An Inverse Two-Columns Updating Method for solving large-scale nonlinear systems of equations

Véra Lucia Rocha Lopes * Luziane Ferreira-Mendonça †
Rosana Pérez ‡

Resumo

Neste trabalho propomos um novo método quase-Newton para solução de sistemas não lineares. Neste método fazemos a atualização de duas colunas por iteração da aproximação da inversa da Jacobiana de maneira a satisfazer (quando possível) as duas últimas equações secantes. Chamamos este método de ITCUM. Propomos uma implementação correta do ponto de vista da álgebra linear e da estabilidade numérica; fazemos a análise teórica do método (convergência local) e apresentamos testes numéricos, onde comparamos o desempenho do ITCUM com o de outros métodos quase-Newton, com ênfase maior em ICUM (método de atualização de uma coluna da Jacobiana inversa) [13].

*Departamento de Matemática Aplicada, IMECC-UNICAMP, Universidade de Campinas, CP 6065, 13081-970 Campinas, SP, Brasil (vlopes@ime.unicamp.br).

†Departamento de Matemática Aplicada, IMECC-UNICAMP, Universidade de Campinas, CP 6065, 13081-970 Campinas, SP, Brasil (luziane@ime.unicamp.br). Essa autora é financiada pela FAPESP (processo 00/00375-4).

‡Departamento de Matemáticas, Universidad del Cauca, Popayán (Cauca), Colombia (rosana@ime.unicamp.br).

Abstract

In this work it is introduced a new quasi-Newton method for solving large-scale nonlinear systems of equations. In this method two columns of the approximation of the inverse Jacobian are updated, in such a way that the two last secant equations are satisfied (when it is possible) at every iteration. The new method is called the Inverse Two-Columns Updating Method (ITCUM). Moreover, it is proposed a right implementation from the point of view of linear algebra and numerical stability. It is presented a local convergence analysis and several numerical tests an a comparison between the performance of this new quasi-Newton method with other quasi-Newton methods, in particular the ICUM (Inverse Column Updating Method) [13].

Key words: Quasi-Newton methods, nonlinear systems, inverse two columns-updating method.

1 Introduction

To solve nonlinear systems of equations is a necessary task in the most applied areas, such as Physics, Engineering, Chemistry and Industry. This problem consists on: given a nonlinear function $F : I\!R^n \to I\!R^n$, continuously differentiable, find a vector $x \in I\!R^n$ such that

$$F(x) = 0. \tag{1}$$

All practical algorithms for solving (1) are iterative. Among them we have Newton method and quasi-Newton methods.

Given an initial approximation $x_0 \in I\!R^n$, Newton's method generate a sequence $\{x_k\}$ of approximations of a solution to (1) by

$$x_{k+1} = x_k - J(x_k)^{-1}F(x_k), \tag{2}$$

where, $J(x_k)$ is the Jacobian matrix of F at x_k. The Newton iteration can be costly, since partial derivatives must be computed and the linear system (2) must be solved at every iteration. This fact motivated the development of quasi-Newton methods, which are defined as the generalization of (2) given by

$$x_{k+1} = x_k - B_k^{-1}F(x_k), \tag{3}$$

where, the matrix B_k is an approximation of $J(x_k)$.

The name "quasi-Newton" was used after 1965 to describe also methods of the form (3) such that the equation below is satisfied:

$$B_{k+1}s_k = y_k = F(x_{k+1}) - F(x_k). \tag{4}$$

Following [4], most authors call quasi-Newton all the methods of the form (3), whereas the class of methods that satisfy (4) are called "secant methods". Accordingly, (4) is called "secant equation".

Among the secant methods, we have Broyden's method [1], the Column Updating Method (CUM) [11] and the Inverse Column Update Method (ICUM) [13], [8].

In Broyden's method and in CUM, the updating of the B_k matrix, is made, respectively, by

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)s_k^T}{s_k^T s_k}, \tag{5}$$

$$B_{k+1} = B_k + \frac{(y^k - B_k s^k){e_{i_k}}^T}{{\mathbf{e}_{i_k}}^T s^k}, \tag{6}$$

where, $|\mathbf{e}_{j_k}^T s_k| = \|s_k\|_\infty$.

In the ICUM, the matrix H_k, an approximation of the inverse Jacobian matrix at x_k, is updated by

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)\mathbf{e}_{j_k}^T}{\mathbf{e}_{j_k}^T y_k}, \tag{7}$$

where, $|\mathbf{e}_{j_k}^T y_k| = \|y_k\|_\infty$.

In a recent numerical work, Lukšan e Vlček [9], conclude that ICUM is the most efficient quasi-Newton method in the solution of large-scale nonlinear systems.

In other works, [12], it has been asked about the importance of the " previous secant equation" with the propose to determine a relative efficiency of different quasi-Newton methods.

The efficiency of ICUM and the aspects mentioned above induced us to introduce another quasi-Newton method similar to ICUM, where we use two columns instead of just one, to update the iteration matrix. In this method, H_k will be equal to H_{k+1} except in two columns, that will be updated in order to satisfy the last two secant equations.

The definition of this method involves diverse situations. It must be observed that the method is not always well defined, because it is possible that the two secant equations may be incompatible. Moreover, it is possible, even being compatible, that the compatibility is so slight that the implementation of the method can be ill-conditioned. For this reason it is necessary a careful analysis of the linear algebra that must be used for its implementation, when it is possible.

Other aspect that is necessary to study is related to the theorical properties of the new method. This is one of the intermediate methods between ICUM and a sequential secant method [10]. These last methods have properties well known, but it is not the case of the intermediate methods.

For large-scale problems, it is clear that the ICUM is more efficient than the sequential secant method, which indeed can not be efficiently implemented for this type of problems.

In this work, we introduce a method that is very closely related to ICUM and which we call the Inverse Two-Columns Updating Method (ITCUM). As we said before, while in ICUM one column of the inverse Jacobian approximation is updated, in order to satisfy in each iteration, the secant equation, in our new method, introduced here, we update two columns of the inverse Jacobian approximation, in such a way that the two last secant equations are satisfied at every iteration.

Moreover, we propose a right implementation in the point of view of Linear Algebra and numerical stability. We present the local convergence analysis and several numerical tests where we compare the performance of the new quasi-Newton method with others quasi-Newton methods, particularly, ICUM (Inverse Column updating Method) [13].

The mathematical description of ITCUM is given in **Section 2** of this paper. In **Section 3** we prove local convergence under standard assumptions. In **Section 4** we discuss the computer implementation and report our numerical experiments. Finally, in **Section 5** we state some conclusions and we discuss some lines for future research.

2 Description of the new quasi-Newton method

The Inverse Two-Columns Updating Method (ITCUM) for solving the problem (1) is defined by

$$x^{k+1} = x^k - H_k F(x^k), \tag{8}$$

where the inverse Jacobian approximation H_k, is updated in such a way that H_{k+1} differs from the previous matrix in two columns and the two last secant equations are satisfied at every iteration, that is:

$$\begin{array}{rcl} H_{k+1}y^k & = & s^k \\ H_{k+1}y^{k-1} & = & s^{k-1}, \end{array} \tag{9}$$

where $s^k = x^{k+1} - x^k$ e $y^k = F(x^{k+1}) - F(x^k)$.

Therefore, the matrix H_{k+1} must be a correction of rank two to H_k, that is,

$$H_{k+1} = H_k + u^k_{i_1}\mathbf{e}^T_{i_1} + u^k_{i_2}\mathbf{e}^T_{i_2}, \tag{10}$$

where, $\mathbf{e}_{i_1}$ and $\mathbf{e}_{i_2}$ belong to the canonical basis of $I\!R^n$ and the n-vectors $u^k_{i_1}$ and $u^k_{i_2}$ must be chosen in such a way that the equations (9) are satisfied. In order to simplify the notation, we suppressed the upper index k in i_1 and i_2.

Observe that equations (9) may be incompatible an therefore the method could be not defined. In order to do an analysis of ITCUM and to determine conditions for a good definition of it, we considered the equations in (9) with H_{k+1} defined by (10),

$$\left\{ \begin{array}{rcrcl} (H_k + u^k_{i_1}\mathbf{e}^T_{i_1} & + & u^k_{i_2}\mathbf{e}^T_{i_2})y^k & = & s^k \\ (H_k + u^k_{i_1}\mathbf{e}^T_{i_1} & + & u^k_{i_2}\mathbf{e}^T_{i_2})y^{k-1} & = & s^{k-1}, \end{array} \right. \tag{11}$$

or, in an equivalent way,

$$\left\{ \begin{array}{rcrcl} u^k_{i_1}(\mathbf{e}^T_{i_1}y^k) & + & u^k_{i_2}(\mathbf{e}^T_{i_2}y^k) & = & s^k - H_k y^k \\ u^k_{i_1}(\mathbf{e}^T_{i_1}y^{k-1}) & + & u^k_{i_2}(\mathbf{e}^T_{i_2}y^{k-1}) & = & s^{k-1} - H_k y^{k-1}. \end{array} \right. \tag{12}$$

The equations (12) represent, for each k, a linear system of $2n$ equations and $2n$ unknowns: the components of the vectors $u^k_{i_1}$ and $u^k_{i_2}$.

Using the notation

$$\begin{array}{rclcrcl} \mathbf{e}^T_{i_1}y^k & = & \alpha^k & & \mathbf{e}^T_{i_2}y^k & = & \beta^k, \\ \mathbf{e}^T_{i_1}y^{k-1} & = & \gamma^k & & \mathbf{e}^T_{i_2}y^{k-1} & = & \delta^k, \end{array} \tag{13}$$

system (12) in matricial form is given by

$$Au^k = \begin{pmatrix} \alpha^k \mathbf{I} & \vdots & \beta^k \mathbf{I} \\ \cdots & & \cdots \\ \gamma^k \mathbf{I} & \vdots & \delta^k \mathbf{I} \end{pmatrix} \begin{pmatrix} u_{i_1}^k \\ \cdots \\ u_{i_2}^k \end{pmatrix} = \begin{pmatrix} s^k - H_k y^k \\ \cdots\cdots\cdots\cdots \\ s^{k-1} - H_k y^{k-1} \end{pmatrix} = \begin{pmatrix} v_1^k \\ \cdots \\ v_2^k \end{pmatrix} = v^k, \quad (14)$$

where, $A \in I\!R^{2n \times 2n}$, I is the $n \times n$ identity matrix, $u^k \in I\!R^{2n}$ and $v^k \in I\!R^{2n}$.

Therefore, the existence of the vectors $u_{i_1}^k$ e $u_{i_2}^k$ satisfying (9) will be determined by the nonsingularity of the matrix $A \in I\!R^{2n \times 2n}$. It is easy to see that the determinant of A is given by

$$det(A) = \left[det \begin{pmatrix} \alpha^k & \beta^k \\ \gamma^k & \delta^k \end{pmatrix} \right]^n = \sigma_k^n.$$

This shows an interesting fact: analyzing the nonsingularity of the $2n \times 2n$ matrix A is equivalent to analyze the nonsingularity of a 2×2 matrix.

If we assume that $\sigma^k = \alpha^k \delta^k - \gamma^k \beta^k \neq 0$, then the matrix A will be nonsingular. In order to find an general expression for the vector u^k in (14), its necessary to solve a linear system, what may be done using, for example LU decomposition which it is the strategy that we use as follows.

Case 1: $|\alpha^k| \geq |\gamma^k| > 0$:

$$A = LU \quad = \quad \begin{pmatrix} \mathbf{I} & \vdots & \mathbf{O} \\ \cdots & \cdots & \cdots \\ \dfrac{\gamma^k}{\alpha^k} \mathbf{I} & \vdots & \mathbf{I} \end{pmatrix} \begin{pmatrix} \alpha^k \mathbf{I} & \vdots & \beta^k \mathbf{I} \\ \cdots & \cdots & \cdots\cdots\cdots \\ \mathbf{O} & \vdots & \dfrac{\alpha^k \delta^k - \beta^k \gamma^k}{\alpha^k} \mathbf{I} \end{pmatrix}.$$

Case 2: $|\alpha^k| < |\gamma^k|$:

$$\begin{aligned} LU &= \begin{pmatrix} \mathbf{I} & \vdots & \mathbf{O} \\ \cdots & \cdots & \cdots \\ \dfrac{\alpha^k}{\gamma^k}\mathbf{I} & \vdots & \mathbf{I} \end{pmatrix} \begin{pmatrix} \gamma^k \mathbf{I} & \vdots & \delta^k \mathbf{I} \\ \cdots & \cdots & \cdots\cdots \\ \mathbf{O} & \vdots & \dfrac{\gamma^k\beta^k - \alpha^k\delta^k}{\gamma^k}\mathbf{I} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{O} & \vdots & \mathbf{I} \\ \cdots & \cdots & \cdots \\ \mathbf{I} & \vdots & \mathbf{O} \end{pmatrix} \begin{pmatrix} \alpha^k \mathbf{I} & \vdots & \beta^k \mathbf{I} \\ \cdots & \cdots & \cdots \\ \gamma^k \mathbf{I} & \vdots & \delta^k \mathbf{I} \end{pmatrix} = PA. \end{aligned}$$

Using this LU decomposition, we solve the system (14), that this,

$$LUu^k = v^k,$$

thus the expression for the vector u^k is given by

$$u^k = \begin{pmatrix} \dfrac{\delta^k v_1^k - \beta^k v_2^k}{\sigma^k} \\ \cdots\cdots\cdots \\ \dfrac{\alpha^k v_2^k - \gamma^k v_1^k}{\sigma^k} \end{pmatrix} = \begin{pmatrix} u_{i_1}^k \\ \cdots \\ u_{i_2}^k \end{pmatrix}. \tag{15}$$

substituting (15) in (10), we obtain

$$H_{k+1} = H_k + \left(\frac{\delta^k v_1^k - \beta^k v_2^k}{\sigma^k}\right)\mathbf{e}_{i_1}^T + \left(\frac{\alpha^k v_2^k - \gamma^k v_1^k}{\sigma^k}\right)\mathbf{e}_{i_2}^T. \tag{16}$$

As we observed previously, the matrix H_{k+1} differ from the matrix H_k only in two columns (i_1 and i_2). From the equality (16), it is possible to write these columns in the following way

$$\begin{aligned} h_{i_1}^{k+1} &= h_{i_1}^k + \frac{\delta^k v_1^k - \beta^k v_2^k}{\sigma^k}, \\ h_{i_2}^{k+1} &= h_{i_2}^k + \frac{\alpha^k v_2^k - \gamma^k v_1^k}{\sigma^k}. \end{aligned} \tag{17}$$

Substituting the expressions of the vectors v_1^k and v_2^k given in (12) in (17) we obtain:

$$\begin{aligned} h_{i_1}^{k+1} &= h_{i_1}^k + \frac{\delta^k}{\sigma^k}\left(s^k - H_k y^k\right) - \frac{\beta^k}{\sigma^k}\left(s^{k-1} - H_k y^{k-1}\right), \\ h_{i_2}^{k+1} &= h_{i_2}^k + \frac{\alpha^k}{\sigma^k}\left(s^{k-1} - H_k y^{k-1}\right) - \frac{\gamma^k}{\sigma^k}\left(s^k - H_k y^k\right). \end{aligned} \tag{18}$$

¿From (18), for each $j = 1, \cdots, n$, the jth-component of the columns to be modified will be updated as follows.

$$\begin{aligned} h_{j\,i_1}^{k+1} &= \frac{\delta^k}{\sigma^k}\left(s_j^k - \sum_{p \neq i_1} h_{j\,p}^k\, y_p^k\right) - \frac{\beta^k}{\sigma^k}\left(s_j^{k-1} - \sum_{p \neq i_1} h_{j\,p}^k\, y_p^{k-1}\right) \\ h_{j\,i_2}^{k+1} &= \frac{\alpha^k}{\sigma^k}\left(s_j^{k-1} - \sum_{p \neq i_2} h_{j\,p}^k\, y_p^{k-1}\right) - \frac{\gamma^k}{\sigma^k}\left(s_j^k - \sum_{p \neq i_2} h_{j\,p}^k\, y_p^k\right). \end{aligned} \tag{19}$$

It is interesting to observe that, computationally, it is more convenient to write the columns i_1 e i_2 of the new matrix in this way:

$$\begin{aligned} h_{j\,i_1}^{k+1} &= \frac{\delta^k}{\sigma^k}\left(s_j^k - \sum_{p \neq i_1,\, i_2} h_{j\,p}^k y_p^k\right) - \frac{\beta^k}{\sigma^k}\left(s_j^{k-1} - \sum_{p \neq i_1,\, i_2} h_{j\,p}^k y_p^{k-1}\right), \\ h_{j\,i_2}^{k+1} &= \frac{\alpha^k}{\sigma^k}\left(s_j^{k-1} - \sum_{p \neq i_1,\, i_2} h_{j\,p}^k y_p^{k-1}\right) - \frac{\gamma^k}{\sigma^k}\left(s_j^k - \sum_{p \neq i_1,\, i_2} h_{j\,p}^k y_p^k\right), \end{aligned}$$

which could be easily obtained from (19).

As it was mentioned previously, the choice of the index i_1 and i_2 of the columns to be modified is restricted to the assumption:

$$\sigma^k = (e_{i_1}^T y^k)(e_{i_2}^T y^{k-1}) - (e_{i_2}^T y^k)(e_{i_1}^T y^{k-1}) \neq 0. \tag{20}$$

Notice that, in the case that y^k becomes a multiple of y^{k-1}, σ^k will be zero. This makes it impossible to choose the columns that must be changed.

We adopted in our numerical tests the following choice for the index i_1 and i_2 :

$$|y_{i_1}^k| = \|y^k\|_\infty \qquad\qquad |y_{i_2}^{k-1}| = \|y^{k-1}\|_\infty.$$

In the case that σ^k becomes zero, we changed the index i_2 in such a way that

$$|(\alpha^k y^{k-1} - \gamma^k y^k)_{i_2}| = \|\alpha^k y^{k-1} - \gamma^k y^k\|_\infty.$$

3 The convergence

¿From now on, we denoted by $\|.\|$ the 2-norm vectors and matrices. Assume that $F : \Omega \subset I\!R^n \to I\!R^n$, $F \in C^1(\Omega)$, Ω an opens and convex set, $x_* \in \Omega$, $F(x_*) = 0$ and

$$\|J(x) - J(x^*)\| \leq L\|x - x^*\|^p, \quad L, p > 0 \tag{21}$$

for all $x \in \Omega$. A inequality (21) implies that for all $u, v \in \Omega$

$$\|F(u) - F(v) - J(x_*)(v - u)\| \leq L\|v - u\|\sigma(u, v)^p, \tag{22}$$

where $\sigma(u, v) = max\{\|u - x_*\|, \|v - x_*\|\}$ (see [1]).

Assume that $J(x_*)$ is nonsingular and define $M = \|J(x_*)^{-1}\|$. By (22), we deduce that for all $u, v \in \Omega$,

$$\|v - u - J(x_*)^{-1}[F(v) - F(u)]\| \leq ML\|v - u\|\sigma(u, v)^p. \tag{23}$$

The local convergence result is stated in the following theorem. It is very similar to the **Theorem 3.1** of [16] but, since its proof involves some interesting adaptations we will present it here.

Theorem 2.1 *Let $\{x^k\}$ e $\{H_k\}$ the sequences generated by the ITCUM and assume that $F(x^k) \neq 0$ and $\sigma^k \neq 0$, for all $k = 0, 1, ...$; let $r \in (0, 1)$. There exist $\varepsilon = \varepsilon_r$, $\eta = \eta_r$ such that, if $\|x^0 - x^*\| \leq \varepsilon$ and $\|H_k - J(x^*)^{-1}\| \leq \eta$, whenever*

$k \equiv 1 \ (mod\ m)$ or $k = 0$, then the sequences $\{x^k\}$ and $\{H_k\}$ are well defined, $\{x^k\}$ converges to x^* and for all $k = 0, 1, \cdots$

$$\|x^{k+1} - x^*\| \leq r\|x^k - x^*\|.$$

Proof:

Define $c_1 = 2n^2M^2L$, $c_2 = n^{5/2}$. Given $\varepsilon,\ \eta > 0$, define $b_i(\varepsilon,\ \eta)$, $i = 0,\ 1,\ \cdots,\ m-1$ by

$$\begin{aligned} b_0(\varepsilon,\ \eta) &= \eta \\ b_1(\varepsilon,\ \eta) &= c_2 b_0(\varepsilon,\ \eta) + c_1\varepsilon^p \\ b_i(\varepsilon,\ \eta) &= R\, c_2\, b_{i-1}(\varepsilon,\ \eta) + R\, c_1\, \varepsilon^p, \qquad i = 2,\ \cdots,\ m-1, \end{aligned} \tag{24}$$

where $R = \dfrac{2\|y^{k-1}\|_\infty \|y^{k-2}\|_\infty}{|\sigma^k|}, \quad k = 2, 3, \cdots$

Clearly, we have, for all $\varepsilon,\ \eta > 0$,

$$0 < b_0(\varepsilon,\ \eta) < b_1(\varepsilon,\ \eta) < \cdots < b_{m-1}(\varepsilon,\ \eta) \quad \text{and} \quad \lim_{\varepsilon,\ \eta \to 0} b_i(\varepsilon,\ \eta) = 0 \tag{25}$$

for $i = 0,\ 1,\ \cdots,\ m-1$.

By (25), we can choose $\varepsilon = \varepsilon_r > 0$ and $\eta = \eta_r > 0$ such that $\varepsilon \leq \varepsilon_1$ and

$$b_i(\varepsilon,\ \eta) + L\varepsilon^p < \frac{r}{M_1}, \tag{26}$$

for $i = 0,\ 1,\ \cdots,\ m-1$, where $M_1 = \max\{\|J(x^*)\|, 2M\}$.

Assume that $\|x^0 - x^*\| \leq \varepsilon$ and $\|H_k - J(x^*)^{-1}\| \leq \eta$ whenever $k \equiv 1 \,(\mathrm{mod}\, m)$ or $k = 0$. We will prove by induction on k that if $k \equiv q \,(\mathrm{mod}\, m)$ then H_k is nonsingular,

$$\|H_k - J(x^*)^{-1}\| \leq b_q(\varepsilon,\ \eta) \tag{27}$$

$$\|H_k\| \leq 2M, \tag{28}$$

$$\|x^{k+1} - x^*\| \leq r\,\|x^k - x^*\|, \tag{29}$$

for all $q = 0,\ 1,\ \cdots, m-1$.

For $k = 0$, by hypothesis,

$$\|H_0 - J(x^*)^{-1}\| \leq \eta = b_0(\varepsilon,\ \eta), \tag{30}$$

thus, by (26) and (30),

$$\begin{aligned} \|H_0\| &\leq \|J(x^*)^{-1}\| + \|H_0 - J(x^*)^{-1}\| \\ &\leq \|J(x^*)^{-1}\| + \eta \\ &\leq \|J(x^*)^{-1}\| + \frac{1}{\|J(x^*)\|} \\ &\leq 2\|J(x^*)^{-1}\| = 2M. \end{aligned}$$

Thus,

$$\|H_0\| \leq 2M. \tag{31}$$

By (22) and (31),

$$\begin{aligned} \|x^1 - x^*\| &= \|x^0 - x^* - H_0F(x^0)\| \\ &= \|x^0 - x^* - H_0[F(x^0) - F(x^*) - J(x^*)(x^0 - x^*)] \\ &\quad + H_0J(x^*)(x^0 - x^*)\| \\ &\leq \|[I - H_0J(x^*)](x^0 - x^*)\| + 2ML\|x^0 - x^*\|^{p+1} \\ &\leq \left(\|J(x^*)^{-1} - H_0\|\,\|J(x^*)\| + 2ML\|x^0 - x^*\|^p\right)\ \|x^0 - x^*\|, \end{aligned}$$

by the definition of M_1, the hypothesis $\|x^0 - x^*\| \leq \varepsilon,\ \|H_0 - J(x^*)^{-1}\| \leq \eta$, and (26), we have:

$$\begin{aligned} \|x^1 - x^*\| &\leq M_1\left(\|J(x^*)^{-1} - H_0\| + L\|x^0 - x^*\|^p\right)\ \|x^0 - x^*\| \\ &\leq M_1\left(\eta + L\varepsilon^p\right)\|x^0 - x^*\| \\ &= M_1\left(b_0(\varepsilon,\ \eta) + L\varepsilon^p\right)\|x^0 - x^*\| \ \leq\ r\,\|x^0 - x^*\|. \end{aligned}$$

Thus, $\|x^1 - x^*\| \leq r\,\|x^0 - x^*\|$.

Consider now $k > 0$, $k \equiv q \pmod{m}$. If $q = 1$, the proofs of (27)-(29) are similar to the case $k = 0$. Assume $q \neq 1$. let us assume now that $q > 0, k \geq 2$ and $k \equiv q \pmod{m}$.

By the hypothesis of induction, H_{k-1} is nonsingular. Let i_1 and i_2, the indexes of the columns to be modified, such that $\sigma^{k-1} \neq 0$. Each component j, $j = 1, 2, \cdots, n$ of the column i_1 is given by:

$$h^k_{j\,i_1} = \frac{\delta^{k-1}}{\sigma^{k-1}} \left(s^{k-1}_j - \sum_{p \neq i_1} h^{k-1}_{j\,p} y^{k-1}_p \right) - \frac{\beta^{k-1}}{\sigma^{k-1}} \left(s^{k-2}_j - \sum_{p \neq i_1} h^{k-1}_{j\,p} y^{k-2}_p \right). \tag{32}$$

so, H_k is well defined.

By addition and subtraction of $\dfrac{\delta^{k-1}}{\sigma^{k-1}} \left(\displaystyle\sum_{p \neq i_r} h^*_{j\,p} y^{k-1}_p \right)$ and $\dfrac{\beta^{k-1}}{\sigma^{k-1}} \left(\displaystyle\sum_{p \neq i_r} h^*_{j\,p} y^{k-2}_p \right)$, respectively, in (32), we have:

$$\begin{aligned} h^k_{j\,i_r} &= \frac{\delta^{k-1}}{\sigma^{k-1}} \left(s^{k-1}_j - \sum_{p \neq i_1} h^*_{j\,p} y^{k-1}_p + \sum_{p \neq i_1} h^*_{j\,p} y^{k-1}_p - \sum_{p \neq i_1} h^{k-1}_{j\,p} y^{k-1}_p \right) \\ &- \frac{\beta^{k-1}}{\sigma^{k-1}} \left(s^{k-2}_j - \sum_{p \neq i_1} h^*_{j\,p} y^{k-2}_p + \sum_{p \neq i_1} h^*_{j\,p} y^{k-2}_p - \sum_{p \neq i_1} h^{k-1}_{j\,p} y^{k-2}_p \right), \end{aligned} \tag{33}$$

Therefore, for all $j = 1, 2, \cdots, n$,

$$\begin{aligned} |h^k_{j\,i_1} - h^*_{j\,i_1}| \;\leq\; & \left| \frac{\delta^{k-1}}{\sigma^{k-1}} \right| |s^{k-1}_j - \sum_{p=1}^{n} h^*_{j\,p} y^{k-1}_p| + \left| \frac{\beta^{k-1}}{\sigma^{k-1}} \right| |s^{k-2}_j - \sum_{p=1}^{n} h^*_{j\,p} y^{k-2}_p| + \\ & \left| \frac{\delta^{k-1}}{\sigma^{k-1}} \right| \sum_{p \neq i_1} |h^*_{j\,p} - h^{k-1}_{j\,p}|\, |y^{k-1}_p| + \left| \frac{\beta^{k-1}}{\sigma^{k-1}} \right| \sum_{p \neq i_1} |h^*_{j\,p} - h^{k-1}_{j\,p}| |y^{k-2}_p|. \end{aligned} \tag{34}$$

Defining $J(x^*)^{-1} = H^*$ and using (23), the inequalities $|y^{k-1}_p| \leq \|y^{k-1}\|_\infty$ and $|y^{k-2}_p| \leq \|y^{k-2}\|_\infty$ in (34), we obtain:

$$|h^k_{j\,i_1} - h^*_{j\,i_1}| \;\leq\; \left| \frac{\delta^{k-1}}{\sigma^{k-1}} \right| \|s^{k-1} - J(x^*)^{-1} y^{k-1}\| + \left| \frac{\beta^{k-1}}{\sigma^{k-1}} \right| \|s^{k-2} - J(x^*)^{-1} y^{k-2}\|$$

$$
\begin{aligned}
&+ \left|\frac{\delta^{k-1}}{\sigma^{k-1}}\right| \|y^{k-1}\|_\infty \sum_{p=1}^{n} |h^*_{j\,p} - h^{k-1}_{j\,p}| + \left|\frac{\beta^{k-1}}{\sigma^{k-1}}\right| \|y^{k-2}\|_\infty \sum_{p=1}^{n} |h^*_{j\,p} - h^{k-1}_{j\,p}| \\
&\leq \left|\frac{\delta^{k-1}}{\sigma^{k-1}}\right| ML\|s^{k-1}\|\varepsilon^p + \left|\frac{\beta^{k-1}}{\sigma^{k-1}}\right| ML\|s^{k-2}\|\varepsilon^p + R\sum_{p=1}^{n} |h^*_{j\,p} - h^{k-1}_{j\,p}| \\
&\leq \left|\frac{\delta^{k-1}}{\sigma^{k-1}}\right| 2M^2L\|y^{k-1}\|\varepsilon^p + \left|\frac{\beta^{k-1}}{\sigma^{k-1}}\right| 2M^2L\|y^{k-2}\|\varepsilon^p \\
&\quad + R\,n\,\|J(x^*)^{-1} - H_{k-1}\|. \qquad (35)
\end{aligned}
$$

By **Lemma 3.1** we have that $\|s^{k-1}\| \leq 2M\,\|y^{k-1}\|$ and $\|s^{k-2}\| \leq 2M\|y^{k-2}\|$. We used this in the two previous inequalities.

Using $|\delta^{k-1}| \leq \|y^{k-2}\| \leq \sqrt{n}\,\|y^{k-2}\|_\infty$ and $|\beta^{k-1}| \leq \|y^{k-1}\| \leq \sqrt{n}\,\|y^{k-1}\|_\infty$, in (35) we have:

$$
\begin{aligned}
|h^k_{j\,i_1} - h^*_{j\,i_1}| &\leq \left(\frac{\|y^{k-1}\|_\infty}{|\sigma^{k-1}|}\|y^{k-2}\|_\infty + \frac{\|y^{k-2}\|_\infty}{|\sigma^{k-1}|}\,\|y^{k-1}\|_\infty\right)\sqrt{n}\;2M^2L\;\varepsilon^p \\
&\quad + R\,n\,\|H_{k-1} - J(x^*)^{-1}\|, \\
&= R\,\sqrt{n}\,2M^2L\,\varepsilon^p + R\,n\,\|H_{k-1} - J(x^*)^{-1}\|. \qquad (36)
\end{aligned}
$$

In similar form, it is easy to proof an analogous result to (36) for the index i_2. That is,

$$
|h^k_{j\,i_2} - h^*_{j\,i_2}| \leq R\,\sqrt{n}\,2M^2L\,\varepsilon^p + R\,n\,\|H_{k-1} - J(x^*)^{-1}\|. \qquad (37)
$$

Also, we have that for the components of the columns that were not modified a inequality (36)is satisfied. i.e, for all $s \neq i_1$ and $s \neq i_2$,

$$
\begin{aligned}
|h^k_{j\,s} - h^*_{j\,s}| &= |h^{k-1}_{j\,s} - h^*_{j\,s}| \\
&\leq \|H_{k-1} - J(x^*)^{-1}\| \\
&\leq R\,\sqrt{n}\,2M^2L\,\varepsilon^p + R\,n\,\|H_{k-1} - J(x^*)^{-1}\|. \qquad (38)
\end{aligned}
$$

Observe that (38) is true because $R > 1$ and $R\sqrt{n}\,2M^2L\,\varepsilon^p > 0$. Moreover, by (36), (37) and(38), we conclude that

$$\|H_k - J(x^*)^{-1}\|_\infty \leq nR\left(\sqrt{n}\,2M^2L\,\varepsilon^p + n\,\|H_{k-1} - J(x^*)^{-1}\|\right). \tag{39}$$

Thus, by (39) and (24), we have:

$$\begin{aligned}
\|H_k - J(x^*)^{-1}\| &\leq \sqrt{n}\,\|H_k - J(x^*)^{-1}\|_\infty \\
&\leq R\left(n^2\,2\,M^2L\,\varepsilon^p + n^{5/2}\,\|H_{k-1} - J(x^*)^{-1}\|\right) \\
&\leq R\left(n^2\,2\,M^2L\,\varepsilon^p + n^{5/2}\,b_{q-1}(\varepsilon,\,\eta)\right) \\
&= R\,(c_2\,b_{q-1}(\varepsilon,\,\eta) + c_1\,\varepsilon^p) \\
&= b_q(\varepsilon,\,\eta).
\end{aligned} \tag{40}$$

Then, $\|H_k - J(x^*)^{-1}\| \leq b_q(\varepsilon,\,\eta)$. Thus, by (26),

$$\|H_k - J(x^*)^{-1}\| \leq \frac{r}{M_1} \leq \frac{1}{2M},$$

therefore, by Banach's Lemma [5], H_k is nonsingular and using the hypothesis, $\sigma^{k-1} \neq 0$, we conclude that for all k, the sequences $\{x_k\}$ and $\{H_k\}$ are well defined.

Moreover

$$\begin{aligned}
\|H_k\| &\leq \|J(x^*)^{-1}\| + \|H_k - J(x^*)^{-1}\| \\
&\leq \|J(x^*)^{-1}\| + \frac{r}{M_1} \leq \|J(x^*)^{-1}\| + \frac{1}{\|J(x^*)\|} \\
&\leq 2\,\|J(x^*)^{-1}\| = 2\,M.
\end{aligned} \tag{41}$$

So, $\|H_k\| \leq 2\,M$ and finally, by (22), (26) and (41),

$$\begin{aligned}
\|x^{k+1} - x^*\| &= \|x^k - x^* - H_kF(x^k)\| \\
&= \|x^k - x^* - H_k\left[F(x^k) - F(x^*) - J(x^*)(x^k - x^*)\right] \\
&\quad - H_kJ(x^*)(x^k - x^*)\| \\
&\leq \|\left[I - H_kJ(x^*)\right](x^k - x^*)\| + 2\,M\,L\,\|x^k - x^*\|^{p+1}
\end{aligned}$$

$$
\begin{aligned}
&\leq \left[\|J(x^*)\|\,\|J(x^*)^{-1} - H_k\| + 2\,M\,L\,\|x^k - x^*\|^p\right]\,\|x^k - x^*\| \\
&\leq M_1\,(b_q(\varepsilon,\,\eta) + L\,\varepsilon^p)\,\|x^k - x^*\| \\
&\leq r\,\|x^k - x^*\|.
\end{aligned}
$$

Thus, $\|x^{k+1} - x^*\| \leq r\,\|x^k - x^*\|$, which completes the proof of the theorem. ∎

4 Computer implementation of ITCUM and numerical experiments

In this section, we present some comparative implementations of ITCUM. For this it was used some test problems from [16], [8], [3].

¿From the equation (16), letting $v_1^p = s^p - H_p\,y^p$ and $v_2^p = s^{p-1} - H_p\,y^{p-1}$, we obtain:

$$
H_k = H_0 + \sum_{p=0}^{k-1}\left(\frac{\delta^p v_1^p - \beta^p v_2^p}{\sigma^p}\right)(\mathbf{e}_{i_1}^p)^T + \sum_{p=0}^{k-1}\left(\frac{\alpha^p v_2^p - \gamma^p v_1^p}{\sigma^p}\right)(\mathbf{e}_{i_2}^p)^T, \tag{42}
$$

or equivalently

$$
H_k = H_0 + \sum_{p=0}^{k-1} w_1^p(\mathbf{e}_{i_1}^p)^T + \sum_{p=0}^{k-1} w_2^p(\mathbf{e}_{i_2}^p)^T, \tag{43}
$$

where

$$
w_1^p = \frac{\delta^p v_1^p - \beta^p v_2^p}{\sigma^p} \quad \text{and} \quad w_2^p = \frac{\alpha^p v_2^p - \gamma^p v_1^p}{\sigma^p}.
$$

The implementation of ITCUM is based on the formula (43). Thus, in each iteration k, the calculus of H_k implies in the storage of two vectors (w_1^k e w_2^k) and two additional indexes (i_1 and i_2). For this reason, the number of consecutive iterations of the method is limited by the availability of memory to the computer.

Considering that there is sufficient space to store m pairs of vectors, then it is possible to do one "Newton" iteration[1] and m consecutive ITCUM iterations.

[1] In the restarts, we did not use the " exact" Jacobian.

Therefore, if $\rho \equiv 0 \,(\mathrm{mod}\, m)$, $\theta \in \{1, ..., m-1\}$, we obtain:

$$H_{\rho+\theta} = H_\rho + \sum_{l=0}^{\theta-1} w_1^{\rho+l} (\mathbf{e}_{i_1}^{\rho+l})^T + \sum_{l=0}^{\theta-1} w_2^{\rho+l} (\mathbf{e}_{i_2}^{\rho+l})^T. \tag{44}$$

Then, the parameter m determines the number of possibles iterations of type ITCUM between two restarts. We used $m = 30$ when we worked with the large-scale problems (for the other problems it was not necessary to do restarts).

The problems tested in this work were organized in two classes: short-scale and large-scale problems according to the number of variables that they have. We present them as follows.

Short-scale problems :

1: Rosenbrock (n=2). [3]. $x_0 = (-1.2,\, 1)^T$.

2: Freudenstein-Roth (n=2). [3]. $x_0 = (0.5,\, -2)^T$.

3: Powell badly scaled function (n=2). [3]. $x_0 = (0.5,\, -2)^T$.

4: Powell singular function (n=4). [3]. $x_0 = (0.5,\, -2)^T$.

5: Extended Rosenbrock (n=50). [3]. $x_0 = (-1.2,\, 1, -1.2,\, 1, ...)^T$.

6: Trigonometric function (n=2). [3]. $x_0 = (1/n,\, ...,\, 1/n)^T$.

7: Discrete boundary value function (n=2). Function 28 in [3].
$x_0 = (\xi_j)$, where $\xi_j = t_j(t_j - 1)$, $h = 1/(n+1)$ e $t_j = jh$.

8: Broyden banded function (n=2). [3]. $x_0 = (-1,\, ...,\, -1)^T$.

9: Linear System (n=50). [8]. $x_0 = (1,\, -1, 1,\, -1, ...)^T$.

10: Chandrasekhar H-equation (n=50). [8]. $x_0 = (0,\, ...,\, 0)^T$.

Large-scale problems:

Problems 11 to 15: Each test is generated as a finite-difference discretization of a Poisson equation in the square [0, 1] × [0, 1]. The number of divisions of the interval is denoted by N (32 and 50). In all cases, the starting point is $x_0 = (-1, -1, ..., -1)^T$ and the number of variables is $n = (N-1)^2$.

The Jacobian of each one of the large-scale problems is sparse with fivediagonal structure; thus it can be considered "well represented" by its the tridiagonal part. Motivated by this fact, if $k \equiv 0 \pmod{m}$, we chose:

$$H_k = [\mathcal{P}_\tau(J(x^k))]^{-1}, \tag{45}$$

where, $\mathcal{P}_\tau$ is the orthogonal projection operator on the subspace of tridiagonal matrices. Thus, the algorithms are restarted using $m = 30$.

For the short-scale problems, when $k = 0, 1$, we chose:

$$H_k = [\mathcal{P}_\mathcal{D}(J(x^k))]^{-1}, \tag{46}$$

where, $\mathcal{P}_\mathcal{D}$ is the orthogonal projection operator on the subspace of diagonal matrices. If one of the elements of this diagonal Jacobian matrix is null, we replaced it by 1.

In according to **Lema 2.1**, near an isolated solution it is not possible that $y^k = 0$. This fact may occur far from x^* which makes $\sigma^{k-1} = 0$, independently of the choice of the index. In the numerical tests, this situation is detected verifying the inequality:

$$\|y^k\| \leq 10^{-6} \|F(x^k)\|. \tag{47}$$

In the case (47) is satisfied, we defined $H_{k+1} = H_k$.

In each iteration k, the choice of indexes i_1 and i_2 of the columns to be modified was done according to the description in **Section 2,** with a small variation to avoid instabilities problems in h_{k+1}. For this, we defined a parameter for changing the value of sigma (tol_σ). Thus, the index i_2 will be altered when:

$$|\sigma^k| \leq tol_\sigma. \tag{48}$$

In our numerical tests we used $tol_\sigma = 10^{-4}$ for the problems 11, 12, 13 and $tol_\sigma = 10^{-6}$ for the other problems.

Based on [16], we used the following convergence criteria:

$$\|F(x^k)\|_\infty \leq 10^{-5}\|F(x^0)\|_\infty \qquad \text{short-scale}$$

$$\|F(x^k)\|_\infty \leq 10^{-3}\|F(x^0)\|_\infty \qquad \text{large-scale.}$$

We also stopped the execution of the numerical tests when the number of iterations exceeded 200 or when $\|F(x^k)\|_\infty \geq 10^4\|F(x^0)\|_\infty$. In the first case, we say that ITCUM did not converge (it is represented, in the tables, by the term NC) and in the second one, we say that the method diverged (which it will be represented in the tables with the term DIV).

We compare the performance of ITCUM with the Newton method and with other quasi-Newton methods: Broyden's method [1], CUM [11], ICUM [16]. The implementation of these methods were done as in [16].

The numerical tests were run in an AMD Athlon - 800 MHZ computer, in the state University of Campinas using the MATLAB 6.0 with single precision.

The numerical results are presented in **Tables 1** to **3.** Each one of them has six columns indicating, respectively, the problem, the number of iterations used for Newton, Broyden, CUM, ICUM and ITCUM. For the large scale problems aditional to the number of iterations used for each method (KON), we present the computer CPU time in seconds (TIME). In this case, the results of each test is represented by a pair (KON; TIME).

Prob	*Newton*	*Broyden*	*CUM*	*ICUM*	*ITCUM*
1	2	12	13	8	5
2	41	Div	Div	19	NC
3	11	33	40	83	22
4	4	60	67	Div	56
5	2	12	13	8	5
6	9	8	8	9	8
7	2	5	5	5	4
8	4	6	5	5	6
9	1	4	5	4	4

Table1: *Short-scale problems.*

c	*Newton*	*Broyden*	*CUM*	*ICUM*	*ITCUM*
0.1	3	3	4	4	3
0.5	3	6	6	6	5
0.9	5	10	10	9	7
0.99	6	12	33	12	11
0.999	7	14	39	13	13
$1-10^{-4}$	8	17	32	15	13
$1-10^{-5}$	9	24	38	16	15
$1-10^{-6}$	10	27	43	17	16
$1-10^{-7}$	10	31	39	17	16
$1-10^{-8}$	10	28	33	17	16
1	10	33	33	17	16

Table 2: *Chandrasekhar H-equation.*

Pr	N	*Newton*	*Broyden*	*CUM*	*ICUM*	*ITCUM*
11	32	(2; 0.65)	(62; 5.55)	(59; 4.89)	(54; 4.67)	(44; 4.55)
	50	(2; 2.97)	(112; 38.83)	(105; 33.89)	(67; 22.30)	(82; 29.33)
12	32	(5; 1.54)	(52; 4.39)	(56; 5.11)	(47; 4.45)	(44; 4.61)
	50	(4; 5.76)	(99; 35.08)	(86; 27.74)	(65; 21.59)	(66; 24.50)
13	32	(9; 2.70)	(65; 5.94)	(64; 5.93)	(55; 5.05)	(56; 5.61)
	50	(9; 12.58)	(149; 52.68)	(75; 24.88)	(64; 21.42)	(62; 23.81)
14	32	(2; 0.71)	(68; 5.99)	(95; 7.96)	(62; 5.54)	(51; 5.06)
	50	(2; 3.08)	(155; 59.65)	(176; 55.42)	(92; 29.55)	(101; 35.70)
15	32	(1; 0.39)	(62; 4.72)	(82; 5.82)	(61; 5.00)	(57; 5.05)
	50	(1; 1.59)	(132; 43.17)	(141; 41.36)	(115; 34.38)	(111; 36.52)

Table 3: *Large-scale Problems.*

Observe that in several cases, the performance of ITCUM is worse that of the ICUM, which is not strange because, in nonlinear problems, it is practically impossible to find the best method (in performance) for all the problems.

For determining the new columns, while ICUM has to manipulate with only one equation, ITCUM needs to solve a linear system, where there exists the possibility of null determinant of the matrix of the system.

When the dimension of the problems increases, the performance of ITCUM becomes, in mean, inferior to that of ICUM (**Table 3**); This fact occurs because for the implementation of this method, the choice of indexes becomes more complicated because of the size of the vectors.

Other numerical tests, different from that mentioned previously, were done. In these tests, we worked with several versions of ITCUM, generated from the dif-

ferent choices for the index i_1 e i_2 and from the various criteria used for to alter them (when it is necessary) in the implementation. Among the versions that we used, the criterion of choice described in **Section 3** was of the best performance in the problems tested in this work, as was expected by the observation??????

References

[1] Broyden, C. G.; Dennis, J. E. Jr; Moré, J. J. (1973). On the local and superlinear convergence of quasi-Newton methods, *J. Inst. Math. Appl.* **12**, pp 223-245.

[2] Cunha, M. C. C. (2000). *Métodos Numéricos*, Editora da UNICAMP, 2.ed., Campinas, SP.

[3] Dennis, J. E. Jr; Moré, J. J.(1997). Quase-Newton methods, motivation and theory, *SIAM Review* **19**, pp 46-89.

[4] Dennis, J. E. Jr; Schnabel, R. B.(1983). *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice Hall, Englewood Cliffs, N.J.

[5] Golub, G. H.; Van Loan, Ch. F.(1995). *Matrix Computations*, The Johns Hopkins University Press, 3nd. edition, Baltimore and London.

[6] Gomes-Ruggiero, M. A. (1990). *Método quase-Newton para resolução de sistemas não lineares esparsos e de grande porte*, Tese de Doutorado, FEE-Unicamp, Campinas, Brasil.

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

[8] Lopes, V. L. R.; Martínez, J. M.(1995). Convergence properties of the inverse column-updating method, *Optimization Methods and Software* **6**, pp 127-144.

[9] Lukšan,L.; Vlček, J.(1998). Computational experience with globally convergent descent methods for large sparse systems of nonlinear equations, *Optimization Methods and Software* **8**, pp 185-199.

[10] Martínez, J. M.(1979). Three new algorithms based on the sequential secant method, *BIT* **19**, pp 236-243.

[11] Martínez, J. M.(1984). A quasi-Newton method with modification of one column per iteration, *Computing* **33**, pp 353-362.

[12] Martínez, J. M.; Ochi, L. S.(1982). Sobre dois métodos de Broyden, *Matemática Aplicada e Computacional* **1**, pp 135-141.

[13] Martínez, J. M.; Zambaldi, M. C.(1992). An inverse column-updating method for solving large-scale nonlinear systems of equations, *Optimization Methods and Software* **1**, pp 129-140.

[14] Moré, J. J.; Garbow, B. S.; Hillstrom, K. E. (1981). Testing unconstrained optimization software, *ACM Transactions on Mathematical Software* **7**, pp 17-41.

[15] Ortega, J. M.; Rheinboldt, W. G. (1970). *Iterative solution of nonlinear equations in several variables*, Academic Press, NY.

[16] Zambaldi, M. C. (1993). *Novos resultados sobre fórmulas secantes e aplicações*, Tese de Doutorado, Departamento de Matemática Aplicada, UNICAMP, Campinas, Brasil.