|  |
| --- |
| http://paginas.fe.up.pt/~anict/Symposium2011/images/LogoFCTUC.jpg |
|  |
| Análise e Transformação de Dados |
| Trabalho Prático nº 1 |
|  |
|  |
|  |

Grupo 23:

João Miguel Rodrigues Jesus nº 2008111667

Rosa Manuela Rodrigues de Faria nº 2005128014

1. Sendo este o grupo número 23, a função é da seguinte forma:
   1. Substituindo as equações trigonométricas:

Teremos

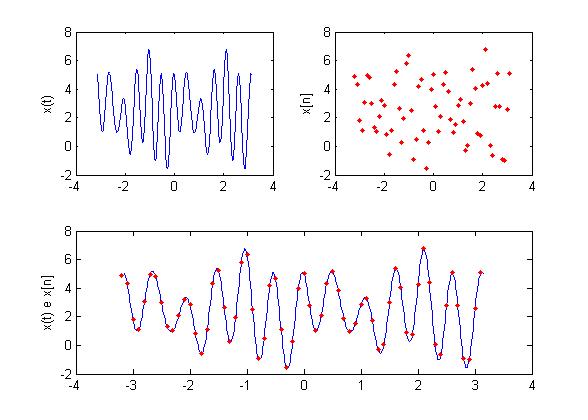
Com as constantes

* 1. Substituindo obtemos

* 1. Para representar o gráfico de foi primeiro criado um vector igualmente espaçado usando a função *linespace*, com mínimo de –π, máximo π, e número de elementos igual a 500. Foi depois calculada a função com o vector criado anteriormente.

No caso de foi também criado um vector, n, com passo 1, início em e término em , ambos arredondados para baixo. Seguidamente foi aplicada a fórmula calculada no exercício anterior. Finalmente foi feito o plot, tendo no caso de o parâmetro ‘.’, para que os seus elementos sejam sinalizados apenas com pontos. A realçar que, para ser possível comparar os gráficos, foi necessário utilizar n\*Ts como eixo dos xx (no caso de x[n]).

O script correspondente a este exercício encontra-se no ficheiro ex13.m, e o resultado é a imagem seguinte:



Resultado da execução do exercício 1.3

* 1. Para a resolução deste exercício foi necessária a criação de 3 funções auxiliares: func23.m, simpson.m e trapézio.m. A primeira calcula simplesmente a função para o(s) valor(es) *t* dado(s), de modo a facilitar o cálculo dos integrais necessários.  
     As duas funções seguintes aplicam as regras de Simpson e do trapézio, respectivamente, para aproximação de integrais. Em ambas as funções são aplicadas as fórmulas directamente, aplicando os somatórios correspondentes por somas de vectores contendo todos os valores a somar através da função *sum* do Matlab, dando como entrada os extremos do intervalo e o número de intervalos a utilizar.

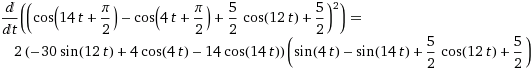
Assim, foi criado o script ex14.m, em que é criada uma variável simbólica, seguida da criação da função correspondente. É depois calculado o valor exacto da energia do sinal no intervalo pedido com a função *int* do módulo simbólico do Matlab. Seguidamente são utilizadas as funções trapézio.m e simpson.m para o cálculo do integral de forma aproximada.

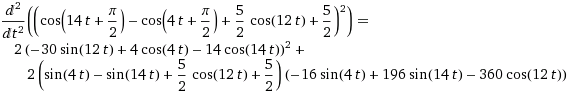
**Determinação do número de intervalos a considerar nas aproximações:**

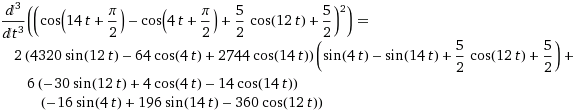
Para determinar o número de intervalos necessários para o erro ser menor que 0,001 seria necessário aplicar, para as regras de Simpson e dos trapézios, respectivamente, as fórmulas

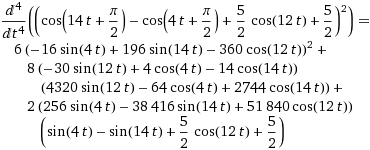
Em que *k* é o majorante de, respectivamente, e .

Foram então calculadas as derivadas de utilizando o website www.wolframalpha.com, obtendo os resultados:









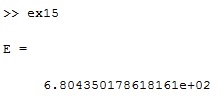
Para encontrar o majorante destas funções considerou-se o pior caso, em que todos os senos e co-senos tomam o valor 1. No entanto o valor obtido foi elevadíssimo e irreal para o efeito pretendido, logo procedeu-se à experimentação com diferentes quantidades de intervalos, chegando-se aos valores 30 (para a Regra de Simpson) e 13 (para a Regra dos Trapézios) como número mínimo de intervalos a utilizar.

Assim, obtiveram-se os seguintes valores:

* Valor real do integral: J
* Valores aproximados:

|  |  |  |  |
| --- | --- | --- | --- |
| Regra de Cálculo | Valor obtido com o Intervalo mínimo (J) | Valor imediatamente anterior (J) | Erro |
| Simpson | 65.188047561988355 | 163.3628179866692 | 1.4E-12 |
| Trapézios | 65.188047561988199 | 55.58399833082965 | 1.42E-11 |

* 1. Para resolver este exercício foi simplesmente criado o script ex15.m, em que é calculado o vector n e xn, como no exercício 1.3, sendo depois calculada a energia do sinal discreto x[n] através do somatório da função x[n]2. O valor obtido foi de 680.4350178618161 J.



**Nota: Compatibilidade com o Octave**

Todas as rotinas do exercício 1 são compatíveis tanto com o software Octave (uma vez instalado o pacote simbólico) como com o Matlab, exceptuando, no exercício 1.4, o cálculo do integral simbólico, que não tem uma função correspondente à do Matlab. Em termos de comparação dos tempos de execução, temos os seguintes resultados:

|  |  |  |
| --- | --- | --- |
| Exercício | Tempo de execução (Octave) | Tempo de execução (Matlab) |
| Ex13.m | 0.22700 s | 1.199808 s |
| Ex14.m (sem cálculo simbólico) | 0.00732 s | 0.064533 s |
| Ex15.m | 0.00025 s | 0.002535 s |

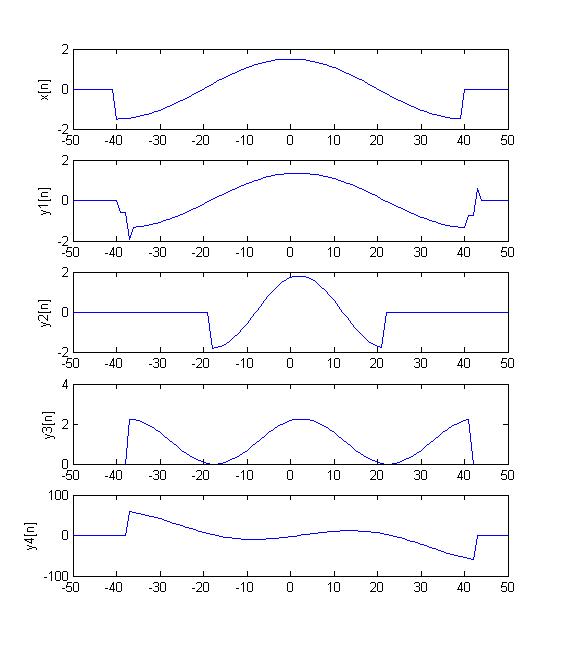
Por este resultado podemos ver que o Octave é bastante mais rápido que o Matlab.

1. Funções a utilizar:

**Sinal de entrada:**

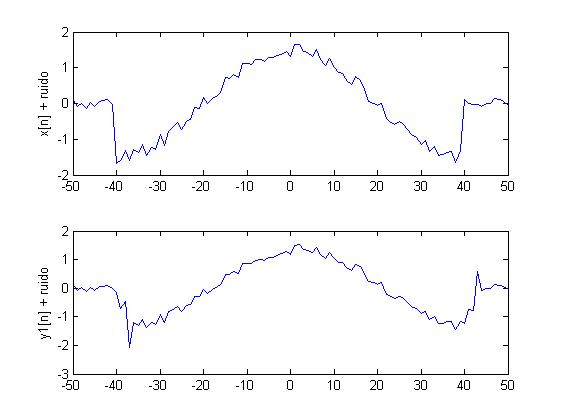
**Respostas do Sistema:**

* 1. Para a realização deste exercício foi necessário criar duas funções auxiliares, u.m e ux.m, para cálculo do degrau unitário e para cálculo da função x[n], respectivamente. Foi depois desenvolvida a rotina ex21.m, em que são criados os vectores y1n, y2n, y3n e y4n representantes das funções dadas, para n no intervalo [-50;50]. É finalmente feito o plot das várias funções.



Representação gráfica do sinal de entrada e das repostas do sistema

* 1. De modo a criar o ruído para adicionar ao sinal foi criado o vector ruido, utilizando a função rand do Matlab, com amplitude 0.4 e centrada em 0. É então calculado o valor do sinal com ruído xnr, e a resposta do sistema com ruído y1nr. Finalmente faz-se o plot dos vectores obtidos.



Representação gráfica do sinal com ruído e da resposta do sistema também com ruído

* 1. Um sistema é linear quando respeita as condições de homogeneidade e aditividade. No caso da homogeneidade, a prova consiste em provar que . Já para a aditividade é necessário provar que com e .
     1. Função
* Homogeneidade:
* Aditividade:
  + 1. Função
* Homogeneidade:
* Aditividade:
  + 1. Função
* Homogeneidade:
* Aditividade:
  + 1. Função
* Homogeneidade:
* Aditividade:

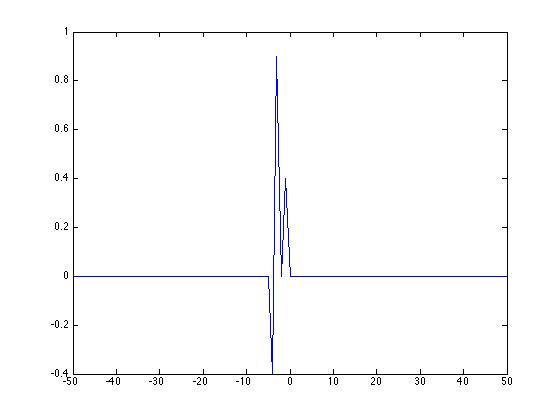
Destas demonstrações se conclui que as funções, e são lineares mas a função não o é.

* 1. Um sistema é invariante no tempo quando verifica .

.

* 1. Para calcular a resposta do impulso do sistema y1[n] temos primeiro de determinar a sua função. Sendo o nosso grupo o 23, foi determinada da seguinte maneira:

No entanto, para percorrer todo o intervalo necessário do sinal, foi necessário definir a escala de n desde a 50 até 50,obtendo o seguinte gráfico:



Impulso do sistema y1[n]

* 1. Para determinarmos a função de transferência do sistema G(z) através de foi preciso calcular a Transformada de z do impulso com condições iniciais nulas, obtendo a seguinte expressão:

Fazendo a Transformada de Z obtivemos a seguinte expressão:

* 1. Para calcular os valores em que k possa ser estável, teremos de utilizar a função de transferência dada por:

Ao ajustarmos o intervalo de 5 a 5 com um passo de 0.1 e após adicionarmos um vector com os valores do grupo foram obtidas as seguintes raízes da função:

-1.11 -1.1 -1.09 -1.08 -1.07 -1.06 -1.05 -1.04 -1.03 -1.02

-1.01 -1 -0.99 -0.98 -0.97 -0.96 -0.95 -0.94 -0.93 -0.92

-0.91 -0.9 -0.89 -0.88 -0.87 -0.86 -0.85 -0.84 -0.83 -0.82

-0.81 -0.8 -0.79 -0.78 -0.77 -0.76 -0.75 -0.74 -0.73 -0.72

-0.71 -0.7 -0.69 -0.68 -0.67 -0.66 -0.65 -0.64 -0.63 -0.62

-0.61 -0.6 -0.59 -0.58 -0.57 -0.56 -0.55 -0.54 -0.53 -0.52

-0.51 -0.5 -0.49 -0.48 -0.47 -0.46 -0.45 -0.44 -0.43 -0.42

-0.41 -0.4 -0.39 -0.38 -0.37 -0.36 -0.35 -0.34 -0.33 -0.32

-0.31 -0.3 -0.29 -0.28 -0.27 -0.26 -0.25 -0.24 -0.23 -0.22

-0.21 -0.2 -0.19 -0.18 -0.17 -0.16 -0.15 -0.14 -0.13 -0.12

-0.11 -0.1 -0.09 -0.08 -0.07 -0.06 -0.05 -0.04 -0.03 -0.02

-0.01 0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09

0.1 0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2

0.21 0.22 0.23 0.24 0.25 0.26 0.27 0.28 0.29 0.3 0.31

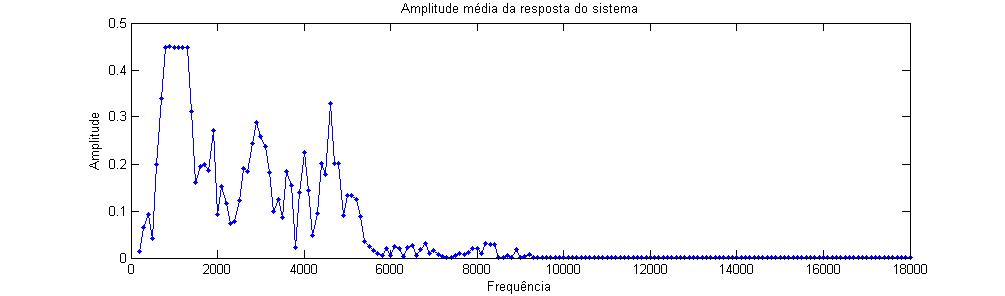
0.32 0.33 0.34 0.35 0.36 0.37 0.38 0.39 0.4 0.41 0.42

0.43 0.44 0.45 0.46 0.47 0.48 0.49 0.5 0.51 0.52 0.53

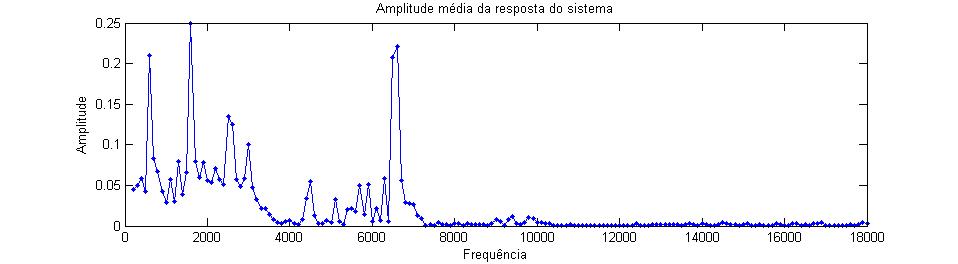
0.54 0.55 0.56 0.57 0.58

1. Para realizar esta experiência começámos por definir os parâmetros necessários: frequência de amostragem, duração da amostra, frequências a reproduzir e tempos de amostragem. De seguida, dentro de um ciclo que percorre as frequências a utilizar, foi criado o sinal na respectiva frequência e usados os comandos *wavplay* e *wavrecord* para a gravação assíncrona do som emitido, armazenando a resposta na matriz *gravacoes*. É depois calculado o máximo de cada período de amostragem e a média desses máximos, correspondendo esta à amplitude das frequências; a imagem obtida resulta dessas mesmas médias.

Realizámos esta experiência em dois computadores diferentes obtendo os resultados seguintes:



Amplitude média de resposta do sistema - Computador A



Amplitude média de resposta do sistema - Computador B

No computador A verifica-se que as colunas filtram os sons abaixo dos 500 Hz, sendo visível uma subida significativa de amplitude entre os 500 e os 800 Hz. Os valores mantêm-se durante algum tempo, voltando a diminuir por volta dos 1300 Hz, e variando bastante no intervalo de 1500 a 5800 Hz. Após os 9100 Hz, os sons já são praticamente inaudíveis, já que muito provavelmente as colunas conseguem reproduzir esses sons (ou o microfone não os consegue captar).

No computador B obtivemos um comportamento semelhante, tendo porém um pico nos 7000 Hz superior ao dos 800 Hz. Aos 1700 Hz dá-se a amplitude máxima da experiência no valor de 0.25. Comparando com o computador A, podemos observar que as colunas e/ou o microfone são de qualidade inferior, dado que no computador A a amplitude máxima é de 0.45. Também verificamos que a partir de 10000 Hz os valores não são nulos, causados pelo facto de as colunas emitirem estalidos durante o teste.