Solving recent applications by quasi-Newton methods

Rosana Pérez * Véra Lucia da Rocha Lopes †

October 29, 2001

Resumo

Em [12], mostramos que existem muitos problemas recentes na pesquisa aplicada para os quais os métodos quase-Newton são a melhor opção para resolver sistemas de equações não lineares. Isto se deve ao fato de possuirem baixo custo computacional [9], [6], [5].

Motivados por esse trabalho e pelo fato do ICUM ter sido considerado recentemente o mais eficiente método quase-Newton para resolver sistemas não lineares de grande porte [7], nosso interesse atual é resolver numericamente alguns problemas reais usando métodos quase-Newton, em particular o ICUM.

Para isso, consideramos quatro problemas que ocorrem frequentemente em aplicações nas areas de Geofísica, Biologia, Engenharia e Física, respectivamente. Duas destas aplicações são descritas neste trabalho e são baseadas em trabalhos recentes [14], [10]. As outras duas aplicações foram descritas em [12] tendo como base [11], [13].

*Department of Mathematics, University of Cauca, Popayán, (Cauca), Colombia (rosana@atenea.ucauca.edu.co).

†Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas, SP, Brazil (vlopes@ime.unicamp.br).

Para resolver cada um dos problemas, devemos solucionar um sistema não linear. Para isto, usamos os métodos quase-Newton: Broyden e ICUM. Apresentamos uma análise comparativa cuidadosa dos resultados obtidos.

Abstract

In [12], we have shown that there are many recent problems in applied research for which the quasi-Newton methods are the best option for solving the nonlinear systems of equations that appear in the solution of such problems. The main reason for using these methods is because they have low computational cost [9], [6], [5].

Motivated by this work and by the fact that the ICUM, was considered recently as the most efficient quasi-Newton method for solving large-scale nonlinear systems [7], we are now interested in implementing it with some real problems.

For this, we consider in this work four problems of common occurrence in applications in Geophysics, Biology, Engineering and Physics, respectively. Two of them are described here based in recent works [14],[10]. The two other applications were described in [12] with base in [11],[13].

For solving each problem, we must solve a nonlinear system of equations. For this, we use the quasi-Newton methods: Broyden and ICUM and present a careful comparative analysis of the results obtained.

1 Introduction

There are many problems in different areas of the applied research for which the quasi-Newton methods [3],[8] are the best option for solving nonlinear systems of equations. In [12] we presented recent applications in areas such as Physics, Chemical Engineering, Electronic Engineering, Astrophysics, Electric Engineering and Mechanical Engineering. In general, these methods are chosen because of their low computational cost.

Solving a nonlinear system of equations consists on: given a nonlinear function $F : \mathbb{R}^n \to \mathbb{R}^n$, continuously differentiable, find a vector $x \in \mathbb{R}^n$ such that

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

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

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

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

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

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

In quasi-Newton methods, the matrices B_k are intended to be approximations of $J(x_k)$. In many methods, the computation of (3) does not involve computing derivatives at all. Moreover, in many particular methods, B_{k+1}^{-1} is obtained from B_k^{-1} using simple procedures thanks to which the linear algebra cost involved in (3) is much less than the one involved in (2).

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 [2], 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] and the Inverse Column Update Method (ICUM) [9], [6]. In the first one, the updating of the matrix B_k, is made by

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

and in the second one, the matrix B_k^{-1} is updated by

$$B_{k+1}^{-1} = B_k^{-1} + \frac{(s_k - B_k^{-1} y_k)\mathbf{e}_{j_k}^T}{\mathbf{e}_{j_k}^T y_k},$$

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

In this work we use the quasi-Newton methods: Broyden and ICUM to solve four real problems in Geophysics, Biology, Engineering and Physics, respectively. We chose these methods because the first one it is the most popular quasi-Newton method used for solving nonlinear systems and the second one because of its excellent performance in the solution of large-scale nonlinear systems [7].

In **Section 2**, based on the works [14], [10], we present a description of two recent applications that were not considered in [12]. In **Section 3**, we present a description of the numerical tests, their results and an analysis of them. Finally, in **Section 3,** the conclusions are presented.

2 Applications

In this section we described two of the four applications that we considered in the numerical test. The first one is a problem of common occurrence in seismological applications called Two-point ray tracing problem [14] and the second one is a recent problem about the interaction between two viruses [10].

The two others applications are related with target location and determination of basin of periodic trajectories of dynamical systems, respectively. We presented a description of them in [12] having as base, the recent works [11] and [13]. For the safe of understanding we make a brief abstract of each problem in the respective section.

We chose these four problems because we think that they are interesting real applications in different areas of the knowledge in which a nonlinear system of equation must be solved.

2.1 Two-point ray tracing problem (2001) [14] [4]

In general form, a Two-point ray tracing problem consists on constructing a ray that joins two given points in the domain.[1]

In [4], the earth structure is modelled by piecewise constant regions of arbitrary shape. The interfaces between regions as well as the free surface of the earth are assumed to be smooth curves. Generally, the convention used in this problem is that i_0 represent the free surface of the earth.

The medium between each successive pair of interfaces is assumed to be homogeneous, isotropic and perfectly elastic. Thus most kinds of signals can be propagate in such media. All the rays must be straight line segments in each region.[2] A ray is determined geometrically by knowing the initial or "source"

[1]In [14], they assume that the earth is represented by a two-dimensional (2D case) domain or by a three dimensional domain with cylindrical symmetry with respect to an axis perpendicular to the plain of interest (2.5D case).

[2]Thus no differential equations need to be solved [4].

point, the final or "receiver" point and each point at which the ray meets an interface. At the contact points Snell's law must hold. It is this condition that permits to determine the intersection points and thus the ray.

Mathematically, A Two-point ray tracing problem consist on, given

- two points: $\mathbf{X_0}$, the source point and $\mathbf{X_{n+1}}$, the receiver point, both located in some fixed interfaces,
- a velocity of the $k-$th region crossed, v_i, $i = 1, \cdots, n+1$,
- a finite sequence of positive integers $i_1, i_2, \cdots, i_n$ that represent the indices of the n interfaces intersected by the ray path, f_{i_k},

to find $\mathbf{X_k}$ for $k = 1, 2, \cdots, n$, a point located in the kth interface, where the Snell's law[14], [4] is satisfied.

Between each of these consecutive points the ray can be described by the line segment $[\mathbf{X_k}, \mathbf{X_{k-1}}]$, $k = 1, 2, \cdots, n+1$. Thus, the ray has n intersections points with the interfaces plus two endpoints: the source and receiver.

Due to the fact that the whole problem is characterized by the intersection points $\mathbf{X_k} = (x_k, f_{i_k}(x_k))^T$ in which the Snell's law must be satisfied, then they can be found solving a nonlinear system of equations. To see this, it is necessary to transform the Snell's law in a vectorial form, using the unitary vectors in the direction of the ray,

$$\frac{\mathbf{X_k} - \mathbf{X_{k-1}}}{\|\mathbf{X_k} - \mathbf{X_{k-1}}\|_2} \qquad \frac{\mathbf{X_{k+1}} - \mathbf{X_k}}{\|\mathbf{X_{k+1}} - \mathbf{X_k}\|_2}$$

and a tangent vector to the kth interface at $\mathbf{X_k}$, $\tau_k = (1, f'_{i_k}(x_k))^T$. Then Snell's law, in the most general form, requires that

$$v_{i_{k+1}} \left\langle \tau_k, \frac{\mathbf{X_k} - \mathbf{X_{k-1}}}{\|\mathbf{X_k} - \mathbf{X_{k-1}}\|_2} \right\rangle = v_{i_k} \left\langle \tau_k, \frac{\mathbf{X_{k+1}} - \mathbf{X_k}}{\|\mathbf{X_{k+1}} - \mathbf{X_k}\|_2} \right\rangle. \qquad (5)$$

Equation (5) represents a nonlinear system of n equations in n unknowns, the scalars $x_1, x_2, \cdots, x_n$, since the source point, $\mathbf{X_0} = (x_0, f_{i_0})^T$ and the receiver point, $\mathbf{X}_{n+1} = (x_n, f_{i_{n+1}})^T$, are assumed to be known. The kth equation is given by

$$\begin{aligned} \phi_v^k &= v_{i_{k+1}} \frac{(x_k - x_{k-1}) + f'_{i_k}(x_k)(f_{i_k}(x_k) - f_{i_{k-1}}(x_{k-1}))}{[(x_k - x_{k-1})^2 + (f_{i_k}(x_k) - f_{i_{k-1}}(x_{k-1}))^2]^{1/2}} \\ &\quad - v_{i_k} \frac{(x_{k+1} - x_k) + f'_{i_k}(x_k)(f_{i_{k+1}}(x_{k+1}) - f_{i_k}(x_k))}{[(x_{k+1} - x_k)^2 + (f_{i_{k+1}}(x_{k+1}) - f_{i_k}(x_k))^2]^{1/2}}. \end{aligned} \tag{6}$$

If we introduce the vectors $\mathbf{x} = (x_1, x_2, ..., x_n)^T$, $\mathbf{v} = (v_1, v_2, ..., v_{n+1})^T$, and define the function $\Phi_{\mathbf{v}}$ by

$$\begin{aligned} \Phi_{\mathbf{v}} : \quad I\!R^n &\longrightarrow I\!R^n \\ \mathbf{x} &\longmapsto \Phi_{\mathbf{v}}(\mathbf{x}) = \begin{pmatrix} \phi_{\mathbf{v}}^1(\mathbf{x}) \\ \vdots \\ \phi_{\mathbf{v}}^n(\mathbf{x}) \end{pmatrix}, \end{aligned}$$

then, solving the two point tracing problem is equivalent to solving the nonlinear system of equations

$$\Phi_{\mathbf{v}}(\mathbf{x}) = 0. \tag{7}$$

2.2 An approach to estimating the transmission coefficients for AIDS and for Tuberculosis using mathematical models (2001) [10]

In this paper, the authors present a mathematical model that describes the interaction between the Human Immunodeficiency Virus (HIV) and Tuberculosis, which is caused by a bacillus of the type *Mycobacterium tuberculosis* (MTB). These infections are considered in a closed environment, like a prision or mental institution.

Using nine working assumptions [10], the dynamics of the model is formulated through a compartment system described by nonlinear ordinary differential equations, which represent the different subpopulations. Therefore, each compartment, in turn, represent one of the stages of the interaction between

Acquired Immunodeficiency Syndrome (AIDS)[3] and tuberculosis (TB). They assume the total population constant.[4]

To clarify the biological process, they introduce the following notation. The state variables are:

- x_1 : the healthy individuals susceptible to both HIV and MTB infections.
- x_2 : the individuals who have been infected with MTB, but have no clinical illness and hence are not infected.
- T_b : individuals with TB disease.
- y_1 : the HIV-positive individuals without MTB infection.
- y_2 : the HIV-positive individuals with MTB infection.
- A : individuals with AIDS but without TB and MTB infections.
- A_{tb} : individuals with AIDS and TB infection.

and the parameters are:

- β : transmission coefficient for HIV infection.
- λ : transmission coefficient for MTB infection.
- ω : the incubation rate for AIDS without MTB infection.
- ξ : the incubation rate for AIDS with MTB infection.
- σ : the reactivation rate of TB disease.
- ρ : is recovery rate of TB.
- μ : is the natural mortality or remaining time in a closed community.
- α : the AIDS mortality rate.
- θ : the TB mortality rate.

The differential equations that govern the process are the following:

$$\frac{dx_1}{dt} = \phi - \beta x_1(y_1 + y_2) - \lambda x_1 T_b - \mu x_1$$

[3] Syndrome characterized by the interaction of the HIV with other infections.
[4] Thus, if one inmate or patient leaves the place, another replaces him.

$$
\begin{aligned}
\frac{dx_2}{dt} &= \rho T_b + \lambda x_1 T_b - \beta x_2(y_1 + y_2) - (\sigma + \mu)x_2 \\
\frac{dT_b}{dt} &= \sigma x_2 - \beta T_b(y_1 + y_2) - (\rho + \mu + \theta)T_b \\
\frac{dy_1}{dt} &= \beta x_1(y_1 + y_2) - \lambda y_1 T_b - (\mu + \omega)y_1 \\
\frac{dy_2}{dt} &= \beta x_2(y_1 + y_2) + \lambda y_1 T_b - (\xi + \mu)y_2 \\
\frac{dA}{dt} &= \omega y_1 - \lambda A A_{tb} - (\mu + \alpha)A \\
\frac{dA_{tb}}{dt} &= \xi y_2 + \lambda A A_{tb} + \beta T_b(y_1 + y_2) - (\mu + \alpha + \theta)A_{tb}
\end{aligned}
\tag{8}
$$

where $\phi = \mu + \theta(T_b + A_{tb}) + \alpha(A + A_{tb})$, accordingly the constant population hypothesis. Therefore, summing up these equations, one gets $\frac{dN}{dt} = 0$, that is, the total population remains constant in all time. Because of the conservation law $x_1(t) + x_2(t) + T_b(t) + y_1(t) + y_2(t) + A(t) + A_{tb}(t) = N(t) = 1$ for any $t \in I\!R$, they can eliminate one of the state variables, by using:

$$
x_1 = 1 - (x_2 + T_b + y_1 + y_2 + A + A_{tb}) = 1 - S. \tag{9}
$$

Hence, the seven dimensional system (8) reduces to the following six dimensional system:

$$
\begin{aligned}
\frac{dx_2}{dt} &= \rho T_b + \lambda(1 - S)T_b - \beta x_2(y_1 + y_2) - (\sigma + \mu)x_2 \\
\frac{dT_b}{dt} &= \sigma x_2 - \beta T_b(y_1 + y_2) - (\rho + \mu + \theta)T_b \\
\frac{dy_1}{dt} &= \beta(1 - S)(y_1 + y_2) - \lambda y_1 T_b - (\mu + \omega)y_1 \\
\frac{dy_2}{dt} &= \beta x_2(y_1 + y_2) + \lambda y_1 T_b - (\xi + \mu)y_2 \\
\frac{dA}{dt} &= \omega y_1 - \lambda A A_{tb} - (\mu + \alpha)A \\
\frac{dA_{tb}}{dt} &= \xi y_2 + \lambda A A_{tb} + \beta T_b(y_1 + y_2) - (\mu + \alpha + \theta)A_{tb}.
\end{aligned}
\tag{10}
$$

Thus the modified model is given by

$$\dot{z} = F(z),$$

where $F : I\!R^6 \rightarrow I\!R^6$ and $z = (x_2, T_b, y_1, y_2, A, A_{tb})^T$.

To begin the analysis of the model given by equations (10) and (9), they examine its disease-free steady state to determine the threshold values for which the diseases will die out and, by biological simplification, they calculate the endemic equilibrium points of the model, that is, vectors z^* such that $F(z^*) = 0$. Using a control technique proposed in their paper, they found seven equilibrium points of the model. Actually, an eight equilibrium point was found but only numerically [10].

Moreover, they present the stability analysis of equilibrium points and observe that the stability conditions for each of these points depend on both: the transmission coefficients for HIV and for MTB (β and λ) which must be estimated. The other parameters haven been evaluated from the literature.

3 Numerical tests

For the numerical tests with the problems and with the quasi-Newton Methods chosen: Broyden and ICUM we will call the four problems:

P1: Two-point ray tracing problem.

P2: AIDS-Tuberculosis problem.

P3: Target location.

P4: Basin Problem.

The codes of the algorithms, functions and Jacobians of each problem was written in MATLAB 6.0. These experiments were run using a computer AMD Athlon-800MHz.

The initial points used were the same suggested by the authors of the applications. In all tests we used the convergence criterion $\|F(x^k)\| \leq 10^{-6}$, excepting in the **P3** problem, in which we use a tolerance equal to 10^{-4} as it was suggested in [11].

We also stopped the iterations when the number of iterations exceeded 300 or when $\|F(x^k)\| \geq 10^5$. In the last case we will say that the method used diverges.

3.1 P1: Two-point ray tracing problem

In order to do the implementation of the Two-point ray tracing problem we consider a particular case:

$$n = 13, \qquad \mathbf{X}_0 = \begin{pmatrix} x_0 \\ f_{i_0}(x_0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \mathbf{X}_{14} = \begin{pmatrix} x_{14} \\ f_{i_{14}}(x_{14}) \end{pmatrix} = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}.$$

The values for the parameter α were $800, 840, 880, \cdots 1600$, each one of them corresponding to increments in the receptor, in Geophysics terms.

A finite sequence, $\{f_{i_1}, f_{i_2}, \cdots, f_{i_{13}}\}$, of the interfaces intersected by the ray path is given by

$$\{f_{i_1}, f_{i_2}, \cdots, f_{i_{13}}\} = \{f_1, f_6, f_7, f_2, f_3, f_4, f_5, f_4, f_3, f_2, f_7, f_6, f_1\}, \tag{11}$$

where, each function $f_i : I\!R \to I\!R,\ i = 1, 2, \cdots, 7,$ is defined, respectively by

$$\begin{aligned} f_1(x) &= 900. \\ f_2(x) &= 8000 - \sqrt{6000^2 + (x - 1000)^2}. \\ f_3(x) &= 8500 - \sqrt{5000^2 + (x - 2500)^2}. \\ f_4(x) &= 5000 + 10^{-6}(x - 1000)^2. \\ f_5(x) &= 8000 - (0.3333)x. \\ f_6(x) &= 1400. \\ f_7(x) &= 1900. \end{aligned} \tag{12}$$

The sequence of velocities, $\{v_k\}$, $k = 1, \cdots, 14$ is represented by a vector:

$$\mathbf{v} = 10^3(1, 2, 3, 5, 6, 8, 10, 10, 8, 6, 5, 3, 2, 1)^T.$$

Thus, we want to find a vector $\mathbf{x}^* = (x_1^*, x_2^*, \cdots, x_{13}^*)^T$ such as $\Phi_{\mathbf{v}}(\mathbf{x}^*) = 0$, where, $\Phi : I\!R^{13} \longrightarrow I\!R^{13}$ is defined using (6), (11) and (12).

The author of [14], gives us the values of α and the initial point[5] x_0 :

$$\begin{aligned} x_0 \;=\; &(50.852,\ 107.626,\ 193.482,\ 237.195,\ 811.580,\ 1197.912,\ 2134.884, \\ &1540.541,\ 1262.964,\ 901.701,\ 883.199,\ 828.718,\ 792.517)^T; \end{aligned}$$

actually, this point is the solution to an nonlinear system of equations and thus it represents a ray. Therefore, for the problem that we are considering it is a good initial point.

Starting with this x_0, we want to solve a nonlinear system for each one of the values of parameter α, using Broyden's method and ICUM , respectively.

[5]This point was found in [14] using continuation methods.

In **Table 1,** for each value of α, we show the number of iteration performed by each quasi-method above mentioned. The solution vector found for each value of α is shown like a column in **Table 2.**

We observe that the solutions found here are exactly the same as the ones found in [14], where they used the continuation method to solve the problems.

α	*Broyden*	*ICUM*
800	5	4
840	6	5
880	6	5
920	6	5
960	7	5
1000	7	6
1040	7	6
1080	7	7
1120	7	6
1160	8	7
1200	8	7
1240	9	7
1280	9	8
1320	9	8
1360	10	8
1400	10	9
1440	11	9
1480	11	9
1520	11	10
1560	12	10
1600	12	10

Table 1: *Iterations Number for* **P1.**

α	800	840	880	920	960	1000	1040
x_1^*	51.115	51.380	51.645	51.910	52.175	52.440	52.706
x_2^*	108.187	108.750	109.314	109.878	110.442	111.007	111.573
x_3^*	194.501	195.523	196.545	197.569	198.594	199.621	200.648
x_4^*	238.415	239.636	240.859	242.083	243.308	244.535	245.763
x_5^*	815.969	820.365	824.768	829.178	833.594	838.017	842.446
x_6^*	1206.559	1215.235	1223.939	1232.671	1241.432	1250.221	1259.037
x_7^*	2151.026	2167.197	2183.397	2199.626	2215.884	2232.169	2248.482
x_8^*	1566.899	1593.250	1619.595	1645.932	1672.262	1698.583	1724.895
x_9^*	1291.900	1320.912	1349.998	1379.155	1408.382	1437.677	1467.039
x_{10}^*	938.430	975.197	1012.000	1048.835	1085.699	1122.591	1159.508
x_{11}^*	920.440	957.676	994.907	1032.133	1069.353	1106.568	1143.779
x_{12}^*	867.185	905.649	944.109	982.566	1021.019	1059.470	1097.917
x_{13}^*	831.793	871.067	910.340	949.611	988.880	1028.148	1067.414

α	1080	1120	1160	1200	1240	1280	1320
x_1^*	52.972	53.239	53.505	53.772	54.038	54.305	54.572
x_2^*	112.139	112.706	113.273	113.841	114.409	114.977	115.546
x_3^*	201.677	202.706	203.737	204.769	205.801	206.834	207.868
x_4^*	246.992	248.222	249.453	250.685	251.918	253.152	254.387
x_5^*	846.881	851.321	855.767	860.219	864.676	869.138	873.605
x_6^*	1267.881	1276.752	1285.650	1294.575	1303.527	1312.505	1321.510
x_7^*	2264.823	2281.191	2297.586	2314.008	2330.455	2346.929	2363.428
x_8^*	1751.198	1777.491	1803.773	1830.045	1856.305	1882.553	1908.788
x_9^*	1496.464	1525.952	1555.500	1585.106	1614.769	1644.486	1674.255
x_{10}^*	1196.446	1233.404	1270.377	1307.364	1344.362	1381.368	1418.379
x_{11}^*	1180.984	1218.184	1255.380	1292.571	1329.757	1366.939	1404.117
x_{12}^*	1136.361	1174.803	1213.241	1251.676	1290.109	1328.538	1366.965
x_{13}^*	1106.679	1145.942	1185.203	1224.464	1263.722	1302.980	1342.236

α	1360	1400	1440	1480	1520	1560	1600
x_1^*	54.839	55.107	55.374	55.641	55.909	56.176	56.444
x_2^*	116.115	116.684	117.254	117.823	118.393	118.963	119.533
x_3^*	208.903	209.939	210.975	212.011	213.048	214.086	215.123
x_4^*	255.622	256.858	258.095	259.332	260.569	261.807	263.045
x_5^*	878.077	882.553	887.034	891.519	896.008	900.501	904.998
x_6^*	1330.540	1339.596	1348.678	1357.785	1366.917	1376.074	1385.256
x_7^*	2379.952	2396.502	2413.076	2429.674	2446.297	2462.943	2479.612
x_8^*	1935.010	1961.219	1987.414	2013.594	2039.758	2065.908	2092.041
x_9^*	1704.075	1733.943	1763.857	1793.816	1823.816	1853.856	1883.934
x_{10}^*	1455.393	1492.406	1529.416	1566.420	1603.414	1640.398	1677.366
x_{11}^*	1441.290	1478.459	1515.625	1552.786	1589.944	1627.097	1664.248
x_{12}^*	1405.390	1443.812	1482.231	1520.648	1559.062	1597.475	1635.885
x_{13}^*	1381.491	1420.744	1459.996	1499.247	1538.497	1577.745	1616.993

Table 2: *The solution vector found for each α.*

3.2 P2: AIDS-Tuberculosis problem

In this section, our propose is determining numerically the equilibrium points of the model (9)-(10), that is, to solve the nonlinear system $F(z) = 0$, using Broyden's method and ICUM. Moreover, we want to determine if these points are stable or not.[6] For this, we implemented these methods and ran them considering, like in [10], the variation of the parameters λ and β and the other constants fixed: $\rho = 0.5$, $\sigma = 0.05$, $\mu = 0.1$, $\theta = 0.05$, $\omega = 0.1$, $\xi = 0.2$, $\alpha = 0.33$.

We variated the parameters λ e β using different values presented in **Figure 1** of [10]. That figure shows attraction regions of the equilibrium points in the space of parameters λ e β. We include here a short biological description of these regions. For a more general information, see [10].

[6]That this, if the real part of the eigenvalues of the Jacobian matrix in this equilibrium points, is negative or not.

R_ϕ: region where both infections die out in the community.
R_1: the HIV infection progress to AIDS disease.
R_3: the HIV infection progress to AIDS with TB disease.
R_4: the MTB infection progress to TB disease.
R_7: HIV and MTB progress to AIDS and TB disease, respectively

We also consider the fact mentioned in [10], that the model does not depend on the initial conditions. Thus the only starting point used is $z_0 = (1, 1, 1, 1, 1, 1)^T$.

We present the results in **Tables 3-7.** The first column of each table, shows the values of λ and β used in each region mentioned above; the second column shows the corresponding equilibrium point to the λ and β parameters.[7] The columns, *Broyden* and *ICUM,* show respectively, the number of iterations performed by each method until an equilibrium point is found and finally, the last column, $\rho_{\mathbf{max}}$, shows the real part of the greatest eigenvalue of the Jacobian matrix of F in z^*. This information helps us in the analysis of stability of the solutions.

In **Table 3,** we can observe that, for any values of λ and β in the region R_ϕ, both methods converge to the trivial equilibrium point z_*, which is a stable equilibrium point. This results corresponds to the theorical analysis in [10].

$(\lambda ; \beta)$	z_*^T	Broyden	ICUM	$\rho_{\mathbf{max}}$
(0.5 ; 0.1)	$(0,0,0,0,0,0)$	10	9	-0.0646
(0.25 ; 0.05)	$(0,0,0,0,0,0)$	8	7	-0.0838
(0.7 ; 0.15)	$(0,0,0,0,0,0)$	11	10	-0.0500
(0.9 ; 0.01)	$(0,0,0,0,0,0)$	10	9	-0.0360
(1 ; 0.19)	$(0,0,0,0,0,0)$	17	15	-0.0100
(1 ; 0.01)	$(0,0,0,0,0,0)$	11	11	-0.0292

Table 3: region R_ϕ.

[7] That is, the solution to the nonlinear system of equation.

In **Table 4,** we can observe an interesting situation in region R_1 : for the values $\lambda = 5$ and $\beta = 0.39$: the methods converge to different equilibrium points. Aother thing is that Broyden's method finds a stable equilibrium point and ICUM finds an unstable one. Observe also that Broyden's method performs twice the number of iterations performed by ICUM.

$(\lambda;\beta)$	$z_* = (0,0,y_1,0,A,0)^T$	Broyden	ICUM	$\rho_{\max}$
(1 ; 0.25)	$(0,0,0.16,0,0.04,0)$	13	14	-0.0513
(2 ; 0.4)	$(0,0,0.41,0,0.09,0)$	15	15	-0.1915
(2.8 ; 0.5)	$(0,0,0.49,0,0.11,0)$	16	18	-0.1630
(3 ; 0.6)	$(0,0,0.54,0,0.13,0)$	15	8	-0.1026
(4 ; 0.45)	$(0,0,0.45,0,0.10,0)$	31	20	-0.0607
(5 ; 0.39)	$(0,0,0.40,0,0.19,0)$	64		-0.0204
	$(0,0,0.39,0,0.10,0)$		32	0.0204

Table 4: region R_1.

Table 5, shows that the particular situation described in the previous paragraph is more common in region R_3 with the difference: in five of the six cases tested, ICUM finds a stable equilibrium point of form $z_* = (0,0,y_1,0,A,A_{\tau b})^T$. According to the theory in [10] only stable equilibrium points of the form of $z_* = (0,0,y_1,0,A,A_{\tau b})^T$ must appear in the region R_3.

$(\lambda\,;\,\beta)$	$z_* = (0,0,y_1,0,A,A_{\tau b})^T$	Broyden	ICUM	$\rho_{\mathbf{max}}$
(7 ; 0.58)	$(0,0,0.54,0,0.07,0.05)$	27	39	-0.2548
(6.4 ; 0.44)	$(0.71,0.05,0,0,0.08,0.08)$	32		0.4543
	$(0,0,0.44,0,0.08,0.03)$		27	-0.1131
(5.5 ; 0.51)	$(0,0,0.49,0,0.11,0)$	25		0.1508
	$(0,0,0.50,0,0.09,0.03)$		45	-0.1508
(8.5 ; 0.42)	$(0.77,0.06,0,0,0,0\ )$	29		0.1040
	$(0,0,0.43,0,0.06,0.04)$		53	-0.0416
(7.7 ; 0.44)	$(0,0,0\ ,0,0.06,\text{-}0.06)$	229		0.4543
	$(0,0,0.45,0,0.06,0.04)$		38	-0.0839
(5 ; 0.49)	$(0,0,0.48,0,0.11,0)$	23		0.0783
	$(0,0,0.48,0,0.10,0.01)$		29	-0.0787

Table 5: region R_3.

Table 6, shows that both methods converge to the same equilibrium point which is stable in the first four cases and unstable in the other two cases. The performance of ICUM is better than Broyden's method.

$(\lambda\,;\,\beta)$	$z_* = (x_2,\tau_b,0,0,0,0)^T$	Broyden	ICUM	$\rho_{\mathbf{max}}$
(2 ; 0.01)	$(0.26,0.02,0,0,0,0)$	12	11	-0.0342
(3 ; 0.1)	$(0.48,0.04,0,0,0,0)$	13	10	-0.0950
(4 ; 0.2)	$(0.59,0.05,0,0,0,0)$	16	13	-0.0847
(5 ; 0.22)	$(0.58,0.04,0,0,0,0)$	15	13	-0.0743
(7 ; 0.25)	$(0.66,0.05,0,0,0,0)$	23	16	0.4543
(8 ; 0.19)	$(0.73,0.06,0,0,0,0)$	25	17	0.4543

Table 6: region R_4.

Finally, in **Table 7,** we can observe that, in all the cases tested, both methods converge to a stable equilibrium point which has all its components positive. For these tests, the performance of ICUM is again better than that of Broyden's method.

$(\lambda\,;\,\beta)$	$z_* = (x_2, \tau_b, y_1, y_2, A, A_{\tau b})^T$	Broyden	ICUM	$\rho_{\mathbf{max}}$
$(5\,;\,0.32)$	$(0.42, 0.03, 0.04, 0.07, 0.01, 0.03)$	28	21	-0.0118
$(4.5\,;\,0.3)$	$(0.54, 0.04, 0.01, 0.03, 0.02, 0.01)$	25	26	-0.0068
$(6\,;\,0.35)$	$(0.27, 0.02, 0.10, 0.10, 0.01, 0.05)$	26	20	-0.0072
$(7\,;\,0.36)$	$(0.39, 0.03, 0.06, 0.11, 0.01, 0.06)$	19	16	-0.0222
$(8\,;\,0.37)$	$(0.43, 0.03, 0.04, 0.12, 0.01, 0.06)$	19	15	-0.0311
$(7\,;\,0.33)$	$(0.60, 0.04, 0.01, 0.06, 0.01, 0.03)$	25	19	-0.0193

Table 7: region R_7.

3.3 P3: Target location problem

We consider the nonlinear system

$$F_n(x_t\,,\,y_t\,,\,z_t) = z_t + \sqrt{(x_t - x_n)^2 + (y_t - y_n)^2 + z_t^2} - r_n = 0.$$

where x_t, y_t and z_t are the unknown target co-ordinates, (x_n, y_n) are the known receiving elements position, located in a plane and r_n is the nth round trip distance of the transmitted pulse. As before, our interest now is solving numerically this system using Broyden's method and ICUM. Its solution gives the position of the target [11].

We consider $N = 5$ and $N = 6$ receiving elements located in a plane. The minimum number of receiving elements needed to calculate the unknown are three, which gives, respectively, ten and twenty possible combinations for an five and six elements planar array.

For each test, we used two initial points: one of them is a point $(0, 0, z)^T$, where z is an approximation[8] to $r_n/2$. The other initial point is an approximation to the solution found with the first initial point. The reason of this choice is that we want to have an idea of the performance of both quasi-Newton methods using a good approximation to the solution, like it was considered in [11]. The authors of [11] used the beam-forming technique to obtain good starting points for their hybrid algorithm.

For each one of ten possible combinations of three elements (case $N = 5$) and twenty possible combinations of three elements (case $N = 6$) we ran the Broyden and ICUM algorithms using the two initial points mentioned previously and $\|F(x)\|_\infty \geq 10^5$.

The **Tables 8** and **10**, show the information about receiving elements position and the values of r_n for $n = 1, 2, \cdots, 5$. Similarly, the **Tables 12** and **14**, show the same information,but in the case $n = 1, 2, \cdots, 6$.

The results obtained are shown in **Tables 9** and **11** for $N = 5$, and in **Tables 13** and **15** for $N = 6$. In the last tables, the notation NC means that the algorithm did not converge, because it reached the maximun number of iterations allowed in the algorithm.

In the case $N = 5$ we observe that, for initial points $x_0 = (0, 0, z)^T$, $z \approx r_n/2$, the performance of Broyden's method is slightly better than ICUM. When a initial approximation is close to the solution, the performance of both method is the same.

In the case $N = 6$, the results show that for initial points $x_0 = (0, 0, z)^T$, $z \approx r_n/2$, the performance of the Broyden's method is better than that of ICUM in most of cases.

[8] Like it was suggested by the authors in [11]

n	1	2	3	4	5
(x_n, y_n)	$(0,0)$	$(2,0)$	$(3,1)$	$(1,1)$	$(0,2)$
r_n	18.85	18.22	18.17	18.54	19.05

Table 8: *Receiving elements and r_n values for $n = 5$.*

Combinations	$x_0 = (0,0,9)$		$x_0 = (3.5,0,8.9)$	
	Broyden	*ICUM*	*Broyden*	*ICUM*
x_1 x_2 x_3	5	6	3	3
x_1 x_2 x_4	5	6	3	3
x_1 x_2 x_5	5	6	3	3
x_1 x_3 x_4	5	5	3	3
x_1 x_3 x_5	5	6	3	3
x_1 x_4 x_5	5	6	3	3
x_2 x_3 x_4	5	5	3	3
x_2 x_3 x_5	15	17	3	5
x_2 x_4 x_5	5	5	3	3
x_3 x_4 x_5	5	5	3	3

Table 9: *Number of iterations performed by the methods.* $x_T = (4,\ 0,\ 9)$

n	1	2	3	4	5
(x_n, y_n)	$(0,0)$	$(4,6)$	$(0,4)$	$(3,0)$	$(1,1)$
r_n	13.48	12.63	12.32	13.28	12.78

Table 10: *Receiving elements and r_n values for $n = 5$.*

Combinations	$x_0 = (0, 0, 6)$		$x_0 = (1, 3, 6)$	
	Broyden	*ICUM*	*Broyden*	*ICUM*
$x_1\ x_2\ x_3$	7	8	4	5
$x_1\ x_2\ x_4$	6	7	4	4
$x_1\ x_2\ x_5$	16	10	5	5
$x_1\ x_3\ x_4$	6	6	4	4
$x_1\ x_3\ x_5$	7	7	4	4
$x_1\ x_4\ x_5$	6	7	4	4
$x_2\ x_3\ x_4$	7	8	4	4
$x_2\ x_3\ x_5$	7	8	4	4
$x_2\ x_4\ x_5$	6	8	4	4
$x_3\ x_4\ x_5$	6	7	4	5

Table 11: *Number of iterations performed by the methods.* $x_T = (2,\ 4,\ 6)$

Table 13 shows that there are five combinations of three elements for which the initial point, $x_0 = (0, 0, 3)$, is already the solution, and this solution is different from the solution found by the other combinations using the same initial point, $x_T = (3,\ 3,\ 3)$. When an initial approximation is close to the solution, the performance of both method is similar again.

n	1	2	3	4	5	6
(x_n, y_n)	$(0, 3)$	$(3, 0)$	$(1, 2)$	$(1, 1)$	$(2, 1)$	$(2, 2)$
r_n	7.24	7.24	6.74	7.12	6.74	6.32

Table 12: *Receiving elements and* r_n *values for* $n = 6$.

Combinations	$x_0 = (0, 0, 3)$		$x_0 = (2.4, 2.7, 3.4)$	
	Broyden	*ICUM*	*Broyden*	*ICUM*
$x_1\ x_2\ x_3$	0	0	4	6
$x_1\ x_2\ x_4$	13	10	4	4
$x_1\ x_2\ x_5$	0	0	4	6
$x_1\ x_2\ x_6$	7	NC	4	5
$x_1\ x_3\ x_4$	8	12	3	5
$x_1\ x_3\ x_5$	0	0	4	6
$x_1\ x_3\ x_6$	11	NC	4	5
$x_1\ x_4\ x_5$	8	10	4	5
$x_1\ x_4\ x_6$	8	10	4	5
$x_1\ x_5\ x_6$	8	10	4	5
$x_2\ x_3\ x_4$	8	10	4	5
$x_2\ x_3\ x_5$	0	0	4	6
$x_2\ x_3\ x_6$	8	10	4	5
$x_2\ x_4\ x_5$	7	10	4	5
$x_2\ x_4\ x_6$	8	10	4	5
$x_2\ x_5\ x_6$	11	11	4	4
$x_3\ x_4\ x_5$	7	10	4	5
$x_3\ x_4\ x_6$	8	10	4	5
$x_3\ x_5\ x_6$	9	10	4	5
$x_4\ x_5\ x_6$	8	10	4	5

Table 13: *Number of iterations performed by methods.* $x_T = (3,\ 3,\ 3)$.

Table 15 shows that there are one combination of three elements for which both methods does not converge for any initial points used. For the other combinations the performance of the methods is similar to that described in **Table 12.**

n	1	2	3	4	5	6
(x_n, y_n)	$(0,0)$	$(0,5)$	$(10,0)$	$(2,15)$	$(5,5)$	$(20,10)$
r_n	25	24.14	21.18	26.25	21.18	25

Table 14: *Receiving elements and r_n values for $n = 6$.*

Combinations	$x_0 = (0,0,10)$		$x_0 = (9,3.5,10)$	
	Broyden	*ICUM*	*Broyden*	*ICUM*
$x_1\ x_2\ x_3$	8	8	3	4
$x_1\ x_2\ x_4$	NC	NC	NC	NC
$x_1\ x_2\ x_5$	8	11	3	4
$x_1\ x_2\ x_6$	8	8	4	4
$x_1\ x_3\ x_4$	8	9	6	6
$x_1\ x_3\ x_5$	7	8	3	4
$x_1\ x_3\ x_6$	9	9	4	5
$x_1\ x_4\ x_5$	8	8	8	8
$x_1\ x_4\ x_6$	7	8	5	5
$x_1\ x_5\ x_6$	11	10	4	5
$x_2\ x_3\ x_4$	9	11	7	8
$x_2\ x_3\ x_5$	8	10	6	8
$x_2\ x_3\ x_6$	8	10	4	4
$x_2\ x_4\ x_5$	9	14	7	9
$x_2\ x_4\ x_6$	8	9	7	7
$x_2\ x_5\ x_6$	8	8	4	4
$x_3\ x_4\ x_5$	NC	NC	5	5
$x_3\ x_4\ x_6$	7	9	5	5
$x_3\ x_5\ x_6$	8	11	4	4
$x_4\ x_5\ x_6$	9	15	6	8

Table 15: *Number of iterations performed by methods.* $x_T = (10,\ 5,\ 10)$

3.4 P4: Basin problem

Here, we are interested in using quasi-Newton methods to find $p-$period points of the Hénon map, $F : I\!R^2 \to I\!R^2$, defined by

$$F\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2.12 - x^2 - 0.3y \\ x \end{pmatrix}.$$

A $p-$period point is a point x_* such that $F^p(x_*) = x_*$. Quasi-Newton methods can be used to find the periodic p point of F by letting $G = F^p - \mathbf{I}$, where $\mathbf{I}$ is the identity mapping, and solving the system $G(x) = 0$.

Following [13], we wish to find all the isolated root of G (all periodic p points of F for a fixed p). For this, we search for all roots of G in a specific bounded region (in our tests, the region is $[0,1] \times [0,1]$), by choosing a large number of initial points x_0, randomly generated.

We consider the cases $p = 3$, $p = 4$ and $p = 5$. For each value of p we generated sixty random initial points in $[0,1] \times [0,1]$ and for each one of them we ran Broyden's and ICUM algorithms.

The results are presented as follows: **Table 16** shows, for each value of p used in the tests, the periodic p points of F found by the Broyden's and ICUM algorithms, and the information about the nonsingularity of the Jacobian matrix of F in these points. This is an important condition used in [13] to conclude that an estimate of basin[9] of x^* in a region with area 1 is an estimate of the probability of selecting an initial point x_0 whose sequence generated by Newton's method converges to x^*.

After each **Table 17** to **19**, we present, for each value of p, the number of initial points in the basin[10] of each x^*. For this, we used the results obtained with ICUM because it converges in all the cases tested. We call this basin as **ICUM-basin,** in analogy to [13].

[9] Initial points set such as a sequence generate by the Newton-type method converge to x^*.

[10] The elements of this set appear in the first column of each table.

p	x^* : periodic p point of F	$\det(J(x^*;p))$
3	$x_1^* = (0.8529; -0.8777)^T$	-18.9937
	$x_2^* = (0.9445; \quad 0.9445)^T$	-0.1068
4	$x_2^* = (0.9445; \quad 0.9445)^T$	1.8988
	$x_3^* = (0.3391; \quad 1.3900)^T$	6.8635
	$x_4^* = (0.7955; -1.3550)^T$	-128.1806
5	$x_2^* = (0.9445; \quad 0.9445)^T$	1.3431
	$x_5^* = (0.9942; -0.7832)^T$	69.8901
	$x_6^* = (1.0551; -0.7512)^T$	46.7556
	$x_7^* = (0.2852; \quad 1.2322)^T$	-5.4186
	$x_8^* = (-0.0458; 1.3666)^T$	10.3533

Table 16: *Solution vectors found for each* $p = 3, 4, 5$.

¿From the results of **Table 16,** we can also see that $x_2^* = (0.9445; 0.9445)^T$ is simultaneously **two, three** and **five-period point** of F.

Tables 17 to **19** show, for each value of p, the number of iterations performed by Broyden's method (k_B) and ICUM (k_I) in case of convergence. The symbol NC means that the algorithm stopped because it reached the maximun number of iterations permitted in the algorithm. Moreover, they have an aditional column that indicates the point of convergence. This is done in order to determinate the basin of each $p-$period point found.

The results permit us to conclude that for each $p\,(3, 4, 5)$ and each one of the sixty random initial points, only ICUM converge in all the cases. In the case where both methods converge the performance of ICUM is better than Broyden's one.

x_0	k_B	k_I	x_1^*
$(0.8364; 0.1453)^T$	8	7	x_1^*
$(0.8240; 0.1340)^T$	8	7	x_1^*
$(0.9636; 0.1205)^T$	53	11	x_1^*
$(0.4128; 0.4014)^T$	9	9	x_1^*
$(0.4210; 0.3770)^T$	9	9	x_1^*
$(0.4768; 0.4688)^T$	8	8	x_1^*
$(0.7486; 0.3741)^T$	NC	12	x_1^*
$(0.4542; 0.0386)^T$	9	8	x_1^*
$(0.5624; 0.3723)^T$	7	8	x_1^*
$(0.7928; 0.7952)^T$	11	9	x_1^*
$(0.3829; 0.2528)^T$	9	9	x_1^*
$(0.3429; 0.9678)^T$	15	11	x_1^*
$(0.4798; 0.3683)^T$	9	8	x_1^*
$(0.7646; 0.3771)^T$	8	8	x_1^*
$(0.9003; 0.1834)^T$	NC	12	x_1^*
$(0.3683; 0.9175)^T$	12	10	x_1^*
$(0.5159; 0.0903)^T$	8	7	x_1^*
$(0.7353; 0.0047)^T$	8	6	x_1^*
$(0.6031; 0.9569)^T$	8	7	x_1^*
$(0.3974; 0.7316)^T$	10	9	x_1^*
$(0.6846; 0.9785)^T$	10	8	x_1^*
$(0.5147; 0.6363)^T$	8	8	x_1^*
$(0.4010; 0.4866)^T$	9	9	x_1^*
$(0.7505; 0.1262)^T$	7	7	x_1^*
$(0.6933; 0.9358)^T$	10	7	x_1^*
$(0.4776; 0.1291)^T$	9	8	x_1^*
$(0.7729; 0.2973)^T$	8	7	x_1^*
$(0.8437; 0.8815)^T$	15	17	x_1^*
$(0.7000; 0.7557)^T$	9	7	x_1^*
$(0.8293; 0.9706)^T$	16	11	x_1^*

x_0	k_B	k_I	x^*
$(0.9745; 0.4022)^T$	11	7	x_2^*
$(0.8995; 0.1707)^T$	NC	11	x_1^*
$(0.9523; 0.4577)^T$	11	8	x_2^*
$(0.5369; 0.0665)^T$	8	7	x_1^*
$(0.4939; 0.4175)^T$	8	8	x_1^*
$(0.6854; 0.9671)^T$	10	8	x_1^*
$(0.7538; 0.0968)^T$	7	7	x_1^*
$(0.7649; 0.6579)^T$	9	8	x_1^*
$(0.8104; 0.3742)^T$	9	8	x_1^*
$(0.4928; 0.0835)^T$	9	8	x_1^*
$(0.7067; 0.1684)^T$	9	6	x_1^*
$(0.8137; 0.4662)^T$	10	8	x_1^*
$(0.7223; 0.9949)^T$	11	7	x_1^*
$(0.3625; 0.7308)^T$	10	10	x_1^*
$(0.6497; 0.6813)^T$	8	7	x_1^*
$(0.9452; 0.6133)^T$	10	8	x_2^*
$(0.7829; 0.0032)^T$	6	7	x_1^*
$(0.7970; 0.6418)^T$	10	9	x_1^*
$(0.4161; 0.1864)^T$	9	9	x_1^*
$(0.3100; 0.9441)^T$	25	20	x_2^*
$(0.9807; 0.5551)^T$	10	9	x_2^*
$(0.9885; 0.6916)^T$	10	16	x_1^*
$(0.4407; 0.0062)^T$	9	8	x_1^*
$(0.6868; 0.2972)^T$	11	7	x_1^*
$(0.6472; 0.4638)^T$	8	7	x_1^*
$(0.3001; 0.9981)^T$	18	13	x_2^*
$(0.6602; 0.3323)^T$	7	7	x_1^*
$(0.9073; 0.6702)^T$	9	9	x_2^*
$(0.9543; 0.8814)^T$	9	9	x_2^*
$(0.4110; 0.4248)^T$	9	9	x_1^*

Tables 17: Case $p = 3$.

Observe that for $p = 3$, the **ICUM-basin** of x_1^* has 52 points, while that in ICUM-basin of x_2^* has 8 points.

x_0	k_B	k_I	x_3^*
$(0.8000; 0.2894)^T$	11	8	x_3^*
$(0.6951; 0.2593)^T$	13	9	x_3^*
$(0.7132; 0.7204)^T$	9	8	x_3^*
$(0.7333; 0.6223)^T$	10	8	x_3^*
$(0.5068; 0.8841)^T$	9	9	x_3^*
$(0.9519; 0.1690)^T$	11	9	x_3^*
$(0.8267; 0.6114)^T$	10	12	x_3^*
$(0.8473; 0.1141)^T$	12	8	x_3^*
$(0.6492; 0.1148)^T$	15	10	x_3^*
$(0.5752; 0.4081)^T$	17	11	x_3^*
$(0.7133; 0.8674)^T$	10	8	x_3^*
$(0.9033; 0.0203)^T$	11	9	x_3^*
$(0.7418; 0.9948)^T$	NC	10	x_3^*
$(0.8667; 0.4858)^T$	12	8	x_3^*
$(0.6368; 0.9441)^T$	10	8	x_3^*
$(0.6773; 0.5862)^T$	10	8	x_3^*
$(0.4879; 0.8915)^T$	9	7	x_3^*
$(0.7623; 0.6553)^T$	9	9	x_3^*
$(0.3645; 0.9757)^T$	10	9	x_3^*
$(0.9016; 0.3242)^T$	11	8	x_3^*
$(0.7379; 0.1118)^T$	14	9	x_3^*
$(0.2457; 0.8976)^T$	NC	32	x_3^*
$(0.7666; 0.0454)^T$	12	8	x_3^*
$(0.5093; 0.6248)^T$	10	9	x_3^*
$(0.6255; 0.9912)^T$	8	8	x_3^*
$(0.6781; 0.5088)^T$	12	8	x_3^*
$(0.8150; 0.6896)^T$	29	14	x_3^*
$(0.6368; 0.7691)^T$	10	7	x_3^*
$(0.8460; 0.1724)^T$	11	8	x_3^*
$(0.5874; 0.9242)^T$	11	7	x_3^*

x_0	k_B	k_I	x^*
$(0.7169; 0.6433)^T$	10	8	x_3^*
$(0.7826; 0.4665)^T$	11	8	x_3^*
$(0.2323; 0.3179)^T$	NC	31	x_3^*
$(0.7888; 0.6344)^T$	10	9	x_3^*
$(0.6598; 0.5376)^T$	10	8	x_3^*
$(0.1360; 0.7552)^T$	25	24	x_3^*
$(0.2033; 0.8193)^T$	NC	14	x_4^*
$(0.2923; 0.0913)^T$	15	18	x_4^*
$(0.6156; 0.0464)^T$	14	22	x_4^*
$(0.5660; 0.2553)^T$	NC	25	x_4^*
$(0.7067; 0.1684)^T$	NC	34	x_4^*
$(0.0650; 0.8792)^T$	NC	26	x_4^*
$(0.5398; 0.9233)^T$	NC	15	x_2^*
$(0.3087; 0.5582)^T$	19	14	x_4^*
$(0.3618; 0.2314)^T$	12	15	x_4^*
$(0.6787; 0.9798)^T$	10	8	x_3^*
$(0.4497; 0.6431)^T$	11	15	x_3^*
$(0.3965; 0.4807)^T$	14	14	x_4^*
$(0.8173; 0.2346)^T$	11	9	x_3^*
$(0.9441; 0.9121^T$	11	7	x_2^*
$(0.9636; 0.0201)^T$	11	9	x_3^*
$(0.3838; 0.8624)^T$	NC	14	x_2^*
$(0.4443; 0.4232)^T$	NC	21	x_2^*
$(0.9288; 0.4851)^T$	18	9	x_2^*
$(0.8506; 0.7131)^T$	17	9	x_2^*
$(0.2452; 0.4680)^T$	NC	21	x_3^*
$(0.9962; 0.6141)^T$	9	11	x_2^*
$(0.9192; 0.7805)^T$	NC	8	x_2^*
$(0.4377; 0.5918)^T$	NC	18	x_3^*
$(0.6219; 0.8946)^T$	9	8	x_3^*

Tables 18: Case $p = 4$.

In this case, the **ICUM-basin** of x_2^*, x_3^* and x_4^* have, respectively, 8, 43 and 9 points.

x_0	k_B	k_I	x_6^*
$(0.8447; 0.3678)^T$	10	6	x_6^*
$(0.6552; 0.8376)^T$	16	8	x_6^*
$(0.8580; 0.3358)^T$	13	7	x_6^*
$(0.9827; 0.8066)^T$	NC	34	x_6^*
$(0.5668; 0.8230)^T$	16	9	x_6^*
$(0.7505; 0.7400)^T$	9	7	x_6^*
$(0.7176; 0.6927)^T$	11	9	x_6^*
$(0.6992; 0.7275)^T$	14	8	x_6^*
$(0.7159; 0.8928)^T$	11	8	x_6^*
$(0.8656; 0.2324)^T$	16	9	x_6^*
$(0.2319; 0.2393))^T$	NC	15	x_6^*
$(0.9016; 0.0056)^T$	17	8	x_6^*
$(0.6252; 0.7334)^T$	20	14	x_6^*
$(0.8214; 0.4447)^T$	13	8	x_6^*
$(0.6154; 0.7919)^T$	14	9	x_6^*
$(0.8318; 0.5028)^T$	28	10	x_6^*
$(0.3784; 0.8600)^T$	11	16	x_6^*
$(0.8216; 0.6449)^T$	NC	9	x_6^*
$(0.8180; 0.6602)^T$	NC	16	x_6^*
$(0.8385; 0.5681)^T$	39	9	x_6^*
$(0.8939; 0.1991)^T$	22	9	x_6^*
$(0.5828; 0.4235)^T$	NC	27	x_6^*
$(0.5798; 0.7604)^T$	9	9	x_6^*
$(0.7942; 0.0592^T$	14	7	x_6^*
$(0.8392; 0.6288)^T$	NC	9	x_6^*
$(0.8214; 0.4447)^T$	13	9	x_6^*
$(0.6154; 0.7919)^T$	14	9	x_6^*
$(0.8318; 0.5028)^T$	28	10	x_6^*
$(0.8939; 0.1991)^T$	22	9	x_6^*
$(0.8385; 0.5681)^T$	39	9	x_6^*

x_0	k_B	k_I	x^*
$(0.7680; 0.9708)^T$	21	16	x_2^*
$(0.9901; 0.7889)^T$	22	23	x_2^*
$(0.9669; 0.6649)^T$	14	9	x_2^*
$(0.0841; 0.4544)^T$	NC	29	x_2^*
$(0.8049; 0.9084)^T$	15	10	x_2^*
$(0.2974; 0.0492)^T$	NC	34	x_2^*
$(0.9517; 0.6400)^T$	18	9	x_2^*
$(0.8699; 0.7694)^T$	13	9	x_2^*
$(0.8295; 0.9561)^T$	13	10	x_2^*
$(0.8983; 0.7546)^T$	12	12	x_2^*
$(0.7939; 0.9200)^T$	16	10	x_2^*
$(0.6085; 0.0158)^T$	12	17	x_8^*
$(0.5869; 0.0576)^T$	18	17	x_8^*
$(0.3676; 0.6315)^T$	14	15	x_8^*
$(0.8704; 0.0099)^T$	32	9	x_8^*
$(0.4103; 0.8936)^T$	23	11	x_7^*
$(0.2722; 0.1988)^T$	NC	18	x_5^*
$(0.8381; 0.0196)^T$	9	7	x_5^*
$(0.6813; 0.3795)^T$	12	8	x_5^*
$(0.7095; 0.4289)^T$	11	9	x_5^*
$(0.3704; 0.7027)^T$	NC	19	x_7^*
$(0.7271; 0.3093)^T$	11	8	x_5^*
$(0.8744; 0.0150)^T$	20	10	x_5^*
$(0.4387; 0.4983)^T$	14	19	x_8^*
$(0.3200; 0.9601)^T$	13	11	x_7^*
$(0.7266; 0.4120)^T$	15	8	x_5^*
$(0.7446; 0.2679)^T$	9	7	x_5^*
$(0.6833; 0.2126)^T$	8	9	x_5^*
$(0.6072; 0.6299)^T$	11	8	x_5^*
$(0.4514; 0.0439)^T$	21	20	x_8^*

Tables 19: Case $p = 5$.

Here, we observe that for $p = 5$, **ICUM-basin** of $x_2^*, x_5^*, x_6^*, x_7^*$ and x_8^* have, respectively, 11, 10, 30, 3 and 6 points..

4 Conclusions

The motivation to try to answer some questions about quasi-Newton methods such us (i) are there problems, in the applied research, for which the quasi-Newton methods are the best option? (ii) which are they? (iii) why? led us to make a rigorous bibliographical research, which we have done with great success in [12].

In that work we selected, among 295 applications that uses quasi-Newton methods, the nine that we found more interesting, covering several applied areas as Physics, Engineering, Biology, Astrophysics, etc [12]. Almost all of them used, as a tool to solve the nonlinear system that appeared in their problems, Broyden's method. The authors justified their choice by the very cheap computational cost and the very easy implementation of Broyden's method.

After the good results obtained with ICUM when applied to several problems from the classical literature [9], [5], and after the affirmation of Lukšan and Vlček [7] that it was the best quasi-Newton Method for large-scale sparse problems, we decided to do this work which basicaly compare these two methods for solving the nonlinear systems appearing in applied problems.

Since solving nonlinear systems is only a small (but important) step in the applied problems, it was not an easy task to obtain the equations, initial guesses and expected solution of the system to develop our project.

The results obtained show that only for problem **P3** [11], Broyden's method has a slightly better performance than ICUM.

Both methods found the expected solution for problem **P1** [14], but in all the cases ICUM performed less iterations than Broyden's method.

Problem **P2** [10] presented the most unexpected results. First of all, in one of the regions each method found a different equilibrium points. In another region, in five among six cases tested only ICUM found the expected solution.

In the last problem, **P4** [13], only ICUM converged in all the cases and in the cases where both methods converged, it had a better performance, in terms of number of iterations performed.

As can be easily seen, all the problems studied here can not be considered large scale problems: **P1:** (thirteen dimensional), **P2:** (seven dimensional), **P3:** (three dimensional) and **P4:** (two dimensional).

So, our last conclusion it that, besides being among the best quasi-Newton methods for solving large scale sparse nonlinear systems, ICUM is also competitive when used to solve the systems (not necessarily large scaled) that appear in real applications.

Acknowledgements

The authors would like to thank, in particular, the colleagues: Lucia Medina who gave us support with the Target problem; Carlos Piedrahíta and Silvia Martorano Raymundo which patiently explained the basic ideas of the Two-point ray tracing and AIDS-tuberculosis problems, respectively, and for giving us the numerical data for the implementation of these problems and finally, Luziane Ferreira de Mendonça for her suggestions to the computational implementation which improved this work.

References

[1] BROYDEN, C. G., Luss, D. [**1965**] **A class of methods for solving nonlinear simultaneus equations**. *Mathematics of Computations* **19,** 577-593 .

[2] DENNIS, Jr. J. E., Schnabel R. B. [**1983**] **Numerical methods for unconstrained optimization and nonlinear equations**. *Prentice-Hall.*

[3] DENNIS, Jr. J. E., Moré J. J. [**1977**] **Quasi-Newton methods, motivation and theory**. *SIAM Review* **19,** 46-89.

[4] KELLER, H. B., Perozzi, D. J. [**1983**] **Fast seismic ray tracing,** *SIAM Journal of Applied Mathematics* **43,** N.4 981-992.

[5] LOPES, V. L. R, Martínez , J. M. and Pérez, R. [**1999**] **On the local convergence of quase-Newton methods for nonlinear complementarity problems**, *Applied Numerical Mathematics* **30,** 3-22.

[6] LOPES, V. L. R. ; Martínez, J. M. [**1995**] **Convergence properties of the inverse Column-Updating method,** *Optimization Methods and Software* **6,** 127-144.

[7] LUKŠAN, L., Vlček, J. [**1998**] **Computational experience with globally convergent descent methods for large sparse systems of nonlineal equations,** *Optimization Methods and Software* **8,** 185-199.

[8] MARTÍNEZ, J. M. [**2000**] **Practical quasi-Newton methods for solving nonlinear systems,** *Journal of Computational and Applied Mathematics* **124,** 97-121.

[9] 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,** 129-140.

[10] RAYMUNDO, S. M, Engel, A. B. Yang, H. M. and Bassanezi, R. C. **[2001] An approach to estimating the transmission coefficients for AIDS and for Tuberculosis using mathematical models,** *Systems Analysis Modelling Simulation* , submitted.

[11] MEDINA, L., Wykes, C. **[2001] Multiple target 3D location airborne ultrasonic system**. *Ultrasonics* **39,** 19-25.

[12] PÉREZ, R., Lopes, V. L. R. **[2001] Recent applications of quasi-Newton methods for solving nonlinear systems of equations**. *Technical Report* **26/01**, Departamento de Matemática Aplicada, IMECC, Universidade Estadual de Campinas, Brazil.

[13] MILLER, J. R., Yorke, J. A. **[2000] Finding all periodic orbits of maps using Newton methods: sizes of basins**. *Physica D* **135,** 195-211.

[14] PIEDRAHÍTA, C. **[2001] Extenções do método de continuação usando combinatória para o traçamento de raios**. *Ph.D. Thesis,* IMECC, UNICAMP, Campinas, SP, Brazil.