

### 3.1 Modelo com EDOs com taxas de transporte

```
%-----  
% ----- Modelo com EDOs com taxas de transporte -----  
%  
% alpha = 0.18; r = 0.97; K = 1; P0 = 0.1; condd = 0.5;  
% Clc = 0.8; C2 = 0.3; C3 = 0.5;  
%  
%-----  
clear all; clc;  
%-----  
% Matriz de taxa de transporte  
  
% Regras para as taxas de transporte:  
%     alpha1  alpha2  alpha3  
%     alpha4      *   alpha5  
%     alpha6  alpha7  alpha8  
  
alpha = 0.18;  
alpha1 = alpha; alpha2 = alpha; alpha3 = alpha; alpha4 = alpha;  
alpha5 = alpha; alpha6 = alpha; alpha7 = alpha; alpha8 = alpha;  
alpha0 = 0;  
  
m = 40; % número de linhas do pomar  
n = 50; % número de colunas do pomar  
p = max(m,n); q = min(m,n);  
M = zeros(m*n,m*n); % matriz de taxa de transporte  
k = 1;  
  
while k<q+1  
    for j=(1+(k-1)*p):k*p           % diagonal principal  
        for i=(1+(k-1)*p):k*p  
            if j==i+1  
                M(i,j) = alpha5;  
            elseif j==i-1  
                M(i,j) = alpha4;  
            else  
                M(i,j)=alpha0;  
            end  
        end  
    end  
    for j=(1+k*p):(k+1)*p         % 1ª diagonal a direita  
        for i=(1+(k-1)*p):k*p  
            if j==i+p  
                M(i,j) = alpha7;  
            elseif j==i+p+1  
                M(i,j) = alpha8;  
            elseif j==i+p-1  
                M(i,j) = alpha6;  
            else  
                M(i,j)=alpha0;  
            end  
        end  
    end  
    for j=(1+(k-1)*p):k*p         % 1ª diagonal a esquerda  
        for i=(1+k*p):(k+1)*p  
            if i==j+p
```

```

        M(i,j) = alpha2;
    elseif i==j+p+1
        M(i,j) = alpha1;
    elseif i==j+p-1
        M(i,j) = alpha3;
    else
        M(i,j)=alpha0;
    end
end
end
k=k+1;
end
M = M([1:m*n],[1:m*n]);

% Considerando 2000 árvores, temos:
%      P1      P2      P3      ... P50
%      P51      P52      P53      ... P100
%      P101     P102     P103     ... P150
%      ...      ...      ...      ... ...
%      P1951   P1952   P1953   ... P2000
%-----

%-----
% Método Runge-Kutta de 4a ordem para a equação do psilídeo de cada árvore

it = 119;
t0 = 0; tf = 12; h = tf/it;
t = linspace(t0,tf,it+1);
r = 0.97; % taxa de crescimento logístico
K = 1;    % capacidade de suporte
P0 = 0.1; % condição inicial
condc = 0.5; % condição para a difusão acontecer (50% da capacidade suporte)
% sem controle
Ps = zeros(it+1,(m*n)); % população de psilídeos P sem controle
Ps(1,1201) = P0;
us = zeros(it+1,m*n);
lambs = zeros(it+1,m*n);
% com controle
Pc = zeros(it+1,(m*n)); % população de psilídeos P com controle
Pc(1,1201) = P0;
uc = zeros(it+1,m*n);
lambc = zeros(it+1,m*n);
C1c = 0.8; % peso de x*rho na eq de estado (com controle)
C1s = 0;   % peso 0 de x*rho na eq de estado (sem controle)
C2 = 0.3; % peso de rho no funcional
C3 = 0.5; % peso de u no funcional
testes = -1; testec = -1; delta = 0.001; % parâmetros para o teste de
convergência

% Sem controle
while(testes < 0)
    oldus = us; oldPs = Ps; oldlambs = lambs;

    % Runge-Kutta para P (de frente para trás)
    for i=1:it; % loop das iterações
        for a=1:(m*n) % loop das árvores
            for x=1:(m*n)

```

```

        if Ps(i,x)>=condd & Ps(i,x)<1
            in(x) = M(x,a)*Ps(i,x);
        else
            in(x) = 0;
        end
    end
    Pin(a) = sum(in);
    k1 = r*Ps(i,a)*(1-(Ps(i,a)/K)) + Pin(a) - C1s*Ps(i,a)*us(i,a);
    for x=1:(m*n)
        if Ps(i,x)>=condd & Ps(i,x)<1
            in(x) = M(x,a)*Ps(i,x);
        else
            in(x) = 0;
        end
    end
    Pin(a) = sum(in);
    k2 = r*(Ps(i,a)+(h/2)*k1)*(1-((Ps(i,a)+(h/2)*k1)/K)) ...
        + Pin(a) - C1s*(Ps(i,a)+(h/2)*k1)*(us(i,a)+us(i+1,a))/2;
    for x=1:(m*n)
        if Ps(i,x)>=condd & Ps(i,x)<1
            in(x) = M(x,a)*Ps(i,x);
        else
            in(x) = 0;
        end
    end
    Pin(a) = sum(in);
    k3 = r*(Ps(i,a)+(h/2)*k2)*(1-((Ps(i,a)+(h/2)*k2)/K)) ...
        + Pin(a) - C1s*(Ps(i,a)+(h/2)*k2)*(us(i,a)+us(i+1,a))/2;
    for x=1:(m*n)
        if Ps(i,x)>=condd & Ps(i,x)<1
            in(x) = M(x,a)*Ps(i,x);
        else
            in(x) = 0;
        end
    end
    Pin(a) = sum(in);
    k4 = r*(Ps(i,a)+h*k3)*(1-((Ps(i,a)+h*k3)/K)) ...
        + Pin(a) - C1s*(Ps(i,a)+h*k3)*us(i+1,a);
    Ps(i+1,a) = Ps(i,a) + (h/6)*(k1 + 2*k2 + 2*k3 + k4);
end
end

% Runge-Kutta para lambda (de trás para frente)
for i=1:it; % loop das iterações
    j = it + 2 - i;
    for a=1:(m*n) % loop das árvores
        k1 = lambs(j,a)*(r*(2*(Ps(j,a)/K)-1) + C1s*us(j,a)) - 2*C2*Ps(j,a);
        k2 = (lambs(j,a)-(h/2)*k1)*(r*(2*((Ps(j,a)+Ps(j-1,a))/2)/K)-1) ...
            + C1s*((us(j,a)+us(j-1,a))/2)) - 2*C2*((Ps(j,a)+Ps(j-1,a))/2);
        k3 = (lambs(j,a)-(h/2)*k2)*(r*(2*((Ps(j,a)+Ps(j-1,a))/2)/K)-1) ...
            + C1s*((us(j,a)+us(j-1,a))/2)) - 2*C2*((Ps(j,a)+Ps(j-1,a))/2);
        k4 = (lambs(j,a)-h*k3)*(r*(2*(Ps(j-1,a)/K)-1) ...
            + C1s*us(j-1,a)) - 2*C2*Ps(j-1,a);
        lambs(j-1,a) = lambs(j,a) - (h/6)*(k1 + 2*k2 + 2*k3 + k4);
    end
end
end

```

```

for a=1:(m*n) % loop das árvores
    uls(:,a) = C1s*(lambs(:,a).*Ps(:,a))/(2*C3);
    us(:,a) = (oldus(:,a) + uls(:,a))/2;
end

% Teste de convergência
temp1s = delta*sum(norm(us)) - sum(norm(oldus - us));
temp2s = delta*sum(norm(Ps)) - sum(norm(oldPs - Ps));
temp3s = delta*sum(norm(lambs)) - sum(norm(oldlambs - lambs));
testes = min(temp1s, min(temp2s, temp3s))
end

% Com controle
while(testec < 0)
    olduc = uc; oldPc = Pc; oldlambc = lambc;

    % Runge-Kutta para P (de frente para trás)
    for i=1:it; % loop das iterações
        for a=1:(m*n) % loop das árvores
            for x=1:(m*n)
                if Pc(i,x)>=condd & Pc(i,x)<1
                    in(x) = M(x,a)*Pc(i,x);
                else
                    in(x) = 0;
                end
            end
            Pin(a) = sum(in);
            k1 = r*Pc(i,a)*(1-(Pc(i,a)/K)) + Pin(a) - C1c*Pc(i,a)*uc(i,a);
            for x=1:(m*n)
                if Pc(i,x)>=condd & Pc(i,x)<1
                    in(x) = M(x,a)*Pc(i,x);
                else
                    in(x) = 0;
                end
            end
            Pin(a) = sum(in);
            k2 = r*(Pc(i,a)+(h/2)*k1)*(1-((Pc(i,a)+(h/2)*k1)/K)) ...
                + Pin(a) - C1c*(Ps(i,a)+(h/2)*k1)*(uc(i,a)+uc(i+1,a))/2;
            for x=1:(m*n)
                if Pc(i,x)>=condd & Pc(i,x)<1
                    in(x) = M(x,a)*Pc(i,x);
                else
                    in(x) = 0;
                end
            end
            Pin(a) = sum(in);
            k3 = r*(Pc(i,a)+(h/2)*k2)*(1-((Pc(i,a)+(h/2)*k2)/K)) ...
                + Pin(a) - C1c*(Pc(i,a)+(h/2)*k2)*(uc(i,a)+uc(i+1,a))/2;
            for x=1:(m*n)
                if Pc(i,x)>=condd & Pc(i,x)<1
                    in(x) = M(x,a)*Pc(i,x);
                else
                    in(x) = 0;
                end
            end
            Pin(a) = sum(in);
            k4 = r*(Pc(i,a)+h*k3)*(1-((Pc(i,a)+h*k3)/K)) ...

```

```

        + Pin(a) - C1c*(Pc(i,a)+h*k3)*uc(i+1,a);
        Pc(i+1,a) = Pc(i,a) + (h/6)*(k1 + 2*k2 + 2*k3 + k4);
    end
end

% Runge-Kutta para lambda (de trás para frente)
for i=1:it; % loop das iterações
    j = it + 2 - i;
    for a=1:(m*n) % loop das árvores
        k1 = lambc(j,a)*(r*(2*(Pc(j,a)/K)-1) + C1c*uc(j,a)) - 2*C2*Pc(j,a);
        k2 = (lambc(j,a)-(h/2)*k1)*(r*(2*((Pc(j,a)+Pc(j-1,a))/2)/K)-1) ...
            + C1c*((uc(j,a)+uc(j-1,a))/2) - 2*C2*((Pc(j,a)+Pc(j-1,a))/2);
        k3 = (lambc(j,a)-(h/2)*k2)*(r*(2*((Pc(j,a)+Pc(j-1,a))/2)/K)-1) ...
            + C1c*((uc(j,a)+uc(j-1,a))/2) - 2*C2*((Pc(j,a)+Pc(j-1,a))/2);
        k4 = (lambc(j,a)-h*k3)*(r*(2*(Pc(j-1,a)/K)-1) ...
            + C1c*uc(j-1,a)) - 2*C2*Pc(j-1,a);
        lambc(j-1,a) = lambc(j,a) - (h/6)*(k1 + 2*k2 + 2*k3 + k4);
    end
end

for a=1:(m*n) % loop das árvores
    ulc(:,a) = C1c*(lambc(:,a).*Pc(:,a))/(2*C3);
    uc(:,a) = (olduc(:,a) + ulc(:,a))/2;
end

% Teste de convergência
temp1c = delta*sum(norm(uc)) - sum(norm(olduc - uc));
temp2c = delta*sum(norm(Pc)) - sum(norm(oldPc - Pc));
temp3c = delta*sum(norm(lambc)) - sum(norm(oldlambc - lambc));
testec = min(temp1c, min(temp2c, temp3c))
end
%-----
%-----
% Matriz mapa do pomar

%           A(1,1)  A(1,2)  ... A(1,50)
%           A(2,1)  A(2,2)  ... A(2,50)
%           ...     ...     ... ..
%           A(40,1) A(40,2) ... A(40,50)

% Sem controle
for k=1:it+1;
    for i=1:m
        for j=1:n
            As(i,j,k) = Ps(k,n*(i-1)+j); % árvore A(i,j) para cada iteração k
        end
    end
end

% Com controle
for k=1:it+1;
    for i=1:m
        for j=1:n
            Ac(i,j,k) = Pc(k,n*(i-1)+j); % árvore A(i,j) para cada iteração k
        end
    end
end

```

```

end
%-----

%-----
% Exibindo a vista por cima do pomar

% Definindo o mapa de cores
c0 = 0; c1 = 1;

mapacor = [1,1,1 ; ... %branco c0
           0,0,0]; ... %preto c1

% Preparando a matriz para receber as cores
% Sem controle
nAs = zeros(m,n,it+1);
nAs(As==0) = c0;
nAs(As>0) = c1;
% Com controle
nAc = zeros(m,n,it+1);
nAc(Ac==0) = c0;
nAc(Ac>0) = c1;

% Exibindo a imagem
z=1;
figure
for k=1:it+1
    k
    w = mod(k, ((it+1)/12));
    if w==0
        z=z+1;
    end

    subplot(1,2,1)
    imagesc(nAs(:,:,k),[0,1]); colormap(mapacor); axis image;
    title(['Orchard without control'];[' ']); drawnow;

    subplot(1,2,2)
    imagesc(nAc(:,:,k),[0,1]); colormap(mapacor); axis image;
    title(['Orchard with control'];[' ']); drawnow;

    % Calculando a porcentagem de árvores com psilídeos
    conts = double(nAs(:,:,k)>0.0001);
    parts = size(find(conts));
    zs(k) = parts(1)/(m*n);
    contc = double(nAc(:,:,k)>0.0001);
    partc = size(find(contc));
    zc(k) = partc(1)/(m*n);

end

% Porcentagem de árvores com psilídeos
zs(it+1) % sem controle
zc(it+1) % com controle

%-----
% Gráfico da primeira árvore e demais

```

```

figure % A(24,1)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1151),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1151),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    legend('Without control','With control')
    title('A 1151')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1151),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

figure % A(24,2)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1152),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1152),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1152')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1152),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

figure % A(24,3)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1153),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1153),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1153')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1153),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

```

```

figure % A(25,1)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1201),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1201),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1201 (primary tree)')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1201),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

```

```

figure % A(25,2)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1202),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1202),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1202')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1202),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

```

```

figure % A(25,3)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1203),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1203),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1203')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1203),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

```



```

figure % A(26,1)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1251),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1251),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1251')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1251),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

figure % A(26,2)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1252),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1252),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1252')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1252),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

figure % A(26,3)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ps(:,1253),'-.','linewidth',2,'color',[0.12 0.36 1.00])
    axis([0 12 0 1.5])
    plot(t,Pc(:,1253),'-','linewidth',2,'color',[1.00 0.27 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 1253')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,uc(:,1253),'linewidth',2,'color',[0.00 0.55 0.00])
    axis([0 12 0 1.5])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control u','FontSize',14)

```

### 3.2 Modelo com método GRASP

```
%-----  
% ----- Modelo com método GRASP -----  
%  
% r = 0.1; K = 1; D = 0.008; lim = 0.5; limV = 0.7; d = 15;  
% C1 = 0.2; C2 = 1; C3 = 0.1;  
%  
%-----  
clc; clear all;  
%-----  
  
% Parâmetros do problema  
m = 40;      % número de linhas  
n = 50;      % número de talhões (colunas)  
it = 360;    % número de iterações (360 dias)  
r = 0.1;     % taxa de crescimento intrínseco 0.33  
K = 1;       % capacidade suporte  
D = 0.008;   % coeficiente de difusão 0.007  
lim = 0.5;   % porcentagem de psilídeos numa árvore para que eles migrem para  
outra árvore  
limV = 0.7;  % porcentagem de psilídeos numa árvore para que seja colocado veneno  
d = 15;      % o veneno é aplicado a cada 15 dias  
% C1 = 0.3; C2 = 0.3; C3 = 0.3;  
C1 = 0.2; C2 = 1; C3 = 0.1;  
  
% Parâmetros do algoritmo  
MVc = zeros(n,it); % matriz de aplicação de veneno por talhão no tempo  
deltac = 10^30;  
Vc = [0:0.05:1]; % com veneno (controle)  
Vs = 0; % sem veneno (controle)  
  
%-----  
% Iniciando o código COM controle  
for x = 1:length(Vc) % varrer todas as opções de veneno  
    qtV = Vc(x);  
    MEc = zeros(m,n,it); % matriz de espalhamento no tempo  
    MEc(25,1,1) = 0.1; % define a condição inicial  
    for k = 1:it-1  
        for j = 1:n  
            a = find(MEc(:,j,k) > limV); % busca alguma árvore (no talhão todo)  
que tenha mais de limV psilídeos  
                if length(a) > 0 % se encontrar alguma árvore com mais de  
limV psilídeos, então...  
                    for w=1:4 % aplica veneno nos 4 talhões à direita  
e à esquerda  
                        if (j+w) < n && mod(k,d)==0  
                            MVc(j+w,k) = qtV;  
                        end  
                        if (j-w) > 0 && mod(k,d)==0  
                            MVc(j-w,k) = qtV;  
                        end  
                    end  
                end  
                if mod(k,d)==0  
                    MVc(j,k) = qtV; % aplica no próprio talhão também  
                end  
            end  
        end  
    end  
end
```

```

% Dinâmica de crescimento dos psilídeos
for i=1:m
    Pc = MEc(i,j,k);
    MEc(i,j,k+1) = Pc + r*Pc*(1 - Pc/K) - C1*MVc(j,k)*Pc;
end

% Dinâmica de espalhamento da doença
for i = 1:m
    if j-1 <=0 || MEc(i,j-1,k) < lim % espalhamento à esquerda
        espE = 0;
    else
        espE = MEc(i,j-1,k);
    end
    if j+1 > n || MEc(i,j+1,k) < lim % espalhamento à direita
        espD = 0;
    else
        espD = MEc(i,j+1,k);
    end
    if i-1 <=0 || MEc(i-1,j,k) < lim % espalhamento acima
        espC = 0;
    else
        espC = MEc(i-1,j,k);
    end
    if i+1 > m || MEc(i+1,j,k) < lim % espalhamento abaixo
        espB = 0;
    else
        espB = MEc(i+1,j,k);
    end
    MEc(i,j,k+1) = MEc(i,j,k+1) + D*(espE + espD + espC + espB -
4*MEc(i,j,k));
end
end

% Escolhendo a 'soma' mínima
soma=0;
for k=1:it
    for j=1:n
        soma = soma + C2*sum(MEc(:,j,k))^2 + C3*MVc(j,k)^2;
    end
end

% Atulizando a melhor soma
if soma < deltac
    melhor_solc = soma;
    qtd_V_otimoc = qtV;
    melhor_matriz_espalhamentoc = MEc;
end

end
%-----
%-----
% Iniciando o código SEM controle
qtV = Vs;
MEs = zeros(m,n,it); % matriz de espalhamento no tempo

```

```

MEs(25,1,1) = 0.1; % define a condição inicial
for k = 1:it-1
    for j = 1:n
        % Dinâmica de crescimento dos psilídeos
        for i=1:m
            Ps = MEs(i,j,k);
            MEs(i,j,k+1) = Ps + r*Ps*(1 - Ps/K);
        end
        % Dinâmica de espalhamento da doença
        for i = 1:m
            if j-1 <=0 || MEs(i,j-1,k) < lim % espalhamento à esquerda
                espE = 0;
            else
                espE = MEs(i,j-1,k);
            end
            if j+1 > n || MEs(i,j+1,k) < lim % espalhamento à direita
                espD = 0;
            else
                espD = MEs(i,j+1,k);
            end
            if i-1 <=0 || MEs(i-1,j,k) < lim % espalhamento acima
                espC = 0;
            else
                espC = MEs(i-1,j,k);
            end
            if i+1 > m || MEs(i+1,j,k) < lim % espalhamento abaixo
                espB = 0;
            else
                espB = MEs(i+1,j,k);
            end
            MEs(i,j,k+1) = MEs(i,j,k+1) + D*(espE + espD + espC + espB -
4*MEs(i,j,k));
        end
    end
end
%-----
%-----
% Exibindo a vista por cima do pomar

% Definindo o mapa de cores
c0 = 0; c1 = 1;

mapacor = [1,1,1 ; ... %branco c0
           0,0,0]; ... %preto c1

% Preparando a matriz para receber as cores
% Sem controle
nMEs = zeros(m,n,it);
nMEs(MEs==0) = c0;
nMEs(MEs>0) = c1;
% Com controle
nMEc = zeros(m,n,it);
nMEc(MEc==0) = c0;
nMEc(MEc>0) = c1;

% Exibindo a imagem

```

```

z = 0;

figure
subplot(1,2,1)
pause(0.1)
imagesc(nMEs(:, :, k), [0,1]); colormap(mapacor); axis image;
title({'Pomar sem controle'}; ['Mês ', num2str(z)]); drawnow;

subplot(1,2,2)
imagesc(nMEc(:, :, k), [0,1]); colormap(mapacor); axis image;
title({'Pomar com controle'}; ['Mês ', num2str(z)]); drawnow; pause(0.2)

z = 1;
for k=1:it
w = mod(k, (it/12));
if w==0
z=z+1;
end

subplot(1,2,1)
imagesc(nMEs(:, :, k), [0,1]); colormap(mapacor); axis image;
title({'Orchard without control'}; [' ']); drawnow;

subplot(1,2,2)
imagesc(nMEc(:, :, k), [0,1]); colormap(mapacor); axis image;
title({'Orchard with control'}; [' ']); drawnow;

% Calculando a porcentagem de árvores com psilídeos
conts = double(MEs(:, :, k)>0.000);
parts = size(find(conts));
zs(k) = parts(1)/(m*n);
contc = double(MEc(:, :, k)>0.000);
partc = size(find(contc));
zc(k) = partc(1)/(m*n);

end

% Porcentagem de árvores com psilídeos
zs(it) % sem controle
zc(it) % com controle

%-----
% Gráfico da primeira árvore e demais

t = linspace(0,12,it);
Ms1 = zeros(1,it); Ms2 = zeros(1,it);
Mc1 = zeros(1,it); Mc2 = zeros(1,it);
for k=1:it
Ms1(k) = MEs(25,1,k);
Ms2(k) = MEs(28,5,k);
Ms01(k) = MEs(24,1,k);
Ms02(k) = MEs(24,2,k);
Ms03(k) = MEs(24,3,k);
Ms05(k) = MEs(25,2,k);
Ms06(k) = MEs(25,3,k);
Ms07(k) = MEs(26,1,k);
Ms08(k) = MEs(26,2,k);

```

```

Ms09(k) = MEs(26,3,k);
Mc1(k) = MEc(25,1,k);
Mc2(k) = MEc(28,5,k);
Mc01(k) = MEc(24,1,k);
Mc02(k) = MEc(24,2,k);
Mc03(k) = MEc(24,3,k);
Mc05(k) = MEc(25,2,k);
Mc06(k) = MEc(25,3,k);
Mc07(k) = MEc(26,1,k);
Mc08(k) = MEc(26,2,k);
Mc09(k) = MEc(26,3,k);
end
v1 = zeros(1,it); v2 = zeros(1,it); v3 = zeros(1,it);
for k=1:it
    v1(k) = MVc(1,k);
    v2(k) = MVc(5,k);
    v01(k) = MVc(1,k);
    v02(k) = MVc(2,k);
    v03(k) = MVc(3,k);
    v05(k) = MVc(2,k);
    v06(k) = MVc(3,k);
    v07(k) = MVc(1,k);
    v08(k) = MVc(2,k);
    v09(k) = MVc(3,k);
end

figure % A(24,1)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ms01,'-','linewidth',2,'color',[0.12 0.36 1.00])
    plot(t,Mc01,'-','linewidth',2,'color',[1.00 0.27 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    legend('Without control','With control')
    title('A 24,1')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v01,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control V','FontSize',14)

figure % A(24,2)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ms02,'-','linewidth',2,'color',[0.12 0.36 1.00])
    plot(t,Mc02,'-','linewidth',2,'color',[1.00 0.27 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 24,2')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v02,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)

```

```

ylabel('Control V','FontSize',14)

figure % A(24,3)
subplot(2,1,1)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,Ms03,'-','linewidth',2,'color',[0.12 0.36 1.00])
plot(t,Mc03,'-','linewidth',2,'color',[1.00 0.27 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Population P','FontSize',14)
title('A 24,3')
subplot(2,1,2)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,v03,'linewidth',2,'color',[0.00 0.55 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Control V','FontSize',14)

figure % A(25,1)
subplot(2,1,1)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,Ms1,'-','linewidth',2,'color',[0.12 0.36 1.00])
plot(t,Mc1,'-','linewidth',2,'color',[1.00 0.27 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Population P','FontSize',14)
title('A 25,1 (primary tree)')
subplot(2,1,2)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,v1,'linewidth',2,'color',[0.00 0.55 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Control V','FontSize',14)

figure % A(25,2)
subplot(2,1,1)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,Ms05,'-','linewidth',2,'color',[0.12 0.36 1.00])
plot(t,Mc05,'-','linewidth',2,'color',[1.00 0.27 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Population P','FontSize',14)
title('A 25,2')
subplot(2,1,2)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,v05,'linewidth',2,'color',[0.00 0.55 0.00])
xlabel('Time (months)','FontSize',14)
ylabel('Control V','FontSize',14)

figure % A(25,3)
subplot(2,1,1)
hold on; grid on;
set(gca,'FontSize',14)
plot(t,Ms06,'-','linewidth',2,'color',[0.12 0.36 1.00])
plot(t,Mc06,'-','linewidth',2,'color',[1.00 0.27 0.00])
xlabel('Time (months)','FontSize',14)

```

```

    ylabel('Population P','FontSize',14)
    title('A 25,3')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v06,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control V','FontSize',14)

figure % A(26,1)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ms07,'-','linewidth',2,'color',[0.12 0.36 1.00])
    plot(t,Mc07,'-','linewidth',2,'color',[1.00 0.27 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 26,1')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v07,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control V','FontSize',14)

figure % A(26,2)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ms08,'-','linewidth',2,'color',[0.12 0.36 1.00])
    plot(t,Mc08,'-','linewidth',2,'color',[1.00 0.27 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 26,2')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v08,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control V','FontSize',14)

figure % A(26,3)
subplot(2,1,1)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,Ms09,'-','linewidth',2,'color',[0.12 0.36 1.00])
    plot(t,Mc09,'-','linewidth',2,'color',[1.00 0.27 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Population P','FontSize',14)
    title('A 26,3')
subplot(2,1,2)
    hold on; grid on;
    set(gca,'FontSize',14)
    plot(t,v09,'linewidth',2,'color',[0.00 0.55 0.00])
    xlabel('Time (months)','FontSize',14)
    ylabel('Control V','FontSize',14)

```



