UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA

7º Exercício Computacional de Algoritmos Numéricos II
Relatório

Matheus Gomes Arante de Souza
Vinícius Lucas dos Reis

Vitória
Julho/2019

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA

Matheus Gomes Arante de Souza
Vinícius Lucas dos Reis

7º Exercício Computacional de Algoritmos Numéricos II
Relatório

Este exercício tem por objetivo implementar o método do Ponto Fixo, o método de Newton e suas variações para solução do problema de Bratu não linear resultante da discretização de um PVC por meio de diferenças finitas. Para a implementação dos métodos, foi utilizado o ambiente de programação Octave.

Vitória
Julho/2019

Sumário

1 Introdução

Para encontrar a aproximação da solução de sistemas não lineares, podemos considerar o método das aproximações sucessivas (MAS) e o método de Newton.

O método das aproximações sucessivas consiste em encontrar $G(u)$ de modo que $F(u) = 0 \Rightarrow u = G(u)$. Desta maneira, podemos definir o MAS, a partir de u^0, da seguinte maneira: $u^{k+1} = G(u^k)$. Este método apresenta convergência linear, ou seja quando o número de iterações $k \to \infty$, o erro $e^{k+1} \to c\, e^k$.

O método de Newton é um dos principais métodos utilizados para solucionar sistemas não-lineares. Uma característica interessante deste método é que quando este converge, a convergência é quadrática, ou seja, quando o número de iterações $k \to \infty$, temos que o erro $e^{k+1} \to c(e^k)^2$. Entretanto, a cada iteração deste método é preciso calcular a solução de um sistema linear e por isso, é necessário utilizar métodos eficientes para resolver os sistemas lineares.

O método de Newton possui variações que buscam otimizar determinados pontos do algoritmo por meio de diferentes estratégias e consequentemente melhorar o tempo computacional. De maneira geral, estas são as diferenças entre um método e outro: Enquanto o método de Newton calcula a cada iteração a matriz Jacobiana, o método de Newton Modificado calcula uma única vez esta matriz. O método de Newton Aproximado calcula a cada iteração a matriz Jacobiana, porém utiliza aproximações por diferenças finitas. Por fim, o método de Newton inexato consiste no método de Newton com a resolução do sistema linear através de um método iterativo.

Nas seguintes seções apresentaremos os resultados e conclusões obtidas para o Problema Não-Linear de Bratu Unidimensional quando aplicado ao método das Aproximações Sucessivas, método de Newton, método de Newton Modificado, método de Newton Aproximado e método de Newton Inexato.

2 Resultados

2.1 Método das Aproximações Sucessivas

	n = 10			n = 100			n = 1000		
λ	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo
3.0	51	7.7e-08	0.0057	45	7.3e-08	0.0035	44	1e-07	0.0091
3.2	66	8.8e-08	0.0041	54	8.8e-08	0.005	54	8.6e-08	0.012
3.4	99	9.1e-08	0.0063	68	9e-08	0.0056	68	8.7e-08	0.014
3.6	100	0.00068	0.0064	91	9.2e-08	0.007	91	8.7e-08	0.02
3.8	100	NaN	0.0079	100	2.2e-06	0.0074	100	2.1e-06	0.018
4.0	100	NaN	0.0052	100	8.7e-05	0.0063	100	7.7e-05	0.017
4.2	100	NaN	0.0053	100	0.00026	0.0063	100	0.00023	0.017
4.4	100	NaN	0.0052	100	2.7e-05	0.0063	100	2.5e-05	0.017
4.6	100	NaN	0.0052	100	7.6e-07	0.0064	100	7.2e-07	0.017
4.8	100	3.5e-06	0.0053	89	9.3e-08	0.0057	89	9e-08	0.016
5.0	88	8.6e-08	0.0046	71	9.8e-08	0.0048	71	9.7e-08	0.013
5.2	68	8.2e-08	0.0038	60	9.2e-08	0.004	60	9.1e-08	0.011
5.4	56	9.3e-08	0.0032	52	9.1e-08	0.0037	52	9e-08	0.0099
5.6	49	7.5e-08	0.0028	46	8.7e-08	0.0032	46	8.7e-08	0.0088
5.8	43	7.5e-08	0.0027	41	8.9e-08	0.003	41	8.8e-08	0.008

Tabela 1: Número de iterações, resíduo relativo e tempo computacional para diferentes valores de λ e n.

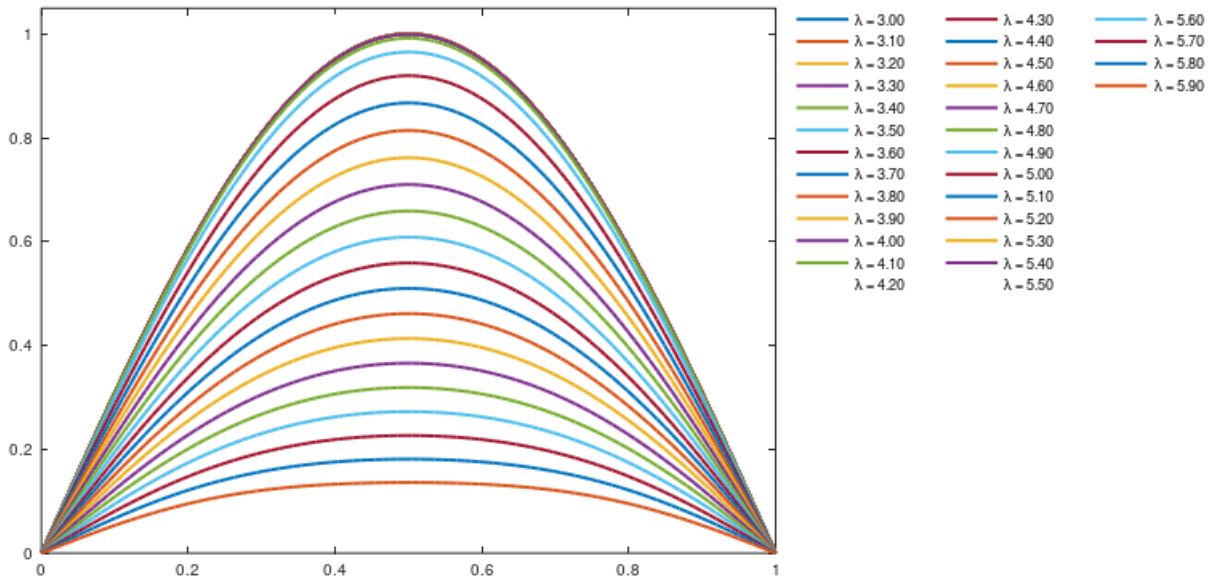

Figura 1: Resultado do Método das Aproximações Sucessivas para $\lambda \in [3.0, 6.0)$ e $n = 100$

2.2 Método de Newton

	n = 10			n = 100			n = 1000		
λ	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo
3.0	5	1.5e-11	0.004	5	6.6e-15	0.026	5	4.2e-16	0.43
3.2	5	1.2e-09	0.0027	5	2.5e-13	0.026	5	2.6e-15	0.4
3.4	6	3.3e-12	0.0034	5	9.4e-12	0.025	5	8.9e-14	0.39
3.6	100	0.0015	0.046	6	1.7e-15	0.025	6	2.4e-16	0.49
3.8	100	0.025	0.047	6	3.5e-12	0.025	6	3e-14	0.47
4.0	100	0.11	0.046	7	6.1e-12	0.036	7	3.1e-14	0.55
4.2	100	6.7e+22	0.072	100	5.3e-08	0.49	9	3.4e-16	0.86
4.4	100	9.6e+18	0.05	7	1.1e-15	0.036	7	2.2e-16	0.56
4.6	100	0.0062	0.047	6	9.1e-15	0.031	6	2.5e-16	0.47
4.8	6	1.2e-11	0.0032	5	4.1e-12	0.024	5	3.8e-14	0.39
5.0	5	1.1e-10	0.0033	5	1.4e-14	0.025	5	3.3e-16	0.39
5.2	5	6.2e-14	0.0032	5	8.5e-17	0.026	5	9.8e-17	0.4
5.4	4	5.5e-10	0.0023	4	1.3e-12	0.02	4	1.3e-14	0.31
5.6	4	3.9e-12	0.0022	4	1.1e-14	0.017	4	1.8e-16	0.31
5.8	4	5.2e-15	0.0022	4	6.1e-17	0.017	4	5.3e-17	0.31

Tabela 2: Número de iterações, resíduo relativo e tempo computacional para diferentes valores de λ e n.

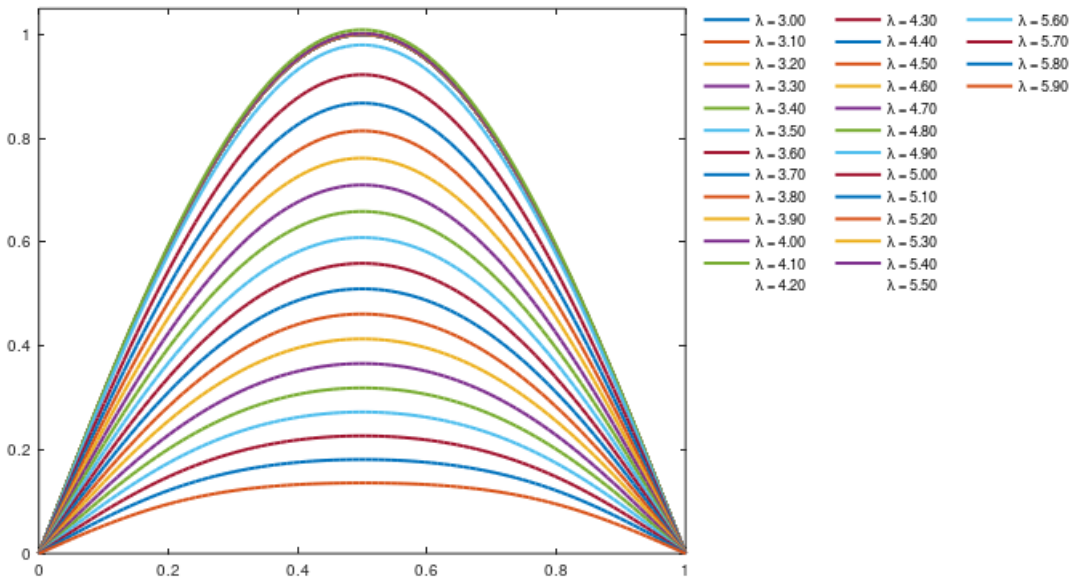

Figura 2: Resultado do Método de Newton para $\lambda \in [3.0, 6)$ e $n = 100$

2.3 Método de Newton Modificado

	n = 10			n = 100			n = 1000		
λ	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo
3.0	34	3.8e-09	0.013	29	3.9e-11	0.036	29	3.8e-13	1.4
3.2	44	4e-09	0.0082	35	3.8e-11	0.035	35	3.6e-13	1.6
3.4	66	4.3e-09	0.013	44	3.4e-11	0.043	44	3.3e-13	2.3
3.6	100	0.00086	0.022	59	3.1e-11	0.056	58	3.6e-13	2.6
3.8	36	NaN	0.0067	88	3.3e-11	0.079	87	3.3e-13	6.1
4.0	30	NaN	0.01	100	6.8e-09	0.16	100	5.1e-11	7.5
4.2	28	NaN	0.0097	100	9.5e-08	0.16	100	6.3e-10	7.7
4.4	31	NaN	0.012	100	1.8e-10	0.18	100	1.4e-12	7.6
4.6	49	NaN	0.016	66	2.3e-11	0.11	66	2.1e-13	5.1
4.8	76	2.1e-09	0.024	45	1.8e-11	0.085	45	1.7e-13	3.4
5.0	41	2.1e-09	0.014	33	1.4e-11	0.061	33	1.4e-13	2.5
5.2	29	1.2e-09	0.01	25	1.2e-11	0.05	25	1.2e-13	1.9
5.4	21	1.2e-09	0.0077	19	1.2e-11	0.04	19	1.2e-13	1.5
5.6	16	7.3e-10	0.0064	15	6.5e-12	0.035	15	6.4e-14	1.2
5.8	12	3.9e-10	0.005	11	6.8e-12	0.026	11	6.6e-14	0.87

Tabela 3: Número de iterações, resíduo relativo e tempo computacional para diferentes valores de λ e n.

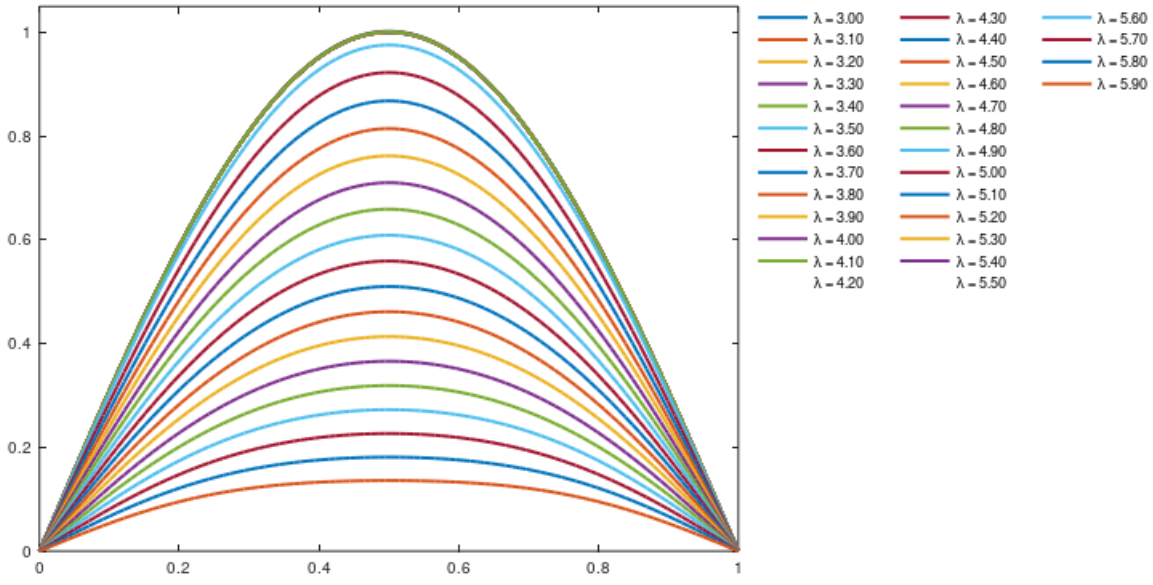

Figura 3: Resultado do Método de Newton Modificado para $\lambda \in [3.0, 6)$ e $n = 100$

2.4 Método de Newton Aproximado

	n = 10			n = 100			n = 1000		
λ	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo
3.0	10	3.5e-09	0.0071	6	8.4e-13	0.034	5	5.5e-15	0.44
3.2	13	3.9e-09	0.01	6	3.1e-12	0.031	5	6.8e-14	0.46
3.4	36	4.6e-09	0.025	6	7.3e-12	0.036	6	1.2e-15	0.53
3.6	100	0.019	0.059	6	2.6e-11	0.038	6	4.5e-15	0.54
3.8	100	34	0.052	8	2e-12	0.047	6	2.4e-13	0.52
4.0	100	6.3e+08	0.052	9	1.4e-11	0.053	7	1.8e-13	0.68
4.2	100	0.038	0.055	100	4.9e-08	0.61	9	4.6e-14	0.86
4.4	100	0.0013	0.06	8	1.3e-11	0.058	7	1e-14	0.65
4.6	100	0.00061	0.064	7	1.9e-12	0.044	6	4.3e-15	0.53
4.8	100	0.0012	0.054	6	1.4e-11	0.034	6	1.1e-15	0.52
5.0	20	1.7e-09	0.011	6	2.2e-12	0.034	5	6e-15	0.44
5.2	12	1.1e-09	0.0067	5	3.7e-12	0.028	5	1.9e-15	0.44
5.4	11	4.8e-10	0.0064	5	2.9e-12	0.029	5	3.4e-16	0.58
5.6	9	1.2e-09	0.01	5	9.1e-13	0.051	4	8.6e-15	0.62
5.8	8	6.9e-10	0.01	4	8e-12	0.039	4	1.1e-15	0.6

Tabela 4: Número de iterações, resíduo relativo e tempo computacional para diferentes valores de λ e n.

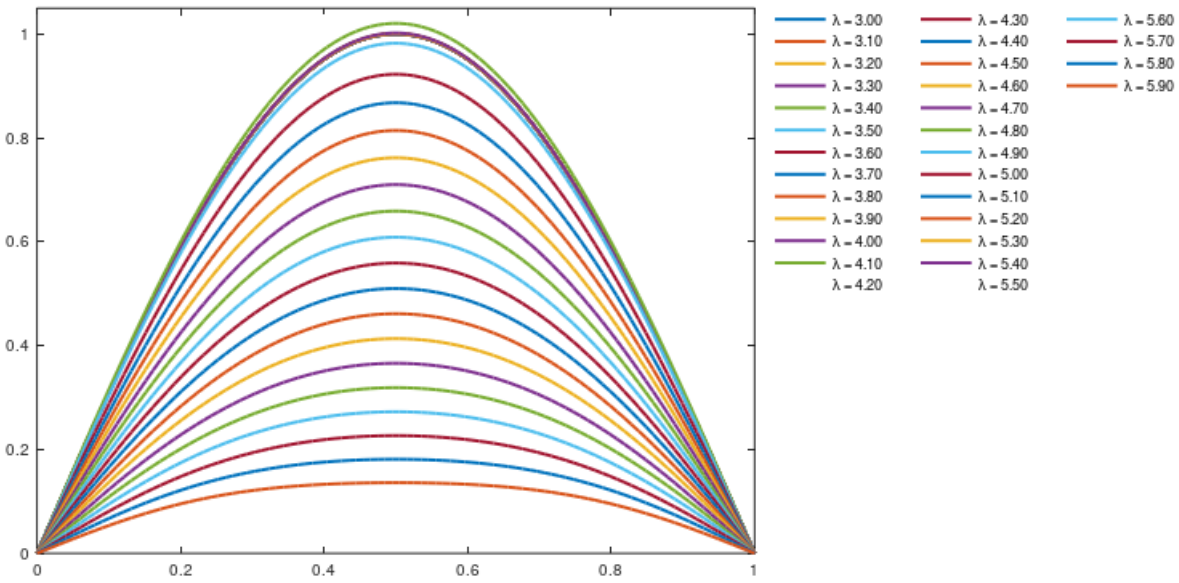

Figura 4: Resultado do Método de Newton Aproximado para $\lambda \in [3.0, 6)$ e $n = 100$

2.5 Método de Newton Inexato

	n = 10			n = 100			n = 1000		
λ	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo	Iter.	Res. Rel.	Tempo
3.0	5	1.5e-11	0.021	5	6.5e-15	0.035	5	2.4e-16	0.21
3.2	5	1.2e-09	0.01	5	2.5e-13	0.027	5	2.6e-15	0.22
3.4	6	3.3e-12	0.013	5	9.4e-12	0.03	5	9e-14	0.26
3.6	100	0.00012	0.25	6	1.7e-15	0.035	6	2.1e-16	0.29
3.8	100	0.001	0.2	6	3.5e-12	0.037	6	3e-14	0.28
4.0	100	0.0087	0.21	7	6.1e-12	0.039	7	3.1e-14	0.41
4.2	100	2.4e+23	0.2	100	3.3e-08	0.6	9	3.4e-16	0.47
4.4	100	6.3e+08	0.32	7	1e-15	0.044	7	2.2e-16	0.34
4.6	100	4.6e+62	0.34	6	9.1e-15	0.034	6	2.6e-16	0.29
4.8	6	1.2e-11	0.017	5	4.1e-12	0.052	5	3.8e-14	0.27
5.0	5	1.1e-10	0.012	5	1.4e-14	0.027	5	3.3e-16	0.25
5.2	5	6.2e-14	0.011	5	9.1e-17	0.03	5	9.9e-17	0.26
5.4	4	5.5e-10	0.011	4	1.3e-12	0.027	4	1.3e-14	0.23
5.6	4	3.9e-12	0.01	4	1.1e-14	0.026	4	2e-16	0.2
5.8	4	5.1e-15	0.013	4	6e-17	0.03	4	5e-17	0.18

Tabela 5: Número de iterações, resíduo relativo e tempo computacional para diferentes valores de λ e n.

Para n = 10, k = 10. Para n = 100, k = 40. Para n = 1000, k = 60 (k é o número de vetores na base para o restart do `gmres`). E o valor da tolerância em cada iteração foi definido pelo Critério Papadrakakis, sendo $n_0 = 10^{-7}$.

$$\eta_k = min\left(\eta_0, \left(\frac{||F(u)||}{r_0}\right)\right)$$

Para acelerar o tempo computacional, quando n = 1000 foi utilizado o pré-condicionador ILU sem preenchimento.

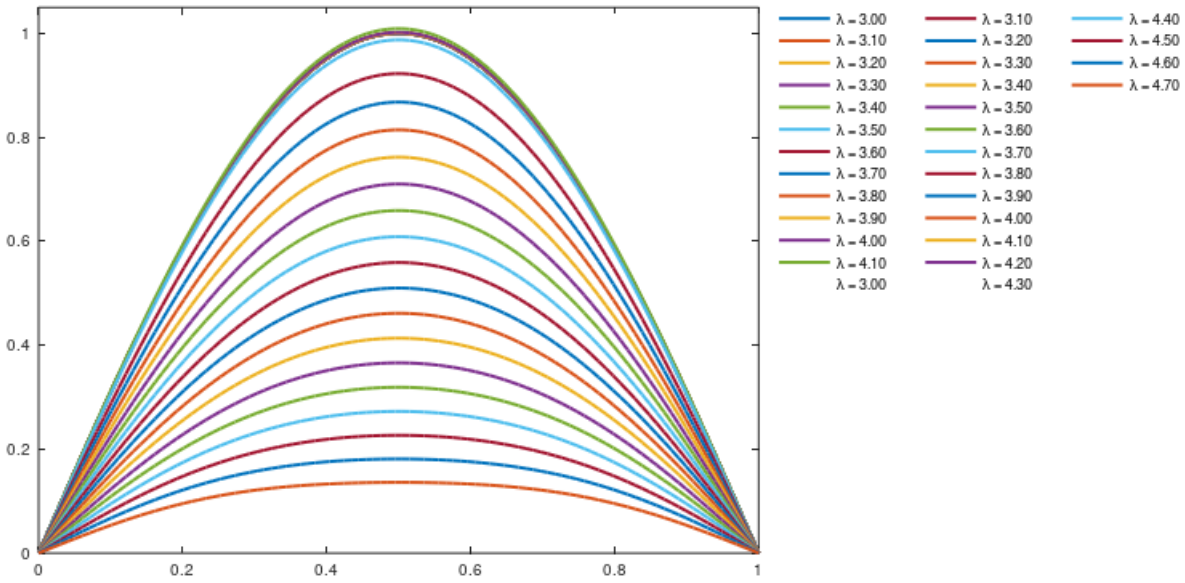

Figura 5: Resultado do Método de Newton Inexato para $\lambda \in [3.0, 6)$ e $n = 100$

3 Conclusão

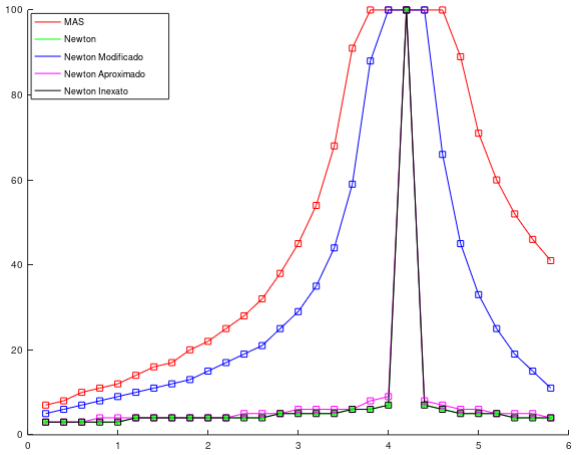

Figura 6: Comparação do Número de Iterações para $\lambda \in (0.0, 6.0)$ e $n = 100$

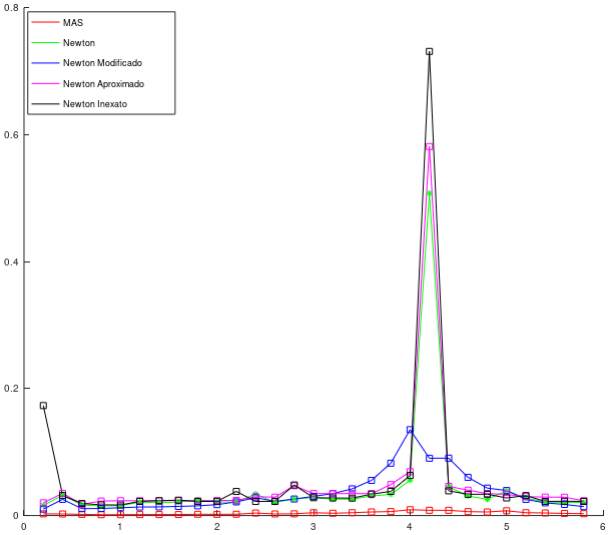

Figura 7: Comparação do Tempo Computacional para $\lambda \in (0.0, 6.0)$ e $n = 100$

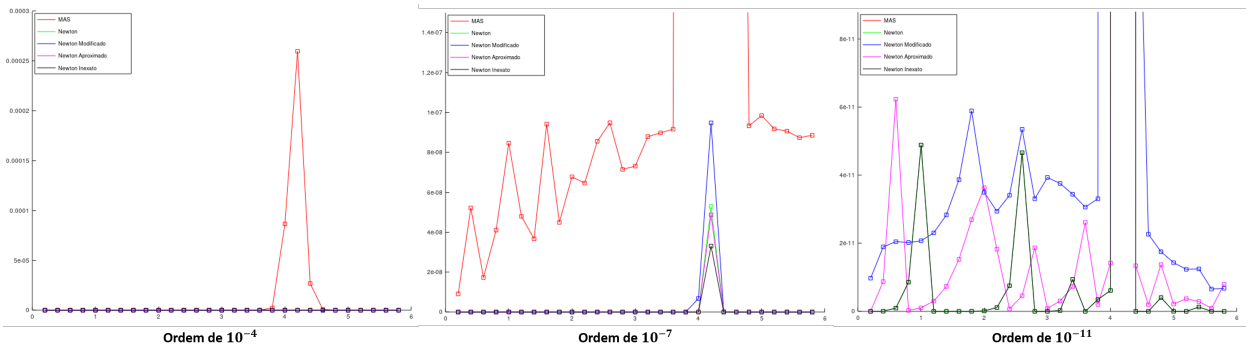

Figura 8: Comparação do Resíduo para $\lambda \in (0.0, 6.0)$ e $n = 100$

3.1 Análise dos Gráficos

Antes de iniciar a análise é preciso esclarecer que os gráficos foram gerados com $n = 100$, pois é onde as características a serem avaliadas ficam mais destacadas. E, por fim, vale ressaltar que no gráfico de comparação de tempo, o Método das Aproximações Sucessivas possui uma vantagem de implementação, uma vez que a cada iteração ele realiza basicamente uma operação de resolução de sistema linear que é dado pela linguagem, diferentemente dos outros métodos que possuem chamadas de funções externas que acabam por afetar seus respectivos desempenhos.

Analisando o primeiro gráfico (Figura 6) conseguimos perceber que conforme o valor de λ se aproxima de 4.2 (valor obtido pela análise das tabelas de cada método), o número de iterações cresce de forma acentuada. Isto mostra que o problema de Bratu possui um comportamento peculiar nesta vizinhança.

Dois algoritmos muito parecidos quanto à implementação são o de Newton e Newton Inexato, sendo que o primeiro calcula o sistema $Ju = F$ de forma exata e o último utiliza um algoritmo iterativo para resolver o sistema. Isto implica o que o uso de um bom algoritmo de resolução aproxima bastante os dois métodos, e isso fica claro quando comparamos as curvas de ambos em todos os gráficos, elas são muito próximas.

Uma boa reflexão a ser feita com base nesses gráficos é a importância de se escolher um algoritmo com base no que se deseja. Por exemplo, apesar do MAS ser o que realizou mais iterações, ele foi o que obteve o menor tempo computacional, e o maior resíduo. Portanto, se a demanda é sobre o tempo, seria melhor escolher este, mas se o desejado é uma resposta com melhor precisão, então uma boa escolha seria o do Newton, pois apesar de ter obtido praticamente a mesma acurácia do Newton Inexato, ele independe do algoritmo de resolução de SL utilizado para garantir sua precisão.

3.2 Análise das Tabelas Individuais

Algo interessante que é revelado ao se fazer uma análise dos resultados obtidos é que cada valor de λ possui um número de iterações diferentes para se obter a convergência. Portanto, caso o número máximo de iterações não seja suficiente para se alcançar a convergência, o alto valor de resíduo irá refletir essa parada "indesejada"por parte do algoritmo de resolução, evidenciando que este não pode terminar sua execução numérica.

Outro ponto a ser destacado é a dependência da qualidade da resposta, não só do valor de λ, mas também do número n de funções f que compõem a matriz de funções F para a resolução do sistema não linear. Ou seja, conforme se aumenta o número de funções do sistema, e consequentemente a matriz Jacobiana, maior é a probabilidade de se obter a convergência para determinada configuração do problema. Em muitos casos, os algoritmos só conseguiram convergir para $\lambda = 4.2$ e sua vizinhança quando $n = 1000$.

4 Referências

EATON, John W. Octave Documentation, 1996.
Disponível em: <https://octave.org/doc/v4.2.2/>.

COMMUNITY, Octave Forge. Octave Forge, 2002.
Disponível em: <https://octave.sourceforge.io/docs.php>.

CATABRIGA, Lucia. Sistemas Não Lineares (SNL), 2019.
Disponível em: <http://inf.ufes.br/~luciac/mn1/191-SNE-AlgoII.pdf>.