by the KEN algorithm are also smaller than those produced by Broyden's method.

The quadratic growth of the number of floating point operations in GMRES becomes more pronounced as the problem size increases. This implies that savings in operations also grow quadratically even though the table shows almost the same relative number of linear iterations among all methods.

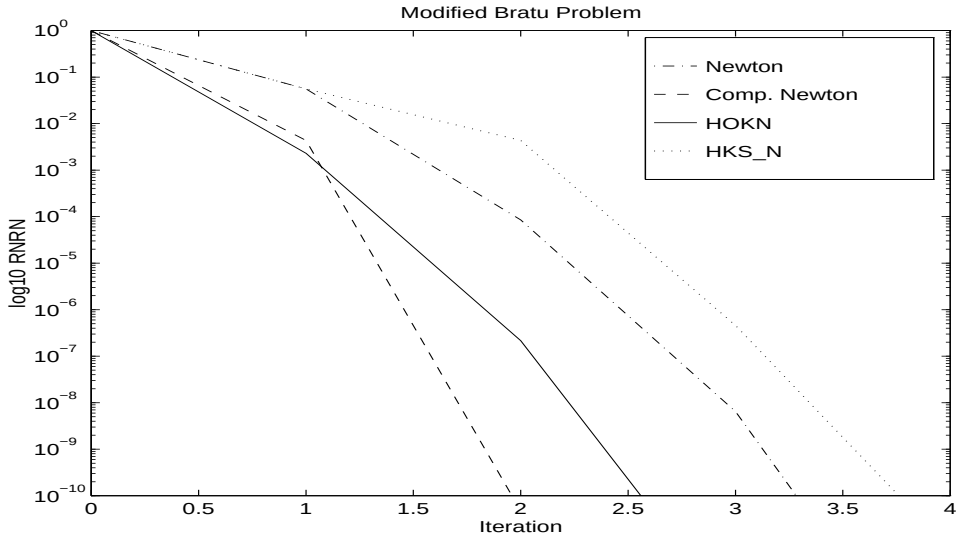

FIG. 10. *Nonlinear iterations vs. Relative Nonlinear Residuals Norms (RNRN) of secant-like methods for the modified Bratu problem on a* 40×40 *grid.*

Figures 10 and 11 show the number of nonlinear iterations taken for all methods to converge to the solution. As in the example cases, higher-order methods appears as the best in terms of total number of nonlinear iterations.

We can observe that HOKN takes less nonlinear iterations than Newton's method

but more than the composite Newton's method. The HKS-N algorithm takes the highest number nonlinear iterations among all. In a similar fashion, the convergence curve described by the nonlinear KEN algorithm falls between that of Broyden's method and that of the NEN algorithm. Note that the HKS-EN performs similarly to the nonlinear KEN algorithm, so that one may expect the use of the Richardson iteration will make the HKS-EN algorithm more efficient whenever Richardson succeeds at every attempt. The HKS-B mimics the behavior of Broyden's method, so this last observation applies as well.

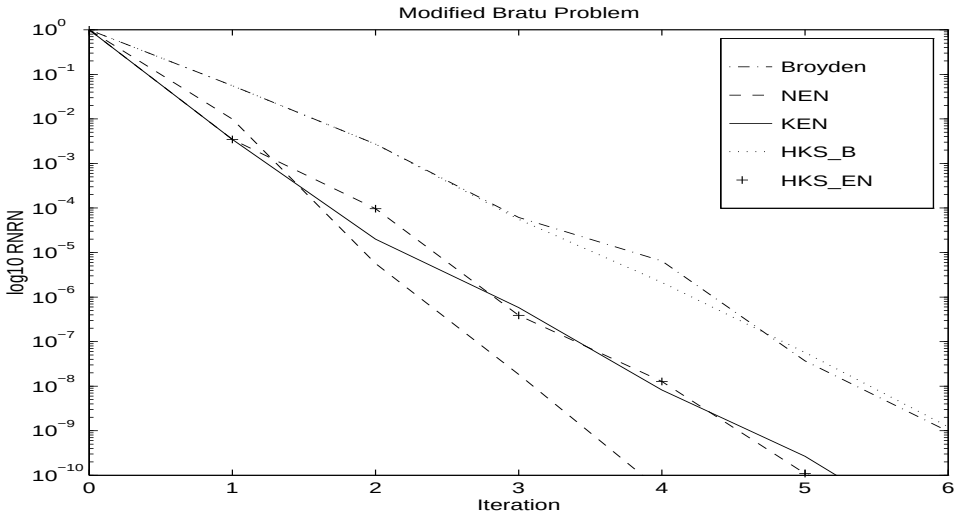

FIG. 11. *Nonlinear iterations vs. Relative Nonlinear Residuals Norms (RNRN) of Newton-like methods for the modified Bratu problem on a* 40×40 *grid.*

Figure 11 calibrates once more the quality of the Krylov-Broyden update compared to the well known Broyden update. The closeness of curves between the HKS-B algorithm and Broyden's method suggests that not much is lost when Broyden's updates are restricted to the current Krylov basis. Under the same light, we can explain the intermediate behavior of the nonlinear KEN algorithm between the NEN algorithm and the HKS-EN algorithm. The nonlinear KEN algorithm alternates Broyden and Krylov-Broyden updates, the NEN performs only Broyden updates and, the HKS-EN performs only Krylov-Broyden updates. All three share the feature of being a higher order version of Broyden's method.

As before, measuring floating point operations instead of number of number of nonlinear iterations provides more conclusive insights. Figures 12 and 13 illustrate the computational efficiency of the new methods.

Figure 12 shows how the HOKN algorithm outperforms the composite Newton's method. The penalty introduced in solving two linear systems with GMRES with the latter method spoils the nice capabilities suggested in Figure 10. The HOKN provides higher convergence rates without incurring in such penalty. Although, it may not be as effective as the composite Newton's method in driving the nonlinear residual norms down, it saves a sensible amount of computation by taking advantage of the underlying Krylov information. In this particular case, however, the quality of the Krylov-Broyden step deteriorates as the solution is approached, making Newton's method more efficient for nonlinear tolerances in the order of 1.0×10^{-7} which may

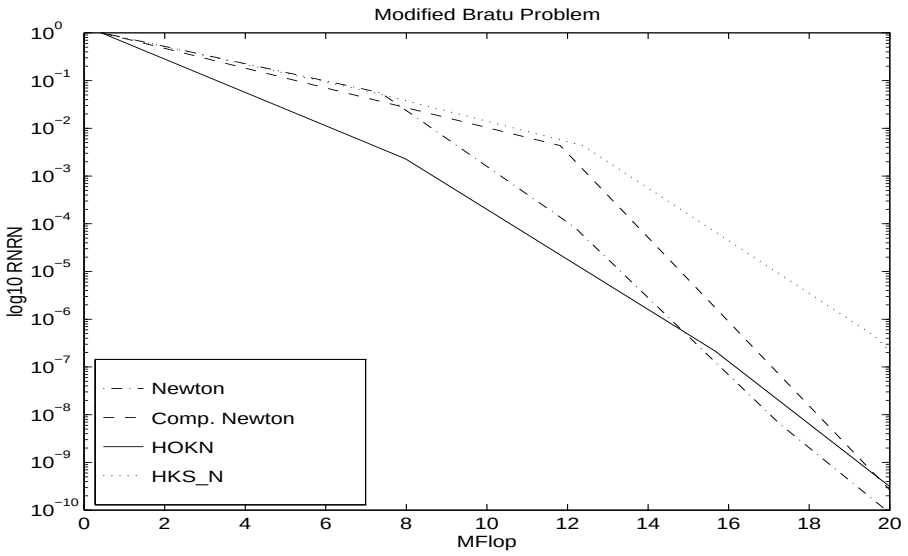

FIG. 12. *Performance in millions of floating point operations of Newton's method, composite Newton's method, the HOKN algorithm and the HKS-N algorithm for solving the modified Bratu problem on a* 40×40 *grid.*

be considered fairly small in most practical situations. The lack of success of the final Krylov-Broyden steps explains the poor results of the HKS-EN algorithm. The Richardson iteration was always able to converge but the nonlinear steps were not as good as those delivered by GMRES.

Figure 13 shows a much closer resemblance among all secant methods. Firstly, they were less effective than those methods evaluating the Jacobian at every nonlinear step in more than 50% of computing work. Secondly, the higher order secant methods based on Krylov-Broyden methods yield the desired pay-off although Broyden's method tends to become more efficient at small relative nonlinear residual norms. This fact stems from the increasing deterioration of the Krylov-Broyden update, but primarily, from the increasing difficulty of the linear systems. The significant savings of GMRES iterations (see Table 4 above) provides a more consistent behavior of computational effort against relative nonlinear residual norms of the HKS-EN and the HKS-B algorithms compared to the nonlinear KEN algorithm.

In general, the contrasting picture of the composite Newton's method and the NEN algorithm is amazing when one looks at Figures 10-13. From being the methods converging in the fewest nonlinear steps they go to being almost the most expensive to use. These two extremes show how a rapid theoretical convergence rate may not sound as promising in terms of a computer implementation. In this sense, the new family of Krylov-secant algorithms maintain a balance that make them attractive for faster nonlinear convergence rates without exceeding the computational cost of the traditional Newton's and Broyden's method.

To end the analysis on the modified Bratu problem, we present how the preconditioner affects the convergence of all methods (see Table 5). In Chapter **??** we devoted a discussion to the use of preconditioning for all the Krylov-secant methods proposed in this dissertation. The high degree of difficulty of the associated Jacobian linear systems makes appropriate an analysis of this kind here.

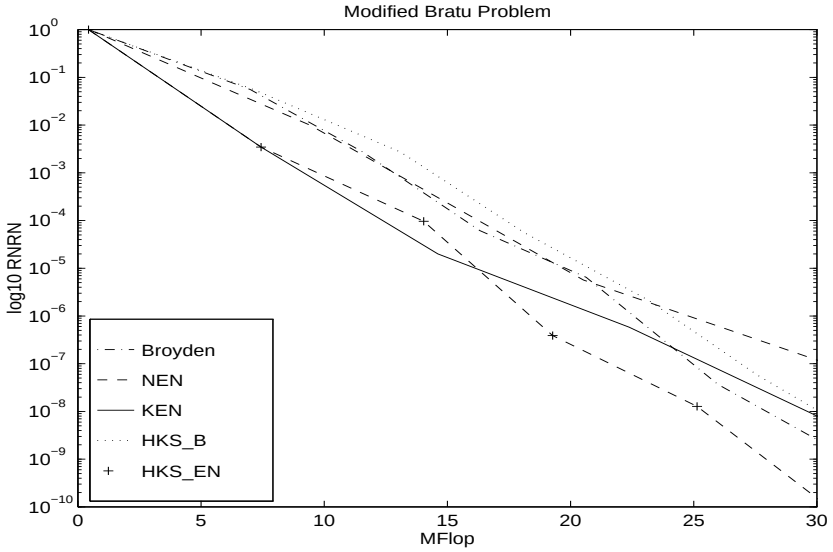

FIG. 13. *Performance in millions of floating point operations of Broyden's method, the nonlinear Eirola-Nevanlinna algorithm, the nonlinear KEN algorithm, the HKS-B algorithm and the HKS-EN algorithm for solving the modified Bratu problem on a* 40×40 *grid.*

TABLE 5

Summary of the total number of linear iterations shown with several preconditioners for all nonlinear methods. The quantities in parentheses indicate the number of Richardson iterations employed by the HKS methods. The problem considered is on a 40×40 *grid.*

Method	Jacobi	Tridiagonal	BJacobi(8)	BJacobi(4)	ILU(0)
Newton	320	98	44	40	138
Comp. Newton	407	131	62	58	173
HOKN	359	83	52	47	163
Broyden	441	156	75	66	236
NEN	431	178	78	74	260
KEN	375	186	90	87	269

We consider a family of standard preconditioners: point Jacobi (i.e., diagonal scaling), tridiagonal preconditioner, block-Jacobi preconditioners with 8 and 4 blocks and, ILU(0) (i.e, incomplete LU with no infill inside the matrix bandwidth).

Both block-Jacobi preconditioners appear to achieve the lowest number of total linear iterations for all methods. They also produce the lowest accumulated number of GMRES and Richardson (in the HKS algorithms) iterations. Note that the ILU(0) is quite poor in this case, due to the strong convective part that makes the inverse of the preconditioner operator not positive stable. (Consequently, some eigenvalues of the preconditioned matrix lie on the left side of the complex plane and the preconditioned system inherits some indefiniteness.)

The main point of Table 5 is to show the stability of the methods for different preconditioners. Recall that Krylov-Broyden updates are applied to the preconditioned system solved by GMRES and that there is no way to reflect (at least in terms of computational cost) the updated and preconditioned system. A large inconsistency between this system and the fixed preconditioner (recall discussion in §§4.2.2) may result in failure in reducing effectively $\|F\|$ as the nonlinear process advances. The table shows that linear iterations reduce consistently according to the quality of the preconditioner. It is worth to add that there were no differences in the total number of nonlinear iterations (which are summarized in Table 4 for this problem size of 40×40.)

5.3. Richards' equation. This example problem models the infiltration of the near-surface underground zone in a vertical cross-section. This is a case of unsaturated flow that takes place in the region between the ground surface and the water table, i.e., the so called *vadose zone*. The flow is driven by gravity and the transport coefficients are modeled by empirical nonlinear functions of the moisture content below saturation conditions. The model equation for this two-dimensional flow is given by

$$\frac{\partial c}{\partial t} - \nabla \cdot [D(c) \cdot \nabla c] - \frac{\partial K(c)}{\partial z} = 0,$$

where c is the underground moisture content, $D(c)$ is the dispersivity coefficient and $K(c)$ is the hydraulic conductivity. This equation is nothing more than a simplification of the well known Richards' equation which also presents nonlinearities in the transient term.

The boundary conditions for this model are of Dirichlet type at the surface, where a constant water content of unity is kept at all times, and of Neumann -no flow- type on the remaining three sides of the model. These conditions are given by

$$c = c_s, \quad \text{at} \quad z = 0, \quad 0 < y < 1,$$

$$\frac{\partial c}{\partial z} = 0, \quad \text{at} \quad z = 1, \quad 0 < y < 1,$$

$$\frac{\partial c}{\partial y} = 0, \quad \text{at} \quad y = 0 \quad \text{and} \quad y = 1, \quad 0 < z < 1,$$

for $t > 0$. Here, z represents the vertical direction and y represents the chosen horizontal direction for the cross section. The surface water content is represented by $c_s = 1$.

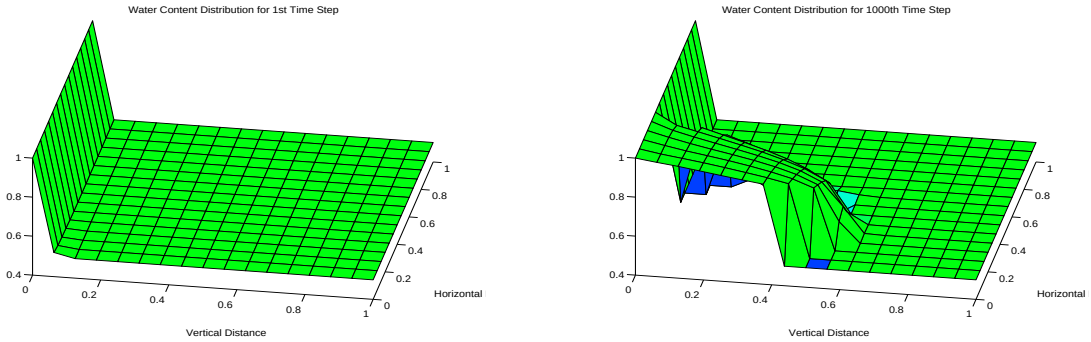

FIG. 14. *Water content distribution at* 1^{st} *and* 1000^{th} *time steps.*

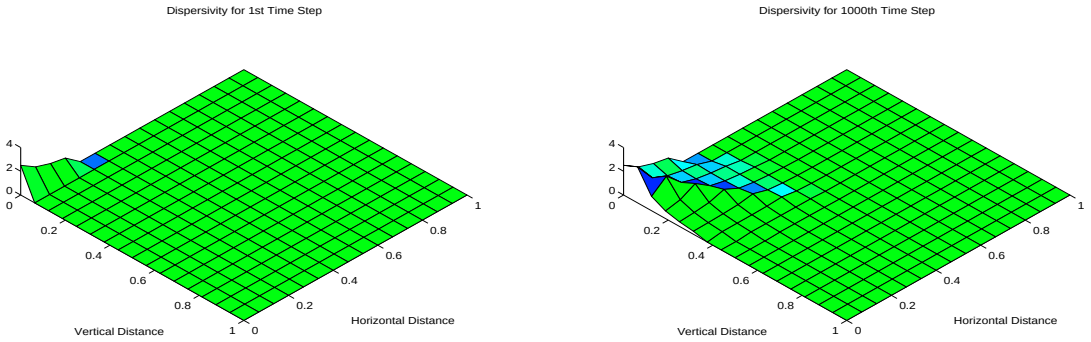

FIG. 15. *Dispersivity at* 1^{st} *and* 1000^{th} *time steps.*

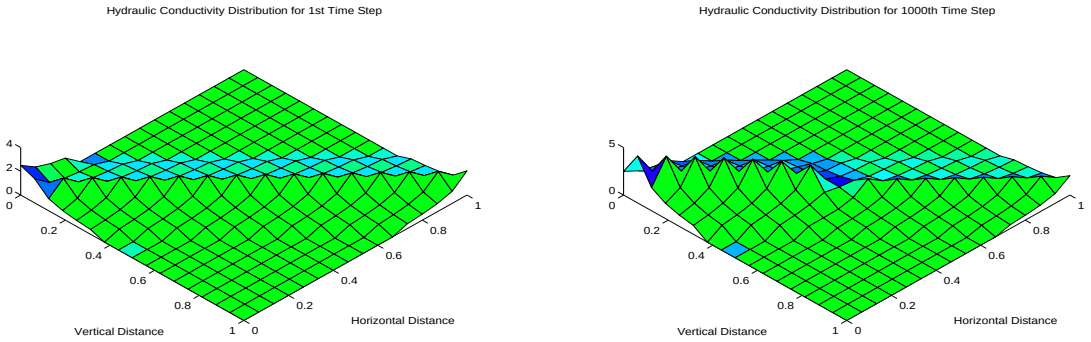

FIG. 16. *Hydraulic Conductivity coefficients at* 1^{st} *and* 1000^{th} *time steps.*

It would be appropriate to say that the same hydraulic conductivity plays a role in both the diffusive and the convective terms, because this model is just the continuity equation for the moisture content, where Darcy's law (with a moisture content dependent hydraulic conductivity) has been replaced for the superficial velocity. In fact, $D(c)$ is often referred to as the capillary diffusivity and is given by

$$D(c) = \frac{K(c)}{-\frac{dc}{d\psi}}, \quad \text{with} \quad \psi = -\frac{p}{\rho},$$

where p and ρ are the groundwater capillary head and density, respectively. However, different functional forms are often used to describe the dependence of both coefficients on the subsurface water content. For this example, our choices of dispersivity and hydraulic conductivity are, respectively,

$$D(c) = K_0 c_e^{\frac{1}{2}} \left[1 - \left(1 - c_e^2\right)^{\frac{1}{2}}\right]^2, \quad \text{and} \quad K(c) = K_0 c_e^{\frac{1}{2}},$$

with

$$c_e = \frac{c - c_0}{c_s - c_0},$$

where c_0 is the irreducible underground water content and the tensor K_0 is a position dependent coefficient that we have chosen according to

$$K_0(i,j) = \frac{5}{\mid i - j \mid +1}, \quad \text{for} \quad 1 \leq i, j \leq 5.$$

where N is the number of gridblocks. This choice of K_0, although contrived, is sometimes found in underground formations and, represents a narrow channel of permeable rock where the moisture is allowed to move. The hydraulic conductivity at saturation is proportional to the rock permeability, which has been shown to change over a few orders of magnitude within relatively short distances in underground formations. In our computational experiments we take $c_0 = 0.25$, which represents a typical value of the irreducible water content. See [1] for a comprehensive discussion of this model.

Figure 14 shows the solution for the moisture content distribution over the two-dimensional domain for a mesh of 16×16 at the 1^{st} and 1000^{th} time steps of simulation. The solution shows the effect of the heterogeneity in the resulting subsurface water content.

A constant time step was used for these simulations, which was chosen small enough to allow the inexact Newton method to converge within 40 nonlinear iterations and given by

$$\Delta t = \frac{1}{16} h^2.$$

This small time step was required in order to use the solution of the previous time level as an acceptable initial guess for the nonlinear iteration.

Figures 15 show the dispersivity coefficient, $D(c)$, over the two-dimensional domain, for the same discretization mesh as in Figure 14, at the 1^{st} and 1000^{th} time steps. Figure 16 shows the distribution of the transport coefficient, $K(c)$, instead.

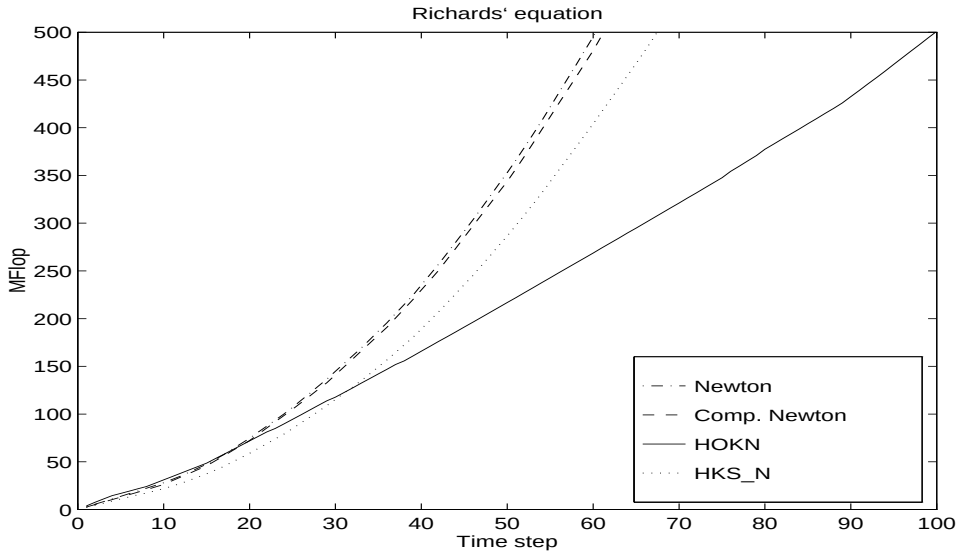

FIG. 17. *Performance in accumulated millions of floating point operations of Newton's method, composite Newton's method, the HOKN algorithm and the HKS-N algorithm for solving Richards' equation.*

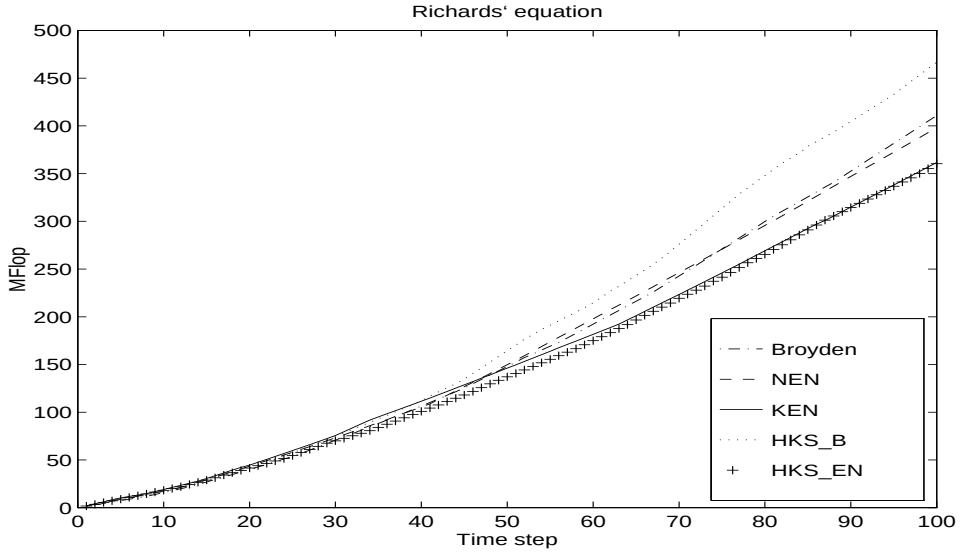

FIG. 18. *Performance in accumulated millions of floating point operations of Broyden's method, the nonlinear Eirola-Nevanlinna algorithm, the nonlinear KEN algorithm, the HKS-B algorithm and the HKS-EN algorithm for solving Richards' equation.*

Both figures are intended to give the reader a feel for the combined effect of heterogeneity and nonlinearities of the model. The coefficients are shown to vary within the interval $(0, 5)$ as a result of having scaled both $K(c)$ and $D(c)$ by $K_{0,max}$. This scaling is hidden in the scaling of the spatial coordinates.

Figure 17 and 18 show the accumulated (million) floating point operations for all the nonlinear methods as the simulation progresses up to 100 time steps, for a discretization mesh of 32×32. No preconditioning was used. The curve clearly exhibits the computational cost trend of all nonlinear methods.

The HOKN algorithm shows a significant saving in computational cost from start to end of this short simulation (see Figure 17). The increasing difficulty of the nonlinear problems as simulation advances produces a superlinear growth in the number of floating point operations. This growth is not only due to the complexity of the nonlinear problems but also to that of the linear problem. This is an example where the region of rapid convergence is far from the initial guess given at every time step, causing unexpected difficulties to Newton's method before reaching that region. Additionally, since derivatives of $D(c)$ and $K(c)$ are approximated by finite differences, ideal conditions for Newton to achieve rapid convergence are violated. (This situation is practically unavoidable in many problems of this type, where coefficients are subject to experimental observation.) On the other hand, as it can be observed in Figure 18, secant methods produce more efficient steps toward the solution, making them more preferable than Newton type of approaches.

The HOKN algorithm delivers between one and two successful Krylov-Broyden steps per nonlinear iteration for this problem case. Note, however, that the HKS-EN algorithm is more efficient than this algorithm during the first 30 time steps. Throughout the whole simulation, the HKS-EN algorithms turns out to be more efficient than Newton's method and the composite Newton's method owing to the substitution of GMRES by Richardson iterations. However, in the absence of that beneficial secant step it shows a similar order cost to that of the other two nonlinear methods.

Figure 18 shows again that the nonlinear KEN algorithm and the HKS-EN are close competitors, perhaps with a marginal advantage for the latter. In this case, the HKS-B performs badly as result of a sequence of poor Krylov-Broyden steps that somehow are corrected in the HKS-EN algorithm. Also, there does not seem to be a clear winner between Broyden and the NEN algorithm (as it also occurs between Newton's method and the composite Newton's method) but, both the nonlinear KEN and the HKS-EN algorithms perform better yet.

TABLE 6

Total nonlinear iterations (NI), GMRES iterations (GI) and (when applicable) Richardson iterations (Rich) for inexact versions of several nonlinear solvers. The problem size considered is of size 16×16 *gridblocks after 100 time steps of simulation.*

Method	NI	GI	Rich.	Backs.
Newton	1627	11890	0	0
Comp. Newton	835	12186	0	0
HOKN	391	2673	0	0
Broyden	631	4046	0	0
NEN	347	4400	0	0
KEN	422	2757	0	0

Table 6 summarizes convergence of the previous plots. The table confirms the excessive work (in terms of nonlinear iterations) carried out by Newton's method, the composite Newton's method and the HKS-N algorithms compared to the HOKN algorithm and all secant methods. The composite Newton's method halves the number of nonlinear iterations of Newton's method but both spend about the same total number of linear iterations. The figures for the HOKN algorithm perfectly justify what is observed in Figure 17. It reduces in half the number of nonlinear iterations taken by the composite Newton's method and, besides, it reduces by an almost 4-fold the total number of linear iterations with respect this higher order method. Hence, the HOKN algorithm not only tackles efficiently the nonlinearities but also leads to much easier linear problems.

The NEN algorithm also halves the number of nonlinear iterations shown by Broyden's method but the number of linear iterations accounts for the similar computational efficiency of both. The KEN algorithm takes an intermediate number of nonlinear iterations between these two algorithms but its efficiency is marked by the fewer number of linear iterations displaid. Note that the HKS-EN algorithm converges in a few more nonlinear iterations than the nonlinear KEN algorithm but reduces in roughly 42 % the total number of GMRES iterations. In this particular situation the savings obtained via Richardson iterations compensate the additional work induced by extra nonlinear iterations. The table clearly exhibits the relative high cost of the HKS-B algorithm compared to Broyden's method: more nonlinear iterations and an overwork of Richardson iterations that did not alleviate the cost of merely relying on GMRES iterations.

One of the other highlights of this table is the reduction in the number of GMRES iterations displayed by the HKS-N and HKS-EN algorithms. These results appear to corroborate those of last section in that the combined number of linear iterations of HKS algorithms is approximately equal to the number of GMRES iterations in the Newton's and Broyden's methods.

6. Evaluating parallel Krylov-secant methods and two-stage preconditioners for systems of coupled nonlinear equations. In this section we experimentally combine and verify two of the main results of this dissertation. That is, the use of the HOKN algorithm together with the 2SGS preconditioner. We begin introducing some features of the simulator and the data model to be analyzed. We then proceed to describe some technicalities that need to be sorted out for the joint implementation of these two ideas. Once this introductory background has been exposed, parallel numerical experiments are presented.

6.1. Brief description of the model. We have chosen the parallel two-phase black-oil reservoir simulator RPARSIM as the common application to evaluate the HOKN method and the 2SGS preconditioner. This not only serves to test both ideas on an real application but also to measure their scalability on a parallel platform. To that end, we perform numerical experiments on an Intel Paragon and an IBM SP2 parallel machines. Both of them are located at the University of Texas, Austin.

The Paragon machine consists of 64 Mbytes of RAM memory plus 16 Kbytes of data cache per node. This specific configuration has 42 nodes arranged in a 2-D mesh topology fashion. Each node can achieve a peak performance of 80 Mflops and a communication speed of 40 Mbytes/s. The SP2 machine consists of 16 nodes, each with 128 Mbytes of RAM. Each node is capable of providing a peak computation

performance of 260 MFlops and a bidirectional communication rate of 50 MBytes/s.

We use MPICH (a public version of MPI developed by Argonne National Lab and Mississippi State University) as the message passing system library. This allows portability of the simulator on different distributed memory architectures.

Most numerical experiments compare the performance of the current inexact Newton solver with the 2SComb preconditioner with the new approach. Details on the performance of this nonlinear solver are given in ([10]). Some of their basic features can be synthesized as follows:

- Inexact Newton method based on the optional choice of GMRES and BiCGSTAB iterative solvers.
- Both GMRES and BiCGSTAB are preconditioned with a two-stage combinative approach (i.e., the 2SComb preconditioner) that uses line correction to solve the partially decoupled pressure system and a tridiagonal preconditioner as $\tilde{M}$ (refer to the notation given in Chapter **??**).
- Line-search backtracking globalization and forcing term criteria based on the work of Eisenstat and Walker.
- The solver is developed to handle a fully-implicit three-dimensional formulation with capabilities to handle a full permeability tensor and general boundary conditions. This implies the manipulation of Jacobian matrices with 64 pressure and concentration coefficient arrays.

TABLE 7
Physical input data.

Initial nonwetting phase pressure at 49 ft	$300psi$
Initial wetting saturation at 49 ft	.5
Nonwetting phase density	$48lb/ft^3$
Nonwetting phase compressibility	$4.2 \times 10^{-5} psi^{-1}$
Wetting phase compressibility	$3.3 \times 10^{-6} psi^{-1}$
Nonwetting phase viscosity	$1.6cp$
Wetting phase viscosity	$0.23cp$
Areal permeability	$150md$
Permeability along 1st and 2nd half of vertical gridblocks	$10md$ and $30md$

Additionally, the data are decomposed in an areal sense (i.e., each processors holds the same original number of gridblocks along the depth direction). This is due to the fact that in most reservoir domains the vertical direction is relatively much smaller than the horizontal plane. where the phases flow. The effective manipulation of a full permeability tensor induces a 19-point stencil discretization for the pressures and concentrations of the linearized wetting phase equation and, a 19-point stencil for pressures and a 7-point stencil for concentrations of the linearized non-wetting phase equation (this gives rise to the 64 coefficient arrays accompanying each gridblock unknown). Therefore, matrix-vector products involves data communication of each node with its four lateral and four corner neighbors (refer to [10] for further details).

In our particular implementation, the 2SGS preconditioner comprises the GMRES solution of each individual block of pressures and concentrations (i.e., the product of densities and saturations of a particular phase). A tridiagonal preconditioner is used to accelerate the convergence rate of this inner GMRES.

Table 7 summarizes the physical parameters for this problem, and Figure **??** shows the associated relative permeability and capillary pressure functions used. The model consist of a water injection well (with bottomhole pressure specified) located at the coordinate $(1,1)$ of the plane and, a production well (with bottomhole pressure specified) at the opposite corner of the plane.

6.2. Considerations for implementing the HOKN algorithm with the 2GSS preconditioner. Before presenting the numerical results, it is important to establish some special considerations arising from the joint implementation of the 2SGS preconditioner and the HOKN nonlinear solver. Since the 2SGS demands previous decoupling of the linear system, the secant equation on which the Krylov-Broyden update is based, is of the form

$$D^{(k)}\left(D^{(k)}\right)^{-1}A^{(k+1)}\left(M^{(k)}\right)^{-1}M^{(k)}s^{(k)} = F^{(k+1)} + r_0^{(k)},$$

for a given kth nonlinear iteration and a GMRES solution $s^{(k)} = s_0^{(k)} + V^{(k)}y^{(k)}$. Here, $M^{(k)}$ represents the inexact block Gauss-Seidel preconditioner acting upon the decoupled matrix $\left(D^{(k)}\right)^{-1}A^{(k)}$. This decoupled matrix expressed as 2×2 blocks has a similar presentation depicted in (**??**).

Using the associated Arnoldi factorization

$$\left(D^{(k)}\right)^{-1}A^{(k)}\left(M^{(k)}\right)^{-1}V^{(k)} = V_{m+1}^{(k)}\overline{H}_m^{(k)},$$

to the Jacobian system

$$\left(D^{(k)}\right)^{-1}A^{(k)}\left(M^{(k)}\right)^{-1}M^{(k)}s^{(k)} = F^{(k1)},$$

one determines that the secant equation for the Hessenberg matrix is given by

$$H_m^{(k+1)}y^{(k)} = \left(V^{(k)}\right)^t\left(D^{(k)}\right)^{-1}\left(F^{(k+1)} + r_0^{(k)}\right).$$

Therefore, Broyden's update of the Hessenberg matrix is given by

$$H_m^{(k+1)} = H_m^{(k)} + \frac{\left[\left(V^{(k)}\right)^t\left(D^{(k)}\right)^{-1}F^{(k+1)} + \beta e_1 - H_m^{(k)}y^{(k)}\right]\left(y^{(k)}\right)^t}{\left(y^{(k)}\right)^t y^{(k)}},$$

with $\beta = \left\|\left(D^{(k)}\right)^{-1}r_0^{(k+1)}\right\|$.

Hence, the value of the function at the new point needs to be decoupled before being projected onto the underlying Krylov subspace. Technically, the Krylov-Broyden update and consequently, the whole HOKN implementation can be carried out in terms of the decoupled linear system.

An efficient implementation is accomplished by carrying out the decoupling operation in place over all arrays holding the matrix coefficients. In order to restore the original Jacobian coefficients, five arrays are employed to hold the main diagonals entries of each block and the vector entries of Δ. This allows to maintain the same standard Euclidean norm in the line-search backtracking strategy, forcing term selection and in the nonlinear stopping criteria. The coefficients values are restored after all Krylov-Broyden steps in the HOKN algorithm have been completed.

As explained in Chapter **??**, there is no need to explicitly form the Jacobian Krylov-Broyden update for the implementation of the HOKN algorithm. All operations can be done in terms of the updated Hessenberg matrix, the orthogonal matrix V_{m+1} and the minimal residual approximation solution $y^{(k)} \in \mathbb{R}^m$. Additionally, the preconditioner $M^{(k)}$ is kept fixed to retrieve the unpreconditioned nonlinear direction, $s^{(k)}$.

We remark that the HOKN algorithm can be easily changed to the standard inexact Newton method with a single flag inhibiting the computation of the Krylov-Broyden steps.

6.3. Numerical results. We compare the effect of the 2SComb and the 2SGS preconditioning on GMRES and BiCGSTAB for two different values of ΔT. This is shown in Table 8.

The table shows that both GMRES and BiCGSTAB algorithms perform similarly for a problem of modest difficulty (i.e., for $\Delta T = .05$). Notice that BiCGSTAB employs almost half of the total number of iterations of GMRES but on the other hand, BiCGSTAB doubles the number of matrix-vector multiplications and preconditioner calls made by GMRES at each linear iteration (cf. Algorithm **??** and Algorithm **??**). The cost associated to the matrix-vector multiplication and the application of any of the two-stage preconditioners makes the performance times comparable between these two linear solvers. In simple problems, Bi-CGSTAB tends to outperform GMRES, whereas in more complex problems the latter method tends to be more robust and efficient as the simulation for $\Delta T = .5$ reveals.

Also remarkable is the performance of both linear solvers with the 2SGS preconditioner in relation to the 2SComb preconditioner. For this particular problem, the 2SGS preconditioner reduces by more than a 10-fold the total number of linear iterations. Since the number of nonlinear iterations is practically unchanged, we improve the computer times by almost 10 times (recall discussion on cost of both schemes). This result corroborates the observations made in the previous section for sample matrices extracted from this physical model.

TABLE 8

Summary of linear iterations (LI), nonlinear iterations (NI), number of backtracks (NB) and execution times of GMRES and Bi-CGSTAB with the use of the 2SComb and the 2SGS preconditioners. The simulation covers 20 time steps with $\Delta T = .05$ and $\Delta T = .5$ for a problem of size $8 \times 24 \times 24$ gridblocks on a mesh of 4×4 nodes of the Intel Paragon. (*): Backtracking method failed after the 17th time step; (**): ΔT was halved after the 16th time step.

ΔT	Linear solver/Prec.	LI	NI	NB	Time(Hrs.)
.05	GMRES/2SComb	1450	45	0	1.10
	GMRES/2SGS	102	49	0	0.11
	Bi-CGSTAB/2SComb	855	45	0	1.19
	Bi-CGSTAB/2SGS	66	44	0	0.07
.5	GMRES/2SComb	6745	100	0	6.37
	GMRES/2SGS	538	107	0	0.51
	Bi-CGSTAB/2SComb(*)	2808	190	41	5.62
	Bi-CGSTAB/2SGS (**)	493	102	12	0.70

BiCGSTAB fails twice for different reasons. For $\Delta t = .5$, the 2SGS preconditioner forces a reduction of the time step due to the high changes of pressures and concen-

trations within the time step. In many reservoir simulation codes it is customary to regulate the next time step according to a maximum allowable change of pressures and saturations within the current time step. This prevents possible loss of material balance due to the deterioration or eventual failure of the nonlinear solution. Shortening the time step increases the chances of convergence for the nonlinear method.

The failure with the 2SComb preconditioner is more serious. The line-search failed because the linear solver was unable to converge at the maximum tolerance allowed (0.1, in our case). Therefore, BiCGSTAB could not provide an acceptable direction for decreasing $\|F\|$. Note that, before breakdown, this execution had undergone a high number of backtracks and nonlinear steps.

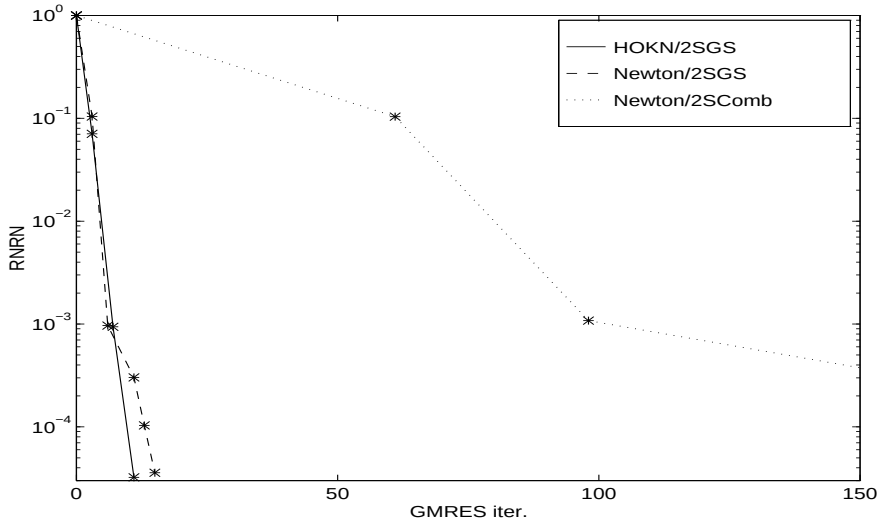

FIG. 19. *Number of accumulated GMRES iterations vs Relative nonlinear residual norms (NRNR) using the HOKN/2SGS, Newton/2SGS and Newton/2SComb solvers on 12 nodes of the IBM SP2 for a problem size of* $16 \times 48 \times 48$ *at the third time step.*

Figure 19 illustrates the relatively strong impact that both HOKN/2SGS and Newton/2SGS solvers have in the simulation. For a moderate problem size, GMRES with 2SComb preconditioning takes above 10 times more linear iterations than with 2SGS preconditioning. The Krylov-Broyden steps in the HOKN method reduce slightly more this number of accumulated iterations. This is accomplished by a more rapid nonlinear convergence.

Figure 20 expresses GMRES iteration effort in terms of computer time. The difference is now less prominent between the 2SComb and 2SGS preconditioning. The line correction in the 2SComb preconditioner contributes to reducing the cost for solving the pressure system. This method was not introduced in the 2SGS in order to preserve the highest possible robustness. The line correction method in the 2SGS has some difficulties due to the lack of diagonal dominance of the pressure block matrix (i.e., this situation does not happen for concentrations coefficients where, contrarily, the system is really easy to solve). This loss of diagonal dominance is observed when there are relative small capillary pressure gradients compared to permeability gradients of the wetting phase at large time steps, violating then the conditions of Theorem **??**. Since the line-search backtracking method allows to handle far guesses to the nonlinear solution, it is preferred to reinforce the robustness of the preconditioner by GMRES

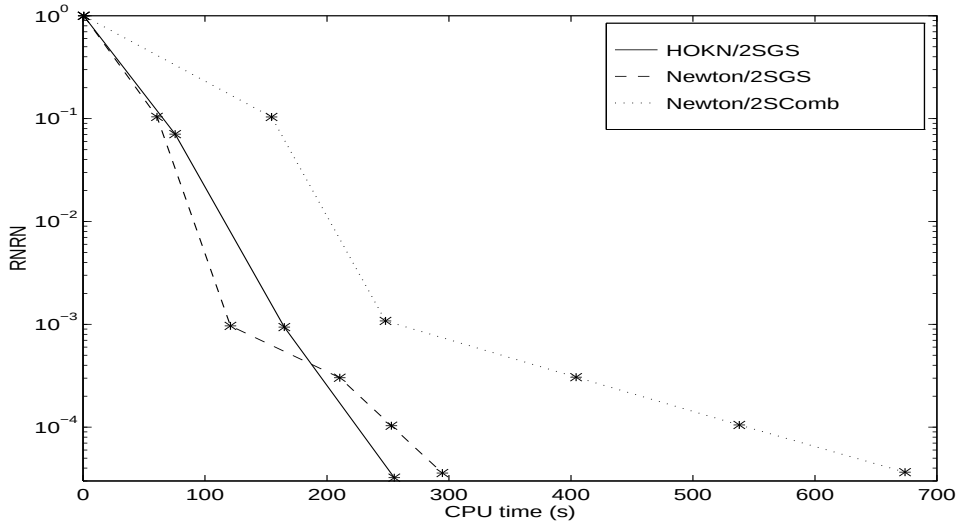

FIG. 20. *CPU time vs Relative nonlinear residual norms (NRNR) using the HOKN/2SGS, Newton/2SGS and Newton/2SComb solvers on 12 nodes of the IBM SP2 for a problem size of* $16 \times 48 \times 48$ *at the third time step.*

in order to be able to take larger time steps. We remark that the line-correction method still works fine in the 2SGS preconditioner for small time steps but with greater restriction than in the 2SComb approach, where the decoupling is partial and more of the elliptic properties of pressures coefficients are preserved.

Despite of this, the HOKN/2SGS solver still outperforms by almost a three-fold the timings of the inexact Newton/2SComb solver.

The previous analysis for a particular time step explains clearly the saving of GMRES iterations for a moderately long simulation. Figure 21 shows that the new HOKN/2SGS spends a considerably smaller amount of GMRES iterations compared to the Newton/2SComb solver.

As before, since one GMRES iteration is more expensive with the 2SGS preconditioner than with the 2SComb preconditioner the Figure 22 exhibits a fairer reality. Nevertheless, the timings of the simulations are reduced in more than a third with the new solver.

Figure 23 shows that not only linear iterations but also nonlinear iterations are reduced. This figure illustrates the effect of using only one Krylov-Broyden step per nonlinear iteration. Although, the HOKN does not imply a noticeable speedup over the inexact Newton method (due in great to its limited parallel capabilities and the ill-conditioning of the Jacobian matrix) its use is still advisable for achieving better material balance. In most cases, relative nonlinear residuals are driven closer to the solution than in those cases where the Krylov-Broyden step was disable. We also, believe that the HOKN effectiveness is attenuated due to the decoupling operation that acts as a left preconditioner of the system. This introduces further concerns in the approximation of the Krylov-Broyden update for simultaneous left and right preconditioning of the Jacobian matrix.

Acknowledgments. The author thank...

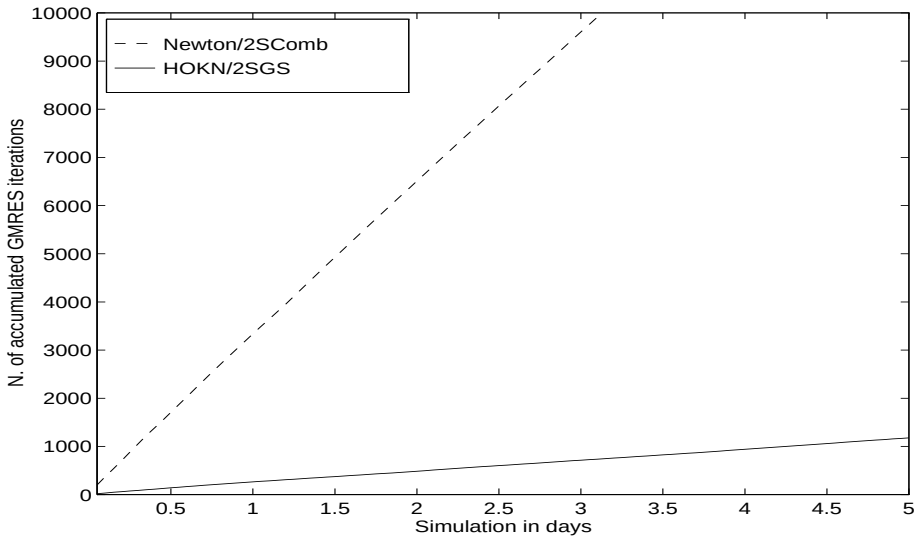

FIG. 21. *Performance in accumulated GMRES iterations of the HOKN/2SGS and Newton/2SComb solvers after 100 time steps of simulation with* $DT = .05$ *of a* $16 \times 48 \times 48$ *problem size on 16 SP2 nodes.*

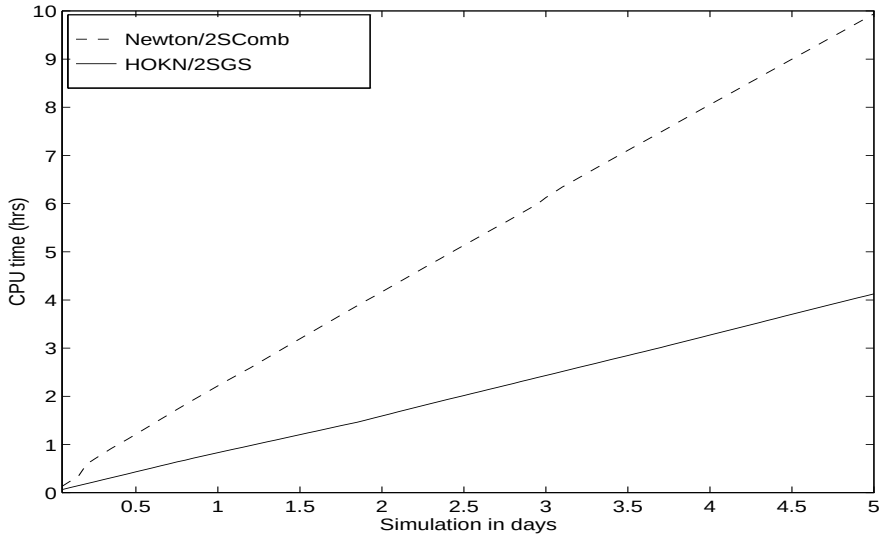

FIG. 22. *Performance in accumulated CPU time of the HOKN/2SGS and Newton/2SComb solvers after 100 time steps of simulation with* $\Delta T = .05$ *of a* $16 \times 48 \times 48$ *problem size on 16 SP2 nodes.*

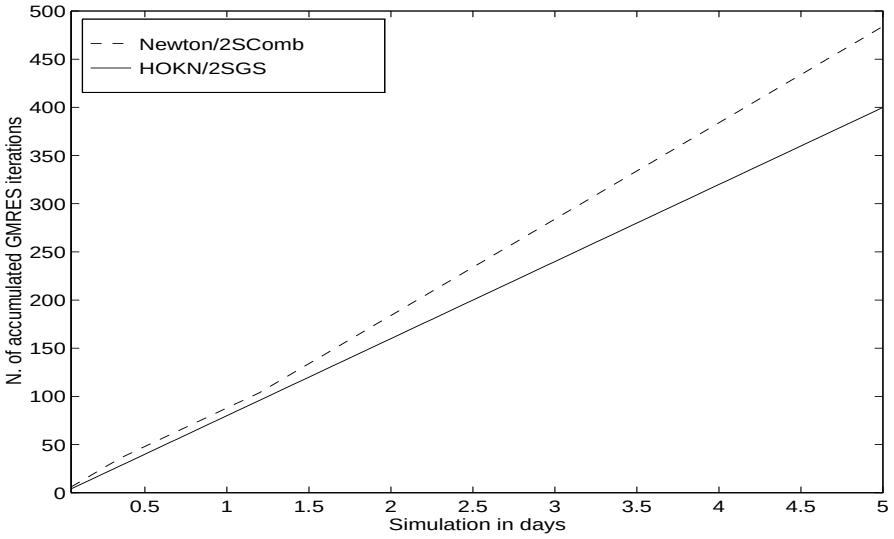

FIG. 23. *Performance in accumulated nonlinear iterations of HOKN/2SGS, Newton/2SGS and Newton/2SComb solvers after 100 time steps of simulation with* $\Delta T = .05$ *of a* $16 \times 48 \times 48$ *problem size on 16 SP2 nodes.*

REFERENCES

[1] J. Bear. *Dynamics of Fluids in Porous Media.* Dover Publications, Inc, 1972.

[2] P.N. Brown. A theoretical comparison of the Arnoldi and GMRES algorithms. *SIAM J. Sci. Statist. Comput.*, 20:58–78, 1992.

[3] P.N. Brown and Y. Saad. Hybrid Krylov methods for nonlinear systems of equations. *SIAM J. Sci. Statist. Comput.*, 11:450–481, 1990.

[4] P.N. Brown, Y. Saad, and H.F. Walker. Preconditioning with low rank updates. Private Communication, 1995.

[5] C.G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19:577–593, 1965.

[6] C.G. Broyden. A new method for solving nonlinear simultaneous equations. *Computing Journal*, 12:94–99, 1969.

[7] R.H. Byrd, J. Nocedal, and R.B. Schnabel. Representations of quasi–Newton matrices and their use in limited memory methods. *Math. Programming*, 63:129–156, 1994.

[8] R.H. Byrd, J. Nocedal, and C. Zhu. Towards a discrete Newton method with memory for large-scale optimization. Technical Report TR OTC 95/01, Optimization Technology Center, 1995.

[9] S. Chandrasekhar. *Radiative Transfer.* Dover, New York, 1960.

[10] C. Dawson, H.M. Klíe, C. San Soucie, and M.F. Wheeler. A parallel, implicit, cell-centered method for two-phase flow. In preparation, 1996.

[11] R.S. Dembo, S.C. Eisenstat, and T. Steihaug. Inexact Newton methods. *SIAM J. Numer. Anal.*, 19:400–408, 1982.

[12] J. E. Dennis and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations.* Prentice–Hall, Englewood Cliffs, New Jersey, 1983.

[13] J.E. Dennis and J.J. Moré. A characterization of superlinear convergence and its applications to quasi-Newton methods. *Math. Comp.*, 228:549–560, 1974.

[14] J.E. Dennis and R.B.Schnabel. Least change secant updates for quasi-Newton methods. *SIAM Review*, 21:443–459, 1979.

[15] J.E. Dennis and H.F. Walker. Convergence theorems for least–change secant updates methods. *SIAM J. Numer. Anal.*, 18:949–987, 1981.

[16] J.E. Dennis and H.F. Walker. Inaccuracy in quasi–Newton methods: Local improvement theorems. In *Mathematical Programming Study 22: Mathematical Programming at Overwolfach.* North-Holland, 1984.

[17] J.E. Dennis and H.F. Walker. Least-change sparse secant updates with inaccurate secant conditions. *SIAM J. Numer. Anal.*, 22:760–778, 1985.

[18] P. Deuflhard, R. Freund, and A. Walter. Fast secant methods for the iterative solution of large nonsymmetric linear systems. *IMPACT of Computing in Science and Engineering*, 2:244–276, 1990.

[19] T. Eirola and O. Nevanlinna. Accelerating with rank–one updates. *Linear Alg. and its Appl.*, 121:511–520, 1989.

[20] S.C. Eisenstat, H.C. Elman, and M.H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 20:345–357, 1983.

[21] S.C. Eisenstat and T. Steihaug. Local analysis of inexact quasi–Newton methods. Technical Report MASC TR 82-7, Dept. of Mathematical Sciences, Rice University, 1982.

[22] S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact Newton method. *SIAM J. Sci. Comput.*, 17:16–32, 1996.

[23] A. Ern, V. Giovangigli, D.E. Keyes, and M. D. Smooke. Towards polyalgorithmic linear system solvers for nonlinear elliptic problems. *SIAM J. Sci. Comput.*, 15:681–703, 1994.

[24] D. Gay. Some convergence properties of Broyden's method. *SIAM J. Numer. Anal.*, 16:623–630, 1979.

[25] D.M. Gay and R.B. Schnabel. Solving systems of nonlinear equations by Broyden's method with projected updates. In O.L. Mangasarian, R.R. Meyer, and S.M. Robinson, editors, *Nonlinear Programming 3*, pages 245–281. Academic Press, N.Y., 1978.

[26] R. Glowinski, H.B. Keller, and L. Reinhart. Continuation-conjugate gradient methods for the least squares solution of nonlinear boundary value problems. *Siam J Sci. Stat. Comput.*, 4:793–833, 1985.

[27] G.H. Golub and C.F. Van Loan. *Matrix Computations*. John Hopkins University Press, 1989.

[28] C.T. Kelley. Iterative methods for linear and nonlinear equations. In *Frontiers in Applied Mathematics*. SIAM, Philadelphia, 1995.

[29] J.M. Martínez. Theory of secant preconditioners. *Math. of Computation*, 60:699–718, 1993.

[30] J.M. Martínez. SOR–secant methods. *SIAM J. Numer. Anal.*, 31:217–226, 1994.

[31] J.J. Moré. A collection of nonlinear problems. In E.L. Allgower and K. Georg, editors, *Lectures in Applied Mathematics, Vol. 26*, pages 723–762. American Mathematical Society, 1990.

[32] S.G. Nash. Newton–type minimization via the Lanczos method. *SIAM J. Num. Anal.*, 21:770–778, 1984.

[33] S.G. Nash. Preconditioning of truncated–Newton methods. *SIAM J. Sci. Stat. Comput.*, 6:599–616, 1985.

[34] S.G. Nash and A. Sofer. *Linear and Nonlinear programming*. McGraw–Hill, 1996.

[35] J. Nocedal. Theory of algorithms for unconstrained optimization. In *Acta Numerica*, pages 199–242. Cambridge University Press, New York, 1991.

[36] J.M. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, 1970.

[37] M. Pernice, L. Zhou, and H.F. Walker. Parallel solution of nonlinear partial differential equations using inexact Newton methods. Technical Report TR48–94, Utah Supercomputing Institute, 1994.

[38] Y. Saad. An overview of Krylov subspace methods with applications to control problems. In M.A. Kaashoek, J.H. vam Schuppen, and A.C. Ran, editors, *Signal Processing, Scattering, Operator Theory and Numerical Methods*, pages 401–410. Birkhauser, 1990.

[39] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company, 1996.

[40] V.E. Shamanskii. A modification of Newton's method. *Ukran. Mat. Zh.*, 19:133–138, 1967. In Russian.

[41] T. Steihaug. Local and superlinear convergence for truncated iterated projection methods. *Mathematical Programming*, 27:176–190, 1983.

[42] J.F. Traub. *Iterative methods for the solution of equations*. Prentice Hall, Englewood Cliffs, 1964.

[43] H.A. van der Vorst and C. Vuik. GMRESR: A Family of Nested GMRES Methods. Technical Report TR91–80, Technological University of Delft, 1991.

[44] C. Vuik. Further experiences with GMRESR. Technical Report TR92–12, Technological University of Delft, 1992.

[45] C. Vuik and H.A. van der Vorst. A comparison of some GMRES–like methods. *Linear Alg. and its Appl.*, 160:131–162, 1992.

[46] J.A. Wheeler and R.A. Smith. Reservoir simulation on a hypercube. In *64th Annual Technical Conference and Exhibition of the Society of Petroleum Engineers*. SPE paper no. 19804, San Antonio, Texas, 1989.

[47] U.M. Yang. *A family of preconditioned iterative solvers for sparse linear systems*. PhD thesis, Dept. of Computer Science, University of Illinois, Urbana-Champaign, 1995.