MEEC
Mestrado em Engenharia Electrotécnica e de Computadores

MCSDI
Modelação e Controlo de Sistemas Dinâmicos

Guião do trabalho laboratorial nº 2

Análise de Sistemas Não Lineares por Plano de Fase Recorrendo ao MATLAB

Análise de Sistemas Não Lineares por Plano de Fase Recorrendo ao MATLAB

Sumário: Pretende-se com este trabalho, fazer uma introdução às capacidades de desenho de trajectórias de equações diferenciais de segunda ordem no Plano de Fase, através da integração pelo Método de Runge-Kutta, recorrendo ao software MATLAB.

Considere-se como exemplo a seguinte equação diferencial de segunda ordem:

$$\frac{d^2x}{dt^2} + \frac{dx}{dt} + x = 0$$

Equação 1: Exemplo de equação diferencial de um sistema de segunda ordem

O objectivo deste trabalho consiste na representação das soluções de uma equação diferencial de segunda ordem no Plano de Fase. Uma equação deste tipo pode ser escrita como:

$$\frac{d^2x}{dt^2} + f\left(x, \frac{dx}{dt}\right) = 0 \Leftrightarrow$$
$$\Leftrightarrow \frac{d^2x}{dt^2} = -f\left(x, \frac{dx}{dt}\right)$$

Equação 2: Forma alternativa da equação diferencial do sistema de segunda ordem

Fazendo:

$$\begin{cases} x_1 = x \\ x_2 = \dfrac{dx}{dt} \end{cases} \Leftrightarrow \begin{cases} \\ x_2 = \dfrac{dx_1}{dt} \end{cases}$$

Equação 3: Escolha das variáveis para representação no Plano de Fase

Substituindo este resultado na equação 2, fica:

$$\begin{cases} \dfrac{dx_1}{dt} = x_2 \\ \dfrac{dx_2}{dt} = -f\left(x_1, x_2\right) \end{cases}$$

Equação 4: Forma alternativa (variáveis de fase) da equação diferencial para representação no Plano de Fase

Considerando agora a equação 1 (exemplo que pretendemos representar no Plano de Fase), e colocando-a na forma apresentada na equação 4, obtém-se o seguinte sistema de duas equações diferenciais de primeira ordem:

$$\begin{cases} \dfrac{dx_1}{dt} = x_2 \\ \dfrac{dx_2}{dt} = -x_1 - x_2 \end{cases}$$

Equação 5: Forma alternativa (variáveis de fase) do exemplo a representar no Plano de Fase

1. Implementação da Equação Diferencial no MATLAB

1. O primeiro passo para a representação desta equação no Plano de Fase, passa pela sua definição no MATLAB.

a) Para esse efeito inicie a aplicação MATLAB.

2. Todo o código necessário para efectuar a representação das soluções de uma equação no Plano de Fase pode ser escrito directamente na janela do MATLAB. Alternativamente, pode-se introduzir todo o código num ficheiro de dados do MATLAB, genericamente chamado M-File (é um ficheiro de texto normal, com extensão *.m), que é depois carregado para o MATLAB sempre que necessário.

a) Esta solução é preferencial quando o código se torna extenso, para evitar ter que o estar a re-escrever de novo sempre que se pretende efectuar uma nova representação.

b) Para criar um novo M-File, escolha a opção "New" do menu "File" do MATLAB.

c) Dentro da opção "New", escolha a subopção "M-File".

d) Aparecer-lhe-á numa nova janela o editor do MATLAB, com o seguinte aspecto:

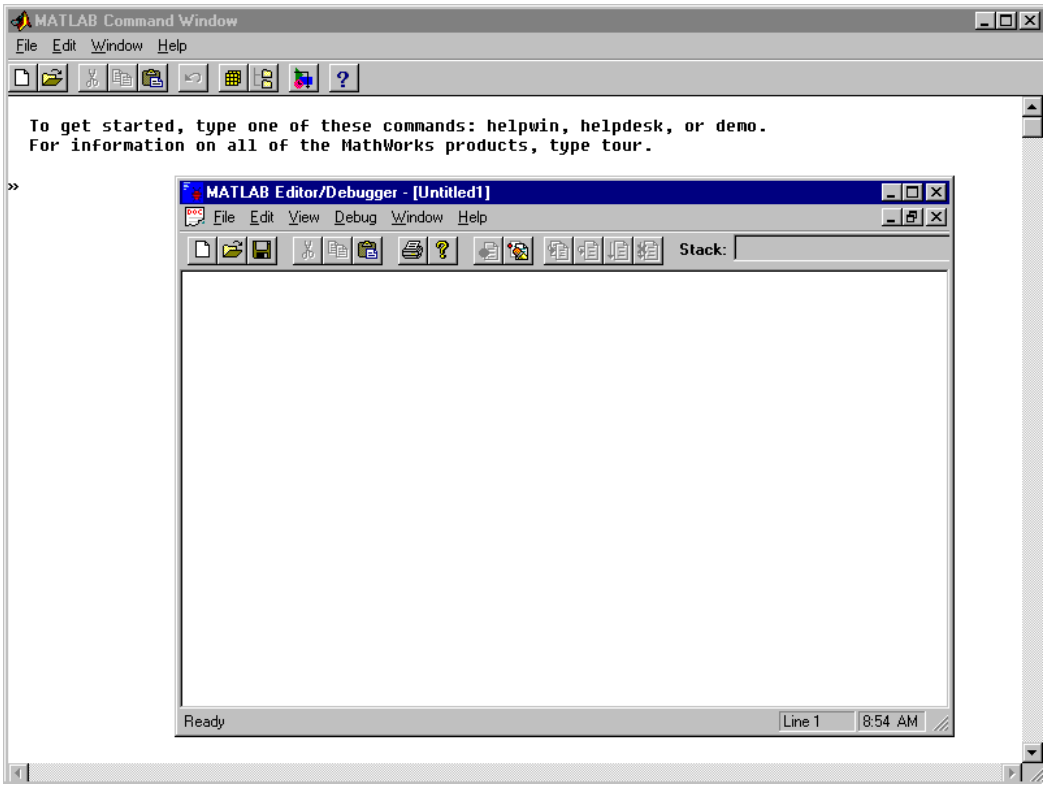

Figura 1: Janela do editor de ficheiros tipo M do MATLAB

e) Comece a introduzir o código que lhe vai sendo dado (poderá introduzir comentários no seu M-File, para o que deverá iniciar os comentários pelo caracter %).

3. O primeiro passo da escrita de um M-File deverá passar pela definição de um cabeçalho que indique claramente a que se refere o ficheiro em questão. Como exemplo, apresenta-se o seguinte código:

```
%-----------------------------------------------------------------
%
%  M-File para traçar as trajectórias no Plano de Fase
%  de Equacoes diferenciais de segunda ordem
%  Dezembro de 1999
%  Manuel Silva, Tenreiro Machado
%
%-----------------------------------------------------------------

%-----------------------------------------------------------------
%
%Definição das variáveis
%
%-----------------------------------------------------------------

   %x(1)=x(t)
   %x(2)=dx/dt

%-----------------------------------------------------------------
%
%Definição da equação diferencial a desenhar no Plano de Fase
%
%-----------------------------------------------------------------

   %
   %Equação 1 - Pag. 22 dos apontamentos
   %
   %DDx+Dx+x=0
   % =>  x(1)=x,x(2)=Dx,Dx(2)=DDx
   %
```

Figura 2: Janela do editor de ficheiros tipo M do MATLAB

4. O primeiro M-File a implementar, servirá para definir as variáveis de fase da equação diferencial a representar no Plano de Fase (ver equação 5).

a) Este M-File será chamado recursivamente pela função `ode45` (que analisaremos à frente), ao longo da execução do programa de cálculo das trajectórias no Plano de Fase, e servirá para efectuar os vários cálculos necessários à determinação de dx_1/dt e dx_2/dt.

b) As variáveis de fase têm que ser definidas num vector coluna (no nosso caso chamado `dx`) em que a primeira linha corresponde à definição da expressão da variável dx_1/dt e a segunda linha à definição da expressão da variável dx_2/dt.

c) Este M-File foi implementado com a estrutura de uma função do MATLAB. A esta função são passados dois vectores, o vector `x` e o vector `t`. O primeiro destes vectores contém os valores de dx_1/dt e dx_2/dt, e o segundo vector contém o instante de tempo no qual estes valores foram calculados. A função retorna o vector `dx` que contém os valores de dx_1/dt e dx_2/dt calculados nessa iteração.

d) É de realçar que o vector `dx` não contém os valores de dx_1/dt e dx_2/dt, mas sim os valores calculados em cada iteração, sendo os valores de dx_1/dt e dx_2/dt armazenados no vector `x`.

e) Sempre que se torne necessário representar no Plano de Fase uma nova equação, este é o único M-File que é necessário alterar.

f) Após terminar a introdução do código grave o M-File, de forma a ser possível voltar a carregá-lo sempre que necessário. Para isso seleccione a opção "Save As..." do menu "File" da janela do Editor do MATLAB e guarde o seu M-file com o nome “caso1.m”.

g) O código completo deste M-File é o apresentado na figura 3, assim como no Anexo 2.

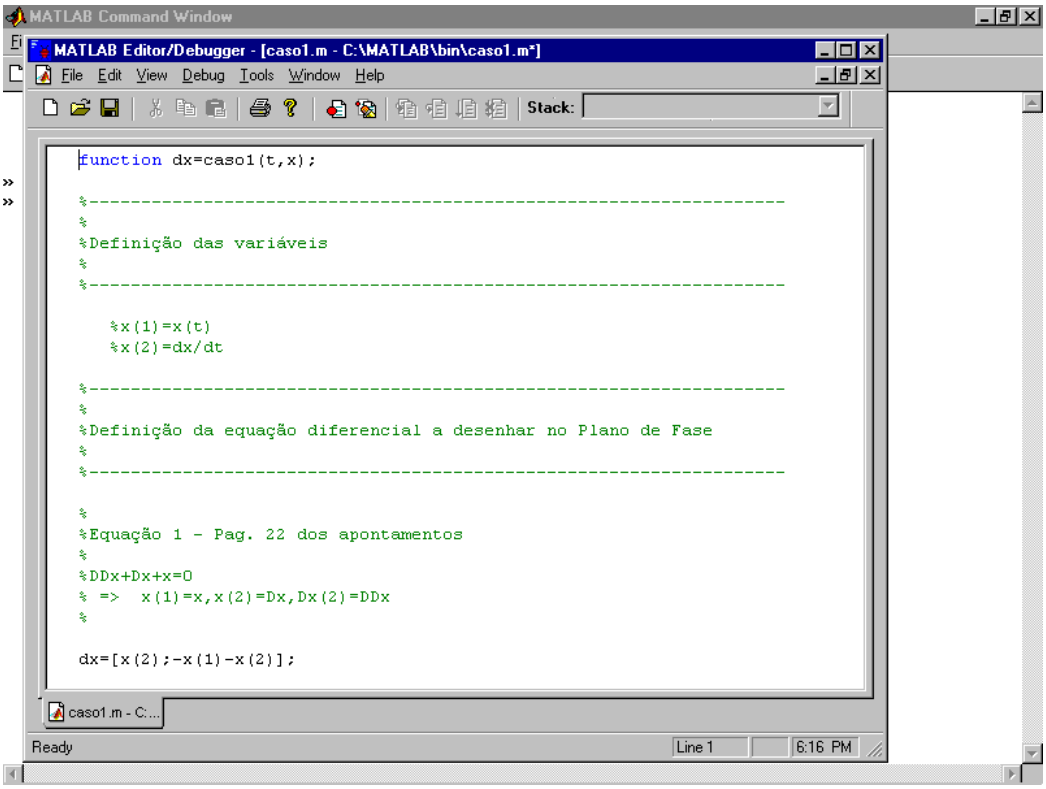

Figura 3: Janela do editor de ficheiros tipo M do MATLAB

5. Crie um novo M-File.

a) Neste M-File vai ser implementado o código para se efectuarem os cálculos necessários à representação da equação 1, no Plano de Fase.

6. Depois de introduzir o cabeçalho do ficheiro, deverão ser definidas as constantes e/ou os parâmetros a serem utilizadas (ver figura 4).

a) Em primeiro lugar, começa por se efectuar a limpeza da memória do MATLAB e a da janela de gráficos, após o que se define e inicializa a variável auxiliar `aux` com o valor 1.

b) De seguida inicializa-se a variável `dt` que corresponde ao incremento temporal entre cada iteração durante o cálculo dos integrais de x_1 e x_2 pelo método de Runge-Kutta. Por sua vez o parâmetro `t_max` corresponde ao máximo tempo durante o qual vamos analisar a evolução das trajectórias no Plano de Fase (neste caso `t_max=` 6 seg).

c) Após isto define-se e inicializa-se o vector que vai conter os instantes de tempo nos quais vão ser calculados os valores dos integrais (`vector_t`), através de um ciclo `for`.

d) Em seguida define-se o parâmetro `n` que indica o número de condições iniciais das trajectórias a considerar, para a representação no Plano de Fase.

e) A definição dos parâmetros conclui-se com a introdução dos valores máximos que as trajectórias podem apresentar no Plano de Fase, `x_max` e `Dx_max`, correspondendo respectivamente aos valores máximos de x_1 e x_2.

```
%-----------------------------------------------------------------
%
%  M-File para traçar as trajectórias no Plano de Fase
%  de Equacoes diferenciais de segunda ordem
%  Dezembro de 1999
%  Manuel Silva, Tenreiro Machado
%
```

```
%-----------------------------------------------------------------

%-----------------------------------------------------------------
%
%Definição e valores por defeito dos parâmetros
%
%-----------------------------------------------------------------

%Limpeza da memória do MATLAB
clear;

%Limpeza da janela com o gráfico
close;

%variavel auxiliar
aux=1;

%incremento temporal
dt=0.01;

%intervalo de tempo a considerar
t0=0;
t_max=6;
for i=t0:dt:t_max
   vector_t(aux)=[i];
   aux=aux+1;
end
aux=1;

%Numero de pontos na grelha
n=6;

%Valores maximos da grelha
x_max=1;
Dx_max=1;
```

Figura 4: Código correspondente à definição dos parâmetros

7. Chegados a esta fase devem-se calcular os valores iniciais das trajectórias.

a) Para este efeito os pontos iniciais da trajectória vão ser calculados de acordo com as equações que se apresentam na figura 5, de forma a que fiquem igualmente distribuídos na janela em que virão a ser apresentados os resultados.

```
%-----------------------------------------------------------------
%
%Desenho da equação diferencial no Plano de Fase
%
%-----------------------------------------------------------------

%Inicializacao da posicao e da velocidade: Condicoes iniciais
for i = 1:n,
   for j = 1:n,

      %Posicao inicial na grelha
      x1=(i-(n+1)/2)*x_max/n;

      %Velocidade inicial na grelha
      x2=(j-(n+1)/2)*Dx_max/n;

      %vector inicial
      x0=[x1;x2];
```

Figura 5: Código correspondente à definição das condições iniciais

8. Estamos agora em condições de iniciar a integração das variáveis de fase (ver equação 5), através do méctodo de Runge-Kutta. Esta integração é efectuada recorrendo ao comando `ode45` do MATLAB.

a) Este comando aceita como entradas o nome do M-File onde se encontram definidas as equações a integrar, o vector de instantes de tempo nos quais vão ser calculados os integrais e as condições inicias.

b) À saída, o comando `ode45` retorna dois vectores, um com os resultados da integração (vector `x`) e outro com os correspondentes instantes de tempo (vector `t`) em que foram determinados os valores dos integrais.

c) Os resultados de `x(1)` e `x(2)` são guardados, respectivamente, na primeira e segunda colunas do vector `x`.

```
%-----------------------------------------------------------------
%
%Integração das equações diferenciais pelo método de Runge Kutta 4(5)
%
%-----------------------------------------------------------------

[t,x]=ode45('caso1',vector_t,x0);
```

Figura 6: Código correspondente à integração das equações diferenciais de primeira ordem pelo método de Runge Kutta 4(5)

9. Quando o cálculo de uma trajectória está completa, esta é representada.

a) São também representados os eixos do gráfico e as respectivas legendas, como se pode observar do código apresentado na figura 7.

```
        %
        %Representação das trajectórias no plano de fase
        %

        if (aux == 1)
            plot(x(:,1),x(:,2))
            hold on
            aux=2;
        else
        plot(x(:,1),x(:,2))
        end

    end %ciclo j
end %ciclo i

hold off

%
%Apresentar a parte importante do gráfico
%
axis([-x_max,x_max,-Dx_max,Dx_max])

%
%Desenho dos eixos do gráfico
%
line([-x_max,x_max],[0,0])
line([0,0],[-Dx_max,Dx_max])

%
%Legendas dos eixos do gráfico
%
xlabel('x1')
ylabel('x2')
```

Figura 7: Código correspondente à representação das trajectórias no Plano de Fase

10. Grave o M-File com o nome “plano_de_fase”.

a) O código completo para a representação das soluções desta equação no Plano de Fase (correspondente ao M-File “plano_de_fase”) é apresentado em anexo (ver Anexo 1).

11. Para executar o M-File “plano_de_fase”, deverá executar o seguinte comando no “prompt” do MATLAB:

a) >plano_de_fase;

b) As trajectórias correspondentes à solução da equação 1 no Plano de Fase são as apresentadas na figura 8 (o ponto singular, cujas coordenadas são (0, 0) corresponde a um foco estável, com valores próprios $\lambda_{1,2}=-\frac{1}{2}\pm j\frac{\sqrt{3}}{2}$).

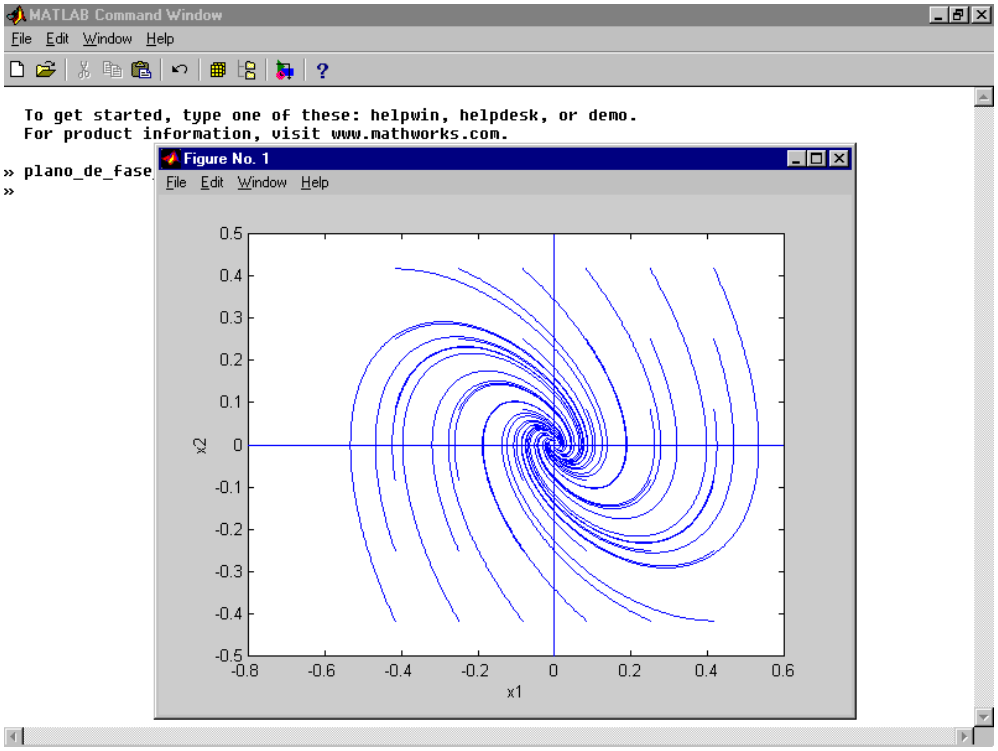

Figura 8: Representação no Plano de Fase correspondente à equação 1

12. Vejamos agora o procedimento necessário caso seja pretendido representar a seguinte equação no Plano de Fase.

$$\frac{d^2x}{dt^2}+\left|\frac{dx}{dt}\right|+x=0$$

Equação 6: Exemplo de equação diferencial de um sistema não linear

a) Tal como indicado no ponto 4., deverá ser criado um M-File que servirá para definir as variáveis de fase da equação diferencial a representar no Plano de Fase (ver equação 7).

$$\begin{cases} \dfrac{dx_1}{dt}=x_2 \\ \dfrac{dx_2}{dt}=-x_1-|x_2| \end{cases}$$

Equação 7: Variáveis de fase para a representação da equação 6 no Plano de Fase

b) No M-file referido na alínea anterior, as variáveis de fase têm que ser definidas num vector coluna (no nosso caso chamado dx) em que a primeira linha corresponde à variável dx_1/dt e a segunda linha à variável dx_2/dt (o código completo deste M-File é o apresentado no Anexo 3).

c) Após terminar a introdução do código grave o M-File com o nome "caso2.m".

d) As trajectórias correspondentes à solução da equação 6 no Plano de Fase são as apresentadas na figura 9 (o ponto singular, cujas coordenadas são (0, 0) corresponde, para o caso em que dx/dt>0, a um foco estável, com valores próprios $\lambda_{1,2}=-\frac{1}{2}\pm j\frac{\sqrt{3}}{2}$ e, para o caso em que dx/dt<0, a um foco instável, com valores próprios $\lambda_{1,2}=\frac{1}{2}\pm j\frac{\sqrt{3}}{2}$).

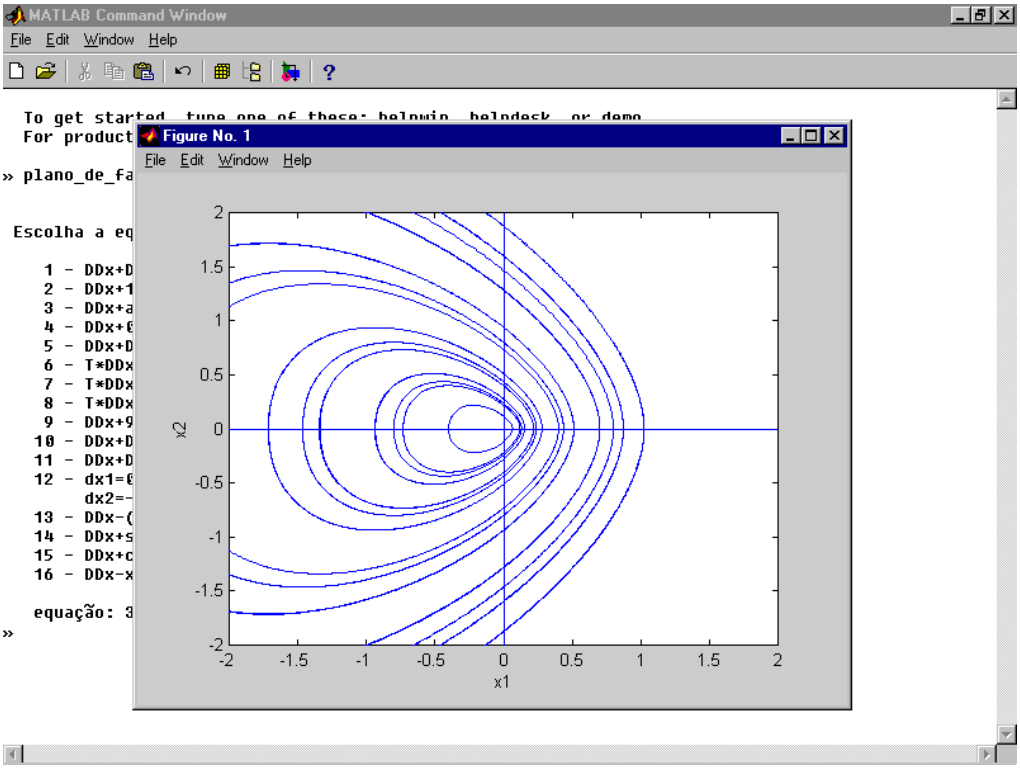

Figura 9: Representação no Plano de Fase correspondente à equação 6

13. Por último, vejamos agora o procedimento necessário caso seja pretendido representar a seguinte equação no Plano de Fase.

$$\frac{d^2x}{dt^2}+\frac{dx}{dt}+\frac{9.8}{1}sen(x)=0$$

Equação 8: Equação diferencial de um sistema não linear

a) Como já referido anteriormente, torna-se necessário desenvolver um M-file para definir as variáveis de fase, de acordo com o indicado na equação 9 (o código completo deste M-File é o apresentado no Anexo 4).

$$\begin{cases} \dfrac{dx_1}{dt}=x_2 \\ \dfrac{dx_2}{dt}=-\dfrac{9.8}{1}sen(x_1)-x_2 \end{cases}$$

Equação 9: Variáveis de fase para a representação da equação 8 no Plano de Fase

b) Após terminar a introdução do código grave o M-File com o nome “caso3.m”.

c) As trajectórias correspondentes à solução da equação 8 no Plano de Fase são as apresentadas na figura 10 (os pontos singulares, cujas coordenadas são (2.n.π, 0) – n = 0, 1, 2, ... - correspondem a focos estáveis, com valores próprios $\lambda_{1,2}=-\frac{1}{2}\pm j\frac{\sqrt{38,2}}{2}$; por sua vez, os pontos singulares, cujas coordenadas são ((2.n.+1).π, 0) – n = 0, 1, 2, ... - correspondem a pontos de sela, com valores próprios λ_1=–3.67 e λ_2=2.67).

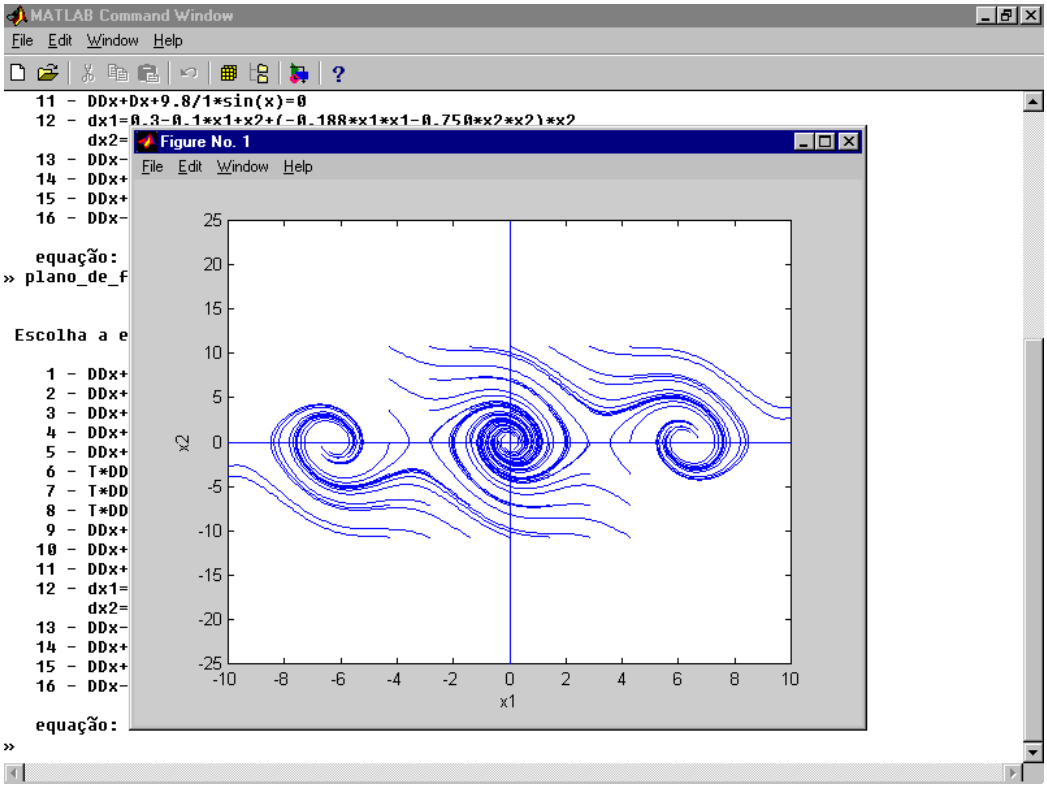

Figura 10: Representação no Plano de Fase correspondente à equação 8

2. Conclusões

Acabamos de ver como é possível recorrendo ao software MATLAB efectuar o estudo de um sistema através da representação no Plano de Fase.

As noções aqui introduzidas, de uma forma necessariamente resumida, podem ser desenvolvidas recorrendo à bibliografia que se apresenta de seguida.

3. Bibliografia

[1] - Ogata, Katsuhiko; Engenharia de Controle Moderno; Prentice-Hall do Brasil; 1982.

4. Anexo 1: Código para Representação das Soluções da Equação Diferencial no Plano de Fase

```
%-----------------------------------------------------------------
%
%  M-File para traçar as trajectórias no Plano de Fase
%  de Equacoes diferenciais de segunda ordem
%  Dezembro de 1999
%  Manuel Silva, Tenreiro Machado
%
%-----------------------------------------------------------------

%-----------------------------------------------------------------
%
%Definição e valores por defeito dos parâmetros
%
%-----------------------------------------------------------------

%Limpeza da memória do MATLAB
clear;

%Limpeza da janela com o gráfico
close;

%variavel auxiliar
aux=1;

%incremento temporal
dt=0.01;

%intervalo de tempo a considerar
t0=0;
t_max=6;
for i=t0:dt:t_max
   vector_t(aux)=[i];
   aux=aux+1;
end
aux=1;

%Numero de pontos na grelha
n=6;

%Valores maximos da grelha
x_max=1;
Dx_max=1;


%-----------------------------------------------------------------
%
%Desenho da equação diferencial no Plano de Fase
%
%-----------------------------------------------------------------

%Inicializacao da posicao e da velocidade: Condicoes iniciais
for i = 1:n,
   for j = 1:n,

      %Posicao inicial na grelha
      x1=(i-(n+1)/2)*x_max/n;

      %Velocidade inicial na grelha
      x2=(j-(n+1)/2)*Dx_max/n;

      %vector inicial
      x0=[x1;x2];
```

```
        %-----------------------------------------------------------------
        %
        %Integração das equações diferenciais pelo método de Runge Kutta 4(5)
        %
        %-----------------------------------------------------------------

        [t,x]=ode45('caso1',vector_t,x0);

        %
        %Representação das trajectórias no plano de fase
        %

        if (aux == 1)
           plot(x(:,1),x(:,2))
           hold on
           aux=2;
        else
        plot(x(:,1),x(:,2))
        end

    end %ciclo j
end %ciclo i

hold off

%
%Apresentar a parte importante do gráfico
%
axis([-x_max,x_max,-Dx_max,Dx_max])

%
%Desenho dos eixos do gráfico
%
line([-x_max,x_max],[0,0])
line([0,0],[-Dx_max,Dx_max])

%
%Legendas dos eixos do gráfico
%
xlabel('x1')
ylabel('x2')
```

5. Anexo 2: Definição das variáveis de fase para representação da equação diferencial 1 no Plano de Fase

```
function dx=caso1(t,x);

%-------------------------------------------------------------------
%
%Definição das variáveis
%
%-------------------------------------------------------------------

    %x(1)=x(t)
    %x(2)=dx/dt

%-------------------------------------------------------------------
%
%Definição da equação diferencial a desenhar no Plano de Fase
%
%-------------------------------------------------------------------

%
%Equação 1 - Pag. 22 dos apontamentos
%
%DDx+Dx+x=0
% =>  x(1)=x,x(2)=Dx,Dx(2)=DDx
```

```
%

dx=[x(2);-x(1)-x(2)];
```

6. Anexo 3: Definição das variáveis de fase para representação da equação diferencial 6 no Plano de Fase

```
function dx=caso2(t,x);

%-------------------------------------------------------------------
%
%Definição das variáveis
%
%-------------------------------------------------------------------

   %x(1)=x(t)
   %x(2)=dx/dt

%-------------------------------------------------------------------
%
%Definição da equação diferencial a desenhar no Plano de Fase
%
%-------------------------------------------------------------------

   %
   %Equação 6 - Pag. 26 dos apontamentos
   %
   %DDx+abs(Dx)+x=0
   % =>  x(1)=x(t),x(2)=Dx,Dx(2)=DDx
   %
   dx=[x(2);-x(1)-abs(x(2))];
```

7. Anexo 4: Definição das variáveis de fase para representação da equação diferencial 8 no Plano de Fase

```
function dx=caso3(t,x);

%-------------------------------------------------------------------
%
%Definição das variáveis
%
%-------------------------------------------------------------------

   %x(1)=x(t)
   %x(2)=dx/dt

%-------------------------------------------------------------------
%
%Definição da equação diferencial a desenhar no Plano de Fase
%
%-------------------------------------------------------------------

   %
   %Equação 8 - Pag. 618 - Pêndulo com atrito
   %
   %DDx+Dx+9.8/1*sin(x)=0
   % =>  x(1)=x(t),x(2)=Dx,Dx(2)=DDx
   %
   dx=[x(2);-9.8/1*sin(x(1))-x(2)];
```