



UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA  
DEPARTAMENTO DE MATEMÁTICA APLICADA



ANDRÉ LUIZ DOS SANTOS

## **Dinâmica intra e interespecífica de duas populações sob o efeito tóxico de uma pluma evolutiva de poluente**

Campinas  
27/11/2019

ANDRÉ LUIZ DOS SANTOS

**Dinâmica intra e interespecífica de duas populações sob o efeito tóxico de uma pluma evolutiva de poluente**

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do(a) Prof. Dr. João Frederico da Costa Azevedo Meyer.

## Resumo

Neste trabalho, a partir da modelagem, da aproximação e da simulação do movimento de uma pluma de poluente num meio, estudam-se os efeitos deste mesmo impacto ambiental sobre duas populações que interagem entre si (a intraespecificidade) e uma com a outra (a interespecificidade). O modelo adotado nesta modelagem matemática é o de um sistema não linear de equações a derivadas parciais de dispersão-migração-dinâmica populacional, incluindo a presença de um material tóxico que afeta as duas populações. No tocante à ação interespecífica, usam-se as clássicas equações de Lotka-Volterra[1][2].

Na aproximação numérica da solução deste sistema não linear, usam-se, respectivamente Diferenças Finitas Centradas na aproximação das variáveis temporais (já que o domínio está num plano), Crank-Nicolson na aproximação da variável temporal e um Método de Ponto Fixo para o tratamento numérico da não-linearidade[3]. A saída é qualitativamente apresentada via superfícies e curvas de nível.

## 1 Introdução

Em estudos de ecologia matemática, muitas vezes a interdisciplinaridade leva a um acoplamento de equações em modelagens. Neste caso, junto com a modelagem do movimento de uma pluma de poluente com efeitos tóxicos para duas populações que interagem num meio limitado, temos a possibilidade de reunir num só modelo as equações de movimento da mancha e das populações consideradas não apenas no tempo mas, também, no espaço. Assim, a passagem da pluma de poluente pelo meio em função de seu transporte advectivo, afeta as duas populações podendo, inclusive, levar a um desequilíbrio destas a curto ou a médio prazos.

Para a condição inicial usam-se algumas diferentes possibilidades, em termos de gerar cenários, e como condição de contorno, usam-se nos três casos, condições de von Neumann homogêneas. No espaço, usam-se diferenças finitas centradas, no tempo, o método de Crank-Nicolson e na aproximação sucessiva (em cada passo no tempo) para as tratar a não-linearidade das equações do sistema usa-se um método iterativo dito de ponto fixo já considerado clássico, devido a Douglas e Dupont.

## 2 Modelo matemático

O modelo adotado nesta modelagem é o de um sistema não linear de equações a derivadas parciais de dispersão-migração-dinâmica populacional. Onde  $C$  denota a concentração de poluentes,  $C = C(x, y, t)$ , com  $(x, y) \in \Omega \subset \mathbb{R}^2$  e  $t \in I = (0, T] \subset \mathbb{R}$ . Com  $C(x, t, 0) = C_0$ . Com isso temos as seguintes equações:

$$\frac{\partial C}{\partial t} - \alpha_c \Delta C + \mathbb{V} \cdot \nabla C + \mu_c = f \quad (1)$$

$$\frac{\partial P}{\partial t} - \alpha_p \Delta P + \mathbb{W} \cdot \nabla P + \mu_p = \lambda_p P \left( 1 - \frac{P+Q}{K} \right) \quad (2)$$

$$\frac{\partial Q}{\partial t} - \alpha_q \Delta Q + \mathbb{U} \cdot \nabla Q + \mu_q = \lambda_q Q \left( 1 - \frac{P+Q}{K} \right) \quad (3)$$

onde:

$\alpha_c, \alpha_p, \alpha_q$  - representa a dispersão do poluente e das espécies

$\mathbb{V}, \mathbb{W}, \mathbb{U}$  - representa o fluxo do poluente e os fluxos migratórios de cada espécie

$\mu_c$  - representa o decaimento do poluente no meio

$f$  - representa as fontes de poluentes

$\mu_p = (\kappa_p C)P$  - é a morbidade da espécie  $P(x, y, t)$  em função da poluição  $C(x, y, t)$

$\mu_q = (\kappa_q C)Q$  - é a morbidade da espécie  $Q(x, y, t)$  em função da poluição  $C(x, y, t)$

Para isso nosso dominio será uma lagoa retangular, e as especies irão ter como ponto inicial as extremidas do dominio.

## 2.1 Condições iniciais

$$P(x, y, 0) \Big|_{\Omega} = P_0, (x, y) \in \Omega \quad (4)$$

$$Q(x, y, 0) \Big|_{\Omega} = Q_0, (x, y) \in \Omega \quad (5)$$

## 2.2 Condição de contorno

Para esse problema usamos a condição de contorno de von Neumann homogênea.

$$\frac{\partial C}{\partial \eta}(x, y, t) \Big|_{\Gamma} = 0, (x, y) \in \Gamma \subset \partial\Omega, t \subset I \quad (6)$$

### 3 Programa

```
% Usando a difusão advectiva e reativa de um poluente, simular a
interação
% competitiva de duas espécies com competição intraespecífica
(Verhulst)
%
% Resolução das equações evolutivas de difusão-advectação-reação com
decaimento
% e de dinâmica espacial e evolutiva das duas espécies
% por diferenças divididas e Crank-Nicolson

clear all
clc

% dados do problema: Poluição
alfc = 0.125e-2; uc = 0.027778; vc = 0.025; muc = 0.1e-7;
%
% dados do problema: Populações
alfa = 0.125e-2; alfb=0.375e-2;
ua = 0.027778; va = 0.025; ub = -0.037778; vb = -0.0125;
mua = 0.1e-1; mub=0.2e-1;
K = 5000;
lba=0.25; lbb=0.125;
roa=0.05; rob=0.05;

%
% fontes de poluição
fc1 = 1; fc2 = 1.25; fc3 = 0.75; fc4 = 0.5;

%dados do dominio
l = 4.5; h = 3; tf = 25;

%dados da discretização
nx = 50; nnx = nx+1; dx = 1/nx;
ny = 30; nny = ny+1; dy = h/ny;
nt = 100; dt = tf/nt; nn = nnx*nny;

% localização das fontes a, b, c, d
%ia=floor(3*nny+nnx/5); ib=floor(7*nny+2*nnx/5);
%ic=floor(12*nny+3*nnx/5); id=floor(14*nny+4*nnx/5);
%ic1=floor(17*nny+2*nny/3); ic2=floor(10*nny+nny/3);
%ic3=floor(12*nny+3*nny/4); ic4=floor(28*nny+nny/4);
% valores iniciais/condição inicial poluente
cz=0.01*ones(nn,1);
% condições iniciais populações a e b
az=0.0001*zeros(nn,1); bz=az/2;
ia1=1; ia2=2; ia3=nny+2; ia4=nny+1; ia5=2*nny+1;
az(ia1)=0.25; az(ia2)=0.125; az(ia3)=0.1875; az(ia4)=0.125;
az(ia5)=0.0675;
ib1=nn; ib2=nn-1; ib3=nn-nny; ib4=nn-nny-1; ib5=nn-2*nny;
bz(ib1)=0.25; bz(ib2)=0.125; bz(ib3)=0.1875; bz(ib4)=0.125;
bz(ib5)=0.0675;
%calculos auxiliares
```

```

ddx = dx*dx; ddy = dy*dy;
d4x = 4*dx; d4y = 4*dy;
% cálculos para fatores nas matrizes
atxc = alfc*dt/ddx; atyc = alfc*dt/ddy;
utxc = uc*dt/d4x; vtyc = vc*dt/d4y;
atxa = alfa*dt/ddx; atya = alfa*dt/ddy;
utxa = ua*dt/d4x; vtya = va*dt/d4y;
atxb = alfb*dt/ddx; atyb = alfb*dt/ddy;
utxb = ub*dt/d4x; vtyb = vb*dt/d4y;
mdtc = muc*dt/2; mdta = 0; mdtb = 0;
%
% preparação dos termos das seis matrizes:
% esquerda de c, esquerda de a, esquerda de b
% direita de c, direita de a, direita de b
dssdec = (-atxc/2)+utxc; dssddc = -dssdec;
dssdea = (-atxa/2)+utxa; dssdda = -dssdea;
dssdeb = (-atxb/2)+utxb; dssddb = -dssdeb;
dsspec = (-atyc/2)+vtyc; dsspc = -dsspec;
dsspea = (-atya/2)+vtya; dsspa = -dsspea;
dsspeb = (-atyb/2)+vtyb; dsspb = -dsspeb;
dpec = 1+atxc+atyc+mdtc;
dpea = 1+atxa+atya;
dpeb = 1+atxb+atyb;
dpdc = 1-atxc-atyc;
dpda = 1+atxa+atya;
dpdb = 1+atxb+atyb;
dsipec = (-atyc/2)-vtyc; dsipdc = -dsipec;
dsipea = (-atya/2)-vtya; dsipda = -dsipea;
dsipeb = (-atyb/2)-vtyb; dsipdb = -dsipeb;
dsidec = (-atxc/2)-utxc; dsidc = -dsidec;
dsidea = (-atxa/2)-utxa; dsidda = -dsidea;
dsideb = (-atxb/2)-utxb; dsiddb = -dsideb;
%
% Números de limiar de Péclét para c, a e b
[uc*dx/alfc vc*dy/alfc];
[ua*dx/alfa va*dy/alfc];
[ub*dx/alfb vb*dy/alfb];
% montar as matrizes e o termo independente
mec = sparse(nn); mdc = sparse(nn);
mea = sparse(nn); mda = sparse(nn);
meb = sparse(nn); mdb = sparse(nn);
mintae=mea; mintbe=meb;
mintad=mda; mintbd=mdb;
cb=zeros(nn,1); ab=zeros(nn,1); bb=zeros(nn,1);
% definir condições iniciais de c, de a e de b
%
cb(ic1)=dt*fc1; cb(ic2)=dt*fc2;
cb(ic3)=dt*fc3; cb(ic4)=dt*fc4;
%ab
%bb

for i = 1 : nn
    mec(i,i) = dpec; mdc(i,i) = dpdc;
    mea(i,i) = dpea; mda(i,i) = dpda;

```

```

    meb(i,i) = dpeb; mdb(i,i) = dpdb;
end

for i = 1 : nn-1
    mec(i+1,i) = dsipec; mdc(i+1,i) = dsipdc;
    mec(i,i+1) = dsspec; mdc(i,i+1) = dsspdcc;
    mea(i+1,i) = dsipea; mda(i+1,i) = dsipda;
    mea(i,i+1) = dsspea; mda(i,i+1) = dsspda;
    meb(i+1,i) = dsipeb; mdb(i+1,i) = dsipdb;
    meb(i,i+1) = dsspeb; mdb(i,i+1) = dsspdb;
end

for i = 1 : nn-nny
    mec(i+nny,i) = dsidec; mdc(i+nny,i) = dsidcc;
    mec(i,i+nny) = dssdec; mdc(i,i+nny) = dssddcc;
    mea(i+nny,i) = dsidea; mda(i+nny,i) = dsidda;
    mea(i,i+nny) = dssdea; mda(i,i+nny) = dssdda;
    meb(i+nny,i) = dsideb; mdb(i+nny,i) = dsidbb;
    meb(i,i+nny) = dssdeb; mdb(i,i+nny) = dssddb;
end

% Correções das bordas horizontais e verticais
%
% Borda Vertical à Esquerda
for i=1:nny
    mec(i,i+nny)=-atxc; mdc(i,i+nny)=atxc;
    mea(i,i+nny)=-atxa; mda(i,i+nny)=atxa;
    meb(i,i+nny)=-atxb; mdb(i,i+nny)=atxb;
end

% Borda Horizontal Superior
for i=1:nnx-1
    ind=i*nny;
    mec(ind,ind+1)=0; mdc(ind,ind+1)=0;
    mec(ind,ind-1)=-atyc; mdc(ind,ind-1)=atyc;
    mea(ind,ind+1)=0; mda(ind,ind+1)=0;
    mea(ind,ind-1)=-atya; mda(ind,ind-1)=atya;
    meb(ind,ind+1)=0; mdb(ind,ind+1)=0;
    meb(ind,ind-1)=-atyb; mdb(ind,ind-1)=atyb;
end

mec(nn,nn-1)=-atyc; mdc(nn,nn-1)=atyc;
mea(nn,nn-1)=-atya; mda(nn,nn-1)=atya;
meb(nn,nn-1)=-atyb; mdb(nn,nn-1)=atyb;

% Borda Horizontal Inferior
for i=2:nnx-1
    ind=(i-1)*nny+1;
    mec(ind,ind+1)=-atyc; mdc(ind,ind+1)=atyc;
    mec(ind,ind-1)=0; mdc(ind,ind-1)=0;
    mea(ind,ind+1)=-atya; mda(ind,ind+1)=atya;
    mea(ind,ind-1)=0; mda(ind,ind-1)=0;
    meb(ind,ind+1)=-atyb; mdb(ind,ind+1)=atyb;
    meb(ind,ind-1)=0; mdb(ind,ind-1)=0;
end

```

```

end
mec(1,2)=-atyc; mdc(1,2)=atyc;
mea(1,2)=-atya; mda(1,2)=atya;
meb(1,2)=-atyb; mdb(1,2)=atyb;

% Borda vertical à direita
for i=1:nny
    ind=(nnx-1)*nny+i;
    mec(ind,ind-nny)=-atxc; mdc(ind,ind-nny)=atxc;
    mea(ind,ind-nny)=-atxa; mda(ind,ind-nny)=atxa;
    meb(ind,ind-nny)=-atxb; mdb(ind,ind-nny)=atxb;
end
a=az; b=bz; c=cz;
% Repetições temporais
for it=1:nt
    aint=(az+a)/2; bint=(bz+b)/2; cint=(c+cz)/2;
    mintae=mea; mintad=mda; mintbe=meb; mintdb=mdb;
    for j=1:5
        aint=(az+a)/2; bint=(bz+b)/2; cint=(c+cz)/2;
        for i=1:nn
            mintae(i,i)=mea(i,i)+(mua*cint(i)-lba
+ (aint(i)+roa*bint(i)))*(dt/2);
            mintad(i,i)=mda(i,i)-(mua*cint(i)-lba
+ (aint(i)+roa*bint(i)))*(dt/2);
            mintbe(i,i)=meb(i,i)+(mub*cint(i)-lbb
+ (bint(i)+rob*aint(i)))*(dt/2);
            mintbd(i,i)=mdb(i,i)-(mub*cint(i)-lbb
+ (bint(i)+rob*bint(i)))*(dt/2);
        end
        c=mec\ (mdc*cz+cb);
        a=mintae\ (mintad*az+ab);
        b=mintbe\ (mintbd*bz+bb);
    end
    % visualização
    for i=1:nny;
        for j=1:nnx
            ind=(j-1)*nny + i;
            verc(i,j)=c(ind);
        end
    end
    % subplot(1,2,1), contour(verc,12), grid on
    % subplot(1,2,2), surf(verc), grid on
    % pause(0.1)
    cz=c;
    az=a;
    bz=b;
end
subplot(1,2,1), contour(verc,12), grid on
subplot(1,2,2), surf(verc), grid on

figure(2)
for i=1:nny;
    for j=1:nnx
        ind=(j-1)*nny + i;

```



```

        vera(i,j)=a(ind);
    end
end
surf(vera), grid

figure(3)
for i=1:nny;
    for j=1:nnx
        ind=(j-1)*nny + i;
        verb(i,j)=b(ind);
    end
end
surf(verb), grid

```

## 4 Resultados

Após a execução do programa no matlab obtivemos os seguintes resultados;

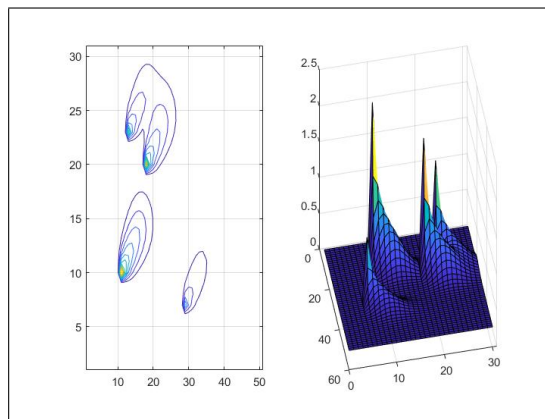


Figura 1: Dispersão do poluente

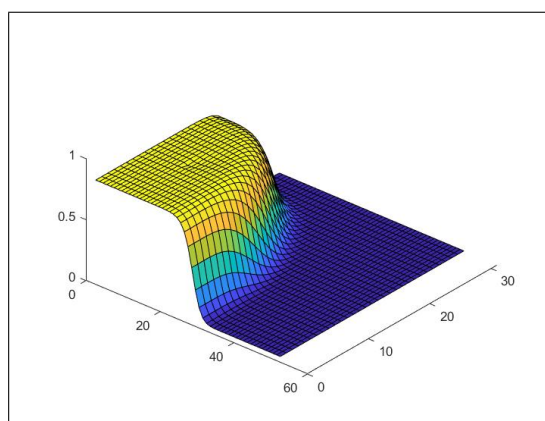


Figura 2: Dispersão da população P

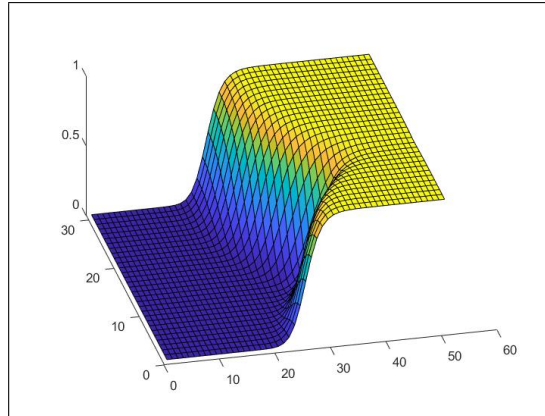


Figura 3: Dispersão da população Q

## 5 Conclusão

Com essas simulações foi possível observar que ambas as espécies sofrem com o poluente, mas olhando individualmente para cada uma é possível observar que, a primeira espécie apesar de crescer bem próximo a uma das fontes, como é possível ver na figura 1 não é tão afetada quanto a segunda, pelo fato que o vento está no sentido do crescimento da espécie. Fazendo com que o poluente evolua rapidamente para fora da área de crescimento, assim não afetando tanto seu crescimento.

Já com a segunda espécie é possível ver claramente que o poluente a afetou mais, por conta da direção do vento ser de encontro ao sentido de crescimento da segunda população, como é possível na figura 3, em que a influência do poluente pode ser vista no vinco da superfície.

Com isso podemos concluir que apesar de nem todas as fontes de poluentes estarem próximas ao ponto inicial de crescimento das espécies, podemos afirmar que se tivermos um vento que aponte no sentido contrário afetará em uma escala bem maior as populações. Esse cenário pode ser pior caso uma das espécies fosse predadora da outra, o que não é o caso dessa simulação, mas isso poderia causar um grande desequilíbrio na cadeia alimentar do ambiente.

## Referências

- [1] Leah Edelstein-Keshet. *Mathematical models in biology*. SIAM, 1988.
- [2] James D Murray. *Mathematical biology*. 1993.
- [3] Márcia A Gomes Ruggiero e Vera Lúcia da Rocha Lopes. *Cálculo numérico: aspectos teóricos e computacionais*. Makron Books do Brasil, 1997.