

ARNALDO CARLOS MÜLLER JUNIOR

INTEGRAÇÃO DA EQUAÇÃO DE MOVIMENTO ATRAVÉS DA  
TRANSFORMADA DE FOURIER COM O USO DE  
PONDERADORES DE ORDEM ELEVADA

Dissertação apresentada à Escola  
de Engenharia de São Carlos da  
Universidade de São Paulo, como  
parte dos requisitos para a  
obtenção do Título de Mestre em  
Engenharia de Estruturas.

Orientador: Prof. Dr. José Elias Laier

São Carlos

2003

Dedico este trabalho aos meus pais, pela sua educação, afeto e dedicação, e a toda minha família, pelo seu encorajamento, incentivo e compreensão.

*“O temor do Senhor é o princípio do conhecimento e da sabedoria.”*

Provérbios 9.10

## RESUMO

MULLER JR., A. C. (2003) *Integração da Equação de Movimento Mediante Transformada de Fourier com Quadraturas de Ordem Elevada*. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2003.

Baseando-se em um sistema com um grau de liberdade, é apresentada neste trabalho a equação de movimento, bem como a sua resolução através das Transformadas de Fourier e da Transformada Rápida de Fourier (FFT). Através da análise da forma como são feitas as integrações nas transformadas, foram estudados e aplicados os ponderadores de Newton-Cotes na resolução da equação de movimento, de forma a aumentar substancialmente a precisão dos resultados em comparação com a forma convencional da Transformada de Fourier.

Palavras-chave: FFT, Fourier, Ponderadores, Equação do Movimento, Dinâmica das Estruturas.

## ABSTRACT

MULLER JR, A. C. (2003) *Integration of the Equation of Motion using Fourier Transform with High Order Quadratures*. M.Sc. Dissertation – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2003.

Based on a single degree of freedom model, this work shows the equation of motion, as well as its solution with the Fourier Transform and the Fast Fourier Transform (FFT). Through the analysis of the methods used in the Fourier Integral, the Newton-Cotes quadratures formulas were studied and applied for the solving of the equation of motion, in order to substantially increase the precision of the results in comparison to the usual Fourier Transform.

Keywords: FFT, Fourier, Quadratures, Equation of Motion, Dynamics of Structures.

## SUMÁRIO

1. INTRODUÇÃO	7
2. EQUAÇÃO DE MOVIMENTO	11
2.1. Introdução	11
2.2. Força de Inércia	12
2.3. Forças Elásticas	12
2.4. Forças de Amortecimento	14
2.5. Equação de Movimento	16
3. EQUAÇÃO DE DUHAMEL	18
4. TRANSFORMADA DE FOURIER	26
4.1. Séries de Fourier	26
4.2. Transformada de Fourier	27
4.3. Aplicação na equação do movimento	30
5. TRANSFORMADA RÁPIDA DE FOURIER (FFT)	32
5.1. Introdução	32
5.2. Transformada numérica de Fourier	33
5.3. Algoritmo da FFT de Hall	35
6. QUADRATURAS DE ORDEM ELEVADA	47
7. EXEMPLOS DE APLICAÇÃO	55
7.1. Introdução	55
7.2. Carregamento retangular	55
7.2. Carregamento Triangular	67
7.3 Carregamento Cossenoidal	77
8. CONCLUSÃO	88
9. BIBLIOGRAFIA	91
ANEXO A – Algoritmos	93

## CAPÍTULO I

### **INTRODUÇÃO**

O objetivo primeiro deste trabalho é obter os deslocamentos de uma estrutura solicitada por um carregamento dinâmico. Existe uma diferença fundamental entre a resposta de uma estrutura a um carregamento dinâmico em comparação a um carregamento estático. Quando se considera uma solicitação constante, o equilíbrio de forças necessário para a obtenção dos esforços internos da estrutura e de seus deslocamentos é feito de maneira simples e direta. Para estruturas solicitadas por carregamentos dinâmicos, infelizmente a situação é outra.

Considera-se como *dinâmico*, uma grandeza variável ao longo do tempo. Carregamentos dinâmicos podem variar tanto de intensidade, como de direção e posição. Tal variação provoca, na estrutura solicitada, deslocamentos igualmente variáveis no tempo – deslocamentos dinâmicos.

Os carregamentos podem ser classificados segundo duas categorias principais. Quando todos os valores do carregamento podem ser determinados dentro de um intervalo de tempo especificado, a solução pode também ser completamente determinada (ainda que não de forma analítica), o que leva a uma análise dita *determinística* do sistema.

Para carregamentos ditos aleatórios e que não podem ser determinados para qualquer tempo dentro de um intervalo especificado,

a solução é tida *não determinística*, pois assim como o carregamento se torna uma análise estatística, assim também é a resposta.

Dentro da análise determinística, dois tipos de carregamentos podem ser considerados:

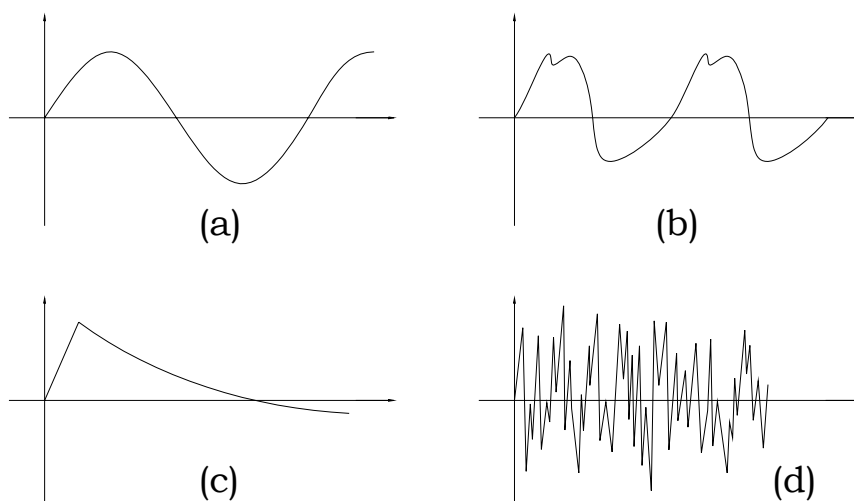


Figura 1.1 – Exemplos de carregamentos

Na figura 1.1, os carregamentos (a) e (b) são periódicos, ou seja, se repetem em ciclos. Podem representar carregamentos simples como o da figura 1.1a, que indica a vibração de um motor sobre uma estrutura, ou podem ser carregamentos complexos, descritos apenas através da Série de Fourier, como a figura 1.1b, indicativa da força de propulsão de uma embarcação.

Os carregamentos explicitados nas figuras 1.1c e 1.1d representam carregamentos *não periódicos*, ou seja, são carregamentos que ocorrem apenas uma vez sobre a estrutura, sem repetição. Carregamentos *não periódicos* típicos são uma explosão (figura 1.1c) e um terremoto (figura 1.1d).

A maneira que a estrutura reage a um carregamento dinâmico, seja ele periódico ou não, também deve ser considerada. Caso o deslocamento de uma estrutura aumente ou diminua de modo proporcional à solicitação, o sistema é dito *linear*. No caso da resposta não variar de modo diretamente proporcional ao carregamento, como



por exemplo, uma resposta em seno ou cosseno, o sistema é dito *não linear*.

Quando o carregamento pode ser descrito através de uma equação simples, o deslocamento do sistema, obtido através da equação do movimento, pode ser obtido analiticamente, usando as técnicas convencionais para resolução de equações diferenciais. Para carregamentos mais complexos, no entanto, não é possível obter-se uma resposta analítica para os deslocamentos. Neste caso, o uso de métodos que resolvam a equação do movimento numericamente se torna necessário. Dos vários métodos disponíveis para resolução numérica da equação do movimento, neste trabalho está descrita a resolução através da Transformada de Fourier. Ou, para o caso de carregamentos discretos, através da Transformada Rápida de Fourier (FFT), um método largamente utilizado nos mais diversos ramos da engenharia, em especial da engenharia elétrica.

A resolução através da FFT traz, no entanto, uma dificuldade cuja mitigação é o objetivo principal deste trabalho. De forma a obter resultados satisfatórios, é freqüentemente necessário utilizar uma discretização do carregamento com um número muito grande de pontos. Quanto maior este número, mais precisa é a solução. Porém, mais demorado é o processamento deste conjunto de pontos, já que são necessárias duas transformadas para a obtenção da solução em deslocamento da estrutura do domínio do tempo. Neste trabalho será feita uma nova consideração sobre a forma como é feito o cálculo da integral na FFT, de forma a, com a utilização de ponderadores de ordem elevada – particularmente os ponderadores de Newton-Cotes – aumentar a precisão da transformada sem que seja necessário aumentar o número de pontos de sua discretização.

Por último, vale ressaltar que este trabalho, além de considerar na resolução da equação do movimento os sistemas como lineares e determinísticos, todas as equações foram desenvolvidas para sistemas com um grau de liberdade (SDOF). Para sistemas com mais de um grau

de liberdade, a resposta pode ser obtida através do princípio da superposição dos efeitos.

## CAPÍTULO II

### EQUAÇÃO DE MOVIMENTO

#### 2.1. Introdução

O mais simples sistema dinâmico possui apenas um grau de liberdade e o seu comportamento é facilmente obtido através do equilíbrio de todas as forças atuantes sobre ele. A figura 2.1 representa esquematicamente tal sistema, onde  $k$  representa o coeficiente de mola, ou a rigidez do conjunto,  $c$  é o coeficiente de amortecimento,  $m$  é a massa e  $p(t)$  a sollicitação atuante ao longo do tempo. O corpo de massa  $m$ , preso a rolamentos ideais, é suposto deslocar-se livremente (sem atrito) ao longo da direção  $x$ .

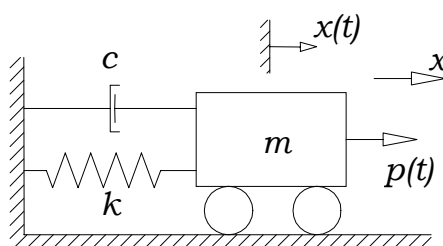


Figura 2.1 – Sistema com um grau de liberdade

Apresenta-se no que se segue uma rápida discussão sobre as ações em jogo no equilíbrio em questão.

## 2.2. Força de Inércia

A segunda lei de Newton prescreve que a ação aplicada sobre um corpo é diretamente proporcional à variação da quantidade de movimento do corpo; ou seja, em termos matemáticos:

$$p(t) = \frac{d}{dt} \left( m \frac{dx}{dt} \right) \quad (2.1)$$

onde emprega-se a notação clássica de derivação no tempo.

Na grande maioria dos problemas da dinâmica das estruturas, a massa é tida como constante e, portanto, a equação (2.1) pode ser reescrita na sua forma mais usual:

$$p(t) = m \frac{d^2 x}{dt^2} = m \ddot{x}(t) \quad (2.2)$$

ou ainda:

$$p(t) - m \ddot{x}(t) = 0 \quad (2.3)$$

onde outra notação também clássica de derivadas no tempo segundo pontos superiores é também empregada.

Em (2.3) o termo  $m \ddot{x}(t)$  pode ser interpretado como a *força de inércia*, dado que consiste na resistência ao movimento do corpo.

Conhecido como *Princípio de d'Alembert*, esta nova escrita também representa uma maneira de se formular o equilíbrio de todas as forças atuantes sobre o corpo. O problema de resposta dinâmica se torna assim equivalente, então, a um problema estático envolvendo equilíbrio de forças considerando-se a força de inércia.

## 2.3. Forças Elásticas

Um corpo elástico, sujeito a uma força que o deforme, desenvolve forças que tendem a restaurar o estado original do corpo. Essas forças, denominadas *forças de restauração elástica*, ocorrem tanto para

carregamentos estáticos quanto para dinâmicos. A maneira mais simples de representar as forças elásticas é através do uso ilustrativo de uma mola (figura 2.2a). Nela, a força restauradora elástica é proporcional à deformação aplicada, dentro de certos intervalos de alongamento (figura 2.2b). Para pequenos intervalos de alongamento (deformação), a resposta é elástico-linear; e, a partir de determinados valores de deformação, a resposta passa a ser não linear, devido ao surgimento de deformações plásticas na mola, que a impossibilitam de restaurar sua condição original.

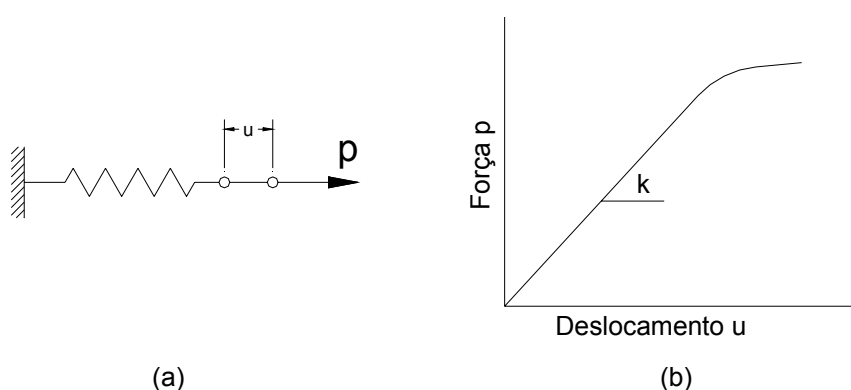


Figura 2.2 - Relação entre deslocamento e força em uma mola

A relação que descreve a força na mola para um deslocamento aplicado pode ser obtida, no intervalo linear, pelo produto entre o deslocamento e a constante de rigidez da mola  $k$  (vide fig. 2.2), ou seja:

$$p = kx \quad (2.4)$$

A constante  $k$  pode ser interpretada como a força necessária para se alcançar um deslocamento unitário. Um raciocínio similar pode ser perfeitamente aplicado a estruturas. Por exemplo, considere-se a barra da figura 2.3, supostamente feita com material de módulo de elasticidade  $E$ , comprimento  $L$  e seção transversal com inércia  $I$ . Neste caso, o deslocamento  $x$ , relacionado à força aplicada  $P$ , obtido pela teoria técnica da flexão resulta (vide fig. 2.3):

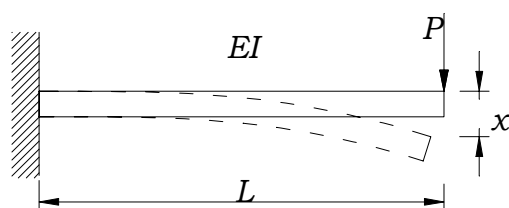


Figura 2.3 – Força elástica em uma barra

$$x = \frac{PL^3}{3EI} \quad (2.5)$$

Por analogia com o expresso em (2.4), percebe-se, pois, facilmente que a constante de mola  $k$  para esta estrutura é dada por:

$$k = \frac{3EI}{L^3} \quad (2.6)$$

Para carregamentos dinâmicos, a única modificação, em relação à equação (2.4), consiste na consideração de funções variáveis no tempo, tanto para a força quanto para a velocidade, resultando-se assim na seguinte relação:

$$F_K(t) = kx(t) \quad (2.7)$$

devendo-se ressaltar tratar-se também de uma força que se opõe ao movimento.

## 2.4. Forças de Amortecimento

Tendo-se em conta estudos puramente no campo da teoria, pode-se facilmente negligenciar o atrito em todas as suas formas. Assim sendo, os movimentos podem eventualmente serem considerados permanentes no tempo. Porém, nos casos reais, a consideração do atrito ou de outras formas de amortecimento, nem sempre podem ser desprezadas, mesmo tendo-se em conta que o amortecimento freqüentemente se revela com uma ordem de grandeza muito pequena

quando comparada às forças de inércia e elásticas. Muitos são os casos em que a consideração de amortecimento nulo pode ser aceita. Em problemas gerais, no entanto, a consideração de sistemas de amortecimento é imprescindível.

No problema de um corpo se movendo ao longo de um eixo, como o da figura 2.1, o amortecimento pode aparecer sob a forma de atrito com o ar ou nos rolamentos do corpo. Em pequenas velocidades, o deslocamento de um corpo através de um fluido como o ar provoca uma força de amortecimento proporcional à sua velocidade, ou seja:

$$F_c(t) = c\dot{x}(t) \quad (2.8)$$

sendo que o fator de proporção  $c$  é chamado *coeficiente de amortecimento*.

Para velocidades maiores em meio fluido, a força de amortecimento torna-se proporcional ao quadrado da velocidade (figura 2.4b). No problema ilustrado na figura 2.1, e nos problemas subseqüentes, considera-se a força de amortecimento como prescrita no modelo de amortecimento viscoso, à exemplo do mostrado na figura 2.4a.

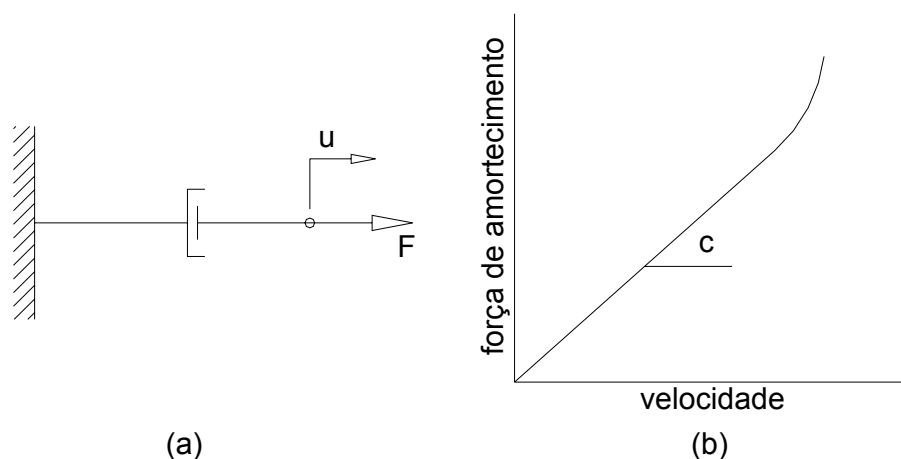


Figura 2.4 Modelo de Amortecimento viscoso

## 2.5. Equação de Movimento

Com todas as forças atuantes sobre o corpo claramente definidas, retorna-se agora ao problema elementar da dinâmica, mostrado novamente na figura 2.5.

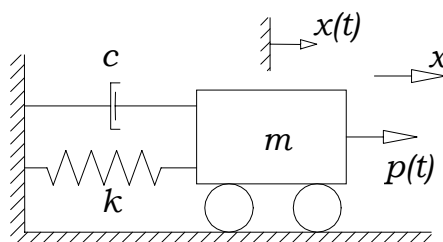


Figura 2.5 – Sistema com um grau de liberdade

Explicitando-se as forças atuantes, e aplicando-se o Princípio de d'Alembert para equilíbrio do conjunto, obtém-se finalmente a equação do movimento:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = p(t) \quad (2.9)$$

A consideração das forças atuantes no sistema na sua forma vetorial é bastante simples neste problema. Porém, em sistemas mais complexos, tal consideração freqüentemente não pode ser aplicada de maneira tão direta. Uma segunda maneira eficiente de se obter a equação de movimento, tem por base a utilização do *Princípio dos Trabalhos Virtuais* que, ao envolver apenas grandezas escalares como trabalho (energia), se mostra nesse aspecto possuir inegáveis vantagens.

Com efeito, exemplicando-se com o problema considerado, seja um deslocamento virtual  $\delta x$  aplicado ao conjunto. O Princípio dos Trabalhos Virtuais prescreve que o trabalho total realizado pelas forças externas e internas do sistema, para este deslocamento, deve ser nulo, ou seja:



$$F_I(t)\delta x + F_D(t)\delta x + F_K(t)\delta x - p(t)\delta x = 0 \quad (2.10)$$

Por outro lado, explicitando-se as forças indicadas e efetuando-se a fatoração tem-se:

$$(m\ddot{x}(t) + c\dot{x}(t) + kx(t) - p(t))\delta x = 0 \quad (2.11)$$

Por fim, como o deslocamento  $\delta x$  é arbitrário, chega-se novamente à equação do movimento:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = p(t) \quad (2.12)$$

encerrando-se o assunto naquilo que interessa.

### CAPITULO III

## EQUAÇÃO DE DUHAMEL

A equação do movimento expressa em (2.9), como sabido, pode ser resolvida analiticamente apenas em certas situações particulares, como carregamentos simplificados, ou mesmo com o desprezo de termos da equação, como o de amortecimento, por exemplo. Para carregamentos genéricos é necessária uma abordagem diferente.

Nesse sentido, considere-se um sistema submetido a uma solicitação arbitrária conforme mostrado na figura 3.1. O carregamento em questão pode ser dividido em uma série de sucessivos carregamentos aplicados segundo pequenos intervalos de tempo (impulsos). Cada impulso elementar provoca uma pequena variação correspondente no deslocamento da estrutura. Como o sistema em consideração é linear, é possível obter a resposta pela soma das respostas individuais de cada impulso.

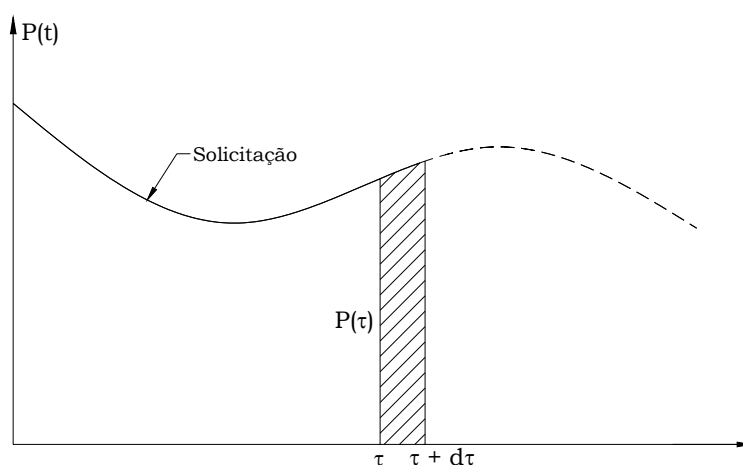


Figura 3.1 – Exemplo de solicitação arbitrária

A chave para a solução reside em se conhecer qual é a influência de cada impulso no comportamento da estrutura, após a aplicação deste. Assim sendo, é interessante ressaltar que, para a determinação da resposta para um tempo  $t$  qualquer, só se deve considerar os impulsos aplicados no tempo  $\tau$  inferior ao tempo considerado ( $\tau < t$ ) – princípio da casualidade, conforme ilustrado na fig. 3.2.

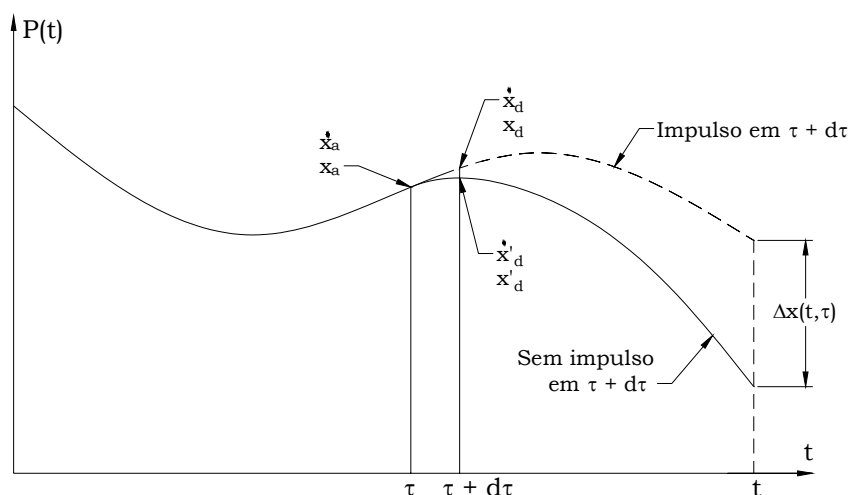


Figura 3.2 – Influência de um impulso em  $t = \tau$

No tempo  $\tau$ , a estrutura possui em deslocamento  $x_a$  e velocidade  $\dot{x}_a$ . No tempo  $\tau + d\tau$ , duas situações devem ser consideradas: Impulso presente em  $\tau + d\tau$ , e neste caso, o sistema possui deslocamento e velocidade definidos como  $x_d$  e  $\dot{x}_d$ , respectivamente. Para esta situação, a aceleração pode ser explicitada, a partir de (2.9), como sendo:

$$\ddot{x} = \frac{P}{m} - \frac{1}{m}(c\dot{x}_a + kx_a) \quad (3.1)$$

lembrando-se que o carregamento é constante no intervalo  $d\tau$ .

A segunda situação é a consideração de impulso nulo no intervalo  $\tau + d\tau$ , quando então o sistema terá deslocamento e velocidade  $x_d'$  e  $\dot{x}_d'$ , respectivamente, e a aceleração é escrita como:

$$\ddot{x}' = -\frac{1}{m}(c\dot{x}_a + kx_a) \quad (3.2)$$

Do cálculo (desenvolvimento das séries de Taylor), pode-se obter os valores para  $x_d$ ,  $x_d'$ ,  $\dot{x}_d$  e  $\dot{x}_d'$ , desprezando-se os infinitésimos de ordem superior:

$$\begin{aligned} x_d &= x_a + \dot{x}_a d\tau \\ x_d' &= x_a + \dot{x}_a d\tau \\ \dot{x}_d &= \dot{x}_a + \ddot{x}_f d\tau \\ \dot{x}_d' &= \dot{x}_a + \ddot{x} d\tau \end{aligned} \quad (3.3)$$

Neste ponto, ainda que não se saiba a variação do deslocamento do sistema no tempo  $t$ , pode-se conhecer a variação do deslocamento e da velocidade no tempo  $\tau + d\tau$ , ou seja:

$$\begin{aligned} \Delta x &= (x_d - x_d') = 0 \\ \Delta \dot{x} &= (\dot{x}_d - \dot{x}_d') = \frac{P}{m} d\tau \end{aligned} \quad (3.4)$$

Por (3.4), nota-se que o impulso não provocou em primeira ordem deslocamento imediato na estrutura, mas provocou variação imediata na velocidade do sistema.

A partir de  $\tau + d\tau$  até o tempo  $t$ , a estrutura entra em movimento livre, cuja solução pode ser obtida a partir da consideração de carregamento nulo na equação de movimento:

$$m\ddot{x} + c\dot{x} + kx = 0 \quad (3.5)$$

A solução de (3.5) é exponencial, na forma:

$$x = Ae^{\beta t} \quad (3.6)$$

A substituição direta de (3.6) e suas derivadas em (3.5), leva a:

$$m(A\beta^2 e^{\beta t}) + c(A\beta e^{\beta t}) + k(Ae^{\beta t}) = 0 \quad (3.7)$$

Dividindo-se (3.7) por  $Ae^{\beta t}$ :

$$m\beta^2 + c\beta + k = 0 \quad (3.8)$$

A equação (3.8) pode ser reescrita de outra maneira, pela consideração de:

$$\left. \begin{aligned} \omega_n &= \sqrt{\frac{k}{m}} \\ \gamma &= \frac{c}{2m\omega_n} \end{aligned} \right\} \quad (3.9)$$

Resultando-se, desta forma em:

$$\beta^2 + 2\gamma\omega_n\beta + \omega_n^2 = 0 \quad (3.10)$$

cujas raízes são:

$$\left. \begin{aligned} \beta_1 &= -\gamma\omega_n + \omega_n\sqrt{\gamma^2 - 1} \\ \beta_2 &= -\gamma\omega_n - \omega_n\sqrt{\gamma^2 - 1} \end{aligned} \right\} \quad (3.11)$$

Dependendo do valor de  $\gamma$ , três cenários para a solução são possíveis:

- $\gamma > 1$ : neste caso, a solução exponencial decai muito rapidamente, eliminando-se toda a vibração do sistema;
- $\gamma = 1$ : esta situação, chamada *amortecimento crítico*, permite calcular o valor do coeficiente de amortecimento crítico, usando-

se para isso a segunda das equação (3.9). Assim como no primeiro caso ( $\gamma > 1$ ), a solução decai muito rapidamente;

$$c_{cr} = 2m\omega_n = 2\sqrt{km} \quad (3.12)$$

- $\gamma < 1$ : a solução possui duas raízes imaginárias e simétricas. É o caso de maior interesse, pois permite uma oscilação de maior duração do sistema até o seu repouso.

Adotando-se, então,  $\gamma < 1$ , e substituindo-se (3.11) em (3.6), a solução resulta em:

$$x(t) = e^{-\gamma\omega_n t} \left[ A_1 e^{\omega_n t \sqrt{\gamma^2 - 1}} + A_2 e^{-\omega_n t \sqrt{\gamma^2 - 1}} \right] \quad (3.13)$$

Efetuando-se as operações trigonométricas pertinentes e tendo-se em conta uma nova notação  $\omega_D$ , resulta:

$$x(t) = e^{-\gamma\omega_n t} [A \operatorname{sen}(\omega_D t) + B \operatorname{cos}(\omega_D t)] \quad (3.14)$$

onde,

$$\omega_D = \omega_n \sqrt{1 - \gamma^2} \quad (3.15)$$

é denominada frequência natural amortecida.

De posse da resposta para movimento livre, é possível retomar as condições iniciais descritas em (3.3) e calcular os termos  $A$  e  $B$  de (3.14):

- Impulso presente em  $\tau + d\tau$ :

$$\begin{aligned} A &= \frac{1}{\omega_D} (\dot{x}_d + \gamma\omega_n x_d) \\ B &= x_d \end{aligned} \quad (3.16)$$

- Sem impulso em  $\tau + d\tau$ :

$$\begin{aligned} A &= \frac{1}{\omega_D} (\dot{x}_d' + \gamma \omega_n x_d') \\ B &= x_d' \end{aligned} \quad (3.17)$$

Para o cálculo das constantes  $A$  e  $B$ , fez-se uso de uma nova variável:

$$\bar{t} = t - (\tau - d\tau) \quad (3.18)$$

Usando-se (3.16) e (3.17) em (3.14), e levando-se em conta a nova variável definida em (3.18), é possível finalmente calcular a variação de deslocamento no sistema no tempo  $t$  devido ao impulso aplicado no tempo  $\tau$ , ou seja:

$$\begin{aligned} \Delta x(t, \tau) &= x(\bar{t}) - x'(\bar{t}) \\ \Delta x(t, \tau) &= e^{-\gamma \omega_n t} \left[ \frac{(\dot{x}_d - \dot{x}_d') + \gamma \omega_n (x_d - x_d')}{\omega_D} \text{sen}(\omega_D \bar{t}) + (x_d - x_d') \cos(\omega_D \bar{t}) \right] \end{aligned} \quad (3.19)$$

Finalmente, usando as equações (3.4) em (3.19), a resposta final é obtida:

$$\Delta x(t, \tau) = \frac{Pd\tau}{m\omega_D} e^{-\gamma \omega_n (t-\tau)} \text{sen}[\omega_D (t - \tau)] \quad (3.20)$$

Para se obter a resposta total  $x(t)$ , deve-se levar em conta o princípio da superposição dos efeitos e calcular a somatória de todos os impulsos até o tempo  $t$ , ou seja:

$$x(t) = \sum \Delta x(t) = \sum \frac{P}{m\omega_D} e^{-\gamma \omega_n (t-\tau)} \text{sen}[\omega_D (t - \tau)] d\tau \quad (3.21)$$

onde já se negligencia o infinitésimo  $d\tau$  no exponencial e no seno. A medida que  $d\tau$  for diminuindo, a resposta se aproxima da resposta exata. No limite  $d\tau \rightarrow 0$ , a somatória se torna uma integral:

$$x(t) = \int_0^t \frac{P(\tau)}{m\omega_D} e^{-\gamma\omega_n(t-\tau)} \text{sen}[\omega_D(t-\tau)] d\tau \quad (3.22)$$

A equação (3.22) é usualmente escrita como:

$$x(t) = \int_0^t P(\tau)h(t-\tau)d\tau \quad (3.23)$$

onde o termo  $h(t-\tau)$  é definido como:

$$h(t-\tau) = \frac{1}{m\omega_D} e^{-\gamma\omega_n(t-\tau)} \text{sen}[\omega_D(t-\tau)] \quad (3.24)$$

A equação (3.23) é conhecida como *Integral de Duhamel*, ou convolução de Duhamel.

O conceito de um carregamento aplicado em um intervalo muito pequeno de tempo (um impulso elementar) pode ser mais bem compreendido através da função *Delta de Dirac*, definida como um carregamento retangular aplicado em um intervalo de tempo  $dt$ , de tal forma que:

$$P(t)dt = 1 \quad (3.25)$$

A integral deste carregamento será sempre unitária, independente do tamanho assumido por  $dt$  (quanto menor o valor de  $dt$ , maior a intensidade de  $P(t)$ ):

$$\int_{-\infty}^{+\infty} P(t)dt = 1, \quad -\infty < t < +\infty \quad (3.26)$$



Usando-se este impulso no tempo  $t = 0$ , é possível calcular a resposta do sistema através da equação (3.14), usando-se como condições iniciais o deslocamento e a velocidade definidos em (3.4):

$$x(t) = \frac{1}{m\omega_D} e^{-\gamma\omega_n t} \text{sen}(\omega_D t) \quad (3.27)$$

Mesmo quando o impulso não é aplicado no tempo  $t = 0$ , a solução (3.27) permanece válida, bastando para isso uma simples mudança de variável. Fazendo:

$$\bar{t} = t - \tau \quad (3.28)$$

obtem-se:

$$x(t - \tau) = \frac{1}{m\omega_D} e^{-\gamma\omega_n(t-\tau)} \text{sen}[\omega_D(t - \tau)] \quad (3.29)$$

Nota-se que a equação (3.29) é idêntica à (3.24). Ou seja, a função  $h(t - \tau)$  pode ser interpretada como a resposta do sistema a um impulso unitário aplicado no tempo  $t = \tau$ .

Outra característica interessante da função  $h(t - \tau)$  é o fato de que, tendo sido obtida através do uso de um impulso unitário, ela representa a influencia das características do sistema (amortecimento, frequência, massa, etc.) no deslocamento total da estrutura.

CAPÍTULO IV  
**TRANSFORMADA DE FOURIER**

**4.1. Séries de Fourier**

Uma das maneiras de representar um carregamento genérico é através de uma série de Fourier, definida como:

$$P(t) = a_0 + \sum_{j=1}^{\infty} a_j \cos(j\omega t) + \sum_{j=1}^{\infty} b_j \text{sen}(j\omega t) \quad (4.1)$$

onde os coeficientes  $a_j$  e  $b_j$  são assumidos reais. A variável  $\omega$ , em (4.1), corresponde à frequência fundamental do carregamento. Considera-se que a função  $P(t)$  é periódica e possui como período:

$$T = \frac{2\pi}{\omega} \quad (4.2)$$

Com maior frequência tem sido mais conveniente expressar a Série de Fourier através de uma somatória de exponenciais complexos, usando para isso as identidades de Euler, ou seja:

$$\begin{aligned} \cos(j\omega t) &= \frac{1}{2}(e^{ij\omega t} + e^{-ij\omega t}) \\ \text{sen}(j\omega t) &= \frac{-i}{2}(e^{ij\omega t} - e^{-ij\omega t}) \end{aligned} \quad (4.3)$$

resultando em (4.1):

$$P(t) = a_0 + \sum_{j=1}^{\infty} \frac{a_j + ib_j}{2} e^{-i\omega jt} + \sum_{j=1}^{\infty} \frac{a_j - ib_j}{2} e^{i\omega jt} \quad (4.4)$$

Os coeficientes  $a_j$  e  $b_j$  podem ser obtidos levando-se em conta as condições de ortogonalidade, ou seja:

$$\begin{aligned} \int_0^T \cos(j\omega t) \cos(k\omega t) dt &= \begin{cases} 0, & j \neq k; \\ T/2, & j = k > 0; \\ T, & j = k = 0; \end{cases} \\ \int_0^T \sen(j\omega t) \sen(k\omega t) dt &= \begin{cases} 0, & j \neq k; \\ T/2, & j = k > 0; \end{cases} \\ \int_0^T \cos(j\omega t) \sen(k\omega t) dt &= 0, \text{ para todo } j, k \end{aligned} \quad (4.5)$$

Desta forma, os coeficientes podem então ser calculados:

$$\begin{aligned} a_0 &= \frac{1}{T} \int_0^T P(t) dt \\ a_j &= \frac{2}{T} \int_0^T P(t) \cos(\omega j t) dt, \quad j > 0; \\ b_j &= \frac{2}{T} \int_0^T P(t) \sen(\omega j t) dt, \quad j > 0; \end{aligned} \quad (4.6)$$

Usando as equações (4.6), a Série de Fourier, já na sua forma exponencial, pode ser escrita como se segue:

$$P(t) = \sum_{j=-\infty}^{+\infty} c_n e^{ij\omega t} \quad (4.7)$$

onde

$$c_n = \frac{1}{T} \int_0^T P(t) e^{-ij\omega t} dt \quad (4.8)$$

## 4.2. Transformada de Fourier

As equações (4.7) e (4.8) assumem que a função  $P(t)$  é periódica. Porém, isso nem sempre é verdade. Para se contornar tal inconveniente,

pode-se assumir que  $P(t)$  se repete após um período definido  $T$ , conforme mostrado na figura 4.1. Quanto maior for o valor de  $T$ , mais precisa será a representação de  $P(t)$ . Quando  $T \rightarrow \infty$ , a série de Fourier fornecerá o resultado exato.

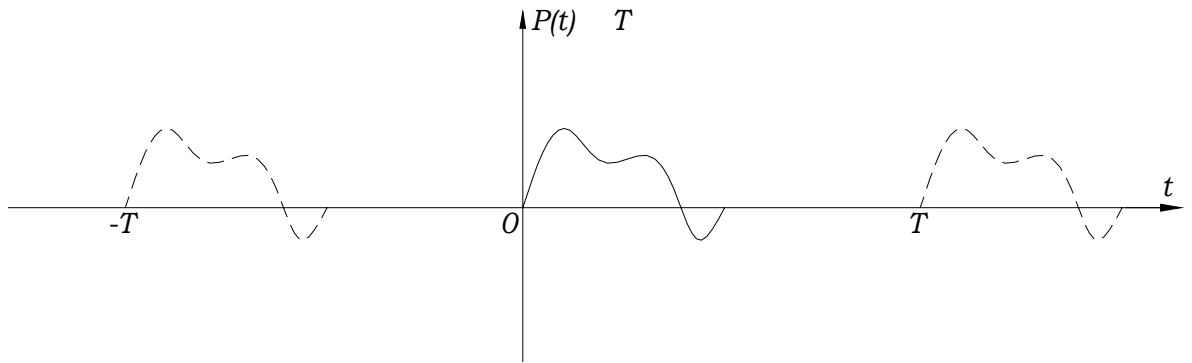


Figura 4.1 – Repetição do carregamento após um período  $T$

Há um problema, no entanto. A medida que  $T$  aumenta, os valores de  $c_n$  diminuem. No limite, com  $T \rightarrow \infty$ , os valores de  $c_n$  serão todos nulos. A solução passa pelo uso de um artifício: Se  $T = \frac{2\pi}{\omega}$ , então:

$$\frac{T\omega}{2\pi} = 1 \quad (4.9)$$

Quando a equação (4.7) é multiplicada por (4.9), torna-se possível aplicar o limite  $T \rightarrow \infty$  para a Série de Fourier, ou seja:

$$P(t) = \frac{1}{2\pi} \sum_{j=-\infty}^{+\infty} Tc_j e^{i\omega t} d\omega = \frac{1}{2\pi} \int_{-\infty}^{+\infty} C(\omega) e^{i\omega t} d\omega \quad (4.10)$$

e

$$C(\omega) = Tc_j = \int_{-\infty}^{+\infty} P(t) e^{-i\omega t} dt \quad (4.11)$$

O conjunto de coeficientes  $C(\omega)$  representa a função  $P(t)$  no domínio da frequência, formando-se o Espectro de Fourier. Quanto maior o valor do período  $T$ , menor será a frequência  $\omega$ , o que significa que o intervalo entre cada coeficiente no domínio da frequência diminui.

As equações (4.11) e (4.10), quando o limite  $T \rightarrow \infty$  é aplicado, representam as equações para transformada de Fourier e transformada inversa de Fourier, respectivamente, e podem ser melhores representadas na sua notação usual:

$$P(\omega) = \int_{-\infty}^{+\infty} P(t)e^{-i\omega t} dt \quad (4.21)$$

e

$$P(t) = \int_{-\infty}^{+\infty} P(\omega)e^{i\omega t} d\omega \quad (4.22)$$

Uma vez que assume-se que o carregamento tem início no tempo  $t = 0$ , pode-se alterar o limite inferior das transformadas de  $-\infty$  para zero, sem que ocorra perda na precisão, ou negligencia a componentes do carregamento ou da transformada.

Uma importante propriedade deve ser observada nas equações da transformada de Fourier para a utilização na resolução da equação do movimento, referente à transformação de derivadas:

$$\dot{P}(\omega) = \int_0^{+\infty} \dot{P}(t)e^{i\omega t} dt \quad (4.23)$$

A resolução de (4.23) se dá usando integração por partes, e resulta:

$$\dot{P}(\omega) = -P(0) + i\omega P(\omega) \quad (4.24)$$

Da mesma forma, é possível calcular a transformação da segunda derivada:

$$\ddot{P}(\omega) = -\dot{P}(0) - i\omega P(0) - \omega^2 P(\omega) \quad (4.25)$$

### 4.3. Aplicação na equação do movimento

As propriedades da transformada de Fourier descritas em (4.24) e (4.25) podem ser usadas diretamente na resolução da equação do movimento:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = p(t)$$

Aplicando-se a transformada de Fourier em ambos os lados da equação, tem-se:

$$m[-\dot{x}(0) - i\omega x(0) - \omega^2 x(\omega)] + c[-x(0) + i\omega x(\omega)] + kx(\omega) = P(\omega) \quad (4.26)$$

Isolando-se o termo  $x(\omega)$  em (4.26), e levando-se em conta condições iniciais nulas ( $x(0) = 0$  e  $\dot{x}(0) = 0$ ), a equação (4.26) resulta:

$$x(\omega) = \frac{P(\omega)}{k - m\omega^2 + i c\omega} \quad (4.27)$$

ou,

$$x(\omega) = P(\omega)H(\omega) \quad (4.28)$$

fazendo-se

$$H(\omega) = \frac{1}{k - m\omega^2 + i c\omega} \quad (4.29)$$

Da mesma forma que o termo  $h(t - \tau)$  descrito em (3.24), o termo  $H(\omega)$ , definido em função das constantes  $k$ ,  $m$  e  $c$ , refere-se apenas às características do sistema. Na verdade, é possível relacionar as funções  $H(\omega)$  e  $h(t - \tau)$ .

Seja a transformada de  $h(t - \tau)$ :

$$H(\omega) = \int_0^{+\infty} \left[ \frac{1}{m\omega_D} e^{-\gamma\omega_n \bar{t}} \text{sen}(\omega_D \bar{t}) \right] e^{-i\omega \bar{t}} d\bar{t} \quad (4.30)$$

onde se considera  $\bar{t} = t - \tau$ , conforme já feito em (3.28).

Aplicando-se a identidade de Euler sobre o seno da equação (4.30), pode-se escrever:

$$H(\omega) = \int_0^{+\infty} \left[ \frac{1}{m\omega_D} e^{-\gamma\omega_n \bar{t}} \left( \frac{e^{i\omega_D \bar{t}} - e^{-i\omega_D \bar{t}}}{2i} \right) \right] e^{-i\omega \bar{t}} d\bar{t} \quad (4.31)$$

Agrupando os exponenciais e resolvendo-se a integral tem-se:

$$H(\omega) = \frac{1}{2im\omega_D} \left[ \frac{e^{(i\omega_D - i\omega - \gamma\omega_n)\bar{t}} + e^{-(i\omega_D + i\omega + \gamma\omega_n)\bar{t}}}{i\omega_D - i\omega - \gamma\omega_n} \right]_0^{\infty} \quad (4.32)$$

que resulta em:

$$H(\omega) = \frac{1}{k - m\omega^2 + i c \omega} \quad (4.33)$$

A equação (4.33) é semelhante à (4.29), o que significa que o termo  $H(\omega)$  nada mais é do que a transformada de Fourier de  $h(t - \tau)$ . Deve-se lembrar que todo o raciocínio aqui exposto baseou-se na hipótese de condições iniciais nulas. Caso isto não se aplique, deve-se acrescentar um termo a mais em (4.27), que corresponde ao termo para vibração livre produzido por tais condições.

## CAPÍTULO V

### **TRANSFORMADA RÁPIDA DE FOURIER (FFT)**

#### **5.1. Introdução**

Quando se desenvolve um algoritmo para a resolução da transformada de Fourier usando-se as equações descritas em (4.21) e (4.22), percebe-se imediatamente a grande quantidade de operações envolvidas no processo. Para se obter a transformada de Fourier de um vetor de ordem  $N$ , são necessárias  $N^2$  operações. Neste capítulo, entende-se por “operação” uma multiplicação entre números complexos, seguida de uma adição igualmente entre números complexos (produto escalar de vetores complexos).

Em 1964, James W. Cooley e John W. Tukey apresentaram um algoritmo baseado na fatoração de uma matriz de ordem  $N \times N$  em  $m$  matrizes esparsas, onde  $m$  é proporcional a  $\log N$ , com apenas dois elementos em cada linha ou coluna. Esta fatoração se mostrou extremamente vantajosa para o cálculo de Séries de Fourier complexas. O algoritmo proposto por Cooley e Tukey ressalta os benefícios operacionais de se escolher um número  $N = 2^m$ , tornando-se o processo mais automatizado. Ele mostra ainda como todo o processo pode ser feito usando apenas um vetor de tamanho  $N$  para armazenar os coeficientes de Fourier calculados. Desta forma, foi desenvolvida a transformada rápida de Fourier (em inglês referida por FFT), em que o número de operações envolvidas foi reduzido de  $N^2$  para  $N \log N$ .



## 5.2. Transformada numérica de Fourier

Antes de podermos analisar como funciona o algoritmo da FFT, é necessário reescrever as equações (4.21) e (4.22) que correspondem às transformadas de Fourier, alterando-as da sua forma analítica, para sua correspondente versão numérica. Para se fazer isto, é necessário o cumprimento de duas etapas: primeiro a formulação das funções contínuas  $P(t)$  e  $P(\omega)$  segundo suas versões discretas, ou seja:

$$P(t) \rightarrow P(j\Delta t) \quad (5.1)$$

e

$$P(\omega) \rightarrow P(k\Delta\omega) \quad (5.2)$$

Esta discretização se dá pela mudança das variáveis contínuas  $t$  e  $\omega$ , em variáveis discretas  $j\Delta t$  e  $k\Delta\omega$ , respectivamente. Os termos  $\Delta t$  e  $\Delta\omega$  correspondem a intervalos discretizados ao longo dos domínios do tempo e da frequência, respectivamente, e os índices  $j$  e  $k$  correspondem às posições das funções ao longo destes domínios discretizados.

A segunda etapa compreende a transformação da integral em uma somatória. Para esta mudança são necessárias algumas considerações:

O período de integração não pode na prática ser considerado como infinito, como é feito na integral. É necessário agora definir um período suficientemente grande, porém finito, ao longo do qual se dará a somatória. Quanto maior este período, maior a precisão dos resultados, com a aproximação da somatória à integral ( $T \rightarrow \infty$ ). Além desta definição, é necessário definir a intensidade da discretização, ou seja, em quantos intervalos ( $N$ ) será dividido o período  $T$ . Da mesma forma, quanto maior o número de intervalos, maior a precisão da somatória. De posse dos valores de  $T$  e  $N$ , é possível então se obter os valores de  $\Delta t$  e  $\Delta\omega$ , como se seguem:

$$\begin{aligned} \Delta t &= \frac{T}{N} \\ \Delta\omega &= \frac{2\pi}{T} \end{aligned} \quad (5.3)$$

Feitas estas considerações, é possível rescrever as equações das Transformadas de Fourier na sua forma numérica, como se segue:

$$P(k\Delta\omega) = \Delta t \sum_{j=0}^{N-1} P(j\Delta t) e^{\frac{-i2\pi jk}{N}} \quad (5.4)$$

$$P(j\Delta t) = \frac{1}{2\pi} \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} P(k\Delta\omega) e^{\frac{i2\pi jk}{N}} \quad (5.5)$$

Na equação (5.4) já foi considerado como limite inferior o valor zero, uma vez que assume-se que o carregamento tem início no tempo  $t = 0$ . Uma característica importante da equação (5.4) refere-se à sua simetria. Considere a resolução de (5.4) para os valores  $k = (N - k)\Delta\omega$  e  $k = (-k\Delta\omega)$ :

$$\begin{aligned} P[(N - k)\Delta\omega] &= \Delta t \sum_{j=0}^{N-1} P(j\Delta t) e^{-2\pi i j(N-k)/N} = \\ &= \Delta t \sum_{j=0}^{N-1} P(j\Delta t) e^{2\pi i jk/N} e^{-2\pi i jN/N} \end{aligned} \quad (5.6)$$

Tendo-se em vista que o termo:

$$e^{-2\pi i j} = 1 \quad (5.7)$$

conforme se verifica da identidade de Euler, para qualquer valor inteiro de  $j$ , a equação (5.6) permite que se escreva:

$$P[(N - k)\Delta\omega] = \Delta t \sum_{j=0}^{N-1} P(j\Delta t) e^{2\pi i jk/N} \quad (5.8)$$

Por outro lado, como:

$$P(-k\Delta\omega) = \Delta t \sum_{j=0}^{N-1} P(j\Delta t) e^{2\pi i jk/N} \quad (5.9)$$

percebe-se que a função  $P(k\Delta\omega)$ , obtida pela equação (5.4), é simétrica sobre o seu termo central, ou seja, os primeiros  $N/2$  valores da função correspondem aos termos positivos de  $\Delta\omega$ , enquanto os termos de  $N/2 + 1$  até  $N - 1$ , correspondem aos termos negativos de  $\Delta\omega$ .

### 5.3. Algoritmo da FFT de Hall

Desde a publicação do algoritmo de Cooley e Tukey, a maior parte das adaptações da FFT têm sido feitas para uso na engenharia elétrica. John F. Hall, no entanto, desenvolveu uma adaptação voltada para a engenharia das estruturas, aplicada especialmente para a resolução da equação do movimento descrita em (2.9). O algoritmo de Hall traz benefícios no armazenamento de vetores e matrizes, e na velocidade de processamento da transformada.

Considere-se o cálculo da transformada numérica de Fourier de um vetor com  $N = 8$ . Da sua definição, pode-se escrever as relações da transformada na forma de um produto matricial como:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & W^8 & W^{10} & W^{12} & W^{14} \\ 1 & W^3 & W^6 & W^9 & W^{12} & W^{15} & W^{18} & W^{21} \\ 1 & W^4 & W^8 & W^{12} & W^{16} & W^{20} & W^{24} & W^{28} \\ 1 & W^5 & W^{10} & W^{15} & W^{20} & W^{25} & W^{30} & W^{35} \\ 1 & W^6 & W^{12} & W^{18} & W^{24} & W^{30} & W^{36} & W^{42} \\ 1 & W^7 & W^{14} & W^{21} & W^{28} & W^{35} & W^{42} & W^{49} \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.10)$$

onde,

$$W = e^{-i2\pi/8} = \cos\left(\frac{2\pi}{8}\right) - i\sin\left(\frac{2\pi}{8}\right) \quad (5.11)$$

e as demais potências de  $W$  são obtidas aplicando-se o expoente sobre o exponencial, ou seja:

$$W^j = e^{-ij2\pi/8} \quad (5.12)$$

A notação  $W$  representa a raiz oitava da unidade, já que  $W^8 = 1$ , e é também um vetor complexo, cujo módulo é unitário e com fase igual a  $-(1/N)$  rotações.

Devido à simetria do vetor transformada, quando o vetor de entrada é um vetor real, pode-se escrever:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(-3\Delta\omega) \\ P(-2\Delta\omega) \\ P(-1\Delta\omega) \end{bmatrix} = \begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} \quad (5.13)$$

Os vetores complexos que intervêm no cálculo da transformada numérica podem ser resumidos como se segue:

$$\begin{aligned} W^0 &= W^8 = W^{16} = W^{24} = 1 \\ W^1 &= W^9 = W^{25} = W^{49} = \cos(\pi/4) - i \sin(\pi/4) \\ W^2 &= W^{10} = W^{18} = W^{42} = -i \\ W^3 &= W^{35} = \cos(3\pi/4) - i \sin(3\pi/4) \\ W^4 &= W^{12} = W^{20} = W^{28} = W^{36} = -1 \\ W^5 &= W^{21} = \cos(5\pi/4) - i \sin(5\pi/4) \\ W^6 &= W^{14} = i \\ W^7 &= W^{15} = \cos(7\pi/4) - i \sin(7\pi/4) \end{aligned} \quad (5.14)$$

Substituindo-se as equações (5.14) em (5.10), obtém-se:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 & W^4 & W^7 & W^2 & W^5 \\ 1 & W^4 & 1 & W^4 & 1 & W^4 & 1 & W^4 \\ 1 & W^5 & W^2 & W^7 & W^4 & W^9 & W^6 & W^{11} \\ 1 & W^6 & W^4 & W^{10} & 1 & W^6 & W^4 & W^{10} \\ 1 & W^7 & W^6 & W^{13} & W^4 & W^{11} & W^2 & W^9 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.15)$$

A matriz (5.15) evidencia a existência de duas propriedades:

### 1ª PROPRIEDADE

As colunas ímpares são simétricas, ou seja, os primeiros  $N/2$  vetores são iguais aos últimos e com a mesma ordenação. Para se perceber essa simetria devem ser consideradas, além das colunas ímpares, as linhas ímpares também, conforme demonstrado em (5.16).

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} \boxed{1} & 1 & \boxed{1} & 1 & \boxed{1} & 1 & \boxed{1} & 1 \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ \boxed{1} & W^2 & \boxed{W^4} & W^6 & \boxed{1} & W^2 & \boxed{W^4} & W^6 \\ 1 & W^3 & W^6 & W^9 & W^4 & W^7 & W^2 & W^5 \\ \boxed{1} & W^4 & \boxed{1} & W^4 & \boxed{1} & W^4 & \boxed{1} & W^4 \\ 1 & W^5 & W^2 & W^7 & W^4 & W^9 & W^6 & W^{11} \\ \boxed{1} & W^6 & \boxed{W^4} & W^{10} & \boxed{1} & W^6 & \boxed{W^4} & W^{10} \\ 1 & W^7 & W^6 & W^{13} & W^4 & W^{11} & W^2 & W^9 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.16)$$

### 2ª PROPRIEDADE

Nas colunas pares, os vetores complexos são exatamente iguais ao produto da coluna anterior pela primeira coluna par. Isto está mostrado em (5.17):

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 & W^4 & W^7 & W^2 & W^5 \\ 1 & W^4 & \boxed{1} & W^4 & \boxed{1} & W^4 & \boxed{1} & W^4 \\ 1 & W^5 & W^2 & W^7 & W^4 & W^9 & W^6 & W^{11} \\ 1 & W^6 & W^4 & W^{10} & \boxed{1} & W^6 & W^4 & W^{10} \\ 1 & W^7 & W^6 & W^{13} & W^4 & W^{11} & W^2 & W^9 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.17)$$

A fim de fazer uso das propriedades 1 e 2, pode-se dividir o vetor de entrada em uma soma de dois vetores, conforme indicado em (5.18):

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(5\Delta\omega) \\ P(6\Delta\omega) \\ P(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} P(0\Delta\omega) \\ 0 \\ P(2\Delta\omega) \\ 0 \\ P(4\Delta\omega) \\ 0 \\ P(6\Delta\omega) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ P(1\Delta\omega) \\ 0 \\ P(3\Delta\omega) \\ 0 \\ P(5\Delta\omega) \\ 0 \\ P(7\Delta\omega) \end{bmatrix} \quad (5.18)$$

Substituindo (5.18) em (5.15), e aplicando-se a propriedade distributiva da multiplicação, é possível reescrever (5.15) em duas parcelas, definidas por (5.19) e (5.20), ou seja:

$$\begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \\ \bar{P}(4\Delta\omega) \\ \bar{P}(5\Delta\omega) \\ \bar{P}(6\Delta\omega) \\ \bar{P}(7\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 & W^4 & W^7 & W^2 & W^5 \\ 1 & W^4 & 1 & W^4 & 1 & W^4 & 1 & W^4 \\ 1 & W^5 & W^2 & W^7 & W^4 & W^9 & W^6 & W^{11} \\ 1 & W^6 & W^4 & W^{10} & 1 & W^6 & W^4 & W^{10} \\ 1 & W^7 & W^6 & W^{13} & W^4 & W^{11} & W^2 & W^9 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ 0 \\ P(2\Delta t) \\ 0 \\ P(4\Delta t) \\ 0 \\ P(6\Delta t) \\ 0 \end{bmatrix} \quad (5.19)$$

e

$$\begin{bmatrix} P'(0\Delta\omega)W^0 \\ P'(1\Delta\omega)W^1 \\ P'(2\Delta\omega)W^2 \\ P'(3\Delta\omega)W^3 \\ P'(4\Delta\omega)W^4 \\ P'(5\Delta\omega)W^5 \\ P'(6\Delta\omega)W^6 \\ P'(7\Delta\omega)W^7 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7 \\ 1 & W^2 & W^4 & W^6 & 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 & W^4 & W^7 & W^2 & W^5 \\ 1 & W^4 & 1 & W^4 & 1 & W^4 & 1 & W^4 \\ 1 & W^5 & W^2 & W^7 & W^4 & W^9 & W^6 & W^{11} \\ 1 & W^6 & W^4 & W^{10} & 1 & W^6 & W^4 & W^{10} \\ 1 & W^7 & W^6 & W^{13} & W^4 & W^{11} & W^2 & W^9 \end{bmatrix} \times \begin{bmatrix} 0 \\ P(1\Delta t) \\ 0 \\ P(3\Delta t) \\ 0 \\ P(5\Delta t) \\ 0 \\ P(7\Delta t) \end{bmatrix} \quad (5.20)$$

Portanto, a transformada numérica passa a ser indicada da seguinte forma:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(-3\Delta\omega) \\ P(-2\Delta\omega) \\ P(-1\Delta\omega) \end{bmatrix} = \begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \\ \bar{P}(4\Delta\omega) \\ \bar{P}(5\Delta\omega) \\ \bar{P}(6\Delta\omega) \\ \bar{P}(7\Delta\omega) \end{bmatrix} + \begin{bmatrix} P'(0\Delta\omega)W^0 \\ P'(1\Delta\omega)W^1 \\ P'(2\Delta\omega)W^2 \\ P'(3\Delta\omega)W^3 \\ P'(4\Delta\omega)W^4 \\ P'(5\Delta\omega)W^5 \\ P'(6\Delta\omega)W^6 \\ P'(7\Delta\omega)W^7 \end{bmatrix} \quad (5.21)$$

Tendo-se em vistas as equações (5.19) e (5.20), conclui-se então que:

$$\begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ P(2\Delta t) \\ P(4\Delta t) \\ P(6\Delta t) \end{bmatrix} \quad (5.22)$$

$$\begin{bmatrix} P'(0\Delta\omega) \\ P'(1\Delta\omega) \\ P'(2\Delta\omega) \\ P'(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} P(1\Delta t) \\ P(3\Delta t) \\ P(5\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.23)$$

Retornando-se as equações (5.22) e (5.23) em (5.21), tem-se pois:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(-3\Delta\omega) \\ P(-2\Delta\omega) \\ P(-1\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^3 \\ 1 & 0 & 0 & 0 & W^4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^5 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^6 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^7 \end{bmatrix} \times \begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(4\Delta\omega) \\ \bar{P}(6\Delta\omega) \\ P'(1\Delta\omega) \\ P'(3\Delta\omega) \\ P'(5\Delta\omega) \\ P'(7\Delta\omega) \end{bmatrix} \quad (5.24)$$

Ao se fazer esta operação, a resolução da transformada numérica passou de uma matriz  $N = 8$  para duas matrizes ((5.22) e (5.23)) com  $N = 4$  cada uma, representando um significativo aumento na velocidade de processamento.

Pode-se aumentar ainda mais a eficiência do algoritmo ao se aplicar as mesmas propriedades aplicadas em (5.15), às matrizes definidas em (5.22) e (5.23), que resultam:

$$\begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \left( \begin{bmatrix} P(0\Delta t) \\ 0 \\ P(4\Delta t) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ P(2\Delta t) \\ 0 \\ P(6\Delta t) \end{bmatrix} \right) \quad (5.25)$$

e

$$\begin{bmatrix} P'(0\Delta\omega) \\ P'(1\Delta\omega) \\ P'(2\Delta\omega) \\ P'(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \left( \begin{bmatrix} P(1\Delta t) \\ 0 \\ P(5\Delta t) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ P(3\Delta t) \\ 0 \\ P(7\Delta t) \end{bmatrix} \right) \quad (5.26)$$

Desmembrando-se (5.25) e (5.26), obtêm-se:

$$\begin{bmatrix} \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \\ \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} P(0\Delta t) \\ 0 \\ P(4\Delta t) \\ 0 \end{bmatrix} \quad (5.27)$$

$$\begin{bmatrix} \ddot{P}(0\Delta\omega)W^0 \\ \ddot{P}(1\Delta\omega)W^2 \\ \ddot{P}(0\Delta\omega)W^4 \\ \ddot{P}(1\Delta\omega)W^6 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} 0 \\ P(2\Delta t) \\ 0 \\ P(6\Delta t) \end{bmatrix}$$

e



$$\begin{aligned}
\begin{bmatrix} P''(0\Delta\omega) \\ P''(1\Delta\omega) \\ P''(0\Delta\omega) \\ P''(1\Delta\omega) \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} P(1\Delta t) \\ 0 \\ P(5\Delta t) \\ 0 \end{bmatrix} \\
\begin{bmatrix} P'''(0\Delta\omega)W^0 \\ P'''(1\Delta\omega)W^2 \\ P'''(0\Delta\omega)W^4 \\ P'''(1\Delta\omega)W^6 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^4 & 1 & W^4 \\ 1 & W^6 & W^4 & W^2 \end{bmatrix} \times \begin{bmatrix} 0 \\ P(3\Delta t) \\ 0 \\ P(7\Delta t) \end{bmatrix}
\end{aligned} \tag{5.28}$$

Note-se que quando são criadas as equações (5.27) e (5.28), a ordem das matrizes foi novamente diminuída. Enquanto as matrizes (5.23) e (5.24) possuíam ordem  $N = 4$ , as equações (5.27) e (5.28) possuem ordem  $N = 2$ .

Utilizando-se (5.27) e (5.28) pode-se escrever:

$$\begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \\ \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \end{bmatrix} + \begin{bmatrix} \ddot{P}(0\Delta\omega)W^0 \\ \ddot{P}(1\Delta\omega)W^2 \\ \ddot{P}(0\Delta\omega)W^4 \\ \ddot{P}(1\Delta\omega)W^6 \end{bmatrix} \tag{5.29}$$

e

$$\begin{bmatrix} P'(0\Delta\omega) \\ P'(1\Delta\omega) \\ P'(2\Delta\omega) \\ P'(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} P''(0\Delta\omega) \\ P''(1\Delta\omega) \\ P''(0\Delta\omega) \\ P''(1\Delta\omega) \end{bmatrix} + \begin{bmatrix} P'''(0\Delta\omega)W^0 \\ P'''(1\Delta\omega)W^2 \\ P'''(0\Delta\omega)W^4 \\ P'''(1\Delta\omega)W^6 \end{bmatrix} \tag{5.30}$$

Reunindo-se as equações (5.27) e (5.28), obtém-se:

$$\begin{bmatrix} \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \\ \ddot{P}(0\Delta\omega) \\ \ddot{P}(1\Delta\omega) \\ P''(0\Delta\omega) \\ P''(1\Delta\omega) \\ P'''(0\Delta\omega) \\ P'''(1\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & W^4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & W^4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & W^4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & W^4 \end{bmatrix} x \begin{bmatrix} P(0\Delta t) \\ P(4\Delta t) \\ P(2\Delta t) \\ P(6\Delta t) \\ P(1\Delta t) \\ P(5\Delta t) \\ P(3\Delta t) \\ P(7\Delta t) \end{bmatrix} \quad (5.31)$$

Simplificando-se as equações (5.29) e (5.30), obtêm-se:

$$\begin{bmatrix} \bar{P}(0\Delta\omega) \\ \bar{P}(1\Delta\omega) \\ \bar{P}(2\Delta\omega) \\ \bar{P}(3\Delta\omega) \\ P'(0\Delta\omega) \\ P'(1\Delta\omega) \\ P'(2\Delta\omega) \\ P'(3\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^2 & 0 & 0 & 0 & 0 \\ 1 & 0 & W^4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & W^4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^6 \end{bmatrix} x \begin{bmatrix} \dot{P}(0\Delta\omega) \\ \dot{P}(1\Delta\omega) \\ \ddot{P}(0\Delta\omega) \\ \ddot{P}(1\Delta\omega) \\ P''(0\Delta\omega) \\ P''(1\Delta\omega) \\ P'''(0\Delta\omega) \\ P'''(1\Delta\omega) \end{bmatrix} \quad (5.32)$$

Finalmente, substituindo-se (5.31), (5.32) em (5.24), e criando-se uma matriz auxiliar para reordenar o vetor de entrada, obtêm-se a expressão simplificada que permite calcular a transformada numérica de Fourier, qual seja:

$$\begin{aligned}
& \begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(-3\Delta\omega) \\ P(-2\Delta\omega) \\ P(-1\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^3 \\ 1 & 0 & 0 & 0 & W^4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^5 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^6 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^7 \end{bmatrix} x \\
& \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^2 & 0 & 0 & 0 & 0 \\ 1 & 0 & W^4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & W^4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^6 \end{bmatrix} x \\
& \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & W^4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & W^4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & W^4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & W^4 \end{bmatrix} x \\
& \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix}
\end{aligned} \tag{5.33}$$

Para se explicitar o algoritmo, toma-se a expressão (5.33) e substituem-se os seguintes valores:

$$\left. \begin{aligned} W^4 &= -1 \\ W^1 &= -W^5 \\ W^2 &= -W^6 \\ W^3 &= -W^7 \end{aligned} \right| \quad (5.24)$$

Resultando:

$$\begin{bmatrix} P(0\Delta\omega) \\ P(1\Delta\omega) \\ P(2\Delta\omega) \\ P(3\Delta\omega) \\ P(4\Delta\omega) \\ P(-3\Delta\omega) \\ P(-2\Delta\omega) \\ P(-1\Delta\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & W^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & W^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & W^3 \\ 1 & 0 & 0 & 0 & -W^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -W^1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -W^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -W^3 \end{bmatrix} x$$

$$\begin{bmatrix} 1 & 0 & W^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & W^2 & 0 & 0 & 0 & 0 \\ 1 & 0 & -W^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -W^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & W^0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & W^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & -W^0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -W^2 \end{bmatrix} x$$

$$\begin{bmatrix} 1 & W^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -W^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & W^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -W^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & W^0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -W^0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & W^0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -W^0 \end{bmatrix} x$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P(0\Delta t) \\ P(1\Delta t) \\ P(2\Delta t) \\ P(3\Delta t) \\ P(4\Delta t) \\ P(5\Delta t) \\ P(6\Delta t) \\ P(7\Delta t) \end{bmatrix}$$

(5.35)

Na expressão (5.35), pode-se perceber claramente, através dos retângulos vermelhos, um conjunto básico e único de operações a ser

realizado em cada matriz. Esse conjunto pode ser explicitado da seguinte forma:

1. Segundo número complexo de entrada é multiplicado pela potência de  $W$  indicada.
2. Produto resultante de 1 é somado ao primeiro número complexo de entrada, e o total representa o primeiro termo complexo de saída.
3. Produto de 1 é subtraído do primeiro complexo de entrada, e o resultado forma o segundo valor complexo de saída.

CAPITULO VI  
**QUADRATURAS DE ORDEM ELEVADA**

A fim de que os resultados fornecidos pela FFT sejam satisfatórios, é freqüentemente necessária uma quantidade muito grande de pontos discriminados ao longo do carregamento. No entanto, por mais que a FFT forneça resultados com elevado ganho de performance, o tamanho da matriz de entrada torna, com freqüência, o uso da solução pouco prático. Por outro lado, é possível melhorar os resultados da FFT, e da própria transformada de Fourier, buscando-se novas formas de efetuar as integrações necessárias, a fim de evitar o aumento do tamanho do vetor de entrada.

A necessidade de integrar um vetor  $f(x)$ , discretizado ao longo de um intervalo  $a - b$ , é comum nos problemas envolvendo métodos numéricos. O erro da integral pode ser minimizado usando-se uma adequada função de interpolação para os pontos definidos por  $f(x)$ . Esta função pode ser escrita como:

$$\int_a^b f(x)dx = (b - a) \sum_{k=0}^n B_k^n f(a + kh) \quad (6.1)$$

onde

$$B_k^n = (b - a)^{-1} \int_a^b \frac{w(x)}{(x - a - kh)w'(a + kh)} dx \quad (6.2)$$

e

$$w(x) = (x - x_1) \dots (x - x_k)$$

A variável  $h$ , definida em (6.1) e (6.2), representa o tamanho do intervalo entre os pontos discretizados de  $f(x)$ , e nada mais é do que a divisão do segmento  $a - b$  em  $n$  sub-segmentos, ou seja:

$$h = \frac{b-a}{n} \quad (6.3)$$

definindo assim os pontos  $a, a + h, a + 2h, \dots, a + nh = b$ .

Introduzindo uma nova variável  $t$  tal que  $x = a + th$ , os termos expressos em (6.2) podem ser reescritos como:

$$B_k^n = \frac{(-1)^{n-k}}{nk!(n-k)!} \int_a^b (t-0)(t-1)\dots(t-k+1)(t-k-1)\dots(t-n)dt \quad (6.4)$$

Para cada valor de  $n$ , se obtém uma série de ponderadores  $B_k^n$  que podem ser usados para aumentar a precisão da integral de  $f(x)$ . Quanto maior o valor de  $n$ , maior a precisão obtida, lembrando-se que nunca se deve aplicar um ponderador com número de termos maior que o número de pontos discretizados.

As duas primeiras séries de ponderadores são bem conhecidas. Para  $n = 1$ , tem-se a regra do trapézio, e para  $n = 2$ , a regra da Integração de Simpson. A tabela 6.1 mostra os ponderadores para outros valores de  $n$ . Estão mostrados apenas os ponderadores com  $k \leq n/2$ , já que existe simetria no termo central.

Como se percebe pelos limites da somatória de (6.1), um ponderador de ordem  $n$  possui  $n + 1$  termos. Para funções que possuam mais de  $n + 1$  termos, o ponderador deve ser aplicado sucessivas vezes, sobrepondo-se o último termo da ponderação anterior com o primeiro termo da próxima ponderação. O termo  $(a - b)$  à esquerda da somatória em (6.1), no entanto, deve ser sempre considerado como o comprimento de um único ponderador e não o comprimento total da função a ser integrada.



Quando existem termos no final da função que não se encaixem perfeitamente no ponderador, pode-se usar um ponderador de ordem menor que  $n$ , a fim de que todos os termos estejam ponderados.

Tabela 6.1

n	$B_0^n$	$B_1^n$	$B_2^n$	$B_3^n$	$B_4^n$	$B_5^n$
1	$\frac{1}{2}$					
2	$\frac{1}{6}$	$\frac{4}{6}$				
3	$\frac{1}{8}$	$\frac{3}{8}$				
4	$\frac{7}{90}$	$\frac{32}{90}$	$\frac{12}{90}$			
5	$\frac{19}{288}$	$\frac{75}{288}$	$\frac{50}{288}$			
6	$\frac{41}{840}$	$\frac{216}{840}$	$\frac{27}{840}$	$\frac{272}{840}$		
7	$\frac{751}{17280}$	$\frac{3577}{17280}$	$\frac{1323}{17280}$	$\frac{2989}{17280}$		
8	$\frac{989}{28350}$	$\frac{5888}{28350}$	$-\frac{928}{28350}$	$\frac{10496}{28350}$	$-\frac{4540}{28350}$	
9	$\frac{2857}{89600}$	$\frac{15741}{89600}$	$\frac{1080}{89600}$	$\frac{19344}{89600}$	$\frac{5778}{89600}$	
10	$\frac{16067}{598752}$	$\frac{106300}{598752}$	$-\frac{48525}{598752}$	$\frac{272400}{598752}$	$-\frac{260550}{598752}$	$\frac{427368}{598752}$

Apesar do erro ser decrescente com o aumento da ordem do ponderador, esta regra exige alguns cuidados. Considere o exemplo ilustrado na figura 6.1:

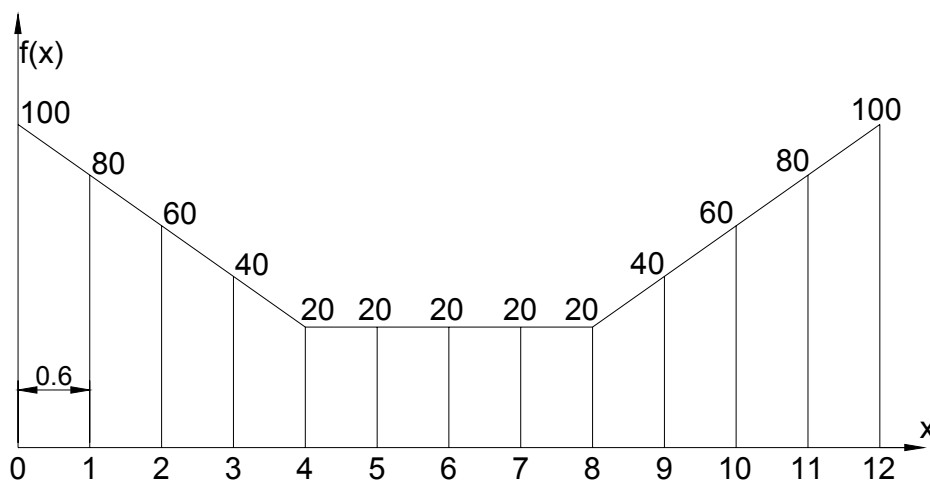


Figura 6.1 – Exemplo de carregamento

Para essa configuração mostrada na figura 6.1, não existe um ponderador que se aplique diretamente a todos os pontos. Ao invés de buscar a formulação de um que seja adequado, é muito mais prático e rápido, dividir a função em  $m$  intervalos onde serão aplicados os ponderadores. Usando-se os ponderadores com  $n = 2$  e  $n = 3$ , a divisão dos intervalos  $m$  é a que se segue:

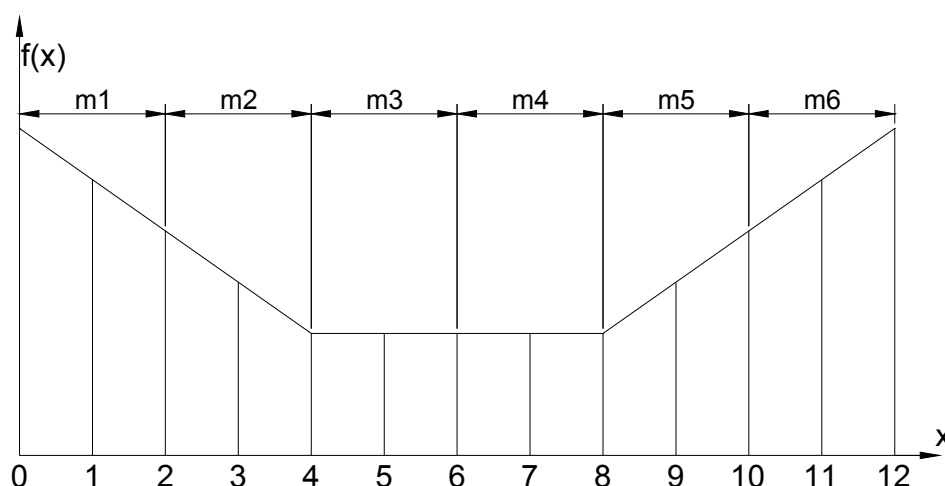


Figura 6.2 – Divisão para  $n = 2$

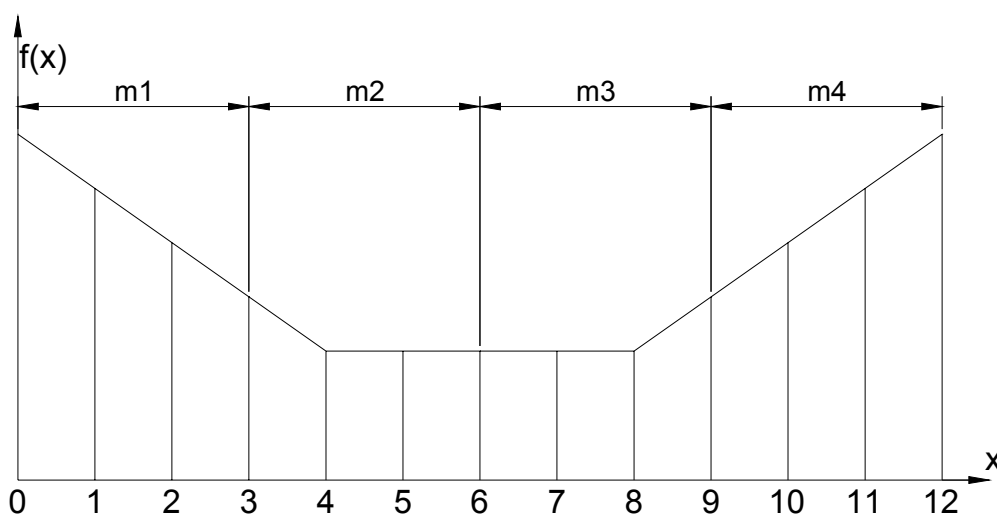


Figura 6.3 – Divisão para  $n = 3$

O resultado da integração usando-se o ponderador de ordem  $n = 2$  é exato, enquanto para  $n = 3$  surge um erro no valor final. A explicação para isso reside nos intervalos  $m_2$  e  $m_3$  da figura 6.3. O ponderador assume que a função  $f(x)$  e sua derivada são contínuas no intervalo onde é aplicado o ponderador. Enquanto nos intervalos  $m_1$  e  $m_4$  essa condição é satisfeita, a primeira derivada de  $f(x)$  não é contínua em  $m_2$  e  $m_3$ . Daí o surgimento do erro. Na figura 6.2, todos os intervalos são

contínuos tanto para  $f(x)$  como para  $f'(x)$ , o que conduz a um resultado mais preciso.

Até o presente momento, foi feita uma abordagem da transformada de Fourier e dos ponderadores de Newton-Cotes de forma separada e independente. No entanto, é possível estabelecer uma ligação entre estes dois assuntos. Considere a equação (5.4) da transformada de Fourier na sua forma numérica:

$$P(k\Delta\omega) = \sum_{j=0}^{N-1} P(j\Delta t) e^{\frac{-i2\pi jk}{N} \Delta t} \quad (6.5)$$

A fim de permitir uma melhor análise da forma de integração numérica usada na transformada, seja uma função  $G_k(j\Delta t)$  definida por:

$$G_k(j\Delta t) = P(j\Delta t) e^{\frac{-i2\pi jk}{N} \Delta t} \quad (6.6)$$

Desta forma, a transformada pode ser escrita como:

$$P(k\Delta\omega) = \sum_{j=0}^{N-1} G_k(j\Delta t) \Delta t \quad (6.7)$$

A forma de integração da transformada, representada em (6.7) pela somatória, corresponde à forma mais simples e elementar de integração numérica, ou seja, o produto dos valores assumidos pela função  $G_k(j\Delta t)$  nos pontos  $j = 0, 1, 2, \dots, N-1$ , pelo intervalo de discretização  $\Delta t$ . A figura 6.4 mostra claramente a forma de cálculo da somatória. Este método é conhecido como a Regra do Retângulo, já que toda a função é dividida em retângulos de base igual, cujas áreas são somadas para se obter a resposta final.

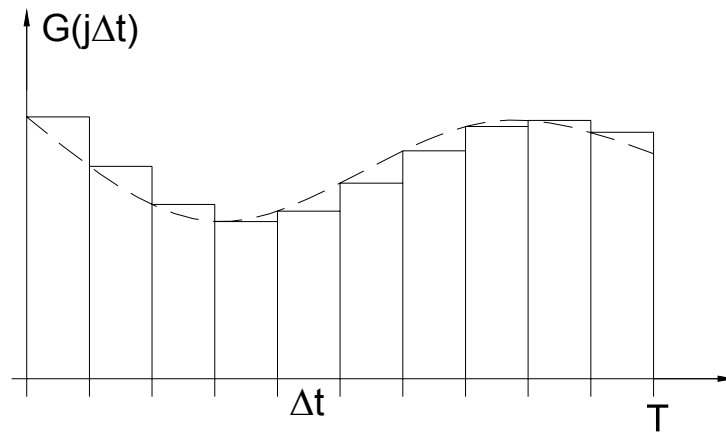


Figura 6.4: Integração por Regra do Retângulo

A partir do momento em que se passa a analisar a transformada de Fourier tendo em vista os métodos de integração numérica conhecidos, pode-se fazer uso dos ponderadores já estudados de Newton-Cotes. A Regra do Trapézio, equivalente ao ponderador de ordem 1 de Newton-Cotes fornece uma precisão maior do que a Regra do Retângulo, que, apesar de sua simplicidade de aplicação, possui o maior erro de todos os métodos de integração numérica conhecidos. Para aumentar a precisão da somatória de (6.7), basta aplicar os ponderadores diretamente na função  $G_k(j\Delta t)$  da mesma forma que se aplica os ponderadores à uma função qualquer:

$$P(k\Delta\omega) = n\Delta t \sum_{j=0}^{N-1} B_j^n G_k(j\Delta t) \quad (6.8)$$

A equação (6.8) deriva da equação (6.1), com algumas diferenças: os termos  $B_j^n$  correspondem aos termos do ponderador, no entanto, caso seja aplicado um ponderador de ordem inferior a  $N$ , estes termos correspondem aos termos obtidos pela sucessiva aplicação de um ponderador de ordem  $n$ . O termo  $(b - a)$  presente na equação (6.1), correspondente ao intervalo onde é aplicado um ponderador, foi reescrito. Como o tamanho do ponderador não varia ao longo do intervalo de integração, este termo pode ser reescrito como:

$$(b - a) = n\Delta t \quad (6.9)$$

onde  $n$  é a ordem do ponderador. Retornando a equação (6.6) em (6.8), tem-se a equação da Transformada de Fourier ponderada por Newton-Cotes:

$$P(k\Delta\omega) = n\Delta t \sum_{j=0}^{N-1} B_j^n P(j\Delta t) e^{\frac{-i2\pi jk}{N}} \quad (6.10)$$

Usar o ponderador diretamente na somatória da transformada, no entanto, traz um complicador que a alteração do algoritmo da FFT. Para escapar desta necessidade, é possível fazer uso da propriedade distributiva da multiplicação e ponderar apenas a função  $P(j\Delta t)$ , e não a função  $G_k(j\Delta t)$ . Levando-se em conta ainda que todos os termos do ponderador, qualquer que seja sua ordem, são termos constantes em função apenas do intervalo de integração, pode-se ponderar a função  $P(j\Delta t)$  antes de usá-la na transformada, mantendo desta forma o algoritmo da FFT intacto.

A ponderação da transformada se dá, então, em três etapas, descritas abaixo:

1. Escolha do ponderador em função da discretização de  $P(j\Delta t)$ ;
2. Aplicação do ponderador sobre  $P(j\Delta t)$ , obtendo um novo vetor de entrada  $P'(j\Delta t)$ .
3. Aplicação do vetor  $P'(j\Delta t)$ , já ponderado, na transformada através do uso de um algoritmo de FFT.

Raciocínio semelhante pode ser feito para a transformada inversa de Fourier, onde os ponderadores de Newton-Cotes são aplicados sobre o vetor  $P(\Delta\omega)$ .

A aplicação dos ponderadores permite que o número de pontos na discretização seja menor, já que os próprios ponderadores garantem um resultado mais preciso. Como será mostrado no próximo capítulo, utilizando-se o ponderador de ordem 2 – Integração por Simpson – é possível obter resultados mais precisos com 32 pontos do que aqueles obtidos pela FFT sem ponderação com 64 pontos.

## CAPITULO VII

### **EXEMPLOS DE APLICAÇÃO**

#### **7.1. Introdução**

Este capítulo é dedicado à exibição de exemplos com a finalidade de mostrar a aplicação dos operadores de integração de ordem elevada na FFT, bem como comparar os resultados obtidos.

São escolhidos alguns exemplos de aplicação bastante simples. Em todos eles foi considerado um carregamento cuja solução pode ser obtida analiticamente (forma fechada), permitindo-se que os resultados, tanto da FFT clássica, quanto da FFT com ordem de integração incrementada possam ser comparados com os da solução exata.

#### **7.2. Carregamento retangular**

A figura 7.1 ilustra um carregamento constante e de valor  $P$  no intervalo de zero até um valor genérico 'a', sendo nulo nos demais tempos.

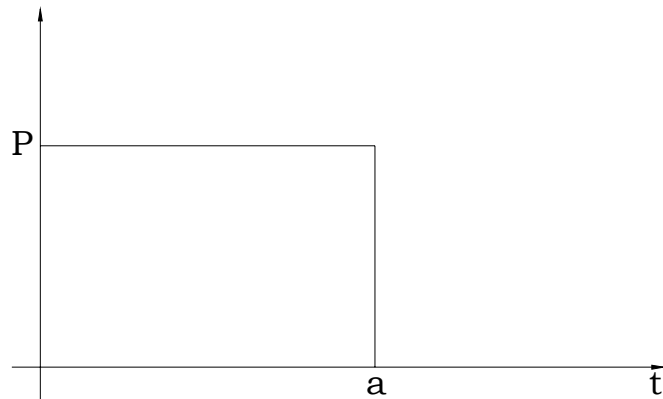


Figura 7.1 – Carregamento Retangular

Em termos analíticos, o carregamento mostrado na figura 7.1 é dado por:

$$P(t) = \begin{cases} 0, & t < 0; \\ P, & 0 \leq t \leq a; \\ 0, & t > a \end{cases} \quad (7.1)$$

Tendo-se em vista que o carregamento só apresenta valores não nulos no intervalo de tempo entre zero e 'a', a transformada de Fourier pode ser expressa na forma:

$$F(\omega) = \int_0^a P e^{-i\omega t} dt \quad (7.2)$$

que resolvida resulta em:

$$F(\omega) = -\frac{P}{i\omega} (e^{-i\omega a} - 1) \quad (7.3)$$

É oportuno assinalar que o expresso em (7.3) é também referido como o carregamento dado no domínio da frequência.

Aplicando-se as identidades de Euler, e separando os componentes Real e Imaginário da solução, obtém-se:

$$\begin{aligned} P_{\text{Re}}(\omega) &= Pa \frac{\text{sen}(\omega a)}{\omega a} \\ P_{\text{Im}}(\omega) &= Pa \frac{[\cos(\omega a) - 1]}{\omega a} \end{aligned} \quad (7.4)$$



No exemplo de aplicação em apreço os valores numéricos dos parâmetros a serem considerados são:

$$T = 20,48s$$

$$\omega = 0,306796 \text{ Hz}$$

$$P = 200$$

$$a = 2,56s$$

Com tais valores, o expresso nas equações (7.4) são lançados nos gráficos das figuras 7.2 e 7.3:

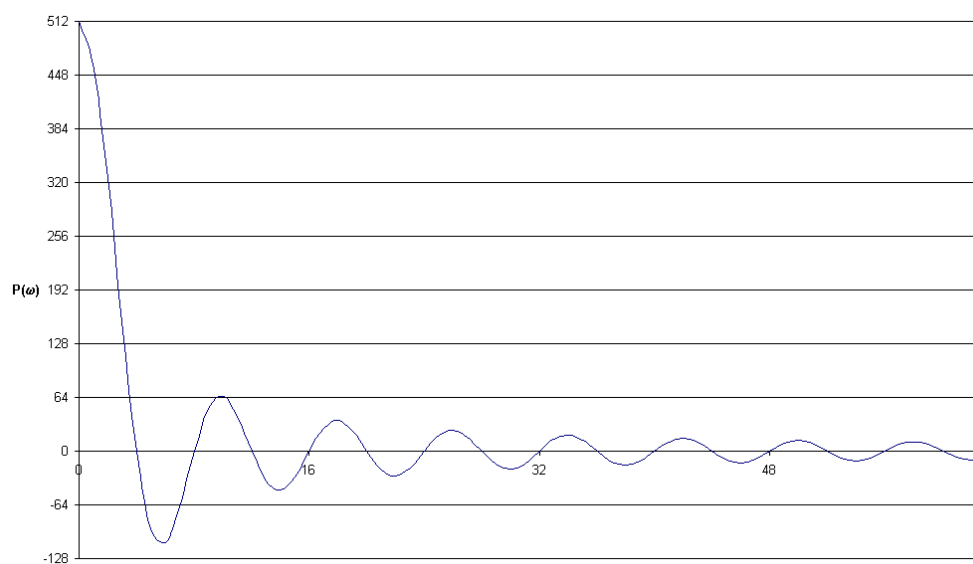


Figura 7.2 – Solução Exata para Carregamento Retangular – Parte Real

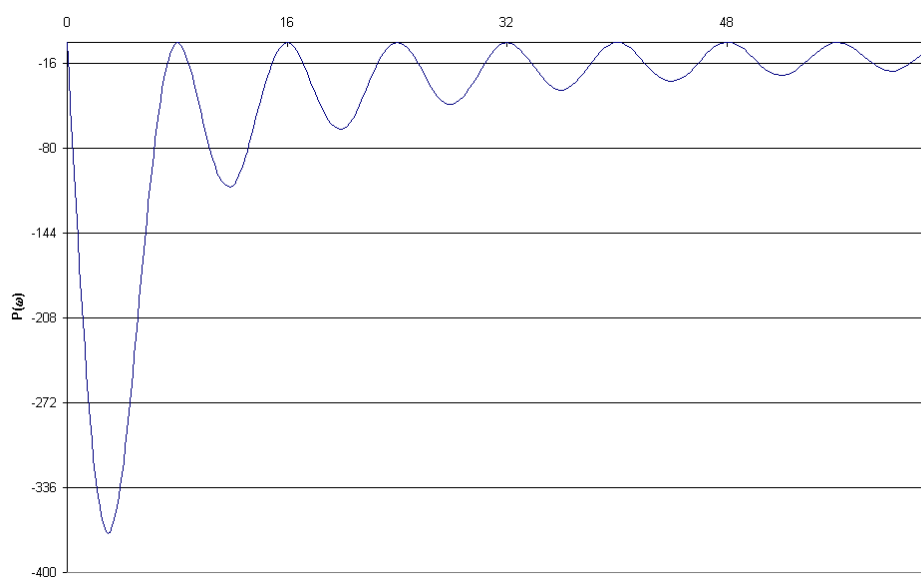


Figura 7.3 – Solução Exata para Carregamento Retangular – Parte Imaginária

São consideradas as discretizações do carregamento com três valores para o passo  $\Delta t$ , quais sejam  $a/16$ ,  $a/32$  e  $a/64$ . Além disso, foram utilizadas três ordens de ponderadores na resolução, quais sejam, quarta, sexta e oitava. As tabelas 7.1 a 7.3 mostram os resultados assim obtidos para a transformada de Fourier:

Número de Pontos discretizados: 16

Intervalo entre cada ponto –  $\Delta t$ : 1,28s

Tabela 7.1

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	512	0	768	0	512	0	512	0	591,6444	0
1	200	200	266,7	200	460,962	-190,9367	673,5325	-278,9863	455,0228	-188,4766	461,024	-190,9624	497,1769	-278,9863
2	200	100	66,67	200	325,9493	-325,9493	437,0193	-437,0193	309,0193	-309,0193	326,6925	-326,6925	260,6638	-437,0193
3	0	0	0	0	153,654	-370,9535	172,9476	-417,5325	135,4573	-327,0228	155,6162	-375,6907	-3,40793	-417,5325
4	0	0	0	0	2E-14	-325,9493	0	-256	0	-256	0	-341,3333	-176,3556	-256
5	0	0	0	0	-92,19239	-222,5721	-22,98629	-55,49382	-60,47663	-146,0035	-105,6291	-255,0111	-199,3419	-55,49382
6	0	0	0	0	-108,6498	-108,6498	74,98066	74,98066	-53,01934	-53,01934	-156,0258	-156,0258	-101,3749	74,98066
7	0	0	0	0	-65,85171	-27,27667	200,5062	83,05238	-18,00349	-7,45729	-169,6778	-70,28283	24,15062	83,05238
8	0	0	0	0	-2E-14	0	256	0	0	0	-170,6667	0	79,64444	0
9	0	0	0	0			200,5062	-83,05238	-18,00349	7,45729	-169,6778	70,28283	24,15062	-83,05238
10	0	0	0	0			74,98066	-74,98066	-53,01934	53,01934	-156,0258	156,0258	-101,3749	-74,98066
11	0	0	0	0			-22,98629	55,49382	-60,47663	146,0035	-105,6291	255,0111	-199,3419	55,49382
12	0	0	0	0			0	256	0	256	0	341,3333	-176,3556	256
13	0	0	0	0			172,9476	417,5325	135,4573	327,0228	155,6162	375,6907	-3,40793	417,5325
14	0	0	0	0			437,0193	437,0193	309,0193	309,0193	326,6925	326,6925	260,6638	437,0193
15	0	0	0	0			673,5325	278,9863	455,0228	188,4766	461,024	190,9624	497,1769	278,9863

Algumas considerações sobre os resultados arrolados na tabela 7.1 e subseqüentes são as seguintes:

- As colunas de título  $p(t)$  representam o carregamento a que é sujeito a FFT. Os ponderadores da quadratura são aplicados já no vetor de carregamento, conforme visto no capítulo 6;
- Foram comparados os resultados obtidos pelo carregamento sem ponderação (coluna “orig.” com resultados indicados na coluna “FFT”), e com ponderadores de ordens 1, 2 e 4 (colunas “ord. 1”, “ord. 2” e “ord. 3” respectivamente). Os resultados da transformada com estes ponderadores são indicados nas colunas “Ponderada ordem 1” até “Ponderada ordem 4”.
- A coluna “Exata” mostra os valores exatos para a transformada obtidos analiticamente. Apesar da resposta analítica poder fornecer os valores da transformada para qualquer tempo, estão indicados apenas os primeiros  $N/2$  valores, já que o algoritmo da transformada retorna apenas os primeiros  $N/2$  valores da resposta.
- O algoritmo da FFT utilizado nas quatro últimas colunas é aquele descrito por Hall, e está explicitado no Anexo A;
- Os resultados da segunda metade da FFT são reprodução da primeira metade, pois os valores para  $N - k$  são iguais aos de  $-k$ . Estes valores estão indicados apenas para efeito de verificação de valores.

Como era esperado, devido ao pouco número de pontos discretizados, a FFT original produz resultados imprecisos. No entanto, mesmo com o ponderador de ordem 1, a precisão é aumentada consideravelmente. O ponderador de ordem 2 produz resultados ainda mais precisos. Para ordem  $N = 4$ , no entanto, o resultado não é preciso. Isso ocorre porque existe uma descontinuidade no intervalo onde é aplicado o ponderador (ponto  $j = 2$ ).

Número de Pontos Discretizados = 32

Intervalo entre cada ponto –  $\Delta t$ : 0,64s

Tabela 7.2

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	512	0	640	0	512	0	512	0	512	0
1	200	200	266,7	284,4	460,962	-190,9367	568,7349	-235,5777	459,48	-190,3229	460,9658	-190,9383	460,9619	-190,9367
2	200	200	133,3	106,7	325,9493	-325,9493	385,7497	-385,7497	321,7497	-321,7497	325,9932	-325,9932	325,9466	-325,9466
3	200	200	266,7	284,4	153,654	-370,9535	167,9304	-405,4198	149,1852	-360,1649	153,7612	-371,2123	153,6375	-370,9137
4	200	100	66,67	62,22	2E-14	-325,9493	0	-309,0193	0	-309,0193	0	-326,6925	0	-325,7164
5	0	0	0	0	-92,19239	-222,5721	-65,92067	-159,1466	-84,66584	-204,4014	-92,72891	-223,8674	-91,8689	-221,7912
6	0	0	0	0	-108,6498	-108,6498	-31,78277	-31,78277	-95,78277	-95,78277	-110,0373	-110,0373	-106,9713	-106,9713
7	0	0	0	0	-65,85171	-27,27667	54,11166	22,41378	-55,14317	-22,84105	-67,52307	-27,96897	-60,71276	-25,14805
8	0	0	0	0	-2E-14	0	128	0	0	0	0	0	11,37778	0
9	0	0	0	0	51,21799	-21,21519	146,3945	-60,63859	37,13968	-15,38376	55,52074	-22,99744	70,53397	-29,21613
10	0	0	0	0	65,18986	-65,18986	106,7634	-106,7634	42,76343	-42,76343	74,69102	-74,69102	90,07214	-90,07214
11	0	0	0	0	41,90563	-101,1691	42,93438	-103,6528	24,18921	-58,39793	52,41116	-126,5317	62,94717	-151,9679
12	0	0	0	0	2E-14	-108,6498	0	-53,01934	0	-53,01934	0	-156,0258	0	-189,1831
13	0	0	0	0	-35,45861	-85,60466	5,01726	12,11274	-13,7279	-33,14209	-63,4563	-153,1971	-78,06113	-188,4563
14	0	0	0	0	-46,56419	-46,56419	51,26961	51,26961	-12,73039	-12,73039	-119,9803	-119,9803	-149,7585	-149,7585
15	0	0	0	0	-30,7308	-12,72911	104,7976	43,4086	-4,45721	-1,84624	-157,6172	-65,28719	-198,86	-82,3705

continua

Continuação Tabela 7.2

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	-2E-14	0	128	0	0	0	-170,6667	0	-216,1778	0
17	0	0	0	0			104,7976	-43,4086	-4,45721	1,84624	-157,6172	65,28719	-198,86	82,3705
18	0	0	0	0			51,26961	-51,26961	-12,73039	12,73039	-119,9803	119,9803	-149,7585	149,7585
19	0	0	0	0			5,01726	-12,11274	-13,7279	33,14209	-63,4563	153,1971	-78,06113	188,4563
20	0	0	0	0			0	53,01934	0	53,01934	0	156,0258	0	189,1831
21	0	0	0	0			42,93438	103,6528	24,18921	58,39793	52,41116	126,5317	62,94717	151,9679
22	0	0	0	0			106,7634	106,7634	42,76343	42,76343	74,69102	74,69102	90,07214	90,07214
23	0	0	0	0			146,3945	60,63859	37,13968	15,38376	55,52074	22,99744	70,53397	29,21613
24	0	0	0	0			128	0	0	0	0	0	11,37778	0
25	0	0	0	0			54,11166	-22,41378	-55,14317	22,84105	-67,52307	27,96897	-60,71276	25,14805
26	0	0	0	0			-31,78277	31,78277	-95,78277	95,78277	-110,0373	110,0373	-106,9713	106,9713
27	0	0	0	0			-65,92067	159,1466	-84,66584	204,4014	-92,72891	223,8674	-91,8689	221,7912
28	0	0	0	0			0	309,0193	0	309,0193	0	326,6925	0	325,7164
29	0	0	0	0			167,9304	405,4198	149,1852	360,1649	153,7612	371,2123	153,6375	370,9137
30	0	0	0	0			385,7497	385,7497	321,7497	321,7497	325,9932	325,9932	325,9466	325,9466
31	0	0	0	0			568,7349	235,5777	459,48	190,3229	460,9658	190,9383	460,9619	190,9367

Número de Pontos Discretizados = 64

Intervalo entre cada ponto –  $\Delta t$ : 0,32s

Tabela 7.3

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	512	0	576	0	512	0	512	0	512	0
1	200	200	266,7	284,4	460,962	-190,9367	515,2191	-213,4107	460,5917	-190,7833	460,9622	-190,9368	460,962	-190,9367
2	200	200	133,3	106,7	325,9493	-325,9493	356,9015	-356,9015	324,9015	-324,9015	325,952	-325,952	325,9493	-325,9493
3	200	200	266,7	284,4	153,654	-370,9535	161,9142	-390,8956	152,5417	-368,2681	153,6605	-370,9692	153,6538	-370,953
4	200	200	133,3	124,4	2E-14	-325,9493	0	-321,7497	0	-321,7497	0	-325,9932	0	-325,9466
5	200	200	266,7	284,4	-92,19239	-222,5721	-80,96113	-195,4575	-90,33371	-218,0849	-92,223	-222,646	-92,18928	-222,5646
6	200	200	133,3	106,7	-108,6498	-108,6498	-73,48986	-73,48986	-105,4899	-105,4899	-108,7256	-108,7256	-108,6381	-108,6381
7	200	200	266,7	284,4	-65,85171	-27,27667	-8,61198	-3,5672	-63,23939	-26,19461	-65,93813	-27,31247	-65,83247	-27,2687
8	200	100	66,67	62,22	-2E-14	0	64	0	0	0	0	0	0	0
9	0	0	0	0	51,21799	-21,21519	102,4691	-42,44408	47,84165	-19,81666	51,40898	-21,2943	51,13486	-21,18075
10	0	0	0	0	65,18986	-65,18986	91,86779	-91,86779	59,86779	-59,86779	65,56924	-65,56924	64,96112	-64,96112
11	0	0	0	0	41,90563	-101,1691	47,12415	-113,7678	37,75156	-91,14034	42,27235	-102,0545	41,59643	-100,4227
12	0	0	0	0	2E-14	-108,6498	0	-95,78277	0	-95,78277	0	-110,0373	0	-106,9713
13	0	0	0	0	-35,45861	-85,60466	-21,13696	-51,02913	-30,50954	-73,65655	-36,10342	-87,16136	-34,27989	-82,75898
14	0	0	0	0	-46,56419	-46,56419	-6,99211	-6,99211	-38,99211	-38,99211	-47,74602	-47,74602	-42,9304	-42,9304
15	0	0	0	0	-30,7308	-12,72911	29,66191	12,28637	-24,96551	-10,34105	-31,80161	-13,17266	-23,4139	-9,69835

Continua.

Continuação Tabela 7.3

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	-2E-14	0	64	0	0	0	0	0	11,37778	0
17	0	0	0	0	27,11541	-11,23157	75,13571	-31,12223	20,5083	-8,49481	28,83013	-11,94183	41,25995	-17,09043
18	0	0	0	0	36,21659	-36,21659	58,26172	-58,26172	26,26172	-26,26172	39,25909	-39,25909	49,87505	-49,87505
19	0	0	0	0	24,26116	-58,57161	26,15422	-63,14187	16,78164	-40,51445	26,95148	-65,06664	32,97867	-79,61755
20	0	0	0	0	-3,79E-14	-65,18986	0	-42,76343	0	-42,76343	0	-74,69102	0	-90,07214
21	0	0	0	0	-21,95057	-52,99336	-4,18977	-10,115	-13,56235	-32,74241	-26,14621	-63,12253	-31,38336	-75,76614
22	0	0	0	0	-29,63176	-29,63176	14,89564	14,89564	-17,10436	-17,10436	-37,06029	-37,06029	-44,51037	-44,51037
23	0	0	0	0	-20,04182	-8,301595	43,92545	18,19452	-10,70197	-4,4329	-26,64919	-11,03846	-32,12718	-13,30752
24	0	0	0	0	-2E-14	0	64	0	0	0	0	0	0	0
25	0	0	0	0	18,43848	-7,637468	62,72364	-25,98098	8,09622	-3,35356	29,17602	-12,0851	35,62262	-14,75537
26	0	0	0	0	25,07302	-25,07302	41,70709	-41,70709	9,70709	-9,70709	44,87038	-44,87038	55,19756	-55,19756
27	0	0	0	0	17,07266	-41,21706	15,04046	-36,31087	5,66787	-13,68346	35,77911	-86,37841	44,34631	-107,0615
28	0	0	0	0	-6,27E-14	-46,56419	0	-12,73039	0	-12,73039	0	-119,9803	0	-149,7585
29	0	0	0	0	-15,89524	-38,3745	6,01612	14,52421	-3,35646	-8,10321	-54,20368	-130,8593	-68,068	-164,3307
30	0	0	0	0	-21,72995	-21,72995	28,84828	28,84828	-3,15172	-3,15172	-111,4522	-111,4522	-140,6152	-140,6152
31	0	0	0	0	-14,86974	-6,159248	53,5158	22,16697	-1,11161	-0,46045	-154,6422	-64,05488	-195,6827	-81,05443

continua.



Continuação da Tabela 7.3

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
32	0	0	0	0	-2E-14	0	64	0	0	0	-170,6667	0	-216,1778	0
33	0	0	0	0			53,5158	-22,16697	-1,11161	0,46045	-154,6422	64,05488	-195,6827	81,05443
34	0	0	0	0			28,84828	-28,84828	-3,15172	3,15172	-111,4522	111,4522	-140,6152	140,6152
35	0	0	0	0			6,01612	-14,52421	-3,35646	8,10321	-54,20368	130,8593	-68,068	164,3307
36	0	0	0	0			0	12,73039	0	12,73039	0	119,9803	0	149,7585
37	0	0	0	0			15,04046	36,31087	5,66787	13,68346	35,77911	86,37841	44,34631	107,0615
38	0	0	0	0			41,70709	41,70709	9,70709	9,70709	44,87038	44,87038	55,19756	55,19756
39	0	0	0	0			62,72364	25,98098	8,09622	3,35356	29,17602	12,0851	35,62262	14,75537
40	0	0	0	0			64	0	0	0	0	0	0	0
41	0	0	0	0			43,92545	-18,19452	-10,70197	4,4329	-26,64919	11,03846	-32,12718	13,30752
42	0	0	0	0			14,89564	-14,89564	-17,10436	17,10436	-37,06029	37,06029	-44,51037	44,51037
43	0	0	0	0			-4,18977	10,115	-13,56235	32,74241	-26,14621	63,12253	-31,38336	75,76614
44	0	0	0	0			0	42,76343	0	42,76343	0	74,69102	0	90,07214
45	0	0	0	0			26,15422	63,14187	16,78164	40,51445	26,95148	65,06664	32,97867	79,61755
46	0	0	0	0			58,26172	58,26172	26,26172	26,26172	39,25909	39,25909	49,87505	49,87505
47	0	0	0	0			75,13571	31,12223	20,5083	8,49481	28,83013	11,94183	41,25995	17,09043

Continua.

Continuação da Tabela 7.3

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
48	0	0	0	0			64	0	0	0	0	0	11,37778	0
49	0	0	0	0			29,66191	-12,28637	-24,96551	10,34105	-31,80161	13,17266	-23,4139	9,69835
50	0	0	0	0			-6,99211	6,99211	-38,99211	38,99211	-47,74602	47,74602	-42,9304	42,9304
51	0	0	0	0			-21,13696	51,02913	-30,50954	73,65655	-36,10342	87,16136	-34,27989	82,75898
52	0	0	0	0			0	95,78277	0	95,78277	0	110,0373	0	106,9713
53	0	0	0	0			47,12415	113,7678	37,75156	91,14034	42,27235	102,0545	41,59643	100,4227
54	0	0	0	0			91,86779	91,86779	59,86779	59,86779	65,56924	65,56924	64,96112	64,96112
55	0	0	0	0			102,4691	42,44408	47,84165	19,81666	51,40898	21,2943	51,13486	21,18075
56	0	0	0	0			64	0	0	0	0	0	0	0
57	0	0	0	0			-8,61198	3,5672	-63,23939	26,19461	-65,93813	27,31247	-65,83247	27,2687
58	0	0	0	0			-73,48986	73,48986	-105,4899	105,4899	-108,7256	108,7256	-108,6381	108,6381
59	0	0	0	0			-80,96113	195,4575	-90,33371	218,0849	-92,223	222,646	-92,18928	222,5646
60	0	0	0	0			0	321,7497	0	321,7497	0	325,9932	0	325,9466
61	0	0	0	0			161,9142	390,8956	152,5417	368,2681	153,6605	370,9692	153,6538	370,953
62	0	0	0	0			356,9015	356,9015	324,9015	324,9015	325,952	325,952	325,9493	325,9493
63	0	0	0	0			515,2191	213,4107	460,5917	190,7833	460,9622	190,9368	460,962	190,9367

Nesse ponto cabem ainda novas considerações sobre os resultados. Por exemplo, analisando-se os valores das tabelas 7.1 a 7.3, percebe-se rapidamente que os resultados fornecidos pela FFT com ponderador de ordem 2 da tabela 7.1 (16 pontos) são mais precisos que aqueles obtidos com a FFT e ponderador ordem 1 na tabela 7.3) (128 pontos).

É importante ressaltar que os resultados mais precisos são sempre aqueles de frequência mais baixa, via de regra. À medida que o número de pontos aumenta, aumenta também o número de frequências que podem ser obtidas com maior precisão.

## 7.2. Carregamento Triangular

Considere-se agora o carregamento com variação linear no tempo como mostrado na figura 7.4.

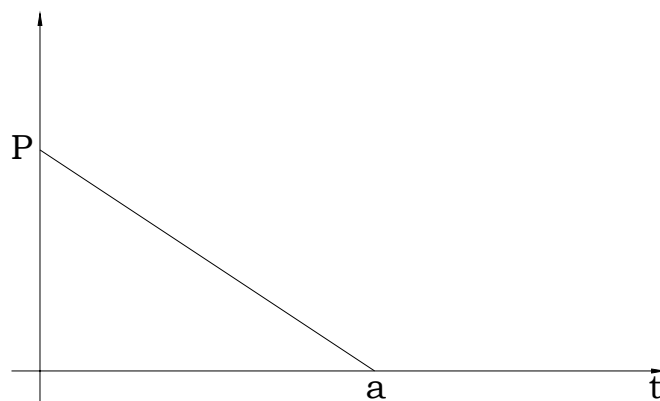


Figura 7.4 – Carregamento Triangular

A forma analítica do carregamento em questão pode assim ser definida:

$$P(t) = \begin{cases} 0, & t < 0; \\ P(1 - \frac{t}{a}), & 0 \leq t \leq a; \\ 0, & t > a; \end{cases} \quad (7.4)$$

Analogamente ao empregado no tratamento da equação (4.21) pode-se escrever:

$$P(\omega) = \int_0^a P(1 - \frac{t}{a}) e^{-i\omega t} dt \quad (7.5)$$

Resolvendo-se a integração redigida em (7.5), e tendo-se em conta as identidades de Euler, obtém-se:

$$\begin{aligned} P_{\text{Re}}(\omega) &= -Pa \frac{\cos(\omega a) - 1}{(\omega a)^2} \\ P_{\text{Im}}(\omega) &= \frac{Pa}{\omega a} \left( \frac{\text{sen}(\omega a)}{\omega a} - 1 \right) \end{aligned} \quad (7.6)$$

Os gráficos das figuras 7.5 e 7.6 mostram os resultados de (7.6) mais claramente. Aqui nesse exemplo novamente são considerados os valores numéricos do exemplo anterior, ou seja:

$$\begin{aligned} T &= 20,48s \\ \omega &= 0,306796 \text{ Hz} \\ P &= 200 \\ a &= 2,56s \end{aligned}$$

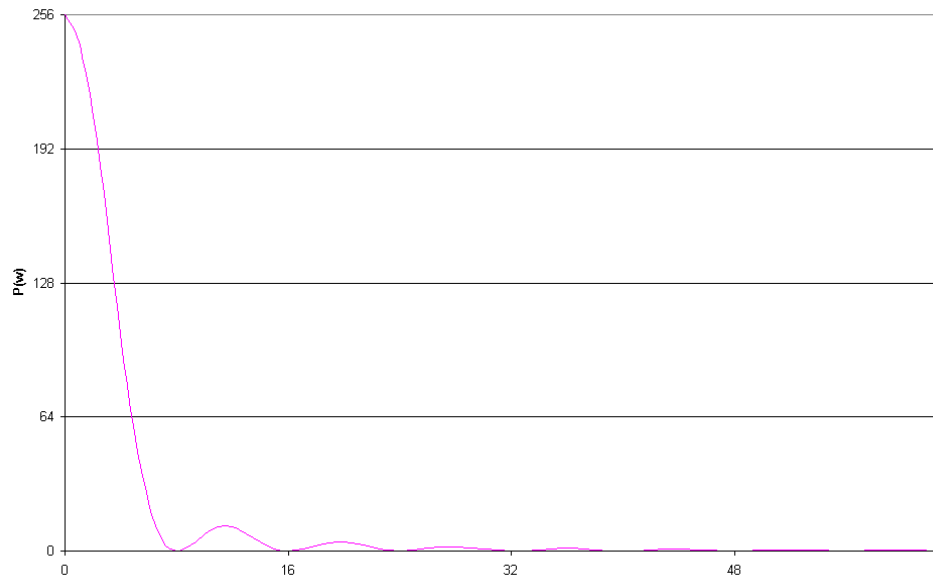


Figura 7.2 – Solução Exata para Carregamento Linear – Parte Real

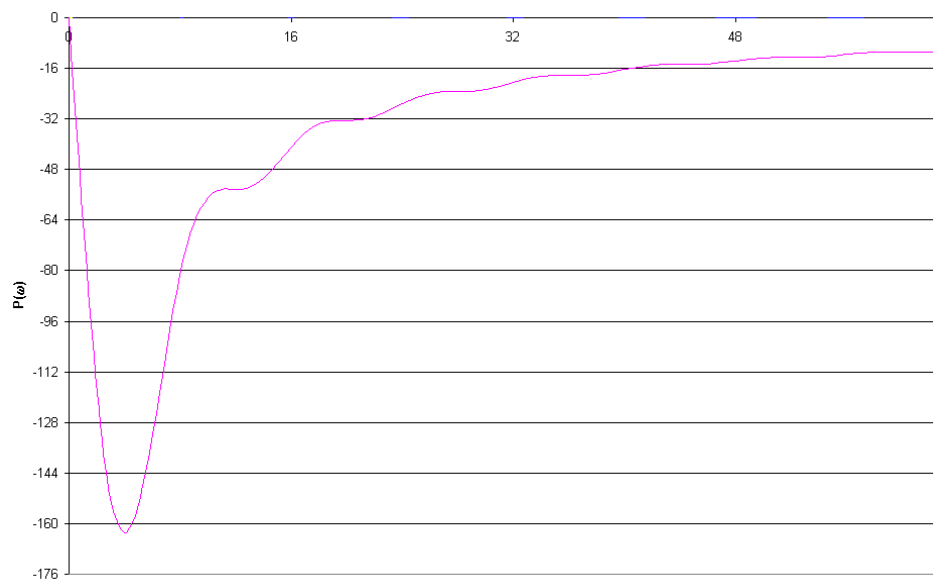


Figura 7.5 – Solução Exata para Carregamento Linear – Parte Imaginária

De modo semelhante ao que foi feito no exemplo precedente, a discretização do carregamento se deu para aqueles mesmos três valores para o passo  $\Delta t$ , e também consideradas três ordens de ponderadores na resolução. As tabelas 7.4 a 7.6 mostram os valores para as transformadas obtidas.

Número de Pontos Discretizados = 16

Intervalo entre cada ponto –  $\Delta t$ : 1,28s

Tabela 7.4

i	P(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	256	0	384	0	256	0	256	0	207,6444	0
1	100	100	133,3	100	243,1081	-64,9837	374,2566	-48,9835	246,2566	-48,9835	243,0088	-65,3113	197,901	-48,9835
2	0	0	0	0	207,5058	-118,444	346,5097	-90,5097	218,5097	-90,5097	206,0129	-120,68	170,1541	-90,5097
3	0	0	0	0	157,4376	-152,087	304,9835	-118,257	176,9835	-118,257	150,6446	-157,675	128,6279	-118,257
4	0	0	0	0	103,7529	-162,975	256	-128	128	-128	85,33333	-170,667	79,64444	-128
5	0	0	0	0	56,67752	-153,856	207,0165	-118,257	79,01652	-118,257	20,02203	-157,675	30,66097	-118,257
6	0	0	0	0	23,0562	-131,706	165,4903	-90,5097	37,49033	-90,5097	-35,3462	-120,68	-10,8652	-90,5097
7	0	0	0	0	4,961391	-105,106	137,7434	-48,9835	9,74342	-48,9835	-72,3421	-65,3113	-38,6121	-48,9835
8	0	0	0	0	0	-81,4873	128	0	0	0	-85,3333	0	-48,3556	0
9	0	0	0	0			137,7434	48,98348	9,74342	48,98348	-72,3421	65,31131	-38,6121	48,98348
10	0	0	0	0			165,4903	90,50967	37,49033	90,50967	-35,3462	120,6796	-10,8652	90,50967
11	0	0	0	0			207,0165	118,2566	79,01652	118,2566	20,02203	157,6754	30,66097	118,2566
12	0	0	0	0			256	128	128	128	85,33333	170,6667	79,64444	128
13	0	0	0	0			304,9835	118,2566	176,9835	118,2566	150,6446	157,6754	128,6279	118,2566
14	0	0	0	0			346,5097	90,50967	218,5097	90,50967	206,0129	120,6796	170,1541	90,50967
15	0	0	0	0			374,2566	48,98348	246,2566	48,98348	243,0088	65,31131	197,901	48,98348

Número de Pontos Discretizados = 32

Intervalo entre cada ponto –  $\Delta t$ : 0,64s

Tabela 7.5

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	256	0	320	0	256	0	256	0	256	0
1	150	150	200	213,3	243,1081	-64,9837	307,8907	-60,9987	243,8907	-60,9987	243,1021	-65,0037	243,1083	-64,9832
2	100	100	66,67	53,33	207,5058	-118,444	274,1931	-111,557	210,1931	-111,557	207,421	-118,572	207,5148	-118,432
3	50	50	66,67	71,11	157,4376	-152,087	226,0699	-143,848	162,0699	-143,848	157,0988	-152,379	157,529	-152,026
4	0	0	0	0	103,7529	-162,975	173,2548	-154,51	109,2548	-154,51	103,0065	-163,346	104,1847	-162,858
5	0	0	0	0	56,67752	-153,856	125,4579	-145,192	61,45787	-145,192	55,60499	-154,171	57,97719	-153,937
6	0	0	0	0	23,0562	-131,706	89,91863	-121,701	25,91863	-121,701	22,0614	-132,099	25,88857	-132,86
7	0	0	0	0	4,961391	-105,106	69,82213	-92,0401	5,82213	-92,0401	4,51504	-106,392	9,63885	-109,131
8	0	0	0	0	0	-81,4873	64	-64	0	-64	0	-85,3333	5,68889	-91,0222
9	0	0	0	0	3,001335	-65,1873	67,92129	-43,0566	3,92129	-43,0566	1,98058	-73,7367	6,93542	-83,0065
10	0	0	0	0	8,300231	-56,8896	75,5717	-31,1917	11,5717	-31,1917	2,93216	-71,7589	5,48405	-84,5881
11	0	0	0	0	11,71023	-54,413	81,55865	-26,9357	17,55865	-26,9357	-2,92731	-75,3331	-4,45727	-90,867
12	0	0	0	0	11,5281	-54,3249	82,74517	-26,5097	18,74517	-26,5097	-17,6731	-78,0129	-24,5402	-94,5915
13	0	0	0	0	8,384249	-53,6189	78,91355	-25,5916	14,91355	-25,5916	-39,1098	-73,541	-51,7601	-88,9554
14	0	0	0	0	4,234812	-50,799	72,31653	-21,0469	8,31653	-21,0469	-61,7479	-58,2325	-79,5986	-70,1599
15	0	0	0	0	1,080481	-46,0684	66,36588	-12,0152	2,36588	-12,0152	-78,931	-32,3481	-100,394	-38,8587

Continua.

Continuação da Tabela 7.5

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	0	-40,7437	64	0	0	0	-85,3333	0	-108,089	0
17	0	0	0	0			66,36588	12,01518	2,36588	12,01518	-78,931	32,34806	-100,394	38,85869
18	0	0	0	0			72,31653	21,04692	8,31653	21,04692	-61,7479	58,23245	-79,5986	70,15992
19	0	0	0	0			78,91355	25,59158	14,91355	25,59158	-39,1098	73,54097	-51,7601	88,9554
20	0	0	0	0			82,74517	26,50967	18,74517	26,50967	-17,6731	78,01289	-24,5402	94,59153
21	0	0	0	0			81,55865	26,93568	17,55865	26,93568	-2,92731	75,3331	-4,45727	90,86701
22	0	0	0	0			75,5717	31,19173	11,5717	31,19173	2,93216	71,75886	5,48405	84,58809
23	0	0	0	0			67,92129	43,05662	3,92129	43,05662	1,98058	73,73665	6,93542	83,00652
24	0	0	0	0			64	64	0	64	0	85,33333	5,68889	91,02222
25	0	0	0	0			69,82213	92,0401	5,82213	92,0401	4,51504	106,3923	9,63885	109,131
26	0	0	0	0			89,91863	121,7014	25,91863	121,7014	22,0614	132,0986	25,88857	132,8599
27	0	0	0	0			125,4579	145,1923	61,45787	145,1923	55,60499	154,1708	57,97719	153,9372
28	0	0	0	0			173,2548	154,5097	109,2548	154,5097	103,0065	163,3462	104,1847	162,8582
29	0	0	0	0			226,0699	143,8482	162,0699	143,8482	157,0988	152,3787	157,529	152,0256
30	0	0	0	0			274,1931	111,5566	210,1931	111,5566	207,421	118,5722	207,5148	118,4317
31	0	0	0	0			307,8907	60,99866	243,8907	60,99866	243,1021	65,00372	243,1083	64,98321



Número de Pontos Discretizados = 64

Intervalo entre cada ponto –  $\Delta t$ : 0,32s

Tabela 7.6

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66,67	62,22	256	0	288	0	256	0	256	0	256	0
1	175	175	233,3	248,9	243,1081	-64,9837	275,3035	-63,9883	243,3035	-63,9883	243,1078	-64,9849	243,1082	-64,9837
2	150	150	100	80	207,5058	-118,444	240,1737	-116,728	208,1737	-116,728	207,5006	-118,451	207,5059	-118,443
3	125	125	166,7	177,8	157,4376	-152,087	190,5806	-150,04	158,5806	-150,04	157,4175	-152,104	157,4387	-152,086
4	100	100	66,67	62,22	103,7529	-162,975	137,0966	-160,875	105,0966	-160,875	103,7105	-162,997	103,7574	-162,973
5	75	75	100	106,7	56,67752	-153,856	89,82944	-151,705	57,82944	-151,705	56,61996	-153,876	56,68763	-153,856
6	50	50	33,33	26,67	23,0562	-131,706	55,73459	-129,224	23,73459	-129,224	23,00658	-131,732	23,06959	-131,708
7	25	25	33,33	35,56	4,961391	-105,106	37,16135	-101,895	5,16135	-101,895	4,94109	-105,179	4,96949	-105,099
8	0	0	0	0	0	-81,4873	32	-77,2548	0	-77,2548	0	-81,6731	0	-81,4291
9	0	0	0	0	3,001335	-65,1873	35,20446	-59,9221	3,20446	-59,9221	2,96552	-65,5439	3,03118	-64,9977
10	0	0	0	0	8,300231	-56,8896	41,0003	-50,8675	9,0003	-50,8675	8,14316	-57,4261	8,49056	-56,4706
11	0	0	0	0	11,71023	-54,413	44,91787	-48,038	12,91787	-48,038	11,37094	-55,0721	12,32416	-53,7214
12	0	0	0	0	11,5281	-54,3249	44,95932	-47,8914	12,95932	-47,8914	11,0307	-55,0186	12,94429	-53,4857
13	0	0	0	0	8,384249	-53,6189	41,62136	-47,1323	9,62136	-47,1323	7,8573	-54,3125	10,98844	-53,0307
14	0	0	0	0	4,234812	-50,799	36,9695	-43,9616	4,9695	-43,9616	3,85383	-51,5999	8,22727	-51,1577
15	0	0	0	0	1,080481	-46,0684	33,29889	-38,4424	1,29889	-38,4424	0,94322	-47,2514	6,26817	-48,245

Continua.

Continuação da Tabela 7.6

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	0	-40,7437	32	-32	0	-32	0	-42,6667	5,68889	-45,5111
17	0	0	0	0	0,841205	-36,3161	33,06699	-26,4272	1,06699	-26,4272	0,63403	-39,2413	5,93836	-44,0139
18	0	0	0	0	2,5618	-33,6548	35,34703	-22,9147	3,34703	-22,9147	1,69053	-37,5686	5,91975	-43,9553
19	0	0	0	0	3,925036	-32,6847	37,29219	-21,5407	5,29219	-21,5407	2,08507	-37,2515	4,83139	-44,6377
20	0	0	0	0	4,150116	-32,5949	37,78585	-21,3817	5,78585	-21,3817	1,46608	-37,3455	2,74203	-45,0361
21	0	0	0	0	3,213012	-32,3737	36,64078	-21,1023	4,64078	-21,1023	0,33482	-37,115	0,5523	-44,6116
22	0	0	0	0	1,714924	-31,3467	34,5714	-19,6758	2,5714	-19,6758	-0,42869	-36,6316	-0,65275	-43,8576
23	0	0	0	0	0,459562	-29,4529	32,71682	-16,8654	0,71682	-16,8654	-0,35133	-36,8395	-0,50679	-44,2112
24	0	0	0	0	0	-27,1624	32	-13,2548	0	-13,2548	0	-39,0065	0	-47,2958
25	0	0	0	0	0,388973	-25,1369	32,66078	-9,85451	0,66078	-9,85451	-1,05967	-43,8194	-1,43132	-53,8335
26	0	0	0	0	1,227845	-23,8452	34,18404	-7,52306	2,18404	-7,52306	-5,72749	-50,5979	-7,58008	-62,7776
27	0	0	0	0	1,943674	-23,3393	35,62843	-6,51264	3,62843	-6,51264	-15,6481	-57,0809	-20,3983	-71,1644
28	0	0	0	0	2,117406	-23,2821	36,15826	-6,3652	4,15826	-6,3652	-30,8739	-59,9902	-39,7993	-74,8792
29	0	0	0	0	1,684825	-23,1771	35,48934	-6,19208	3,48934	-6,19208	-49,3709	-56,2055	-63,1355	-70,1111
30	0	0	0	0	0,922248	-22,6522	34,0194	-5,17113	2,0194	-5,17113	-67,3718	-44,0804	-85,6914	-54,9239
31	0	0	0	0	0,252974	-21,6397	32,5872	-2,98969	0,5872	-2,98969	-80,514	-24,3191	-102,088	-30,274

Continua.

Continuação da Tabela 7.6

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
32	0	0	0	0	0	-20,3718	32	0	0	0	-85,3333	0	-108,089	0
33	0	0	0	0			32,5872	2,98969	0,5872	2,98969	-80,514	24,31913	-102,088	30,27399
34	0	0	0	0			34,0194	5,17113	2,0194	5,17113	-67,3718	44,08036	-85,6914	54,92387
35	0	0	0	0			35,48934	6,19208	3,48934	6,19208	-49,3709	56,2055	-63,1355	70,11111
36	0	0	0	0			36,15826	6,3652	4,15826	6,3652	-30,8739	59,99015	-39,7993	74,87924
37	0	0	0	0			35,62843	6,51264	3,62843	6,51264	-15,6481	57,08094	-20,3983	71,16439
38	0	0	0	0			34,18404	7,52306	2,18404	7,52306	-5,72749	50,59787	-7,58008	62,77764
39	0	0	0	0			32,66078	9,85451	0,66078	9,85451	-1,05967	43,81939	-1,43132	53,8335
40	0	0	0	0			32	13,25483	0	13,25483	0	39,00645	0	47,29576
41	0	0	0	0			32,71682	16,86544	0,71682	16,86544	-0,35133	36,83946	-0,50679	44,2112
42	0	0	0	0			34,5714	19,67576	2,5714	19,67576	-0,42869	36,63159	-0,65275	43,85762
43	0	0	0	0			36,64078	21,10234	4,64078	21,10234	0,33482	37,11501	0,5523	44,61155
44	0	0	0	0			37,78585	21,38172	5,78585	21,38172	1,46608	37,34551	2,74203	45,03607
45	0	0	0	0			37,29219	21,54072	5,29219	21,54072	2,08507	37,25149	4,83139	44,63765
46	0	0	0	0			35,34703	22,91469	3,34703	22,91469	1,69053	37,56857	5,91975	43,9553
47	0	0	0	0			33,06699	26,42717	1,06699	26,42717	0,63403	39,24128	5,93836	44,01391

Continua.

Continuação da Tabela 7.6

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 1	ord. 2	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
48	0	0	0	0			32	32	0	32	0	42,66667	5,68889	45,51111
49	0	0	0	0			33,29889	38,44235	1,29889	38,44235	0,94322	47,2514	6,26817	48,24496
50	0	0	0	0			36,9695	43,96161	4,9695	43,96161	3,85383	51,59985	8,22727	51,15767
51	0	0	0	0			41,62136	47,1323	9,62136	47,1323	7,8573	54,31254	10,98844	53,03065
52	0	0	0	0			44,95932	47,89138	12,95932	47,89138	11,0307	55,01862	12,94429	53,48567
53	0	0	0	0			44,91787	48,03802	12,91787	48,03802	11,37094	55,07213	12,32416	53,7214
54	0	0	0	0			41,0003	50,86749	9,0003	50,86749	8,14316	57,42608	8,49056	56,47056
55	0	0	0	0			35,20446	59,92206	3,20446	59,92206	2,96552	65,54387	3,03118	64,99769
56	0	0	0	0			32	77,25483	0	77,25483	0	81,67311	0	81,4291
57	0	0	0	0			37,16135	101,8946	5,16135	101,8946	4,94109	105,1795	4,96949	105,0986
58	0	0	0	0			55,73459	129,2245	23,73459	129,2245	23,00658	131,7321	23,06959	131,7077
59	0	0	0	0			89,82944	151,7049	57,82944	151,7049	56,61996	153,8758	56,68763	153,8561
60	0	0	0	0			137,0966	160,8749	105,0966	160,8749	103,7105	162,9966	103,7574	162,9733
61	0	0	0	0			190,5806	150,0402	158,5806	150,0402	157,4175	152,1043	157,4387	152,086
62	0	0	0	0			240,1737	116,7277	208,1737	116,7277	207,5006	118,4514	207,5059	118,4434
63	0	0	0	0			275,3035	63,98834	243,3035	63,98834	243,1078	64,98491	243,1082	64,98365

Os resultados aqui encontrados para a transformada de Fourier do carregamento da figura 7.4 são muito similares àqueles obtidos para no exemplo precedente no que tange aos erros observado nas transformadas. Uma vez que este exemplo também possui uma descontinuidade na derivada de  $p(t)$  no tempo  $t=a$ , o resultado da transformada com ponderador de ordem 4 é mais impreciso do que o resultado com ponderador de ordem 2. No entanto, a medida que o número de pontos na discretização aumenta, a influencia das descontinuidade diminui. Para 64 pontos, os valores nas freqüências mais baixas com ponderador de ordem 4 são mais precisos que os obtidos com ponderador de ordem 2, apesar de divergirem mais rapidamente a medida que a freqüência aumenta.

Pode-se perceber, também, que os resultados para baixa freqüência obtidos com ponderador de ordem 2 na discretização com 16 pontos são mais precisos do que aqueles obtidos com ponderador de ordem 1 na discretização com 128 pontos.

### 7.3 Carregamento Cossenoidal

Como terceiro exemplo de aplicação toma-se o caso de um carregamento variando com o cosseno no intervalo de zero até ' $a$ ', como mostra a figura 7.7.

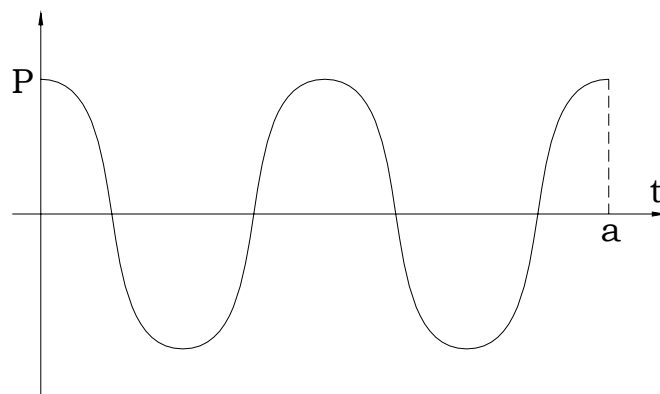


Figura 7.7 – Carregamento cossenoidal

Nesse caso, carregamento pode ser analiticamente expresso como:

$$P(t) = \begin{cases} 0, & t < 0; \\ P \cos(\omega_0 t), & 0 \leq t \leq a; \\ 0, & t > a; \end{cases} \quad (7.7)$$

que conduz à seguinte equação para transformada de Fourier:

$$P(\omega) = \int_0^a P \cos(\omega_0 t) e^{-i\omega t} dt \quad (7.8)$$

resultando em

$$\begin{aligned} P_{\text{Re}}(\omega) &= \frac{Pa}{2} \left[ \frac{\text{sen}(\omega - \omega_0)a}{(\omega - \omega_0)a} + \frac{\text{sen}(\omega + \omega_0)a}{(\omega + \omega_0)a} \right] \\ P_{\text{Im}}(\omega) &= \frac{Pa}{2} \left[ \frac{\cos(\omega - \omega_0)a - 1}{(\omega - \omega_0)a} + \frac{\cos(\omega + \omega_0)a - 1}{(\omega + \omega_0)a} \right] \end{aligned} \quad (7.9)$$

Os valores numéricos dos parâmetros de interesse são novamente dados por:

$$\begin{aligned} T &= 20,48s \\ \omega &= 0,306796 \text{ Hz} \\ P &= 200 \\ a &= 2,56s \\ \omega_0 &= \frac{2\pi}{a} \end{aligned}$$

A resposta exata, obtida por (7.9) está mostrada nas figuras 7.8 e 7.9.

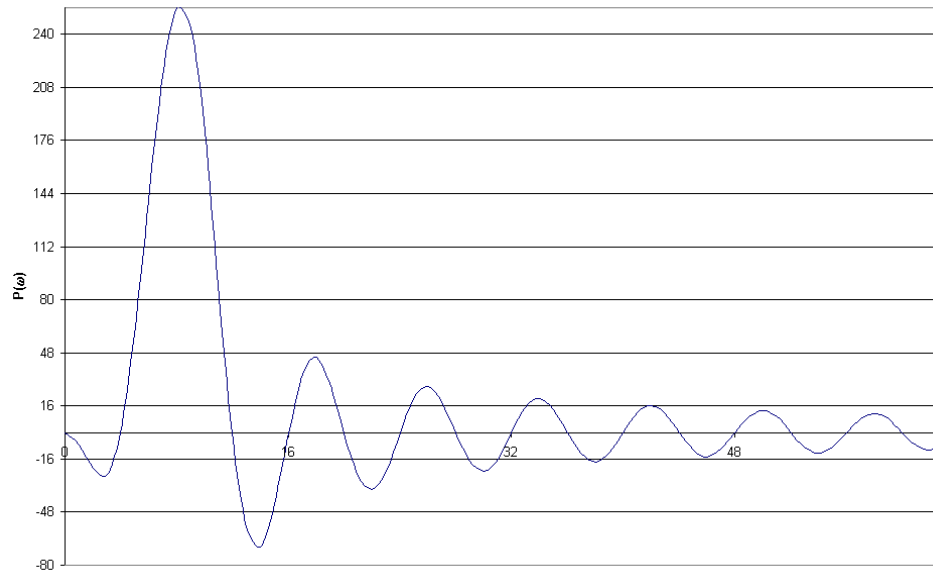


Figura 7.8 – Solução Exata para Carregamento Cosseniodal – Parte Real

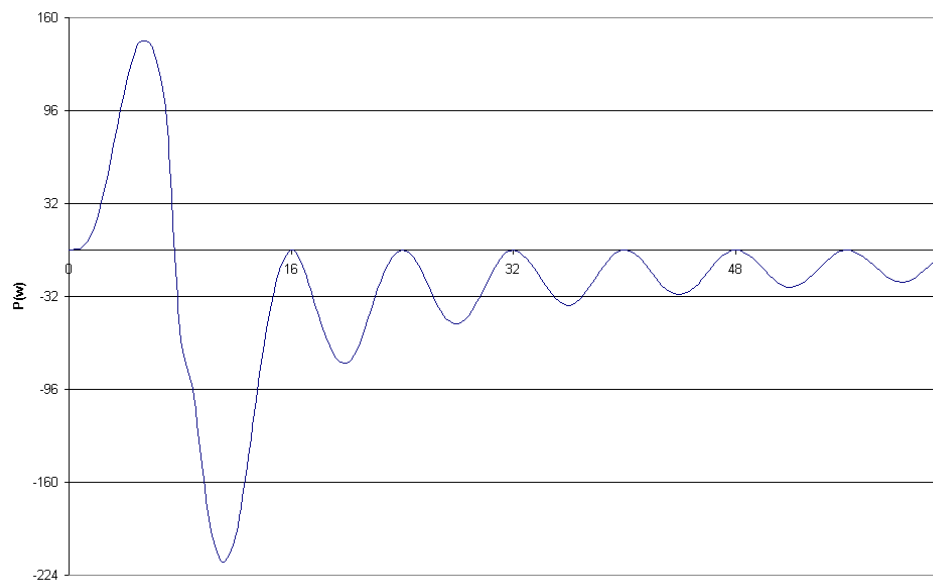


Figura 7.9 – Solução Exata para Carregamento Cosseniodal – Parte Imaginária

As tabelas 7.7 a 7.9 mostram os valores da transformada de Fourier do carregamento descrito em (7.7), levando-se em conta três valores para o passo  $\Delta t$ , e igualmente três ordens de ponderadores.

Número de Pontos Discretizados = 16

Intervalo entre cada ponto –  $\Delta t$ : 1,28s

Tabela 7.7

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66.67	62.22	-2E-14	0	256	0	0	0	-170.667	0	79.64444	0
1	-200	-200	-266.7	-200	-7.31686	3.030741	200.5062	-83.0524	-18.0035	7.45729	-169.678	70.28283	24.15062	-83.0524
2	200	100	66.67	200	-21.73	21.72995	74.98066	-74.9807	-53.0193	53.01934	-156.026	156.0258	-101.375	-74.9807
3	0	0	0	0	-25.1434	60.70149	-22.9863	55.49382	-60.4766	146.0035	-105.629	255.0111	-199.342	55.49382
4	0	0	0	0	2E-14	108.6498	0	256	0	256	0	341.3333	-176.356	256
5	0	0	0	0	59.09769	142.6744	172.9476	417.5325	135.4573	327.0228	155.6162	375.6907	-3.40793	417.5325
6	0	0	0	0	139.6926	139.6926	437.0193	437.0193	309.0193	309.0193	326.6925	326.6925	260.6638	437.0193
7	0	0	0	0	215.1156	89.10379	673.5325	278.9863	455.0228	188.4766	461.024	190.9624	497.1769	278.9863
8	0	0	0	0	256	-54.3249	768	0	512	0	512	0	591.6444	0
9	0	0	0	0			673.5325	-278.986	455.0228	-188.477	461.024	-190.962	497.1769	-278.986
10	0	0	0	0			437.0193	-437.019	309.0193	-309.019	326.6925	-326.692	260.6638	-437.019
11	0	0	0	0			172.9476	-417.533	135.4573	-327.023	155.6162	-375.691	-3.40793	-417.533
12	0	0	0	0			0	-256	0	-256	0	-341.333	-176.356	-256
13	0	0	0	0			-22.9863	-55.4938	-60.4766	-146.003	-105.629	-255.011	-199.342	-55.4938
14	0	0	0	0			74.98066	74.98066	-53.0193	-53.0193	-156.026	-156.026	-101.375	74.98066
15	0	0	0	0			200.5062	83.05238	-18.0035	-7.45729	-169.678	-70.2828	24.15062	83.05238



Número de Pontos Discretizados = 32

Intervalo entre cada ponto –  $\Delta t$ : 0,64s

Tabela 7.8

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66.67	62.22	-2E-14	0	128	0	0	0	0	0	11.37778	0
1	1E-14	0	0	0	-7.31686	3.030741	100.2531	-41.5262	-9.00175	3.72865	-6.00116	2.48576	4.91061	-2.03404
2	-200	-200	-133.3	-106.7	-21.73	21.72995	37.49033	-37.4903	-26.5097	26.50967	-17.6731	17.67311	-8.4496	8.4496
3	-4E-14	0	0	0	-25.1434	60.70149	-11.4932	27.74691	-30.2383	73.00175	-20.1589	48.66783	-14.4609	34.91161
4	200	100	66.67	62.22	2E-14	108.6498	0	128	0	128	0	85.33333	0	68.26667
5	0	0	0	0	59.09769	142.6744	86.47381	208.7663	67.72865	163.5114	45.15243	109.0076	37.78818	91.22874
6	0	0	0	0	139.6926	139.6926	218.5097	218.5097	154.5097	154.5097	103.0065	103.0065	88.09405	88.09405
7	0	0	0	0	215.1156	89.10379	336.7663	139.4932	227.5114	94.23831	151.6743	62.82554	131.051	54.28309
8	0	0	0	0	256	-54.3249	384	0	256	0	170.6667	0	147.9111	0
9	0	0	0	0	244.0387	-101.084	336.7663	-139.493	227.5114	-94.2383	151.6743	-62.8255	131.051	-54.2831
10	0	0	0	0	181.083	-181.083	218.5097	-218.51	154.5097	-154.51	103.0065	-103.006	88.09405	-88.0941
11	0	0	0	0	88.95757	-214.763	86.47381	-208.766	67.72865	-163.511	45.15243	-109.008	37.78818	-91.2287
12	0	0	0	0	2.72E-14	-195.57	0	-128	0	-128	0	-85.33333	0	-68.2667
13	0	0	0	0	-57.0715	-137.783	-11.4932	-27.7469	-30.2383	-73.0018	-20.1589	-48.6678	-14.4609	-34.9116
14	0	0	0	0	-69.1408	-69.1408	37.49033	37.49033	-26.5097	-26.5097	-17.6731	-17.6731	-8.4496	-8.4496
15	0	0	0	0	-42.9468	-17.7891	100.2531	41.52619	-9.00175	-3.72865	-6.00116	-2.48576	4.91061	2.03404

Continua.

Continuação da Tabela 7.8

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	-2E-14	0	128	0	0	0	0	0	11.37778	0
17	0	0	0	0			100.2531	-41.5262	-9.00175	3.72865	-6.00116	2.48576	4.91061	-2.03404
18	0	0	0	0			37.49033	-37.4903	-26.5097	26.50967	-17.6731	17.67311	-8.4496	8.4496
19	0	0	0	0			-11.4932	27.74691	-30.2383	73.00175	-20.1589	48.66783	-14.4609	34.91161
20	0	0	0	0			0	128	0	128	0	85.33333	0	68.26667
21	0	0	0	0			86.47381	208.7663	67.72865	163.5114	45.15243	109.0076	37.78818	91.22874
22	0	0	0	0			218.5097	218.5097	154.5097	154.5097	103.0065	103.0065	88.09405	88.09405
23	0	0	0	0			336.7663	139.4932	227.5114	94.23831	151.6743	62.82554	131.051	54.28309
24	0	0	0	0			384	0	256	0	170.6667	0	147.9111	0
25	0	0	0	0			336.7663	-139.493	227.5114	-94.2383	151.6743	-62.8255	131.051	-54.2831
26	0	0	0	0			218.5097	-218.51	154.5097	-154.51	103.0065	-103.006	88.09405	-88.0941
27	0	0	0	0			86.47381	-208.766	67.72865	-163.511	45.15243	-109.008	37.78818	-91.2287
28	0	0	0	0			0	-128	0	-128	0	-85.3333	0	-68.2667
29	0	0	0	0			-11.4932	-27.7469	-30.2383	-73.0018	-20.1589	-48.6678	-14.4609	-34.9116
30	0	0	0	0			37.49033	37.49033	-26.5097	-26.5097	-17.6731	-17.6731	-8.4496	-8.4496
31	0	0	0	0			100.2531	41.52619	-9.00175	-3.72865	-6.00116	-2.48576	4.91061	2.03404

Número de Pontos Discretizados = 64

Intervalo entre cada ponto –  $\Delta t$ : 0,32s

Tabela 7.9

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
0	200	100	66.67	62.22	-2E-14	0	64	0	0	0	0	0	0	0
1	141.4	141.4	188.6	201.1	-7.31686	3.030741	46.92855	-19.4384	-7.69887	3.18898	-7.26458	3.00909	-7.34881	3.04398
2	1E-14	0	0	0	-21.73	21.72995	9.18896	-9.18896	-22.811	22.81104	-21.5782	21.57816	-21.8385	21.8385
3	-141.4	-141.4	-188.6	-201.1	-25.1434	60.70149	-16.9185	40.84485	-26.2911	63.47227	-24.9753	60.29577	-25.2964	61.07097
4	-200	-200	-133.3	-124.4	2E-14	108.6498	0	112.9835	0	112.9835	0	107.978	0	109.4876
5	-141.4	-141.4	-188.6	-201.1	59.09769	142.6744	70.38864	169.9332	61.01606	147.3058	58.77853	141.9039	59.68693	144.097
6	-4E-14	0	0	0	139.6926	139.6926	174.9547	174.9547	142.9547	142.9547	139.103	139.103	141.5094	141.5094
7	141.4	141.4	188.6	201.1	215.1156	89.10379	272.4405	112.8486	217.8131	90.22113	214.5803	88.88207	218.774	90.61917
8	200	100	66.67	62.22	256	-54.3249	320	0	256	0	256	0	261.6889	0
9	0	0	0	0	244.0387	-101.084	295.1774	-122.266	240.55	-99.6391	244.8962	-101.439	251.111	-104.014
10	0	0	0	0	181.083	-181.083	207.5816	-207.582	175.5816	-175.582	182.6056	-182.606	187.9122	-187.912
11	0	0	0	0	88.95757	-214.763	94.03423	-227.019	84.66165	-204.391	90.30598	-218.018	93.31622	-225.285
12	0	0	0	0	2.72E-14	-195.57	0	-182.257	0	-182.257	0	-200.342	0	-208.009
13	0	0	0	0	-57.0715	-137.783	-42.5755	-102.786	-51.948	-125.414	-59.1846	-142.884	-61.7863	-149.165
14	0	0	0	0	-69.1408	-69.1408	-29.2971	-29.2971	-61.2971	-61.2971	-72.8929	-72.8929	-76.5742	-76.5742
15	0	0	0	0	-42.9468	-17.7891	17.65673	7.31366	-36.9707	-15.3138	-46.2937	-19.1755	-48.9798	-20.2881

Continua.

Continuação da Tabela 7.9

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
16	0	0	0	0	-2E-14	0	64	0	0	0	0	0	0	0
17	0	0	0	0	34.82824	-14.4263	82.59635	-34.2125	27.96894	-11.5851	40.2925	-16.6897	43.37874	-17.9681
18	0	0	0	0	45.13144	-45.1314	66.78744	-66.7874	34.78744	-34.7874	55.21981	-55.2198	60.07934	-60.0793
19	0	0	0	0	29.48915	-71.1931	31.0823	-75.0393	21.70972	-52.4119	39.02573	-94.2165	42.97137	-103.742
20	0	0	0	0	-7E-14	-77.607	0	-54.2566	0	-54.2566	0	-115.009	0	-128.365
21	0	0	0	0	-25.6769	-61.9896	-7.56042	-18.2525	-16.933	-40.8799	-45.1536	-109.01	-51.174	-123.545
22	0	0	0	0	-34.1471	-34.1471	10.92808	10.92808	-21.0719	-21.0719	-79.5991	-79.5991	-91.7728	-91.7728
23	0	0	0	0	-22.8003	-9.44418	41.58886	17.22667	-13.0386	-5.40075	-93.2219	-38.6138	-109.548	-45.3764
24	0	0	0	0	-5.6E-14	0	64	0	0	0	-85.3333	0	-102.4	0
25	0	0	0	0	20.54198	-8.50877	64.32576	-26.6446	9.69834	-4.01718	-62.906	26.05653	-77.2114	31.982
26	0	0	0	0	27.69504	-27.695	43.555	-43.555	11.555	-11.555	-36.0966	36.09656	-45.3701	45.37009
27	0	0	0	0	18.71575	-45.1838	16.08517	-38.833	6.71259	-16.2056	-13.6261	32.89631	-17.5447	42.35657
28	0	0	0	0	-4.1E-14	-50.7032	0	-15.0165	0	-15.0165	0	22.64464	0	29.84317
29	0	0	0	0	-17.2045	-41.5353	5.42534	13.09794	-3.94724	-9.52948	4.81645	11.62794	6.48147	15.64766
30	0	0	0	0	-23.3935	-23.3935	28.30137	28.30137	-3.69863	-3.69863	3.90505	3.90505	5.34359	5.34359
31	0	0	0	0	-15.9307	-6.5987	53.32454	22.08775	-1.30288	-0.53967	1.26341	0.52332	1.74772	0.72393

Continua

Continuação da Tabela 7.9

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
32	0	0	0	0	8.98E-15	0	64	0	0	0	0	0	0	0
33	0	0	0	0			53.32454	-22.0878	-1.30288	0.53967	1.26341	-0.52332	1.74772	-0.72393
34	0	0	0	0			28.30137	-28.3014	-3.69863	3.69863	3.90505	-3.90505	5.34359	-5.34359
35	0	0	0	0			5.42534	-13.0979	-3.94724	9.52948	4.81645	-11.6279	6.48147	-15.6477
36	0	0	0	0			0	15.01652	0	15.01652	0	-22.6446	0	-29.8432
37	0	0	0	0			16.08517	38.83304	6.71259	16.20562	-13.6261	-32.8963	-17.5447	-42.3566
38	0	0	0	0			43.555	43.555	11.555	11.555	-36.0966	-36.0966	-45.3701	-45.3701
39	0	0	0	0			64.32576	26.6446	9.69834	4.01718	-62.906	-26.0565	-77.2114	-31.982
40	0	0	0	0			64	0	0	0	-85.3333	0	-102.4	0
41	0	0	0	0			41.58886	-17.2267	-13.0386	5.40075	-93.2219	38.61377	-109.548	45.37639
42	0	0	0	0			10.92808	-10.9281	-21.0719	21.07192	-79.5991	79.59911	-91.7728	91.77282
43	0	0	0	0			-7.56042	18.25246	-16.933	40.87988	-45.1536	109.0103	-51.174	123.5448
44	0	0	0	0			0	54.25658	0	54.25658	0	115.0088	0	128.3649
45	0	0	0	0			31.0823	75.03931	21.70972	52.4119	39.02573	94.21645	42.97137	103.7421
46	0	0	0	0			66.78744	66.78744	34.78744	34.78744	55.21981	55.21981	60.07934	60.07934
47	0	0	0	0			82.59635	34.21253	27.96894	11.58511	40.2925	16.6897	43.37874	17.96806

Continua.

Continuação da Tabela 7.9

i	p(t)				Exata		FFT		Ponderada ordem 1		Ponderada ordem 2		Ponderada ordem 4	
	orig.	ord. 2	ord. 3	ord. 4	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.	Real	Imag.
48	0	0	0	0			64	0	0	0	0	0	0	0
49	0	0	0	0			17.65673	-7.31366	-36.9707	15.31376	-46.2937	19.17546	-48.9798	20.28811
50	0	0	0	0			-29.2971	29.29711	-61.2971	61.29711	-72.8929	72.89292	-76.5742	76.57424
51	0	0	0	0			-42.5755	102.7862	-51.948	125.4136	-59.1846	142.8843	-61.7863	149.1654
52	0	0	0	0			0	182.2566	0	182.2566	0	200.3421	0	208.0094
53	0	0	0	0			94.03423	227.0187	84.66165	204.3913	90.30598	218.0179	93.31622	225.2853
54	0	0	0	0			207.5816	207.5816	175.5816	175.5816	182.6056	182.6056	187.9122	187.9122
55	0	0	0	0			295.1774	122.2665	240.55	99.63906	244.8962	101.4393	251.111	104.0136
56	0	0	0	0			320	0	256	0	256	0	261.6889	0
57	0	0	0	0			272.4405	-112.849	217.8131	-90.2211	214.5803	-88.8821	218.774	-90.6192
58	0	0	0	0			174.9547	-174.955	142.9547	-142.955	139.103	-139.103	141.5094	-141.509
59	0	0	0	0			70.38864	-169.933	61.01606	-147.306	58.77853	-141.904	59.68693	-144.097
60	0	0	0	0			0	-112.983	0	-112.983	0	-107.978	0	-109.488
61	0	0	0	0			-16.9185	-40.8449	-26.2911	-63.4723	-24.9753	-60.2958	-25.2964	-61.071
62	0	0	0	0			9.18896	9.18896	-22.811	-22.811	-21.5782	-21.5782	-21.8385	-21.8385
63	0	0	0	0			46.92855	19.43844	-7.69887	-3.18898	-7.26458	-3.00909	-7.34881	-3.04398

Para o carregamento cossenoidal descrito em (7.7), uma discretização com apenas 3 pontos, como aquela feita na tabela 7.7, é muito pobre, resultado em resultados muito aquém do esperado. Na medida que o número de pontos aumenta, os resultados se tornam mais precisos e novamente prevalecem aqueles obtidos com ponderadores de ordem maior.

## CAPITULO VIII

### **CONCLUSÃO**

O uso da Transformada de Fourier e da sua versão computacional, a Transformada Rápida de Fourier – FFT vem se tornando cada vez mais comum no âmbito da dinâmica das estruturas. Graças a computadores cada vez mais potentes, tais técnicas têm se tornado vantajosas para um número cada vez maior de problemas. Esta é uma verdade não apenas para a Dinâmica das Estruturas, mas também em diversos outros ramos da engenharia, especialmente na Engenharia Elétrica, onde das Transformadas de Fourier e Laplace, entre outras, tem encontrado terreno fértil para seu desenvolvimento há muitos anos.

No entanto, por mais que os computadores atuais possam processar vetores de carregamento cada vez maiores, ainda é fundamental o desenvolvimento de algoritmos cada vez mais eficientes para a resolução das transformadas.

O uso dos ponderadores de ordem superior no processo de resolução das Transformadas de Fourier resolve alguns dos problemas relacionados à parte computacional do processo, mas cria outros. Com o uso deles é possível obter resultados muito mais precisos comparativamente ao uso das transformadas de Fourier convencionais, conforme mostrado no capítulo VII.

Mesmo utilizando-se um número pequeno de pontos – 16 ou 32 – foi possível obter respostas mais precisas que aquelas obtidas com uma discretização maior, de 64 ou até mesmo de 128 pontos. Considerando-



se que a aplicação do ponderador limita-se à multiplicação do vetor de entrada por um conjunto de ponderadores, o número de operações necessárias para se atingir a precisão desejada é reduzido de forma considerável – resultados onde era necessário um vetor de 128 pontos, com o uso de ponderadores podem ser atingidos com apenas 32 pontos. Tal diminuição produz um aumento expressivo na velocidade do processo de resolução das Transformadas.

Deve-se lembrar, no entanto, que a redução no número de pontos acarreta numa solução mais limitada no domínio da frequência. Apenas os movimentos e excitações de baixa frequência podem ser plenamente analisados, uma vez que o domínio de frequências analisado depende diretamente do intervalo  $\Delta t$ , e este é função do número de pontos presentes na integração. Para uma análise de frequências mais altas, o uso de uma discretização mais apurada – menores valores para  $\Delta t$  – ainda se faz necessário.

Por outro lado, os ponderadores acarretam numa limitação ao problema no que tange ao leque de funções que podem ser analisadas. Os problemas inerentes aos métodos de integração numérica relacionados ao uso dos ponderadores de Newton-Cotes permitem que apenas funções contínuas e de primeira derivada contínua no intervalo analisado possam ser utilizadas.

Funções que possuam descontinuidades no intervalo a ser analisado impõem a aplicação dos ponderadores individualmente em cada subintervalo onde a função e sua derivada são contínuas. Uma vez que, a partir de um intervalo já discretizado, não é possível determinar com segurança os pontos de descontinuidade na função original, torna-se extremamente difícil o desenvolvimento de um algoritmo genérico, que possa ser aplicado a qualquer função.

Funções que possuam carregamentos contínuos, como o efeito de motores sobre estruturas, são casos onde os ponderadores podem ser aplicados. Porém, funções de intensidade aleatória no tempo, muito

comuns no estudo da dinâmica das estruturas, como o efeito de ventos ou terremotos, dificilmente poderão se valer dos métodos abordados neste trabalho.

No entanto, existem aplicações para o que foi estudado aqui em outros ramos da engenharia. Na engenharia elétrica e eletrônica, tanto o uso da FFT, utilizada largamente em processadores digitais de sinais (DSP), por exemplo, quanto o uso de funções contínuas que atendem as exigências e limitações dos ponderadores de Newton-Cotes, são muito comuns. Com as técnicas descritas neste trabalho é possível, por exemplo, ampliar a precisão e a velocidade de processamento de sistemas que usam algoritmos e/ou processadores baseados na FFT, como é o caso de analisadores de espectro ou analisadores de imagem.

## CAPÍTULO IX

**BIBLIOGRAFIA**

- CLOUGH, R. W.; PENZIEN, J. (1975) *Dynamics of Structures*. McGraw-Hill. Nova Iorque. p.634
- COOLEY, J. W. & TUKEY, J. W. (1965) *An Algorithm for the Machine Calculation of Complex Fourier Series*. Math Comput. v. 19. p. 297-301.
- HALL, J. F. (1982) *An FFT Algorithm for Structural Dynamics*. Earthquake Engineering and Structural Dynamics. v. 10. p. 797-811.
- HASSAN, J. C. M. (1987) *Sobre o Emprego dos Métodos Numéricos na Solução da Equação Diferencial da Dinâmica das Estruturas*. Dissertação (Mestrado). Escola de Engenharia de São Carlos. Universidade de São Paulo. São Carlos. 1987
- HUMAR, J. L. (1990) *Dynamics of Structures*. Prentice Hall. Englewood Cliffs. p. 780.
- KRYLOV, V. I. (1962) *Aproximate Calculation of Integrals*. Tradução de Arthur H. Stroud. The MacMillan Company. Nova Iorque. p. 357.

LAIER, J. E. (1978) *Análise das vibrações livres de Edifícios pela Técnica do Meio Contínuo*. Tese (doutorado). Escola de Engenharia de São Carlos. Universidade de São Paulo. 1978.

PAPOULIS, A. (1962) *The Fourier Integral and Its Applications*. McGraw-Hill. Nova Iorque. p.318.

POLLOCK, D.S.G. (1999) *A Handbook of Time-Series Analysis, Signal Processing and Dynamics*. Academic Press. San Diego. p.733.

**ANEXO A – Algoritmos**

Segue a seguir, uma transcrição dos algoritmos de Hall, convertidos para a linguagem C, e o algoritmo para aplicação dos ponderadores de Newton-Cotes em um vetor real.

### ALGORITMO DE HALL

```
#include "math.h"
```

```
void FFT_Hall(double R[], int M, int L, int k, int IT, int TW);
void LPSTOR(int NS, double R[], int LS, double LF, int INCD,
            int INC, double CC);
void Sort2(double R[], int Nk, int TW, int Ini);
void Sort3(double R[], int MK, int ISRT, int TW);
```

```
void FFT_Hall(double R[], int M, int L, int k, int IT, int TW)
{
  /*
```

Calcula tanto a transformada de fourier ( $E^-$ ) de um vetor real R de dimensão real  $N=(2^M)*L$  onde  $L=2$  ou  $3$  ( $IT=0$ ,  $K=0$ ,  $TW$ =intervalo de tempo= $T/N$  onde  $T$  é o período de cálculo) ou a transformada inversa de fourier ( $E^+$ ) de um vetor complexo simétrico de dimensão complexa  $N$  ( $IT=1$ ,  $K$ =veja abaixo,  $TW$ =intervalo de frequência em  $Hz=1/T$ ).

Chamando  $IT=0$ :  $R(1), R(2), R(3), \dots, R(N)$  = valores reais correspondentes aos tempos  $0, TW, 2TW, \dots, (N-1)TW$ .

Chamando  $IT=1$  ou retornando  $IT=0$ : apenas os termos de frequências não negativos são armazenados.  $R(1)$ =termo real correspondente à frequência  $0$ ;  $R(2)$ =parte real do termo de valor complexo correspondendo à máxima frequência em  $Hz$  de  $N/2/T$ ;

$R(3), R(5), R(7), \dots, R(N-1)$ =partes reais dos termos correspondendo às frequências intermediárias;  $R(4), R(6), R(8), \dots, R(N)$ =as partes imaginárias.

Retornando  $IT=1$ :  $R(1), R(2), R(3), \dots, R(N/J)$ =os  $NK=(2^{(M-K)})*L$  valores reais correspondentes aos tempos  $0, J*TW, 2J*TW, \dots, (N-J)*TW$  onde  $J=2^K$  e onde  $M$  deve ser maior que  $K$ . Desconsidere  $R(N/J+1) \dots R(N)$  se  $K$  exceder  $0$ .

Os termos exponenciais são computados recursivamente em grupos; cada termo  $J_e$  calculado usa uma chamada a seno ou cosseno. Um  $J_e$  pequeno gasta tempo de execução e um  $J_e$  grande introduz erros inaceitáveis.  $J_e$  foi setado para  $16$ , mas pode ser alterado como desejado ( $J_e=1, 2, 3, \dots$ ).

```
*/
```

```
int y, YF, Z, Z2, Z3, Z4, M1;
int JE, JI;
double S2, S3, PI;
int Nk, i, j, INC, i2, INCD, jf, LS;
double pid, c, s, cc, ss, ang, ct;
```

```

double R1, R2, R3, R4;

JE = 16;
JI = 2 * JE;
Nk = 2 ^ (M - k) * L;
if (Nk < 4) return;
S2 = 1 / sqrt(2);
S3 = sqrt(3) / (2 - IT);
PI = 4 * atan(1);
if (IT != 1)
{
    if (L == 2) Sort2(R, Nk, TW, 0);
    if (L == 3) Sort3(R, M, 0, TW);
    LPSTOR(L - 1, R, 0, Nk, 0, 0, S3);
}
if (M != 1)
{
    M1 = M - 1;
    for (i = 1; i <= M1; i++)
    {
        if (IT == 0) INC = 2 ^ i * L;
        if (IT == 1) INC = 2 ^ (M - i) * L;
        i2 = INC - 2;
        INCD = INC * 2;
        if (k >= i) LPSTOR(3, R, 0, i2, INCD, INC, 0);
        if (k >= i) break;
        pid = PI / INCD;
        c = cos(2 * pid);
        s = sin(2 * pid);
        LPSTOR(4 + IT, R, 0, Nk, INCD, INC, 0);
        if (INC != 4)
        {
            jf = INC / 2 - 1;
            for (j = 3; j <= jf; j += JI)
            {
                ang = (j - 3) * pid;
                cc = cos(ang);
                ss = sin(ang);
                if (j + JI - 2 <= jf) YF = j + JI - 2;
                else YF = jf;
                for (y = j; y <= YF; y += 2)
                {
                    Z3 = -i2 - y;
                    ct = cc;
                    cc = ct * c - ss * s;
                    ss = ss * c + s * ct;
                    if (IT != 1)
                    {
                        for (Z = y; Z <= Nk; Z += INCD)
                        {
                            Z2 = Z + 1;
                            Z3 = Z3 + INCD;
                            Z4 = Z3 + 1;
                            R1 = R[Z + INC] * c + R[Z2 + INC] * ss;
                            R2 = R[Z2 + INC] * c - R[Z + INC] * ss;
                            R3 = R[Z3 + INC] * ss + R[Z4 + INC] * c;
                            R4 = R[Z4 + INC] * ss - R[Z3 + INC] * c;
                            R[Z + INC] = R[Z3] - R3;
                            R[Z2 + INC] = -R[Z4] + R4;
                            R[Z3 + INC] = R[Z] - R1;
                            R[Z4 + INC] = -R[Z2] + R2;
                        }
                    }
                }
            }
        }
    }
}

```

```

        R[Z] = R[Z] + R1;
        R[Z2] = R[Z2] + R2;
        R[Z3] = R[Z3] + R3;
        R[Z4] = R[Z4] + R4;
    }
}
else
{
    for (Z = y; Z <= Nk; Z += INCD)
    {
        Z2 = Z + 1;
        Z3 = Z3 + INCD;
        Z4 = Z3 + 1;
        R1 = R[Z] - R[Z3 + INC];
        R2 = R[Z2] + R[Z4 + INC];
        R3 = R[Z3] - R[Z + INC];
        R4 = R[Z4] + R[Z2 + INC];
        R[Z] = R[Z] + R[Z3 + INC];
        R[Z2] = R[Z2] - R[Z4 + INC];
        R[Z3] = R[Z3] + R[Z + INC];
        R[Z4] = R[Z4] - R[Z2 + INC];
        R[Z + INC] = R1 * c - R2 * ss;
        R[Z2 + INC] = R2 * c + R1 * ss;
        R[Z3 + INC] = R3 * ss - R4 * c;
        R[Z4 + INC] = R4 * ss + R3 * c;
    }
}
}
}
if (INC == 6) break;

}
LS = INC / 2 + 1;
LPSTOR(6 + IT, R, LS, Nk, INCD, INC, S2);
}
}
if (IT == 0) return;
LPSTOR(6 + L, R, 0, Nk, 0, 0, S3);
if (L == 2) Sort2(R, Nk, TW, 0);
if (L == 3) Sort3(R, M - k, 1, TW);
}

void LPSTOR(int NS, double R[], int LS, double LF, int INCD,
            int INC, double CC)
{
    int L, L1, l2;
    double R1, R2, R3, R4, R5, R6;
    switch(NS)
    {
    case 1:
        for (L = 1; L <= LF; L += 4)
        {
            R1 = R[L] + R[L + 1];
            R2 = R[L + 2] + R[L + 3];
            R[L + 3] = -R[L + 2] + R[L + 3];
            R[L + 2] = R[L] - R[L + 1];
            R[L] = R1 + R2;
            R[L + 1] = R1 - R2;
        }
        break;
    case 2:

```



```

for (L = 1; L <= LF; L += 6)
{
  R1 = R[L] + R[L + 3];
  R2 = R[L] - R[L + 3];
  R3 = R[L + 1] + R[L + 2];
  R4 = R[L + 1] - R[L + 2];
  R5 = R[L + 4] + R[L + 5];
  R6 = R[L + 4] - R[L + 5];
  R[L] = R1 + R3 + R5;
  R[L + 1] = R2 + R4 + R6;
  R[L + 2] = R2 - (R4 + R6) * 0.5;
  R[L + 3] = (-R3 + R5) * CC;
  R[L + 4] = R1 - (R3 + R5) * 0.5;
  R[L + 5] = (R4 - R6) * CC;
}
break;
case 3:
R[1] = R[1] + R[2];
R[2] = R[INC + 1] + R[INC + 1];
for (L = 1; L <= LF; L += 2)
{
  R[L + 2] = R[L + 2] + R[INCD - L];
  R[L + 3] = R[L + 3] - R[INCD - L + 1];
}
break;
case 4:
for (L = 1; L <= LF; L += INCD)
{
  L1 = L + INC;
  R3 = R[L1];
  R[L1] = R[L + 1];
  R[L1 + 1] = -R[L1 + 1];
  R[L + 1] = R[L] - R3;
  R[L] = R[L] + R3;
}
break;
case 5:
for (L = 1; L <= LF; L += INCD)
{
  L1 = L + INC;
  R3 = R[L1];
  R[L1] = R[L] - R[L + 1];
  R[L1 + 1] = -R[L1 + 1] - R[L1 + 1];
  R[L] = R[L] + R[L + 1];
  R[L + 1] = R3 + R3;
}
break;
case 6:
for (L = LS; L <= LF; L += INCD)
{
  l2 = L + 1;
  R1 = (R[L + INC] + R[l2 + INC]) * CC;
  R2 = (R[l2 + INC] - R[L + INC]) * CC;
  R[L + INC] = R[L] - R1;
  R[l2 + INC] = -R[l2] + R2;
  R[L] = R[L] + R1;
  R[l2] = R[l2] + R2;
}
break;
case 7:
for (L = LS; L <= LF; L += INCD)

```

```

    {
        l2 = L + 1;
        R1 = R[L] - R[L + INC];
        R2 = R[l2] + R[l2 + INC];
        R[L] = R[L] + R[L + INC];
        R[l2] = R[l2] - R[l2 + INC];
        R[L + INC] = (R1 - R2) * CC;
        R[l2 + INC] = (R1 + R2) * CC;
    }
    break;
case 8:
    for (L = 1; L <= LF; L += 4)
    {
        R1 = R[L] + R[L + 1];
        R2 = R[L] - R[L + 1];
        R3 = R[L + 2] + R[L + 2];
        R4 = R[L + 3] + R[L + 3];
        R[L] = R1 + R3;
        R[L + 1] = R1 - R3;
        R[L + 2] = R2 - R4;
        R[L + 3] = R2 + R4;
    }
    break;
case 9:
    for (L = 1; L <= LF; L += 6)
    {
        R1 = R[L] + R[L + 1];
        R2 = R[L] - R[L + 1];
        R3 = R[L + 2] + R[L + 4];
        R4 = R[L + 2] - R[L + 4];
        R5 = (R[L + 3] + R[L + 5]) * CC;
        R6 = (R[L + 3] - R[L + 5]) * CC;
        R[L] = R1 + R3 + R3;
        R[L + 1] = R1 - R3 - R6;
        R[L + 4] = R1 - R3 + R6;
        R[L + 2] = R2 + R4 - R5;
        R[L + 3] = R2 - R4 - R4;
        R[L + 5] = R2 + R4 + R5;
    }
    break;
}
}

void Sort2(double R[], int Nk, int TW, int Ini)
{
    int nf, nt, i, j, ic, nk2, nk8, nsub;
    double stor;
    int NVEC[8] = {4, 2, 6, 1, 5, 3, 7, 0};

    nf = 1;
    nk2 = Nk / 2;
    nk8 = Nk / 8;
    if (Nk < 32) nk8 = Nk - 1;
    for (i = 1; i <= nk8; i++)
    {
        ic = 0;
        nt = nf;
        for (j = i; j <= Nk; j += nk8)
        {
            if (nt <= j) break;
            stor = R[Ini + nt];

```

```

    R[Ini + nt] = R[Ini + j];
    R[Ini + j] = stor;
    nt = nf + NVEC[ic];
    ic = ic + 1;
}
nsub = nk2;
while (!(nsub >= nf))
{
    nf = nf - nsub;
    nsub = nsub / 2;
}
nf = nf + nsub;
}
for (i = 1; i <= Nk; i++)
    R[Ini + i] = R[Ini + i] * TW;
}

void Sort3(double R[], int MK, int ISRT, int TW)
{
    int NL, Nk, i, L, i2, i21, js, JI, i3, j, kf, k;
    double rst;

    Nk = 2 ^ MK * 3;
    NL = 1 + 2 * Nk / 3;
    if (ISRT == 0) Sort2(R, 2 * Nk / 3, TW, 0);
    if (ISRT == 1) Sort2(R, Nk / 3, TW, NL);
    for (i = 1; i <= MK; i++)
    {
        L = 1;
        if (ISRT == 0) L = MK - i + 1;
        if (L != 1)
        {
            i2 = 2 ^ (L - 1);
            i21 = i2 - 1;
            js = 2 * i2 + 1;
            JI = 6 * i2;
            if (ISRT == 0) i2 = i2 * 2;
            i3 = JI / 2 - i2;
            for (j = js; j <= Nk; j += JI)
            {
                kf = j + i21;
                for (k = j; k <= kf; k++)
                {
                    rst = R[k];
                    R[k] = R[k + i2];
                    R[k + i2] = R[k + i3];
                    R[k + i3] = rst;
                }
            }
        }
    }
    if (ISRT == 1) Sort2(R, 2 * Nk / 3, TW, 0);
    if (ISRT == 1) Sort2(R, Nk / 3, TW, NL);
}

```

## ALGORITMO PARA PONDERAÇÃO

```

#include "math.h"

void Ponderador(double R[], int N, int Lim);
bool GeraCoef(double Lis[], int N);

bool GeraCoef(double Lis[], int N)
{
    //armazena no vetor Lis os coeficientes relativos ao ponderador de
    ordem N
    if (N < 1) return false;
    switch(N)
    {
    case 1:
        Lis[0] = 1 / 2;
        Lis[1] = 1 / 2;
        break;
    case 2:
        Lis[0] = 1 / 6;
        Lis[1] = 4 / 6;
        Lis[2] = 1 / 6;
        break;
    case 3:
        Lis[0] = 1 / 8;
        Lis[1] = 3 / 8;
        Lis[2] = 3 / 8;
        Lis[3] = 1 / 8;
        break;
    case 4:
        Lis[0] = 7 / 90;
        Lis[1] = 32 / 90;
        Lis[2] = 12 / 90;
        Lis[3] = 32 / 90;
        Lis[4] = 7 / 90;
        break;
    case 5:
        Lis[0] = 19 / 288;
        Lis[1] = 75 / 288;
        Lis[2] = 50 / 288;
        Lis[3] = 50 / 288;
        Lis[4] = 75 / 288;
        Lis[5] = 19 / 288;
        break;
    case 6:
        Lis[0] = 41 / 840;
        Lis[1] = 216 / 840;
        Lis[2] = 27 / 840;
        Lis[3] = 272 / 840;
        Lis[4] = 27 / 840;
        Lis[5] = 216 / 840;
        Lis[6] = 41 / 840;
        break;
    case 7:
        Lis[0] = 751 / 17280;
        Lis[1] = 3577 / 17280;
        Lis[2] = 1323 / 17280;
        Lis[3] = 2989 / 17280;

```

```

    Lis[4] = 2989 / 17280;
    Lis[5] = 1323 / 17280;
    Lis[6] = 3577 / 17280;
    Lis[7] = 751 / 17280;
    break;
case 8:
    Lis[0] = 989 / 28350;
    Lis[1] = 5888 / 28350;
    Lis[2] = -928 / 28350;
    Lis[3] = 10496 / 28350;
    Lis[4] = -4540 / 28350;
    Lis[5] = 10496 / 28350;
    Lis[6] = -928 / 28350;
    Lis[7] = 5888 / 28350;
    Lis[8] = 989 / 28350;
    break;
case 9:
    Lis[0] = 2857 / 89600;
    Lis[1] = 15741 / 89600;
    Lis[2] = 1080 / 89600;
    Lis[3] = 19344 / 89600;
    Lis[4] = 5778 / 89600;
    Lis[5] = 5778 / 89600;
    Lis[6] = 19344 / 89600;
    Lis[7] = 1080 / 89600;
    Lis[8] = 15741 / 89600;
    Lis[9] = 2847 / 89600;
    break;
case 10:
    Lis[0] = 16067 / 598752;
    Lis[1] = 106300 / 598752;
    Lis[2] = -48525 / 598752;
    Lis[3] = 272400 / 598752;
    Lis[4] = -260550 / 598752;
    Lis[5] = 427368 / 598752;
    Lis[6] = -260550 / 598752;
    Lis[7] = 272400 / 598752;
    Lis[8] = -48525 / 598752;
    Lis[9] = 106300 / 598752;
    Lis[10] = 16067 / 598752;
    break;
}
return true;
}

void Ponderador(double R[], int N, int Lim)
{
    double *Coef = new double(N);
    int i, j;
    GeraCoef(Coef, N);
    for (i = 1; i <= (Lim + 1) - N; i += N)
    {
        if (i == 1)
            R[i] = Coef[0] * R[i] * N;
        else
            R[i] = 2 * Coef[0] * R[i] * N;
        for (j = 1; j <= N - 1; j++)
            R[j + i] = Coef[j] * R[j + i] * N;
    }
    R[i] = Coef[0] * R[i] * N;
}

```