



08 a 11 de Outubro de 2018
Instituto Federal Fluminense
Búzios - RJ

MÉTODOS ITERATIVOS PARA SOLUÇÃO DE SISTEMAS NÃO LINEARES E SUAS BACIAS DE ATRAÇÃO

Cauã da Silva Chagas dos Santos¹ - cauatunas@gmail.com

Josecley Fialho Góes¹ - josecley.goes@ufopa.edu.br

Raimundo Oscar de Sousa Júnior¹ - oscarfilho26@hotmail.com

¹Universidade Federal do Oeste do Para, Instituto de Engenharia e Geociências - Santarém, PA, Brasil

Resumo. Neste trabalho são apresentados alguns métodos iterativos clássicos e outros recentes para a resolução de Sistemas Não Lineares. Um problema comum dos métodos nesta área está relacionado em saber para qual solução o problema é atraído, dado um ponto inicial. Com o intuito de visualizar essas atrações, os problemas escolhidos são do tipo que geram fractais. A escolha por esse tipo de problema tem como benefício uma fácil visualização de como os métodos podem ser sensíveis ao valor inicial, mesmo quando a solução está próxima a esse valor, podendo o método divergir ou até mesmo convergir para uma solução que não é tão próxima. O objetivo geral deste trabalho é o estudo comparativo entre os métodos iterativos apresentados, através da análise das imagens das bacias de atração e tabelas de resultados obtidas. Os métodos são implementados na linguagem Julia e aplicados em diferentes sistemas não lineares cujas soluções já são conhecidas. Desta forma, é possível visualizar a eficiência dos métodos. Os dados obtidos são organizados em imagens e tabelas para a realização da análise comparativa, a fim de avaliar a convergência e a eficiência de cada método. Entre os métodos, se destacam o método de Newton, Halley e super-Halley.

Palavras-Chave: Sistemas Não Lineares, Métodos Iterativos, Fractal, Bacias de Atração

1. INTRODUÇÃO

A maior parte dos fenômenos naturais relacionados às mais diferentes áreas da ciência são não-lineares (Savi, 2006). Muitas consistem em resolver um sistema de equações do tipo

$$\begin{cases} f_1(\mathbf{x}) = 0 \\ f_2(\mathbf{x}) = 0 \\ \vdots \\ f_n(\mathbf{x}) = 0 \end{cases} \Rightarrow F(\mathbf{x}) = \mathbf{0} \quad (1)$$

onde cada $f_i, i = 1, 2, \dots, n$ é uma função real de n variáveis reais e F é uma função vetorial em $\mathbf{x} \in \mathbb{R}^n$, sendo que os métodos mais utilizados para resolver este problema são os métodos iterativos, pois em geral não é possível encontrar uma solução analítica.

Dentre os métodos iterativos clássicos, podemos destacar o método de Newton. O método consiste em resolver um sistema linear a cada iteração, cuja matriz dos coeficientes é a jacobiana de F avaliada no iterado atual. Uma das vantagens desse método, sob condições adequadas, é a taxa de convergência quadrática (Kelley, 1995).

Nos últimos anos vem surgindo novos métodos para resolver Sistemas Não Lineares (SNLs), com o objetivos de obter alta velocidade de convergência, convergência global ou eficiência computacional. Percebe-se claramente que o método de Newton e suas variações estão entre os métodos numéricos mais notáveis para encontrar soluções de SNLs. No entanto, geralmente, os autores geralmente resolvem problemas tradicionais afim de validar a qualidade de um determinado método. Portanto, afirmar se o método é estável e globalmente convergente pode não ser satisfatoriamente convincente.

A teoria desses métodos é geralmente apresentada em duas partes: uma com ênfase na prova de convergência do método; a outra abordando a derivação da velocidade assintótica de convergência. O quadro que assim se obtém da literatura é, no entanto, um pouco incompleto. Neste trabalho, será abordado uma metodologia que pode ser utilizada para garantir a qualidade dos métodos de maneira visual, usando problemas de duas variáveis, delimitando o domínio em quadrado onde todas as raízes do problema estão presentes.

Os problemas proposto neste trabalho geram uma estrutura fractal (Falconer, 1986), ao mapear as bacias de atração dos métodos. Desde a década de 1980, os fractais têm atraído grande interesse (Falconer, 2013). Praticamente todas as áreas da ciência foram examinadas do ponto de vista fractal, com a geometria fractal se tornando uma área importante da matemática, tanto como um assunto de interesse em si mesmo quanto como uma ferramenta para uma ampla gama de aplicações. Por meio da estrutura fractal dos métodos iterativos, é possível visualizar suas qualidades, principalmente a sensibilidade em relação ao valor inicial para inicialização dos métodos. O comparativo dos métodos será realizado através dessas imagens e tabelas de resultados numéricos.

Problema de Cayley Este problema remonta a 1879 e foi proposto por *Arthur Cayley* em um artigo intitulado: “*The Newton-Fourier imaginary problem*” (Peitgen, 1986). *Cayley* sugere a extensão do que ele chama de método de *Newton-Fourier*. Seja $P : \mathbb{C} \rightarrow \mathbb{C}$ um polinômio complexo de grau de pelo menos dois e seja α um dos seus zeros. Se $z^{(0)} \in \mathbb{C}$ é um ponto de partida para o método de *Newton* definido por

$$z^{(n+1)} = z^{(n)} - \frac{P(z^{(n)})}{P'(z^{(n)})}, \quad n = 0, 1, 2, \dots \quad (2)$$

então a sequência $\{z^{(n)}\}_{n=0}^{\infty}$ converge, ou não, para um zero de P . Se a sequência converge para α então dizemos que $z^{(0)}$ é atraído para α . *Cayley* reconheceu que, se conhecemos as raízes de uma função, o método de *Newton* sugere outro problema: que ponto de partida e para quais raízes que a iteração de *Newton* não convergem? Apesar de não haver uma tentativa de responder de forma explicita, este problema é uma das principais motivações deste trabalho.

2. Métodos Iterativos para Sistemas Não Lineares

Os dez métodos iterativos aqui tratados são do tipo $\mathbf{x}^{(k+1)} = \phi(\mathbf{x}^{(k)})$. Estes fornecem uma sequência de vetores $\mathbf{x}^{(k)} \xrightarrow{\infty} 0$, sendo k o índice do passo (ou iteração) e $\mathbf{x}^{(k)}$ a aproximação da solução na k -ésima iteração. A partir de uma estimativa inicial $\mathbf{x}^{(0)}$, esta sequência poderá divergir ou convergir para uma i -ésima raiz $\mathbf{x}^{*,i}$ do sistema $F(\mathbf{x}) = 0$.

- **Newton** (Kelley, 1995):

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - F'(\mathbf{x})^{-1}F(\mathbf{x}^{(k)}). \quad (3)$$

- **Classe Halley** (Gutierrez & Hernandez, 1997):

$$\begin{aligned} F'(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)} &= -F(\mathbf{x}^{(k)}) \\ [F'(\mathbf{x}^{(k)}) + \alpha F''(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)}]\mathbf{s}_2^{(k)} &= -\frac{1}{2}[F''(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)}]\mathbf{s}_1^{(k)} \\ \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \mathbf{s}_1^{(k)} + \mathbf{s}_2^{(k)} \end{aligned} \quad (4)$$

- **Schorder** (Steihaug & Suleiman 2013):

$$\begin{aligned} F'(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)} &= -F(\mathbf{x}^{(k)}) \\ [F'(\mathbf{x}^{(k)}) + F''(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)}]\mathbf{s}_2^{(k)} &= -[F''(\mathbf{x}^{(k)})\mathbf{s}_1^{(k)}]\mathbf{s}_1^{(k)} \\ \mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \mathbf{s}_1^{(k)} + \mathbf{s}_2^{(k)}. \end{aligned} \quad (5)$$

- **Halley-Cuyt** (Cuyt & Rall, 1985):

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + (\mathbf{a}^{(k)} \odot \mathbf{a}^{(k)}) \oslash \left(\mathbf{a}^{(k)} + \frac{1}{2}\mathbf{b}^{(k)} \right), \quad (6)$$

onde

$$\mathbf{a}^{(k)} = -F'(\mathbf{x})^{-1}F(\mathbf{x}^{(k)})$$

e

$$\mathbf{b}^{(k)} = F'(\mathbf{x}^{(k)})^{-1}[F''(\mathbf{x}^{(k)})\mathbf{a}^{(k)}]\mathbf{a}^{(k)}.$$

- **Traub** (Traub, 1964):

$$\begin{aligned} \mathbf{y}^{(k)} &= \mathbf{x}^{(k)} - F'(\mathbf{x})^{-1}F(\mathbf{x}^{(k)}) \\ \mathbf{x}^{(k)} &= \mathbf{y}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{y}^{(k)}) \end{aligned} \quad (7)$$

- **Hueso-Martínez-Torregrosa** (Hueso et al., 2009):

$$\begin{aligned} \mathbf{y}^{(k)} &= \mathbf{x}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{x}^{(k)}) \\ \mathbf{z}^{(k)} &= \mathbf{y}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{y}^{(k)}) \\ \mathbf{x}^{(k)} &= \mathbf{z}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{z}^{(k)}) \end{aligned} \quad (8)$$

- **Waseem-Noor-Noor** (Waseem et al., 2016):

$$\begin{aligned} \mathbf{y}^{(k)} &= \mathbf{x}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{x}^{(k)}) \\ \mathbf{z}^{(k)} &= \mathbf{y}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{y}^{(k)}) \\ \mathbf{w}^{(k)} &= \mathbf{z}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{z}^{(k)}) \\ \mathbf{x}^{(k)} &= \mathbf{w}^{(k)} - F'(\mathbf{x}^{(k)})^{-1}F(\mathbf{w}^{(k)}) \end{aligned} \quad (9)$$

3. Problemas escolhidos

Para obtenção dos resultados, foram utilizados sistemas cujas raízes são previamente conhecidas, para assim, validar as raízes fornecidas pelos métodos numéricos. As raízes aqui apresentadas foram obtidas utilizando-se o pacote SymPy para cálculos simbólicos do software Julia (Bezanson et al., 2017; Meurer et al., 2017). Os sistemas não lineares analisados foram obtido usando uma equação com número complexo

$$z = x_1 + ix_2, \quad (10)$$

onde a parte real será a primeira equação e parte imaginária será a segunda equação do sistema. Por Exemplo $p(z) = z^2 - 1 = 0$ é equivalente à $p(z) = x_1^2 + 2ix_1x_2 - x_2^2 - 1 = 0$. Dessa forma, obtemos:

$$F(\mathbf{x}) = \begin{bmatrix} x_1^2 - x_2^2 - 1 \\ 2x_1x_2 \end{bmatrix} = \mathbf{0}.$$

3.1 Problema 1

$$p(z) = z^3 - 1$$

$$F(\mathbf{x}) = \begin{bmatrix} x_1^3 - 3x_1x_2^2 - 1 \\ 3x_1^2x_2 - x_2^3 \end{bmatrix} = \mathbf{0}. \quad (11)$$

Tabela 1 - Classificação de cores para soluções de $h(z) = z^3 - z$

| Solução | Cor |
|---------------------------------------|---------|
| (1, 0) | ciano |
| $(-\frac{1}{2}, \sqrt{\frac{3}{2}})$ | verde |
| $(-\frac{1}{2}, -\sqrt{\frac{3}{2}})$ | amarelo |

3.2 Problema 2

$$p(z) = z^5 - z = 0$$

$$F(\mathbf{x}) = \begin{bmatrix} x_1^5 - 10x_1^3x_2^2 + 5x_1x_2^4 - x_1 \\ 5x_1^4x_2 - 10x_1^2x_2^3 + x_2^5 - x_2 \end{bmatrix} = \mathbf{0}. \quad (12)$$

Tabela 2 - Classificação de cores para soluções de $h(z) = z^5 - z$

| Solução | Cor |
|---------|----------|
| (1, 0) | ciano |
| (0, 0) | amarelo |
| (-1, 0) | verde |
| (0, -1) | magenta |
| (0, 1) | vermelho |

3.3 Problema 3

$$p(z) = z^5 - 1 = 0$$

$$F(\mathbf{x}) = \begin{bmatrix} x_1^5 - 10x_1^3x_2^2 + 5x_1x_2^4 - 1 \\ 5x_1^4x_2 - 10x_1^2x_2^3 + x_2^5 \end{bmatrix} = \mathbf{0}. \quad (13)$$

Tabela 3 - Classificação de cores para soluções de $h(z) = z^5 - 1$

| Solução | Cor |
|--|----------|
| (1, 0) | verde |
| $-\frac{1}{4}(\sqrt{5} + 1); -\frac{1}{2}\sqrt{\frac{1}{2}(5 - \sqrt{5})}$ | ciano |
| $-\frac{1}{4}(\sqrt{5} + 1); \frac{1}{2}\sqrt{\frac{1}{2}(5 - \sqrt{5})}$ | amarelo |
| $\frac{1}{4}(\sqrt{5} - 1); -\frac{1}{2}\sqrt{\frac{1}{2}(5 + \sqrt{5})}$ | magenta |
| $\frac{1}{4}(\sqrt{5} - 1); \frac{1}{2}\sqrt{\frac{1}{2}(5 + \sqrt{5})}$ | vermelho |

4. Bacias de Atração

Para um domínio $D \subset \mathbb{R}^2$, inicia-se cada método iterativo fornecendo as estimativas iniciais $\mathbf{x}^{(0)} \in D$. Para a geração das estimativas iniciais $\mathbf{x}^{(0)}$, utilizou-se o domínio quadrado $D = [-2, 2] \times [-2, 2]$, sendo este discretizado em uma malha de 512×512 pontos. As raízes de todos os sistemas utilizados se encontram dentro do domínio D . Os critérios de parada usados foram $\|F(\mathbf{x}^{(k)})\| < 10^{-10}$, sendo considerado convergente neste caso, e no máximo 40 iterações. Se um processo iterativo alcançar 40 iterações, este é considerado divergente no ponto inicial dado.

Cada raiz do sistema é caracterizada por uma cor e cada ponto da malha será pintado da cor que representa a raiz a qual o ponto inicial convergiu. As bacias de atração irão identificar através dos pontos da malha os diferentes tipos de soluções através das cores que caracterizam cada raiz. Suponhamos que uma função $F(\mathbf{x})$ possua três raízes, $\mathbf{x}^{*,1}$, $\mathbf{x}^{*,2}$ e $\mathbf{x}^{*,3}$, as quais são caracterizadas pelas cores vermelho, verde e azul, respectivamente. Dado um ponto inicial que convergiu para a raiz $\mathbf{x}^{*,1}$ este ponto é pintado de vermelho, caso convirja para $\mathbf{x}^{*,2}$ é pintado de verde e se convergir para $\mathbf{x}^{*,3}$ é pintado de azul.

Foi implementado um algoritmo na linguagem *Julia* que gera as bacias descritas acima. Neste algoritmo é necessário a entrada de todas as raízes, e a escolha das cores que cada raiz representará, os intervalos para gerar a malha e o método a ser usado. A partir de um ponto inicial é gerada uma sequência $\{\mathbf{x}^{(k)}\}_{k=0}^{\infty}$ pelos métodos. A saída dos algoritmos será um $\mathbf{x}^{(k+1)}$, se atingir o critério de convergência, ou um valor lógico *false*, se atingir o número de iterações. Supondo que a saída é convergente, então $\mathbf{x}^{(k+1)}$ é filtrado com $\|\mathbf{x}^{(k+1)} - \mathbf{x}^{*,i}\|_2 < 10^{-3}$. O menor valor desta norma obviamente indica para que raiz $\mathbf{x}^{*,i}$ a sequência convergiu e o ponto inicial é pintado com a cor correspondente à essa raiz. Nos casos que atingirem o número máximo de iterações o filtro não é usado e a cor neste ponto será representado como branco, sendo considerado como ponto de divergência.

Afim de exibir a velocidade com que os métodos convergem para uma raiz a partir do ponto inicial, foram usados tonalidades diferentes para cada cor. Assim, as cores claras indicam um número menor de iterações para convergir para uma raiz, enquanto as mais escuras indicam um número maior de iterações.

Gerar essas imagens é a principal motivação deste trabalho, tendo como objetivo fazer um comparativo visual entre os métodos, apontando quais geram mais pontos de divergência, quais são mais convergentes e quais são os mais rápidos. Com essas imagens é possível apontar métodos mais estáveis, ou seja, pode-se afirmar se são ou não sensíveis aos valores iniciais. Um exemplo é mostrado na Figura 1. Em resumo, as imagens servem para, de uma maneira eficaz, avaliar a qualidade de um método iterativo.

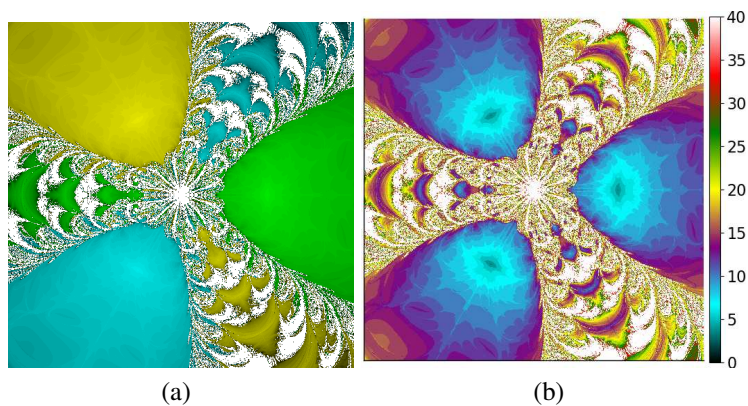


Figura 1 - Bacia de Atração (a) e Mapa de Iterações (b) do Método de Broyden

5. Resultados

Em cada processo iterativo foi utilizada o máximo de 40 iterações e o critério de convergência usado foi $\|F(\mathbf{x}^{(k)})\| < 10^{-10}$. Os métodos serão identificados pelas seguintes abreviações: Nw (Newton), Br (Broyden), Ch (Chebyshev), Ha (Halley), s-Ha (super-Halley), HaC (Halley-Cuyt), Tr (Traub), Hue (Hueso-Martínez-Torregrosa), WNN (Waseem-Noor-Noor).

5.1 Resultados do Problema 1

A partir da Tabela 4 podemos constatar que os métodos mais eficientes são Ha, s-Ha, pois não apresentaram pontos de divergência, seguidos por Nw, Sc, HaC e Ch.

Tabela 4 - Resultados para o Problema 1

| Método | Branco | Verde | Amarelo | Ciano |
|--------|--------|--------|---------|-------|
| Nw | 8 | 92562 | 84787 | 84787 |
| Br | 52588 | 62142 | 73707 | 73707 |
| Ch | 160 | 88920 | 86532 | 86532 |
| Ha | 0 | 90902 | 85621 | 85621 |
| s-Ha | 0 | 91640 | 85252 | 85252 |
| Sc | 22 | 75434 | 93344 | 93344 |
| HaC | 40 | 102062 | 80039 | 80039 |
| Tr | 1542 | 88394 | 86104 | 86104 |
| Hue | 11452 | 83356 | 83668 | 83668 |
| WNN | 30170 | 75422 | 78276 | 78276 |

5.2 Resultados do Problema 2

Com os resultados Tabela 5, pode ser afirmado que os métodos mais eficientes são Nw, Ha, s-Ha, Sc, HaC e Nw, que não obtiveram pontos divergentes, seguidos por Ch e Tr.

Tabela 5 - Resultados para o Problema 2

| Método | Branco | Verde | Amarelo | Ciano | Vermelho | Magenta |
|---------------|---------------|--------------|----------------|--------------|-----------------|----------------|
| Nw | 0 | 30550 | 139944 | 30550 | 30550 | 30550 |
| Br | 39124 | 17030 | 154904 | 17030 | 17028 | 17028 |
| Ch | 136 | 33742 | 127040 | 33742 | 33742 | 33742 |
| Ha | 0 | 39356 | 104720 | 39356 | 39356 | 39356 |
| s-Ha | 0 | 49548 | 63952 | 49548 | 49548 | 49548 |
| Sc | 0 | 15554 | 199928 | 15554 | 15554 | 15554 |
| HaC | 0 | 37664 | 111488 | 37664 | 37664 | 37664 |
| Tr | 608 | 28528 | 147424 | 28528 | 28528 | 28528 |
| Hue | 8672 | 26552 | 147264 | 26552 | 26552 | 26552 |
| WNN | 15376 | 24678 | 148056 | 24678 | 24678 | 24678 |

5.3 Resultados do Problema 3

Os resultados Tabela 6 indicam que os métodos mais eficientes são Ha, s-Ha, seguidos por Hac, Nw, Sc.

Tabela 6 - Resultados para o Problema 3

| Método | Branco | Verde | Amarelo | Ciano | Vermelho | Magenta |
|---------------|---------------|--------------|----------------|--------------|-----------------|----------------|
| Nw | 4156 | 45154 | 55743 | 55743 | 50674 | 50674 |
| Br | 132752 | 21022 | 30875 | 30875 | 23310 | 23310 |
| Ch | 9986 | 44092 | 54829 | 54829 | 49204 | 49204 |
| Ha | 0 | 47214 | 55573 | 55573 | 51892 | 51892 |
| s-Ha | 0 | 46944 | 55370 | 55370 | 52230 | 52230 |
| Sc | 4900 | 42772 | 60792 | 60792 | 46444 | 46444 |
| HaC | 2922 | 66842 | 47147 | 47147 | 49043 | 49043 |
| Tr | 36268 | 39064 | 49904 | 49904 | 43502 | 43502 |
| Hue | 64400 | 34142 | 44252 | 44252 | 37549 | 37549 |
| WNN | 82370 | 29576 | 41085 | 41085 | 34014 | 34014 |

Semelhante ao Problemas 1, os métodos Ha e s-Ha, não obtiveram pontos de divergência.

5.4 Ordem de Convergência

A Ordem de Convergência Computacional (OCC), de uma sequência $\{\mathbf{x}^{(k)}\}_{k=0}^{\infty}$ é definido por

$$OCC \approx \frac{\ln(\|\mathbf{x}^{(k+1)} - \alpha\|/\|\mathbf{x}^{(k)} - \alpha\|)}{\ln(\|\mathbf{x}^{(k)} - \alpha\|/\|\mathbf{x}^{(k-1)} - \alpha\|)}, \quad (14)$$

onde $\mathbf{x}^{(k-1)}$, $\mathbf{x}^{(k)}$ e $\mathbf{x}^{(k+1)}$ são três iterações consecutivas próximas a raiz α e podendo $\|\cdot\|$ ser qualquer norma em \mathbb{R}^n .

Um dos principais problemas de usar a OCC é que ela envolve uma raiz exata α , que em situações reais não é conhecida. Para contornar esse problema, surgiram variações da OCC que não usam uma raiz exata (Grau-Sánchez, 2012). A mais famosa utilizada é a Ordem de Convergência Computacional Aproximada (OCCA), que usa as quatro últimas iterações sucessivas da sequência $\{\mathbf{x}^{(k)}\}_{k=0}^{\infty}$, onde $\mathbf{x}^{(k)} \in \mathbb{R}^n$. A OCCA é dada por

$$OCCA \approx \frac{\ln(\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|/\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|)}{\ln(\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|/\|\mathbf{x}^{(k-1)} - \mathbf{x}^{(k-2)}\|)}, \quad (15)$$

sendo mais vantajoso em relação a OCC, pois não é necessário o conhecimento de qualquer raiz para o cálculo.

Para o cálculo da Ordem de Convergência, o valor inicial para todos os problemas será $\mathbf{x}^{(0)} = (1.2, -0.2)^t$. Este valor foi o escolhido por conta de todos os problemas abordados terem $(1, 0)^t$ como raiz. O critério de parada usado foi $(\|F(\mathbf{x}^{(k+1)})\|_2 + \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2) < 10^{-10}$. Este critério visa tentar garantir melhores resultados para o cálculo da OCCA.

Nos resultados encontrados na Tabela 7 também constam o número de iterações atingidas e o tempo médio gasto pelos métodos para atenderem o critério de parada. Além disso, para amenizar erros de precisão de máquina nos cálculos da OCC e da OCCA, os valores foram obtidos com precisão arbitrária (Bigfloat) da linguagem *Julia*. A precisão escolhida foi de 1000 dígitos (*setprecision(1000)*).

Tabela 7 - Ordem de Convergência Computacional dos métodos

| Mét. | Problema 1 | | | | Problema 2 | | | | Problema 3 | | | |
|------|------------|------|------|---------------|------------|------|------|---------------|------------|------|------|---------------|
| | Iter. | OCC | OCCA | Tempo | Iter. | OCC | OCCA | Tempo | Iter. | OCC | OCCA | Tempo |
| Nw | 6 | 2.00 | 2.00 | 6.95 μ s | 7 | 2.00 | 2.00 | 13.82 μ s | 6 | 2.07 | 2.00 | 12.05 μ s |
| Br | 8 | 1.59 | 1.63 | 11.86 μ s | 11 | 1.20 | 1.36 | 19.51 μ s | 10 | 1.21 | 1.62 | 17.97 μ s |
| Ch | 4 | 3.00 | 2.80 | 10.15 μ s | 5 | 3.00 | 2.95 | 18.02 μ s | 5 | 3.00 | 2.99 | 17.9 μ s |
| Ha | 4 | 3.00 | 2.91 | 10.12 μ s | 4 | 3.00 | 3.00 | 13.21 μ s | 4 | 3.00 | 2.88 | 13.21 μ s |
| s-Ha | 4 | 3.00 | 3.00 | 10.0 μ s | 4 | 3.00 | 2.58 | 13.02 μ s | 4 | 3.00 | 2.73 | 13.21 μ s |
| Sc | 7 | 2.00 | 2.00 | 15.73 μ s | 7 | 2.00 | 2.00 | 24.52 μ s | 7 | 2.00 | 2.00 | 23.92 μ s |
| HaC | 5 | 2.79 | 2.74 | 12.5 μ s | 5 | 2.41 | 2.99 | 17.13 μ s | 4 | 3.00 | 2.96 | 13.93 μ s |
| Tr | 4 | 3.00 | 3.00 | 8.92 μ s | 5 | 3.00 | 3.00 | 16.01 μ s | 5 | * | 3.28 | 16.04 μ s |
| HMT | 4 | 4.00 | 4.00 | 12.67 μ s | 4 | 4.00 | 4.03 | 18.60 μ s | 4 | 4.03 | 4.02 | 18.38 μ s |
| WNN | 3 | 5.01 | 5.01 | 12.38 μ s | 4 | 5.00 | 5.04 | 21.76 μ s | 4 | * | 5.14 | 21.79 μ s |

Legenda: Os valores com "*" significam que estes são indefinidos. Uma das razões para isso acontecer é o valor da última iteração ser igual a raiz.

A partir desta tabela, temos uma melhor noção de qual rápido os métodos são, a partir do valor inicial $\mathbf{x}^{(0)}$. Os resultados coincidem com da literatura (Noor et al., 2015; Waseem et al., 2016; Gutierrez & Hernandez, 1997; Hueso et al., 2009). Pode-se destacar os métodos de Newton, Tr, Ha e s-Ha.

O método Nw foi o mais rápido no Problema 1, sendo seguido por Tr, e no Problema 3, a qual Ha e s-Ha empataram na segunda colocação. Já para o problema 2, o mais rápido foi s-Ha e em segundo foi Ha.

Ressalta-se que, apesar do tempo computacional do HMT e WNN serem os mais altos, estes métodos são mais eficientes para problemas maiores, tendo como justificava a reutilização da matriz Jacobiana em seu múltiplos passos. Portanto, são mais competitivos quando usado decomposição matricial, por exemplo, a LU, na matriz Jacobiana no início de cada iteração para encontrar as soluções dos sistemas lineares em cada passo. Dessa forma, estes métodos terão um menor tempo e custo computacional.

6. CONCLUSÕES

Foram apresentados neste trabalho um número expressivo de métodos iterativos para a solução de sistemas não lineares. Os métodos apresentados são de fácil implementação. Também são apresentadas definições básicas de bacias de atração, que foi as ferramenta para a análise dos métodos iterativos apresentados. Os métodos iterativos estão implementados na linguagem *Julia* sendo aplicados na resolução de três diferentes sistemas não lineares cujas soluções são previamente conhecidas.

Os problemas escolhidos surgiram da equação complexa $z = x_1 + ix_2$. Utilizando como aproximação inicial uma malha 512×512 no intervalo $[-2, 2] \times [-2, 2]$ em \mathbb{R}^2 , são obtidos as bacias de atração de cada método nos problemas escolhidos. Desta forma, obteve-se dados que foram organizados em imagens e tabelas para auxiliar na análise comparativa, a fim de avaliar a convergência, eficiência e eficácia de cada método. Também foram calculados a Ordem de Convergência Computacional (OCC) e a Ordem de Convergência Computacional Aproximada (OCCA) e avaliar de a velocidade dos métodos em cada problema proposto.

Entre os métodos apresentados, os métodos Tensoriais (Classe Halley, Schroder, Halley-Cuyt) e o Newton (Nw) apresentaram os melhores resultados. Entre os ditos Tensoriais, destacam-se o método de Halley (Ha) e super-Halley (s-Ha). O método Ha apresentou estruturas fractais mais lisas (Figura 2(c) e Figura 4(d)), o que pode significar um método mais estável e menos suscetíveis a comportamentos inesperados por conta dos valores iniciais. Entretanto, o método s-Ha mostrou maiores regiões de atração e, conseqüentemente, sendo o mais convergente entre todos os métodos, além de abranger uma maior área onde o método converge com menos iterações (Figura 3(d), Figura 4(e) e Figura 5(e)).

Apesar de bons resultados, os métodos Tensoriais não são recomendados, devido seu alto custo computacional, principalmente, por conta do cálculo de Hessianas. A convergência destes métodos é cúbica, podendo ser substituídos, por exemplo, pelo método de Traub, que possui ordem 4 de convergência, sem a necessidade do cálculo de Hessianas.

O método Nw, também se mostrou eficiente. Além disso, se comparado aos métodos Tensoriais, o método de Newton é mais econômico computacionalmente, já que ele consiste em calcular a matriz de derivadas de primeira ordem (Jacobiana) e resolver um sistema linear a cada iteração.

Futuramente, pretende-se estender a metodologia para problemas de otimização, por conta de uma maior quantidade de métodos, com diferentes características, sendo possível usar métodos que não usam informação de derivadas e outros com um comportamento baseados em fenômenos físicos e biológicos.

REFERÊNCIAS

- Broyden, C. G. (1965), “A class of methods for solving nonlinear simultaneous equations”. *Mathematics of computation*, vol 19, n. 92, 577-593.
- Hueso, J. L.; Martínez, E. and Torregrosa, J. R. (2009), “Third and fourth order iterative methods free from second derivative for nonlinear systems”. *Applied mathematics and Computation*, vol 211, 190-197.
- Gutierrez, J. M & Hernandez, M. A. (1997), “A family of chebyshev-halley type methods in banach spaces”. *Bulletin of the Australian Mathematical Society*, vol 55, 113-130.
- Cuyt, A. A. M. & Rall, L. B. (1985), “Computational implementation of the multivariate Halley method for solving nonlinear systems of equations”, *ACM*, vol 11, n. 1, 20-36
- Waseem, M.; Noor, M.A. and Noor, K. I. (2016), “Efficient method for solving a system of nonlinear equations”, *Applied Mathematics and Computation*, vol 275, 134-146
- Kelley, C. T. (1995), “*Iterative methods for linear and nonlinear equations*”, SIAM, Philadelphia.
- Traub, J. F. (1964), “*Iterative methods for the solution of equations*”, Prentice-Hall, Englewood Cliffs, New Jersey.
- Noor, M. A.; Waseem, M. and Noor, K. I. (2015), “New iterative technique for solving a system of nonlinear equations”. *Applied Mathematics and Computation*, vol 271, 446-466.
- Steihaug, T. & Suleiman, S. (2013), “Rate of convergence of higher order methods”. *Applied Numerical Mathematics*, vol 67, 230-242.
- Gundersen, G. & Steihaug, T. (2010), “On large-scale unconstrained optimization problems and higher order methods”, *Optimization Methods & Software*, vol 25, n. 3, 337-358.
- Grau-Sánchez, M; Noguera, M.; Grau, À. and Herrero, J. R. (2012), “On new computational local orders of convergence”, *Applied Mathematics Letters*, vol 25, n. 12, 2023-2030.
- Bezanson J.; Edelman A.; Karpinski. S. and Shah V.B. (2017), “Julia: A fresh approach to numerical computing”, *SIAM*, vol 59, n. 1, 65-98.
- Meurer, Aaron et al. (2017), “SymPy: symbolic computing in Python”, *PeerJ Computer Science*, vol 3.
- Falconer, K. (2013), “*Fractals: a very short introduction*”, Oxford University Press, New York.
- Falconer, K. (1986), “*The geometry of fractal sets*”, Cambridge university press, vol 85.
- Peitgen, H. O.; Jürgens, H. and Richter, P.H. (1986), “*The beauty of fractals: images of complex dynamical systems*”, Springer.
- Savi, M.A. (2006), “*Dinâmica não-linear e caos*”, Editora E-papers.

APÊNDICE A

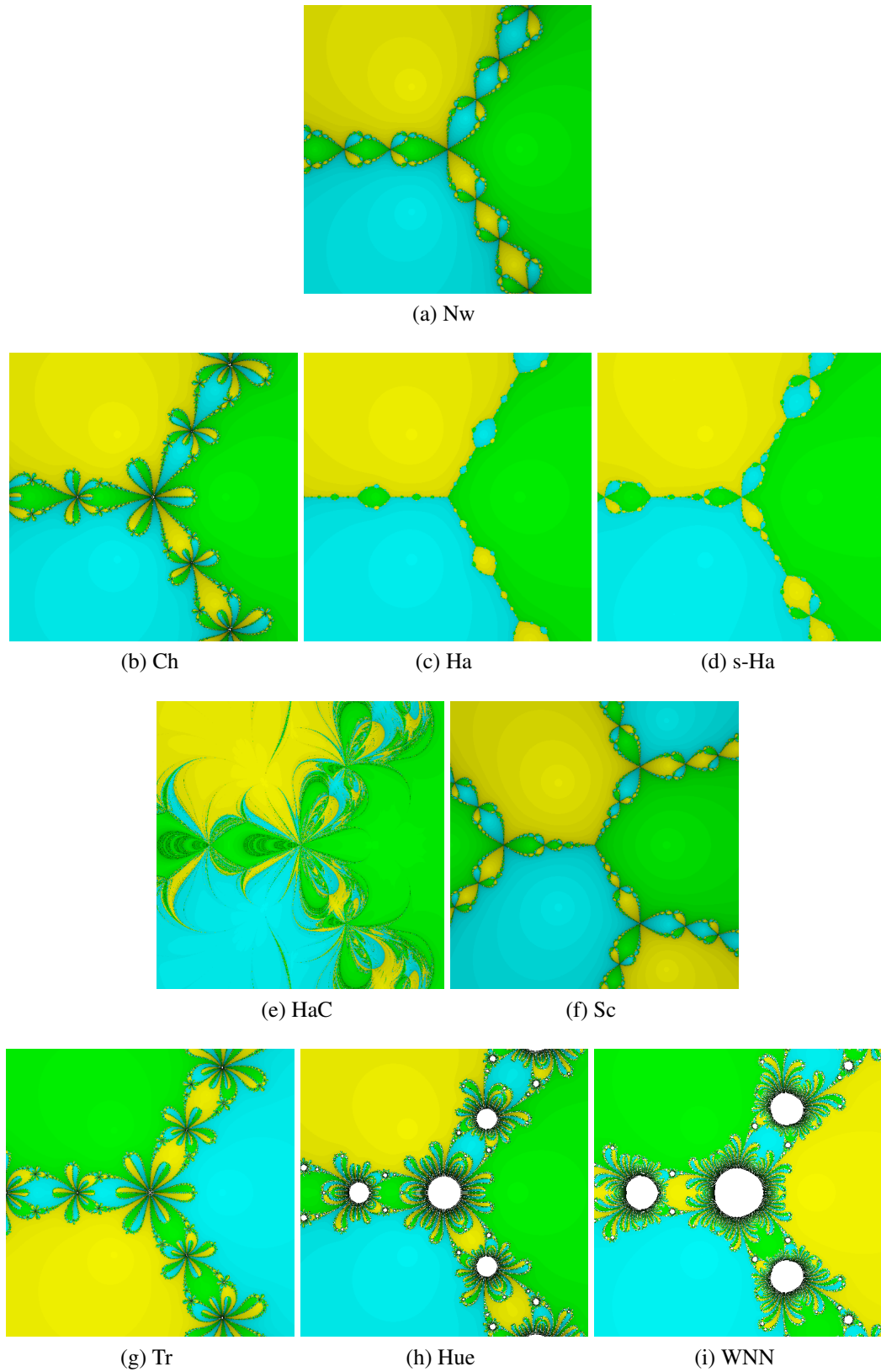


Figura 2 - Bacias de atração dos Métodos de Newton e Broyden para o problema 1

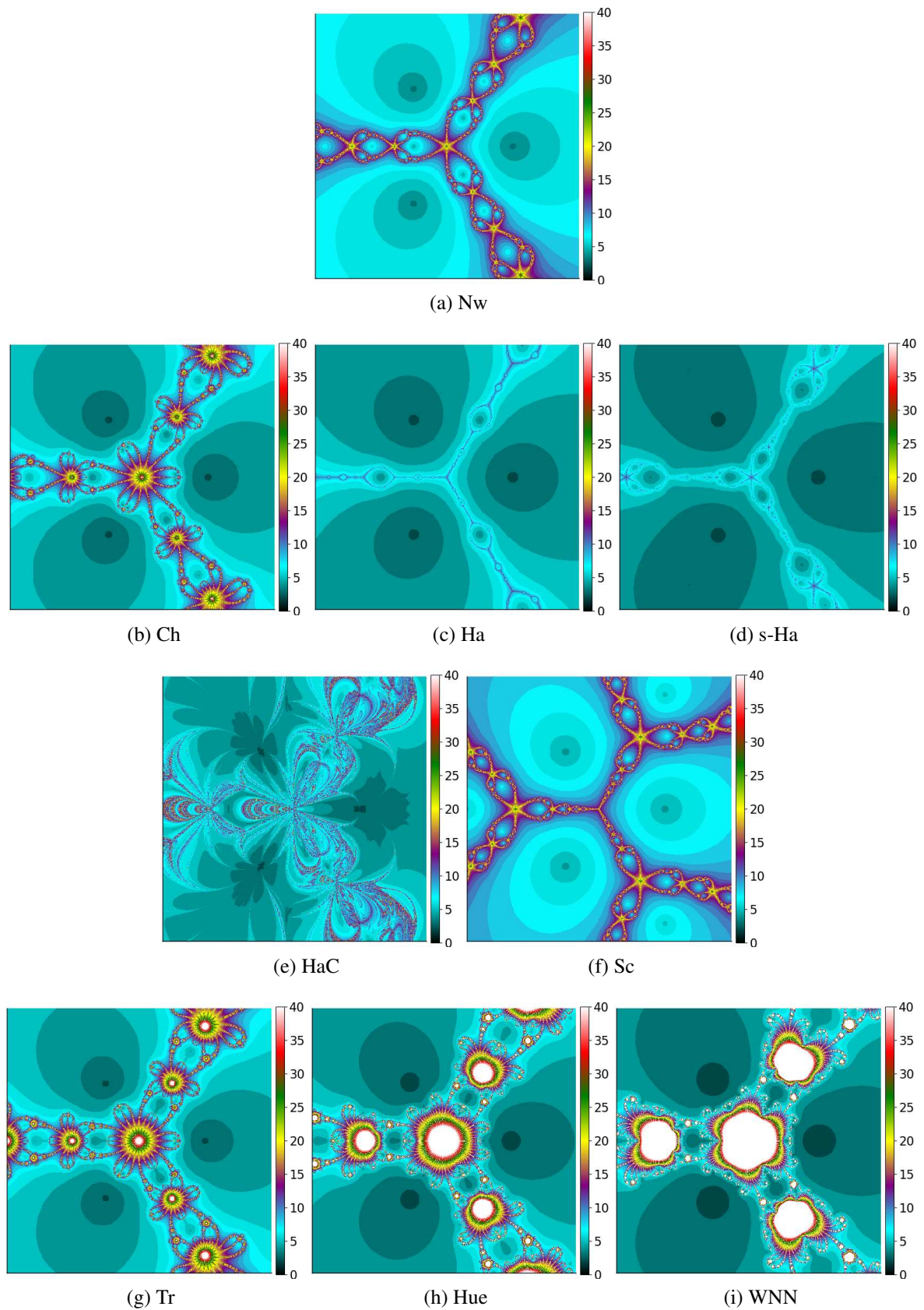


Figura 3 - Mapa de Iterações dos Métodos de Newton e Broyden para o problema 1

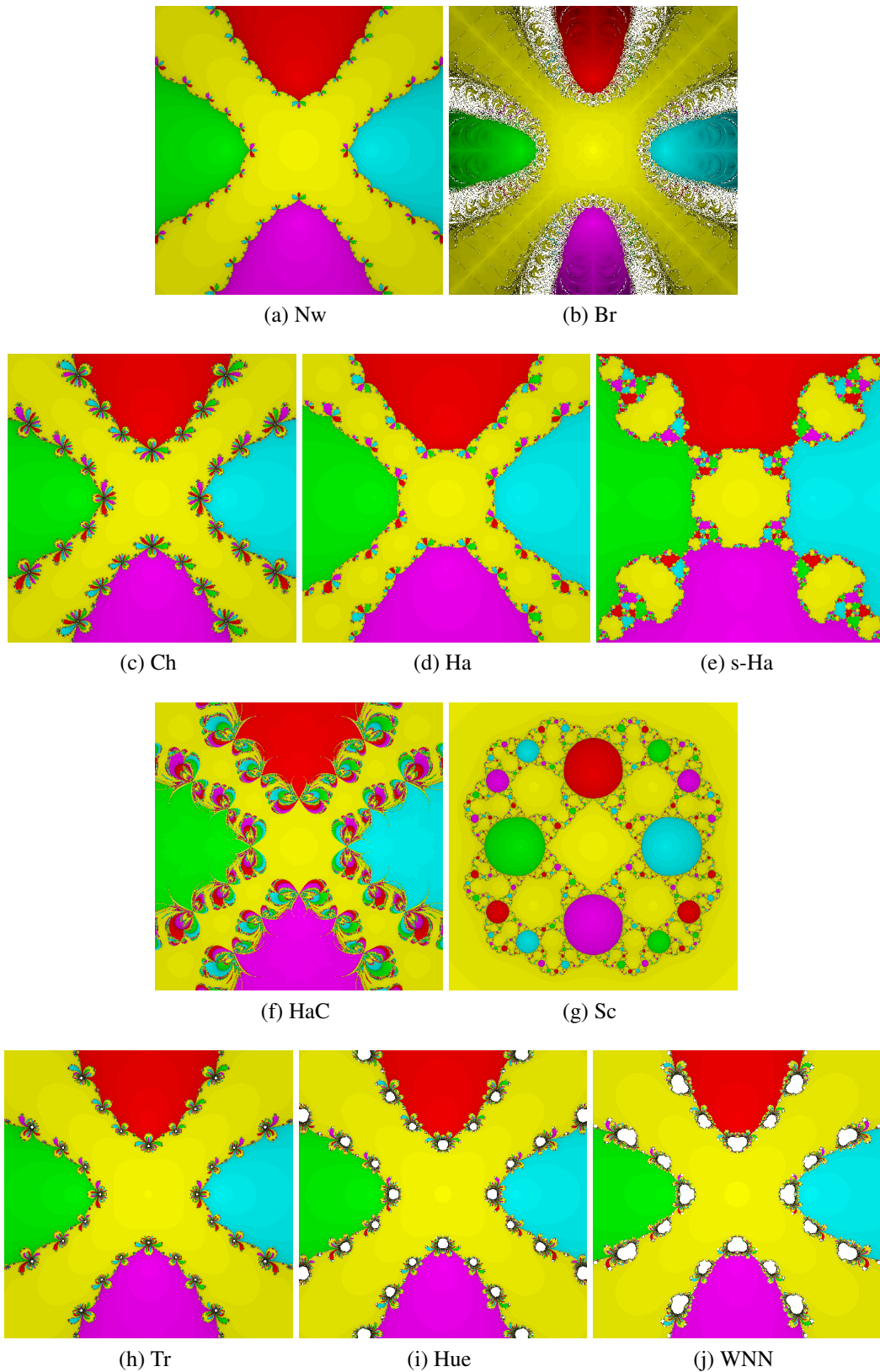


Figura 4 - Bacias de atração dos Métodos Clássicos para o problema 2

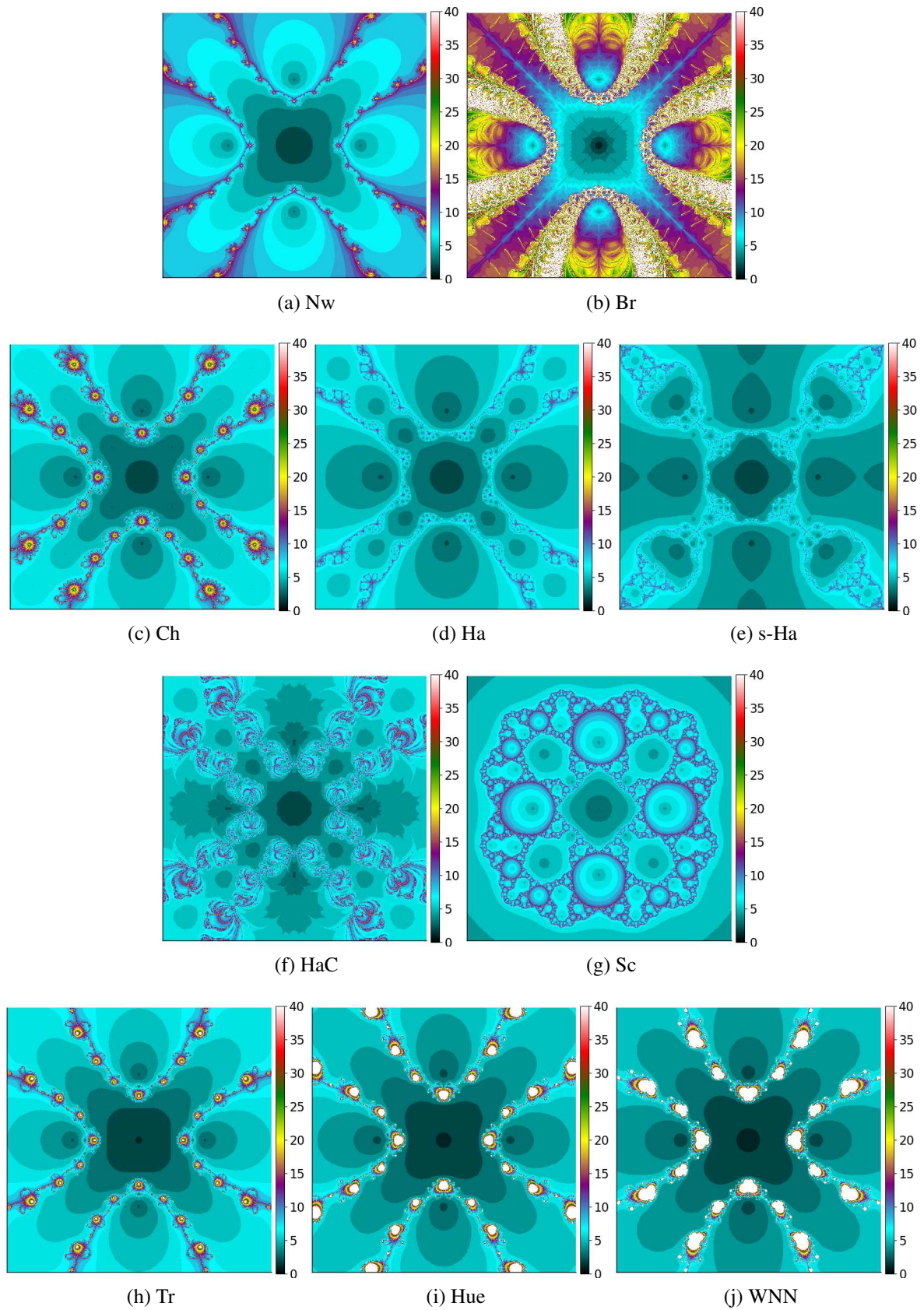


Figura 5 - Mapa de Iterações dos Métodos de Newton e Broyden para o problema 2

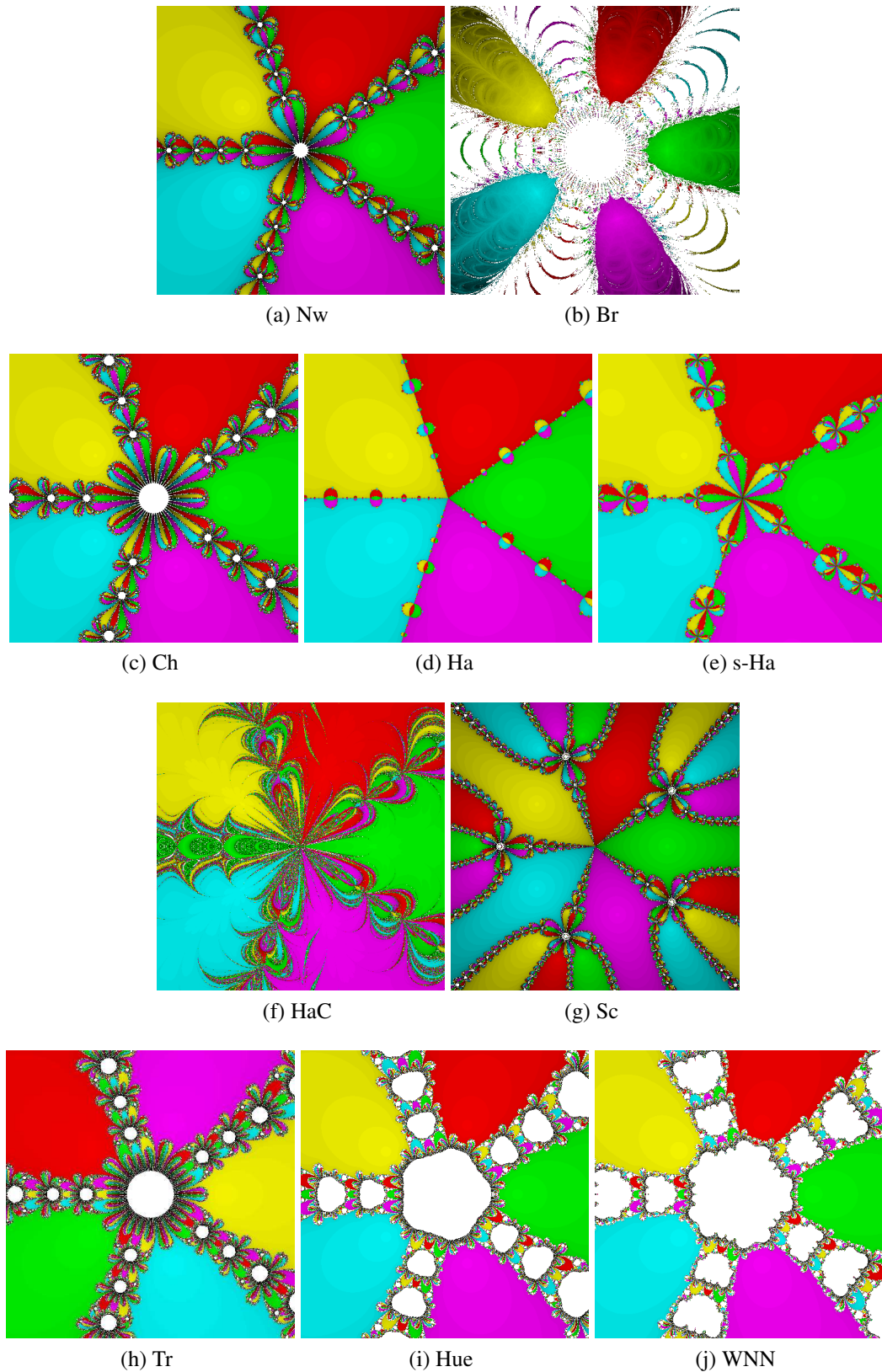


Figura 6 - Bacias de atração dos Métodos Clássicos para o problema 1

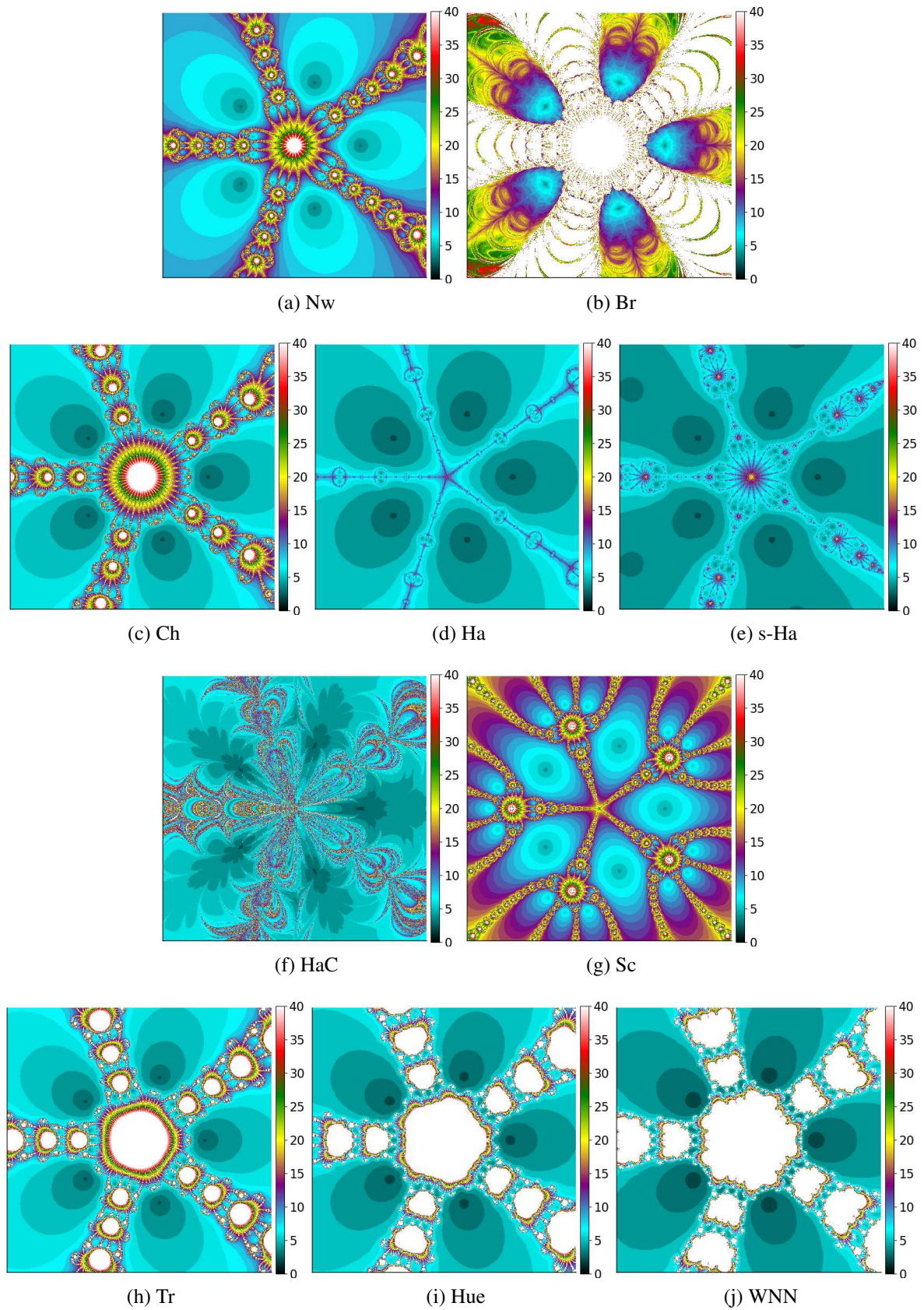


Figura 7 - Mapa de Iterações dos Métodos de Newton e Broyden para o problema 1

ITERATIVE METHODS FOR THE SOLUTION OF NONLINEAR SYSTEMS AND THEIR BASINS OF ATTRACTION

Abstract. *In this paper we present some classic and recent iterative methods for solving nonlinear systems. A common problem of methods in this area is related in knowing for which solution the attracted problem, given a starting point. In order to visualize these attractions, the problems chosen are of the type that generate fractals. The choice of this type of problem has the benefit of an easy visualization of how the methods can be sensitive to the initial value, even when the solution is close to that value, and the method can diverge or even converge to a solution that is not so close. The general objective of this work is the comparative study between the iterative methods presented, through the analysis of the images of the attraction basins and tables of results obtained. The methods are implemented in the Julia language and applied to different nonlinear systems whose solutions are already known. In this way, it is possible to visualize the efficiency of the methods. The obtained data are organized in images and tables for the accomplishment of the comparative analysis, in order to evaluate the convergence and the efficiency of each method. Among the methods, the method of Newton, Halley and super-Halley are the most outstanding.*

Keywords: *Nonlinear Systems, Iterative Methods, Fractal, Basins of Attraction*