Estudo comparativo de três algoritmos de inversão numérica da transformada de Laplace

Noé Franco de Jesus ¹ Camila Pinto da Costa ² Leslie D. Pérez Fernández ² Julián Bravo Castillero ³

Resumo: Inúmeras aplicações na ciência e na tecnologia têm mostrado a utilidade da transformada de Laplace como ferramenta que auxilia na resolução dos modelos matemáticos correspondentes. Contudo, em diversas situações práticas, surge a necessidade de abordagens numéricas na inversão desta transformada por causa da dificuldade em se calcular analiticamente a integral de Bromwich que a define. No entanto, não existe nenhum método universal para a inversão numérica que funcione satisfatoriamente em todas as situações. Neste trabalho, apresenta-se um estudo comparativo do desempenho de três algoritmos de inversão numérica da transformada de Laplace, a saber, o algoritmo Talbot Fixo de Abate e Valkó, o de Stehfest, e o de Zakian.

Palavras-chave: Algoritmo de Stehfest. Algoritmo de Zakian. Algoritmo Talbot Fixo. Inversão numérica. Transformada de Laplace.

Abstract: Numerous applications in science and technology have shown the usefulness of the Laplace transform as a tool that assists in the resolution of the corresponding mathematical models. However, in several practical situations, the need for numerical approaches in the inversion of this transform arises because of the difficulty in analytically calculating the Bromwich integral that defines it. However, there is no universal method for numerical inversion that works satisfactorily in all situations. In this work, a comparative study of the performance of three numerical inversion algorithms of the Laplace transform, namely, the Abate and Valkó Fixed-Talbot algorithm, the Stehfest algorithm, and the Zakian algorithm, is presented.

Keywords: Fixed-Talbot algorithm. Laplace transform. Numerical inversion. Stehfest algorithm. Zakian algorithm.

1 Introdução

As equações diferenciais descrevem o modo como certas grandezas variam com o tempo e/ou no espaço, tal como a corrente em um circuito elétrico, a oscilação de uma membrana vibrante ou ao fluxo de calor através de um condutor isolado. Essas equações geralmente estão associadas às condições iniciais que descrevem o estado do sistema no instante de tempo inicial ou de maneira mais geral, às condições de contorno [1]. Essas equações são aplicadas em diversas áreas da ciência, sendo extensamente utilizada em Engenharia elétrica e Engenharia química.

Dentre as técnicas mais utilizadas na resolução de equações diferenciais encontra-se a transformada de Laplace, que é uma ferramenta operacional usualmente empregada na resolução de problemas de valor inicial que permite, por exemplo, transformar equações diferenciais ordinárias em equações algébricas.

http://dx.doi.org/10.5335/rbca.v9i1.6700

¹Universidade Federal de Pelotas, Instituto de Física e Matemática, Programa de Pós-Graduação em Modelagem Matemática (PPGMMAT), Campus Capão do Leão - Caixa Postal 354, CEP 96010-971 - Pelotas (RS) - Brasil.

²Universidade Federal de Pelotas, Instituto de Física e Matemática, Departamento de Matemática e Estatística, Campus Capão do Leão - Caixa Postal 354. CEP 96010-971 - Pelotas (RS) - Brasil

[{]camila.costa@ufpel.edu.br; leslie.fernandez@ufpel.edu.br}

³Universidad de La Habana, Facultad de Matemática y Computación, San Lázaro y L, Vedado, CP 10400, La Habana, Cuba. {jbravo@matcom.uh.cu}

Denota-se por F(s) a transformada de Laplace de uma função integrável f(t) definida para todo número real $t \geq 0$. Tipicamente, a função F(s) existe para todo s tal que Re(s) > a, em que a é uma constante que depende do comportamento de crescimento de f(t). Assim, a função original f(t) é recuperada mediante a inversão da transformada F(s). Contudo, em estudos mais detalhados, pode ser impossível realizar analiticamente a inversão da transformada, pelo qual é necessário usar algum algoritmo computacional que forneça uma boa aproximação da solução exata na maioria dos casos.

Muitos problemas de Matemática, envolvendo funções transcendentes, podem ser transformados em problemas algébricos usando a transformada de Laplace. Para obter a solução procurada é preciso inverter a transformada de Laplace. Infelizmente, nem sempre é possível calcular a sua inversa por métodos exatos. Além disso, quando os problemas algébricos obtidos têm graus elevados, a função que precisamos de inverter já é uma aproximação. Assim sendo, e se a sua inversa não puder ser calculada facilmente, não faz muito sentido calcular uma inversa exata de uma solução aproximada. Calcula-se, neste caso, uma solução numérica da transformada inversa procurada.

O objetivo do presente trabalho é comparar três algoritmos de inversão numérica da transformada de Laplace, a saber, os algoritmos de Abate e Valkó (ou Talbot Fixo) [2], de Stehfest [3], e de Zakian [4], com relação ao custo computacional, à qualidade da aproximação da função original, e à facilidade de implementação. Assim, este trabalho traz uma análise sucinta dos métodos estudados através de uma pesquisa direcionada.

Buscou-se conhecer o desempenho destes algoritmos e sua precisão nas funções analisadas e também a variação do parâmetro livre estabelecido em cada um deles. A implementação foi realizada em *Scilab* [5], um software livre para computação numérica semelhante ao Matlab que fornece um poderoso ambiente computacional aberto para aplicações científicas. É uma linguagem de programação de alto nível, orientada à análise numérica. A linguagem provê um ambiente para interpretação, com diversas ferramentas numéricas.

Este trabalho está organizado como segue: na Seção 2, se apresentam os três algoritmos de inversão numérica da transformada de Laplace mencionados acima; na Seção 3, o desempenho desses algoritmos é analisado; e as considerações finais são apresentadas na Seção 4.

2 Três algoritmos de inversão numérica da transformada de Laplace

Seja f(t) uma função integrável definida para todo número real t>0. A transformada de Laplace de f(t) é a função F(s) definida pela integral imprópria

$$F(s) = L\{f(t)\} = \int_0^\infty f(t)e^{-st}dt,$$
 (1)

em que $s \in \mathbb{C}$ é a variável no domínio de Laplace. A transformada inversa de Laplace denotada por $f(t) = L^{-1}\{F(s)\}$. A Transformada de Laplace e sua inversa têm algumas propriedades que as fazem útil para analisar sistemas dinâmicos lineares.

Uma fórmula da integral da transformada inversa de Laplace, chamada de integral de Bromwich, integral de Fourier-Mellin, e fórmula da inversa de Mellin, é dada pela integral de linha

$$f(t) = L^{-1}\{F(s)\} = \frac{1}{2\pi i} \lim_{T \to \infty} \int_{a-iT}^{a+iT} F(s)e^{ts}ds,$$
 (2)

em que a integração é feita ao longo da linha vertical Re(s)=a no plano complexo, sendo a maior do que a parte real de todas as singularidades de F(s). Isto garante que o caminho de integração esteja na região de convergência da integral em (2). Ainda, se todas as singularidades de F(s) estão no semi-plano esquerdo, ou se F(s) é uma função diferenciável em $-\infty < Re(s) < \infty$ (isto é, sem singularidades), então, ao fazer a=0 em (2), a transformada inversa de Laplace torna-se idêntica à transformada inversa de Fourier.

2.1 Algoritmo Talbot Fixo

Em 1979, Talbot [6] foi pioneiro na abordagem de métodos para inversão numérica da transformada de Laplace, propondo uma deformação do caminho de integração padrão $B = \{s \in \mathbb{C} : Re(s) = a\}$ na integral de

Bromwich dada por (2). De fato, pode-se deformar a reta B em qualquer caminho aberto que envolva o semieixo real negativo, garantindo que nenhuma singularidade de F(s) seja atravessada na deformação de B. Em outras palavras, a transformação é analítica na região do plano complexo à direita do caminho B.

Portanto, pelo teorema de Cauchy, o contorno deformado é válido. A brilhante contribuição de Talbot é a cuidadosa escolha do caminho:

$$s(\theta) = r\theta(\cot \theta + i), \quad -\pi < \theta < \pi \tag{3}$$

em que r é um parâmetro. Considere a transformada inversa em (2) para $F(s) = 1/s^{\alpha}$ com $\alpha > 0$:

$$f(t) = \frac{1}{2\pi i} \int_{B} \exp\left\{t\left(s - \beta \log s\right)\right\} ds,\tag{4}$$

em que $\beta = \alpha/t$. Em geral, a integral em (4) é difícil de avaliar por causa do comportamento numericamente oscilatório do integrando. Pode-se contornar este problema mediante a seguinte propriedade:

$$Im(s - \beta \log s) = 0. (5)$$

Seja s = x + iy. Em seguida, de (5) obtém-se a equação do caminho mais íngreme:

$$x = y \cot \frac{y}{\beta}. (6)$$

O algoritmo Talbot Fixo [2] é um caso particular do algoritmo de Talbot [6], onde de (2) e (3) segue que:

$$f(t) = \frac{1}{2\pi i} \int_{-\pi}^{\pi} \exp(ts(\theta)) F(s(\theta)) s'(\theta) d\theta. \tag{7}$$

Note que $s'(\theta) = ir(1 + \sigma(\theta))$ onde $\sigma(\theta) = \theta + (\theta \cot \theta - 1) \cot \theta$. Logo,

$$f(t) = \frac{r}{\pi} \int_0^{\pi} Re[\exp(ts(\theta))F(s(\theta))(1+i\sigma(\theta))]d\theta.$$
 (8)

Para aproximar o valor da integral em (8), se utiliza a regra dos trapézios com o tamanho de passo π/M , $M \in \mathbb{N}$, e $\theta_k = k\pi/M$, $k = 0, \dots, M$, obtendo:

$$f_{FT}(t) = \frac{r}{M} \left\{ \frac{1}{2} F(r) \exp(rt) + \sum_{k=1}^{M-1} Re\left[\exp(ts(\theta_k)) F(s(\theta_k)) (1 + i\sigma(\theta_k)) \right] \right\}. \tag{9}$$

Com base em experimentos numéricos, o parâmetro r é fixado no valor r=2M/5t. Assim, $f_{FT}(t)$, que denota o algoritmo de Talbot Fixo, dado por (9) possui somente um parâmetro livre, M que será o número de termos do somatório.

2.2 Algoritmo de Stehfest

O algoritmo de Stehfest [3, 7] tem sido utilizado na resolução de muitos problemas de transporte, por exemplo problemas de dispersão radial [8, 9, 10]. Segundo [11], a versão original desse algoritmo [3] apresentava erros que foram corrigidos em [7]. A versão definitiva desse algoritmo é como segue:

$$f_S(t) = \frac{\ln 2}{t} \sum_{j=1}^{N} V_j F\left(\frac{\ln 2}{t} j\right)$$

$$\tag{10}$$

Sendo que $f_S(t)$ é a aproximação para inversa exata de f(t) e V_i é dada por:

$$V_{j} = (-1)^{\left(\frac{N}{2}+j\right)} \sum_{k=\frac{j+1}{2}}^{\min\left(j,\frac{N}{2}\right)} \frac{k^{\left(\frac{N}{2}\right)}2k!}{\left(\frac{N}{2}-k\right)!k!(k-1)!(j-k)!(2k-j)!}$$
(11)

Em que N é um parâmetro livre a ser determinado e representa o número de termos utilizados no somatório em (11), sendo este um número inteiro par. Para que se tenha uma boa precisão N precisa variar de 10 a 14.

2.3 Algoritmo de Zakian

O algoritmo de Zakian [4] deriva uma fórmula explícita para inversa numérica da transformada de Laplace usando as funções ponderadas para aproximar a função do domínio de tempo obtendo assim:

$$f_Z(t) = \frac{2}{t} \sum_{j=1}^{N_z} Re\left\{k_j F\left(\frac{\alpha_j}{t}\right)\right\}$$
 (12)

onde α_j e k_j são constantes, que pode ser real ou pares de conjugado complexos. Mas para obter o valor numérico se fará uso da Tabela 1 que traz a variação de N_z , sendo que para este parâmetro aplica-se uma variação de 1 a 5, com os valores das constantes que serão aplicadas na implementação do algoritmo. Portanto estes valores tabelados não podem ser alterados, pois através deles é que se obtêm os resultados necessários. Muitos estudos têm mostrado que o erro de truncamento pode ser ignorado quando N=5 [12] usando a Tabela 1 para os valores correspondentes na mesma. Este algoritmo tem um aprofundamento em [13] onde explicita os valores entre s e t para os cálculos. O algoritmo de Stehfest usa $(j \ln 2)/t$ para calcular s, como mostrado em (11) e no algoritmo de Zakian em (12) é substituído por α_j/t .

Tabela 1: Os valores de α_j e de k_j para o algoritmo de Zakian

j	$lpha_j$	k_{j}
1	12.83767675 + i 1.666063445	-36902.08210 + i 196990.4257
2	12.22613209 + i 5.012718792	61277.02524 - i 95408.62551
3	10.93430308 + i 8.409673116	-28916.56288 + i 18169.18531
4	8.776434715 + i 11.92185389	4655.361138 - i 1.901528642
5	5.225453361 + i 15.72952905	-118.7414011 - i 141.3036911

3 Análise de desempenho dos algoritmos apresentados

Avaliou-se os três métodos da inversão numérica de transformada de Laplace e seus respectivos parâmetros livres mostrados na seção anterior. Os algoritmos foram implementados em Scilab, e é mostrado o desempenho dos métodos com a apresentação dos resultados através de gráficos e numericamente.

Utilizou-se a função $f(t)=e^{-t}$ cuja a transformada de Laplace é F(s)=1/(s+1) para testar e comparar os métodos. Na Tabela 2 são apresentados os resultados para o algoritmo Talbot Fixo de Abate e Valkó, na Tabela 3 estão os resultados do algoritmo de Stehfest e na Tabela 4 estão os resultados do algoritmo de Zakian.

Por fim a construção gráfica desta função no software matemático Scilab, apresentada na Figura 1, mostra o desempenho destes métodos. Nota-se, então que os métodos apresentam uma boa aproximação da solução da função estudada.

No algoritmo de Stehfest, comparado com os outros métodos apresentados no decorrer deste trabalho, notou-se que o parâmetro livre N deste algoritmo para a obtenção de uma solução aproximada precisa estar num intervalo de 10 a 14. Ao passar deste intervalo o resultado se afasta da solução real como pode-se notar na Tabela 3. Apresenta-se a aproximação da função através deste método para os valores N=8 e N=16 que estão fora do intervalo estabelecido e N=12, sendo a solução mais aproximada. E assim se observa que N=12 é a solução aproximada da exata.

Tabela 2: Comparação de $f(t) = e^{-t}$ versus $f_{FT}(t)$ para M = 20, 40, 80. $\overline{M=20}$ f(t)M = 40M = 80t0.3678794411 0.3678794411 1 0.36787944386 0.3678794411 2 0.1353352832 0.13533528557 0.1353352832 0.1353352832 3 0.0497870683 0.04978706579 0.0497870683 0.0497870683 4 0.0183156388 0.01831563887 0.0183156388 0.0183156388 5 0.0067379469 0.00673794764 0.0067379469 0.0067379469 0.0024787521 0.00247875288 0.0024787521 0.0024787521 6 7 0.0009118819 0.00091188072 0.0009118819 0.0009118819 8 0.0003354626 0.00033545944 0.0003354626 0.0003354626 9 0.0001234098 0.00012340948 0.0001234098 0.0001234098 0.0000453999 0.00004540296 0.0000453999 0.0000453999 10

Tal	bela 3: Comparaç	can de $f(t) = e^{-t}$	versus $f_S(t)$ para	N = 8, 12, 16.
t	f(t)	N = 8	N = 12	N = 16
	0.06=0=0.1111	0.01100100=11	0.06.00001111	

	* \ /			
1	0.3678794411	0.34198432741	0.3678794411	0.3732198773
2	0.1353352832	0.12954321542	0.1353352832	0.1353352832
3	0.0497870683	0.03985643222	0.0497870683	0.0534455676
4	0.0183156388	0.02123348756	0.0183156388	0.0297554021
5	0.0067379469	0.00509874532	0.0067379469	0.0073477352
6	0.0024787521	0.00243345677	0.0024787521	0.0032122435
7	0.0009118819	0.00089987634	0.0009118819	0.0009883223
8	0.0003354626	0.00023345666	0.0003354626	0.0004098875
9	0.0001234098	0.00012340948	0.0001234098	0.0001323445
10	0.0000453999	0.00003540296	0.0000453999	0.0000543225

Tabela 4: Comparação de $f(t) = e^{-t}$ versus $f_Z(t)$ para $N_z = 1, 3, 5$.

			72(°) F 2 = 7 = 7 = 7	
\overline{t}	f(t)	$N_z = 1$	$N_z = 3$	$N_z = 5$
1	0.3678794411	0.42123343231	0.6543234543	0.3678794411
2	0.1353352832	0.23435567801	0.2134568723	0.1353352832
3	0.0497870683	0.05978787679	0.0697843283	0.0497870683
4	0.0183156388	0.03234354666	0.0487656556	0.0183156388
5	0.0067379469	0.00673794764	0.0067379469	0.0067379469
6	0.0024787521	0.00324345563	0.0025164544	0.0024787521
7	0.0009118819	0.00124434654	0.0009118819	0.0009118819
8	0.0003354626	0.00043237687	0.0004432337	0.0003354626
9	0.0001234098	0.00023112215	0.0001433232	0.0001234098
10	0.0000453999	0.00005434433	0.0000665444	0.0000453999

O algoritmo de Zakian necessita usar uma tabela que estipula os valores de k_j e α_j , com o parâmetro variando de 1 a 5. E como se pode notar através do gráfico, que se encontra na Figura 1, que este algoritmo também se aproxima do resultado da função f(t), mesmo não sendo tão eficiente como os outros métodos. Numericamente existe uma diferença bem considerável da solução aproximada para a solução exata da função.

Para solidificar o estudo sobre estes métodos, aplicou-se em outra função, que corresponde a $f(t) = \cos t$, cuja a transformada é $F(s) = s/(s^2+1)$ e as mesmas análises feitas no estudo da função anterior foram realizadas, portanto para esta também.

No algoritmo de Talbot Fixo, foram utilizados os parâmetros livres iguais a M=20, 40 e 80 apresentados

Figura 1: Comparação de $f(t) = e^{-t}$ versus $f_{FT}(t)$ para M = 40, $f_S(t)$ para N = 12, e $f_Z(t)$ para $N_z = 5$.

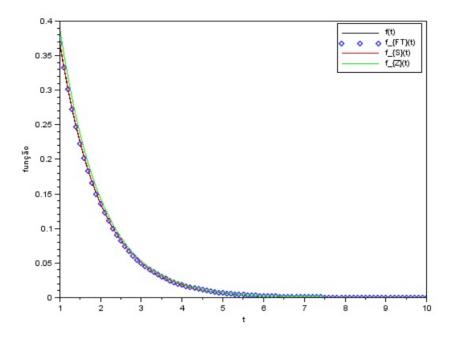


Tabela 5: Comparação da eficiência dos três algoritmos para $f(t) = e^{-t}$.

	1 3		1 0 \ /
	Parâmetro	erro relativo (máximo)	tempo computacional (s)
	M = 20	10^{-9}	0.018
Algoritmo FT	M = 40	10^{-13}	0.021
	M = 80	10^{-7}	0.032
	N = 8	10^{-6}	0.036
Stehfest	N = 12	10^{-8}	0.041
	N = 16	10^{-4}	0.025
	$N_z = 1$	10^{-1}	0.016
Zakian	$N_z = 3$	10^{-1}	0.022
	$N_z = 5$	10^{-9}	0.026

na Tabela 6. E como se pode notar, através da tabela a seguir, o parâmetro livre M para o algoritmo de Talbot está entre 20 e 70, pois quando o parâmetro ultrapassa o limite estabelecido pelo algoritmo a solução apresenta uma diferença significativa da solução real.

Utilizando a mesma função, aplicou-se o algoritmo de Stehfest, sendo os resultados apresentados na Tabela 7, com parâmetro livre igual a $N=8,\,12$ e $16,\,$ e assim se observa que está dentro do intervalo estipulado. E nota-se que a solução do método é praticamente a mesma da f(t).

Por último o algoritmo de Zakian, Tabela 8, apresenta uma solução não tão precisa para a última função analisada, onde se pode ver na Figura 2. Com os coeficientes tabelados muitas das funções ao aplicar este método não apresentaram bons resultados como os demais métodos. Isso levando em conta outras funções estudadas que pode ser encontradas em qualquer tabela de transformada de Laplace.

Note que nos gráficos apresentados, os índices estão denotados assim: f(t), f_{FT} (t), $f_{S}(t)$, $f_{Z}(t)$ que respectivamente se refere à função exata, à inversa numérica do algoritmo de Talbot Fixo, à inversa do algoritmo

Tabela 6: Comparação de $f(t) = \cos t$ versus $f_{FT}(t)$ para M = 20, 40, 80.

	1 3	• ()	V \ / I	, ,
\overline{t}	f(t)	M = 20	M = 40	M = 80
1	0.54030233212	0.54030233655	0.54030233212	0.54030233221
2	- 0.41614686543	- 0.41614686877	- 0.41614686543	- 0.41614686532
3	- 0.98999253421	- 0.98999253425	- 0.98999253421	- 0.98999253422
4	- 0.65364362321	- 0.65364362366	- 0.65364362321	- 0.65364362325
5	0.28366222334	0.28366222543	0.28366222334	0.28366222343
6	0.96017034343	0.96017034377	0.96017034343	0.96017034346
7	0.75390236553	0.75390236654	0.75390236553	0.75390236554
8	- 0.14550000874	- 0.14550000832	- 0.14550000874	- 0.14550000885
9	- 0.91113032343	- 0.91113032655	- 0.91113032343	- 0.91113032354
10	- 0.83907152334	- 0.83907152554	- 0.83907152334	- 0.83907152398

Tabela 7: Comparação de $f(t) = \cos t$ versus $f_S(t)$ para N = 8, 12, 16.

\overline{t}	f(t)	N = 8	N = 12	N = 16
1	0.54030233212	0.54030243265	0.54030233212	0.54030329876
2	- 0.41614686543	- 0.41614665432	- 0.41614686543	- 0.41614432004
3	- 0.98999253421	- 0.98999254325	- 0.98999253421	- 0.98999232456
4	- 0.65364362321	- 0.65364365553	- 0.65364362321	- 0.65364387665
5	0.28366222334	0.28366222577	0.28366222334	0.28366222765
6	0.96017034343	0.96017654432	0.96017034343	0.96017432236
7	0.75390236553	0.75390766537	0.75390236553	0.75390265449
8	- 0.14550000874	- 0.14550000654	- 0.14550000874	- 0.14550087543
9	- 0.91113032343	- 0.91113032763	- 0.91113032343	- 0.91113543998
10	- 0.83907152334	- 0.83907124559	- 0.83907152334	- 0.83907143667

Tabela 8: Comparação de $f(t) = \cos t$ versus $f_Z(t)$ para $N_z = 1, 3, 5$.

	1	3 0 ()	7 Z \ / I	~ / /
\overline{t}	f(t)	$N_z = 1$	$N_z = 3$	$N_z = 5$
1	0.54030233212	0.44030533436	0.44030233233	0.54030233212
2	- 0.41614686543	- 0.31614676541	- 0.51614684331	- 0.41614686543
3	- 0.98999253421	- 1.13435678932	- 0.91232447890	- 0.98999253421
4	- 0.65364362321	- 0.55364362547	- 0.65364362321	- 0.75365879986
5	0.28366222334	0.38366222332	0.28366222334	0.38366225446
6	0.96017034343	1.14017034343	1.14017034321	0.96017034343
7	0.75390236553	0.85390236557	0.81119023633	0.75390236553
8	- 0.14550000874	- 0.24552133456	- 0.21344537892	- 0.14550000874
9	- 0.91113032343	- 0.82345334343	- 0.82345543398	- 0.91113032343
10	- 0.83907152334	- 0.92334297865	- 0.92330988745	- 0.83907152334

de Stehfest e por último ao algoritmo de Zakian.

Os algoritmos apresentam um desempenho satisfatório em relação às funções analisadas. O que se notou quando se trabalhou cada método, que o parâmetro livre possui um intervalo de convergência que faz com que o algoritmo se aproxime da solução desejada. O algoritmo de Talbot Fixo é um algoritmo fácil de implementação, mas que conforme o aumento do parâmetro livre M, este perde a consistência, ficando o parâmetro entre 20 a 70.

O algoritmo de Stehfest não é tão simples de implementar do que outros, sendo que este usa valores reais para s e só precisa estimar um parâmetro livre, sendo que o parâmetro livre N só pode variar de 10 a 14, passando deste intervalo os resultados não são tão precisos.

Figura 2: Desempenho dos algoritmos na função $f(t) = \cos t$.

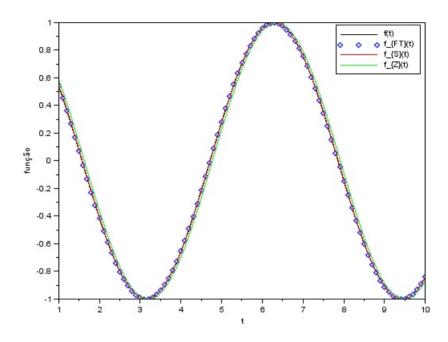


Tabela 9: Tabela comparando os algoritmos e sua eficiência através da $f(t) = \cos t$.

		C	3 ()
	Parâmetro	erro relativo (máximo)	tempo computacional (s)
	M = 20	10^{-10}	0.019
Algoritmo FT	M = 40	10^{-12}	0.022
	M = 80	10^{-8}	0.029
	N = 8	10^{-6}	0.044
Stehfest	N = 12	10^{-8}	0.042
	N = 16	10^{-4}	0.029
	$N_z = 1$	10^{-1}	0.019
Zakian	$N_z = 3$	10^{-1}	0.020
	$N_z = 5$	10^{-9}	0.023

O algoritmo de Zakian não é tão eficiente quanto aos demais algoritmos, mostrando graficamente uma solução com um afastamento maior numericamente da solução real. Isso, sendo com relação aos demais algoritmos estudados neste artigo.

Quando se menciona menor custo computacional está se atribuindo a precisão desejada, a capacidade do método em conduzir aos resultados desejados (velocidade de convergência) e esforço computacional utilizado (tempo de processamento, economia de memória necessária para a resolução). Como se pode ver na Tabela 5 e 9.

4 Conclusões

Neste artigo, investigou-se três métodos de inversão numérica de transformada de Laplace e a eficiência de seus algoritmos. O desenvolvimento dos algoritmos não constam no artigo. Essencialmente o algoritmo de cada método está definido nas Equações (9),(10) e (12). Sendo explicitado os resultados de algumas funções graficamente. O que pode notar é que o algoritmo de Talbot Fixo converge mais rápido, tendo uma precisão mais

eficiente do que os demais.

Com este estudo percebe-se que para obter a inversão numérica da transformada de Laplace é preciso ter acesso aos valores da transformada em certos pontos do plano complexo. Supondo conhecer a expressão analítica da transformada a obter a solução e calcular valores aproximados da função original e pontos igualmente espaçados dos intervalos pelos algoritmos aqui estudados.

Como mencionado acima, existem muitos problemas cuja solução pode ser encontrada em termos da transformada de Laplace. Contudo, se torna bastante complicado o processo de inversão desta transformada através de técnicas de análises complexas. Ou melhor, a dificuldade principal em aplicar-se a técnica da transformada de Laplace.

Ao realizar este estudo, pode se perceber que os métodos numéricos têm sido desenvolvidos já que na maioria dos casos os métodos analíticos são insuficientes. Os melhores métodos encontrados na literatura, conhecidos para inversão da transformada de Laplace, são baseados na integração numérica da integral de Bromwich, mostrado em (2).

Portanto é preciso que se observe antes de tudo, certos critérios para avaliação de diferentes técnicas numéricas de inversão, como a aplicação a uma variedade de tipos comuns de problemas de inversão; precisão numérica; tempo relativo de computação; dificuldade de programação e implementação. É importante salientar de que nenhum método é definitivamente superior em todos os critérios. Existem muitos problemas para os quais inverter numericamente a transformada de Laplace requer um tipo especial de método ou pode ser grandemente facilitado pelo uso de um determinado método.

Portanto, as técnicas numéricas de inversão de Laplace não são exata e muitas vezes dependem da escolha de um parâmetro livre que é desconhecido, é vantajoso ou usar mais uma técnica de inversão ou realizar uma experimentação e estudar o efeito do parâmetro livre na solução. O que se observa nas duas funções onde a $f(t) = e^{-t}$ é uma função monótona decrescente com relação a t e a $f(t) = \cos t$ apresenta singularidades.

Como se apresentou no decorrer deste artigo, onde se investigou os algoritmos e sua eficiência para inverter numericamente a transformada de Laplace e seus resultados que são apresentados nas tabelas e gráficos que se encontram na Seção 3 que mostra a performance nas funções estudadas.

Agradecimentos

Agradecemos o projeto CAPES No. 88881.030424/2013-01 pelo suporte financeiro.

Referências

- [1] SCCHIFF, J. The Laplace Transform: Theory and Applications. New York: Springer-Verlag, 1999.
- [2] ABATE, J.; VALKÓ, P. P. Multi-precision Laplace transform inversion. *International Journal for Numerical Methods in Engineering*, Wiley, v. 60, n. 5-7, p. 973–93, 2004. ISSN 1097-0207. Disponível em: http://dx.doi.org/10.1002/nme.995>. Acesso em: 30 abr. 2017.
- [3] STEHFEST, H. Algorithm 368: Numerical inversion of Laplace transforms. *Communications of the ACM*, Association for Computing Machinery, v. 13, n. 1, p. 47–49, 1970. ISSN 0001-0782. Disponível em: http://dx.doi.org/10.1145/361953.361969>. Acesso em: 30 abr. 2017.
- [4] ZAKIAN, V. Numerical inversion of Laplace transform. *Electronic Letters*, Institution of Engineering and Technology, v. 5, n. 6, p. 120–121, 1969. ISSN 0013-5194. Disponível em: http://dx.doi.org/10.1049/el:19690090. Acesso em: 30 abr. 2017.
- [5] SCILAB. *Open source software for numerical computation*. Scilab Enterprises, 2015. Disponível em: http://www.scilab.org. Acesso em: 30 abr. 2017.

- [6] TALBOT, A. The accurate inversion of Laplace transforms. *IMA Journal of Applied Mathematics*, Oxford University Press, v. 23, n. 1, p. 97–120, 1979. ISSN 0272-4960. Disponível em: http://dx.doi.org/10.1093/imamat/23.1.97. Acesso em: 30 abr. 2017.
- [7] STEHFEST, H. Remark on algorithm 368: Numerical inversion of Laplace transforms. *Communications of the ACM*, Association for Computing Machinery, v. 13, n. 10, p. 624, 1970. ISSN 0001-0782. Disponível em: http://dx.doi.org/10.1145/355598.362787. Acesso em: 30 abr. 2017.
- [8] MOENCH, A. F.; OGATA, A. A numerical inversion of the Laplace transform solution to radial dispersion in a porous medium. *Water Resources Research*, Wiley, v. 17, n. 1, p. 250–252, 1981. ISSN 1944-7973. Disponível em: http://dx.doi.org/10.1029/WR017i001p00250>. Acesso em: 30 abr. 2017.
- [9] CHEN, C. S. Analytical and approximate solutions to radial dispersion from an injection well to a geological unit with simultaneous diffusion into adjacent strata. *Water Resources Research*, Wiley, v. 21, n. 8, p. 1069–1076, 1985. ISSN 1944-7973. Disponível em: http://dx.doi.org/10.1029/WR021i008p01069>. Acesso em: 30 abr. 2017.
- [10] WANG, Q. R.; ZHAN, H. B. Radial reactive solute transport in an aquifer-aquitard system. *Advances in Water Resources*, Elsevier, v. 61, p. 51–61, 2013. ISSN 0309-1708. Disponível em: http://dx.doi.org/10.1016/j.advwatres.2013.08.013>. Acesso em: 30 abr. 2017.
- [11] WANG, Q.; ZHAN, H. On different numerical inverse Laplace methods for solute transport problems. *Advances in Water Resources*, Elsevier, v. 75, p. 80–92, 2015. ISSN 0309-1708. Disponível em: http://dx.doi.org/10.1016/j.advwatres.2014.11.001>. Acesso em: 30 abr. 2017.
- [12] HALSTED D., J.; BROWN, D. Zakian's technique for inverting Laplace transforms. *The Chemical Engineering Journal*, Elsevier, v. 3, p. 312–313, 1972. ISSN 1385-8947. Disponível em: http://dx.doi.org/10.1016/0300-9467(72)85037-8. Acesso em: 30 abr. 2017.
- [13] KUHLMAN, K. L. Review of inverse Laplace transform algorithms for Laplace-space numerical approaches. *Numerical Algorithm*, Springer, v. 63, n. 2, p. 339–355, 2013. ISSN 1017-1398. Disponível em: http://dx.doi.org/10.1007/s11075-012-9625-3. Acesso em: 30 abr. 2017.