

```

1 %-----
2 %%%% "fadiga_maincode"
3 %%%% IMPLEMENTED BY SERGIO CUSTODIO - engsergiocustodio@gmail.com
4 %-----
5
6 clear all; close all; clc;
7 %% Tempo e Velocidade
8 tf=10000; %tempo de analise
9 turbulence=0.05;%turbulencia em relacao a velocidade media
10 seriehistorica=load('seriehistorica.txt');
11
12 %% Dinâmica
13
14 dinamica=load('dinamica.txt');
15 vel_dinamic = dinamica(:,1)';
16 omega_dinamic = dinamica(:,2)';
17
18 %% BET-FEM
19 BET_FEM=load('BET_FEM.txt');
20 Velocidade=BET_FEM(:,1)'; %m/s
21 Scasca=BET_FEM(:,2)';
22
23 %% S-N
24
25 Su=310; %MPa %tensao de ruptura %MPa %Aluminium AA6061-T6 MPa Zakaria 2013
26 % Curva de Wholer sigma_a_equi=C_wholer*(N_wholer1^(1/m_wholer))
27 C_wholer=650.8; %MPa %Aluminium AA6061-T6 MPa Zakaria 2013
28 m_wholer=-1/0.12; %Aluminium AA6061-T6 Zakaria 2013
29
30 %% Resultados
31 R_lista = [0.5 0.33 0.1 -1.0];
32
33 V_lista = [1.0 1.5 2.0 2.33 2.5];
34 Resultado = zeros(size(R_lista,2),8);
35
36 contador=1;
37 for k=1:size(seriehistorica,1)
38 for i=1:size(R_lista,2)
39 vm=seriehistorica(k,1);
40 R=R_lista(1,i); %razao de tensoes
41 Resultado(contador,1)=vm;
42 Resultado(contador,4)=R;
43 [Resultado(contador,2),Resultado(contador,5),Resultado(contador,6)]=fatigue_turbine_m2(tf,
turbulence,vm,Velocidade,Scasca,R,Su,C_wholer,m_wholer,vel_dinamic,omega_dinamic);
44 Resultado(contador,7)=seriehistorica(k,2);
45 Resultado(contador,3)=Resultado(contador,7).*0.01./Resultado(contador,6);
46 contador=contador+1;
47 end
48 end
49
50 %% Vida por Velocidade
51 figure(1)
52
53 for j=1:size(R_lista,2)
54 for i=1:size(seriehistorica,1)
55 cont=size(R_lista,2)*(i-1)+j;
56 vidaciclos(j,i)=Resultado(cont,5);
57 end
58 h=plot(seriehistorica(:,1),vidaciclos(j,:));
59 set(h,'LineWidth',1.0);
60 hold on
61 end
62 legend('R=0.5','R=0.33','R=0.1','R=-1.0','LOCATION','BEST')
63 set(h,'LineWidth',1.0);
64 set(gca, 'YScale', 'log')

```

```

65 set(gca,'ylim',[1e3 1e21])
66 set(h,'LineWidth',1.5);
67 xlabel('River Velocity (m/s)')
68 ylabel('Life (cicles)')
69 xlim([min(seriehistorica(:,1)) max(seriehistorica(:,1))])
70
71 figure(2)
72
73 for j=1:size(R_lista,2)
74 for i=1:size(seriehistorica,1)
75 cont=size(R_lista,2)*(i-1)+j;
76 vidaanos(j,i)=Resultado(cont,6);
77 end
78 h=plot(seriehistorica(:,1),vidaanos(j,:));
79 set(h,'LineWidth',1.0);
80 hold on
81 end
82 vidaanos(end+1,:)=30;
83 h=plot(seriehistorica(:,1),vidaanos(end,:), '--');
84 hold on
85 vidaanos(end+1,:)=10;
86 h=plot(seriehistorica(:,1),vidaanos(end,:), '--');
87 hold on
88 vidaanos(end+1,:)=1;
89 h=plot(seriehistorica(:,1),vidaanos(end,:), '--');
90 hold on
91
92 legend('R=0.5','R=0.33','R=0.1','R=-1.0','30 years','10 years','1 year','LOCATION','BEST')
93 set(h,'LineWidth',1.0);
94 set(gca, 'YScale', 'log')
95 set(gca,'ylim',[1e-5 1e15])
96 set(h,'LineWidth',1.5);
97 xlabel('River Velocity (m/s)')
98 ylabel('Life (years)')
99 xlim([min(seriehistorica(:,1)) max(seriehistorica(:,1))])
100 %%
101 figure(3)
102
103 for j=1:size(R_lista,2)
104 for i=1:size(seriehistorica,1)
105 cont=size(R_lista,2)*(i-1)+j;
106 danos(j,i)=Resultado(cont,3);
107 end
108 vidatotal(j,1)=1/sum(danos(j,:),2);
109 h=stem(seriehistorica(:,1),danos(j,:), 'filled');
110 hold on
111 end
112
113 legend('R=0.5','R=0.33','R=0.1','R=-1.0','LOCATION','BEST')
114 set(gca, 'YScale', 'log')
115 xlabel('River Velocity (m/s)')
116 ylabel('Damage per year')
117 xlim([min(seriehistorica(:,1)) max(seriehistorica(:,1))])
118
119 vidatotal
120
121

```