The method of lines for the numerical solution of a mathematical model for capillary formation: The role of tumor angiogenic factor in the extra-cellular matrix

Arzu Erdem a, Serdal Pamuk b,*

a Applied Mathematical Sciences Research Center, University of Kocaeli, Ataturk Bulvari, 41300 Kocaeli, Turkey
b Department of Mathematics, University of Kocaeli, Umuttepe Campus, 41800 Kocaeli, Turkey

Abstract

In this paper we present the method of lines for the numerical solution of a mathematical model for capillary formation in two space dimensions x, y. We study the tumor angiogenic factor (TAF) equation, and suppose there is no tumor source supplied to the system for simplicity. This method is an approach to the numerical solution of partial differential equations that involve a time variable t and space variables x, y, ... We provide computer programs written in Matlab for linear and nonlinear case, and provide figures that show the TAF progression in time.

Keywords: Method of lines; Capillary formation; Tumor angiogenic factor; Porosity; Numerical solution

1. Introduction

The geometry of the “coupled” problem studied in [3,6] can be envisaged as in Fig. 1. In the x – y plane we envisage a capillary segment of unit length located along the y-axis with a TAF source located somewhere along the line x = 1. Basically, the problem consists of two parts: (i) the progression of EC’s on the y-axis, namely in the capillary (1D problem) studied in [5], (ii) the progression of TAF in the unit square, namely in the extra-cellular matrix (ECM) (2D problem). The second case is the focus of this paper.

In the ECM we consider the following initial boundary-value problem originally presented in [3,6]:

\begin{align}
    & u_t = \nabla (D(x,y) \nabla (u^n)), \quad (x,y,t) \in \Omega_T := \{0 < x, y < 1, 0 < t \leq T\}, \\
    & u(x,y,0) = 0, \quad (x,y) \in (0,1) \times (0,1), \\
    & u_t(0,y,t) - zu(0,y,t) = 0, \quad 0 < y < 1, \quad 0 < t \leq T,
\end{align}

* Corresponding author.
E-mail addresses: aerdem@kou.edu.tr (A. Erdem), spamuk@kou.edu.tr (S. Pamuk).
\[ u(1, y, t) = 1 - \varepsilon \cos(2\pi y), \quad 0 < y < 1, \ 0 < t \leq T, \]  
\[ u_0(x, 0, t) = u_0(x, 1, t) = 0, \quad 0 < x < 1, \ 0 < t \leq T, \]  
where \( u(x, t) \) is TAF concentration and \( D(x, y) \) is the TAF diffusion coefficient that can depend on the space variables \( x, y \). Here, \( n \) is what we call “porosity” constant and \( \varepsilon \) and \( \varepsilon \) are some positive constants. Also, the form of the TAF source is given by the Eq. (4).

2. Method

As we mentioned in [5], the method of lines (MOL) is a general way of viewing a partial differential equation (PDE) as a system of ordinary differential equations (ODE). The partial derivatives with respect to the space variables are discretized to obtain a system of ODE’s in the variable \( t [1,2,7] \). We now apply this method for our problem (1)–(5).

We proceed uniform grid \( \omega_h := \{(x_n, y_n, t_k) : x_n = ih_1, y_n = jh_2, h = (h_1, h_2), t_k = k\tau, 0 < i < N_1, 0 < j < N_2, 0 < k < k_0, k_0 = T/\tau \} \) and write Eqs. (1)–(5) in the form

\[ \tilde{u}_t = (D(x, y)(\tilde{u})_x)_x + (D(x, y)(\tilde{u})_y)_y, \quad (x, y, t) \in \omega_h, \]
\[ \tilde{u}(x, y, 0) = 0, \quad (x, y) \in (0, 1) \times (0, 1), \]
\[ \tilde{u}_0(0, y, t) - \tilde{u}_0(0, y, t) = 0, \quad 0 < y < 1, \ 0 < t \leq T, \]
\[ \tilde{u}(1, y, t) = 1 - \varepsilon \cos(2\pi y), \quad 0 < y < 1, \ 0 < t \leq T, \]
\[ \tilde{u}_0(x, 0, t) = \tilde{u}_0(x, 1, t) = 0, \quad 0 < x < 1, \ 0 < t \leq T. \]

We subdivide \( x - y \) plane into sets of equal rectangles of sides \( \delta x = h_1 \) and \( \delta y = h_2 \), as shown in Fig. 2 below, and let the co-ordinates (\( x, y \)) of the representative mesh point be

\[ x = ih_1, \quad y = jh_2, \]

where \( i \) and \( j \) are positive integers. We denote the value of \( u \) at \((i, j)\) by \( \tilde{u}_{i,j} \) where \( 0 \leq i \leq N_1, \ 0 \leq j \leq N_2 \). For numerical purposes we take \( D(x, y) = D = \text{constant} \), and write Eqs. (6)–(10) by using the standard difference equations as:

\[ \frac{d\tilde{u}_{i,j}}{dt} = D \left( \frac{\tilde{u}_{i+1,j} - 2\tilde{u}_{i,j} + \tilde{u}_{i-1,j}}{h_1^2} + \frac{\tilde{u}_{i,j+1} - 2\tilde{u}_{i,j} + \tilde{u}_{i,j-1}}{h_2^2} \right), \quad 0 \leq i \leq N_1 - 1, \ 0 \leq j \leq N_2, \]
\[ \tilde{u}_{i,j} = 1, \quad 0 \leq i \leq N_1, \ 0 \leq j \leq N_2, \ t = 0, \]
\[ (\tilde{u}_x)_{i,j} = 0, \quad 0 \leq j \leq N_2, \ t > 0, \]
\[ (\tilde{u}_y)_{i,j} = 1 - \varepsilon \cos(2\pi h_2 j), \quad 0 \leq j \leq N_2, \ t > 0, \]
\[ (\tilde{u}_t)_{i,j} = (\tilde{u}_{i,j})_{i,N_2} = 0, \quad 0 \leq i \leq N_1 - 1, \ t > 0. \]
Using the central difference for the boundary conditions (13) and (15) and taking into account (11), we obtain the following system of ordinary differential equations:

\[
\frac{d\tilde{u}_{0,0}}{dt} = \frac{2D}{h_1^2} \tilde{u}_{1,0}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} + \frac{x}{h_1} \right) \tilde{u}_{0,0}^n + \frac{2D}{h_2^2} \tilde{u}_{0,1}^n,
\]

\[
\frac{d\tilde{u}_{0,j}}{dt} = \frac{2D}{h_1^2} \tilde{u}_{1,j}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} + \frac{x}{h_1} \right) \tilde{u}_{0,j}^n + \frac{D}{h_2^2} \tilde{u}_{0,j+1}^n + \frac{D}{h_2^2} \tilde{u}_{0,j-1}^n, \quad 1 \leq j \leq N_2 - 1,
\]

\[
\frac{d\tilde{u}_{0,N_2}}{dt} = \frac{2D}{h_1^2} \tilde{u}_{1,N_2}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} + \frac{x}{h_1} \right) \tilde{u}_{0,N_2}^n + \frac{2D}{h_2^2} \tilde{u}_{0,N_2-1}^n,
\]

\[
\frac{d\tilde{u}_{i,0}}{dt} = \frac{D}{h_1^2} \tilde{u}_{i+1,0}^n + \frac{D}{h_1^2} \tilde{u}_{i-1,0}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{i,0}^n + \frac{2D}{h_2^2} \tilde{u}_{i,1}^n, \quad 1 \leq i \leq N_1 - 1,
\]

\[
\frac{d\tilde{u}_{N_1-1,0}}{dt} = \frac{D}{h_1^2} \tilde{u}_{N_1-2,0}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{N_1-1,0}^n + \frac{2D}{h_2^2} \tilde{u}_{N_1-1,1}^n,
\]

\[
\frac{d\tilde{u}_{N_1-1,j}}{dt} = \frac{D}{h_1^2} \left( 1 - \cos(2\pi h_2 j) \right) + \frac{D}{h_1^2} \tilde{u}_{N_1-2,j}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{N_1-1,j}^n + \frac{D}{h_2^2} \tilde{u}_{N_1-1,j+1}^n + \frac{D}{h_2^2} \tilde{u}_{N_1-1,j-1}^n, \quad 1 \leq j \leq N_2 - 1,
\]

\[
\frac{d\tilde{u}_{i,j}}{dt} = \frac{D}{h_1^2} \tilde{u}_{i+1,j}^n + \frac{D}{h_1^2} \tilde{u}_{i-1,j}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{i,j}^n + \frac{D}{h_2^2} \tilde{u}_{i,j+1}^n + \frac{D}{h_2^2} \tilde{u}_{i,j-1}^n, \quad 1 \leq i < N_1 - 1, \quad 1 \leq j \leq N_2 - 1,
\]

\[
\frac{d\tilde{u}_{i,N_2}}{dt} = \frac{D}{h_1^2} \tilde{u}_{i+1,N_2}^n + \frac{D}{h_1^2} \tilde{u}_{i-1,N_2}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{i,N_2}^n + \frac{2D}{h_2^2} \tilde{u}_{i,N_2-1}^n + \frac{D}{h_2^2} \tilde{u}_{i,j-1}^n, \quad 1 \leq i < N_1 - 1,
\]

\[
\frac{d\tilde{u}_{N_1-1,N_2}}{dt} = \frac{D}{h_1^2} \tilde{u}_{N_1-2,N_2}^n - 2D \left( \frac{1}{h_1^2} + \frac{1}{h_2^2} \right) \tilde{u}_{N_1-1,N_2}^n + \frac{2D}{h_2^2} \tilde{u}_{N_1-1,N_2-1}^n.
\]

Once the system of ODE’s (16)–(24) is solved, the solution to the initial-boundary value problem (1)–(5) is obtained, so that the MOL is performed.

### 3. Numerical example and computer codes

For numerical purposes we take \( D = 0.0036; \, \alpha = 0.05; \, \varepsilon = 1; \, n = 1,2. \) We write the following computer programme in Matlab, and proceed as follows. Firstly, we define the input data in terms of global variables.
Secondly, we code the ODE system (16)–(24) as a m-file called “iparablik2”. Finally, we solve the ODE system using ODE23, an ODE solver that is built up in Matlab and construct the graphics of TAF concentration at time $t = 200, 400, 650, 1100$ for $n = 1$ (linear case) and $t = 10, 30, 60, 100, 500$ for $n = 2$ (nonlinear case). Plots of the TAF concentration for the linear case and nonlinear case are drawn in Figs. 3 and 4, respectively.

```matlab
figure(1) % Input Data
D=0.0036; alpha=0.05; epsilon=1; N=21;M=21;
id=[D alpha epsilon N M];%input data
clc;clear all;

figure(2) % Output Data%
% v=v(x,t) — The Solution of the problem
% Input Data%
% D=0.0036; n=1,2; alpha=0.05; epsilon=1; N=21;M=21;
% Output Data%
% v=v(x,t) — The Solution of the problem
% Input Data%
% D=0.0036; n=1,2; alpha=0.05; epsilon=1; N=21;M=21;
% Output Data%

figure(3) % Input Data
D=0.0036; alpha=0.05; epsilon=1; N=21;M=21;
id=[D alpha epsilon N M];%input data
clc;clear all;

figure(4) % Output Data%
% v=v(x,t) — The Solution of the problem
```

Fig. 3. Solution of the problem (1)-(5) at $t = 200, t = 400, t = 650, t = 1100$ by using the method of lines for $n = 1$ (linear case).
n=input('input n for the problem \( v_t = \nabla(D(x,y)\nabla(v^n)) \)';) if n==1
T=1100;
dt=50;
TT=[200,400,650,1100]; % time for plotting the figure, n=1
ft=TT/dt+1;
else
T=500;
dt=10;
TT=[10,30,60,100,500]; % time for plotting the figure, n=2
ft=TT/dt+1;
end
hx=1/(N-1);hy=1/(M-1);
x(1:N)=((1:N)-1)*hx;y(1:M)=((1:M)-1)*hy;
t=0:dt:T;
v0(1:N*M)=0;v0=v0'; % initial condition
[t,v]=ode23('fparabolik2',t,v0);
if n==1
subplot(2,2,1)
mesh(x,y,reshape(v(ft(1,:),M,N)))
title(['t = ',num2str(TT(1))])
subplot(2,2,2)
mesh(x,y,reshape(v(ft(2,:),M,N)))
title(['t = ',num2str(TT(2))])
subplot(2,2,3)
mesh(x,y,reshape(v(ft(3),:),M,N))
title(\['t= ',num2str(TT(3))\])
subplot(2,2,4)
mesh(x,y,reshape(v(ft(4),:),M,N))
title(\['t= ',num2str(TT(4))\])
else
    subplot(3,2,1)
    mesh(x,y,reshape(v(ft(1),:),M,N))
title(\['t= ',num2str(TT(1))\])
subplot(3,2,2)
    mesh(x,y,reshape(v(ft(2),:),M,N))
title(\['t= ',num2str(TT(2))\])
subplot(3,2,3)
    mesh(x,y,reshape(v(ft(3),:),M,N))
title(\['t= ',num2str(TT(3))\])
subplot(3,2,4)
    mesh(x,y,reshape(v(ft(4),:),M,N))
title(\['t= ',num2str(TT(4))\])
    subplot(3,2,5)
    mesh(x,y,reshape(v(ft(5),:),M,N))
title(\['t= ',num2str(TT(5))\])
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function z=fparabolik2(t,y);
global id hx hy n
% id=[D alpha epsilon N M]; % input data
z(1)=id(1)*(2*y(id(4)+1)ˆn-2*(1+id(2)*hx)*y(1)ˆn)/hxˆ2+id(1)*(2*y(2)ˆn-2*y(1)ˆn)/hyˆ2;% x=0, y=0; (i=1, j=1)
for i=1:id(4)-2
    z(i*id(5)+1)=id(1)*(y((i+1)*id(5)+1)ˆn-2*y(i*id(5)+1)ˆn+...+y((i-1)*id(5)+1)ˆn)/hxˆ2
    % 0<x<1, y=0; (j=1, 1<= i<= N-2)
    z(id(5)*(id(4)-1)+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*(id(4)-1)+i)ˆn+y(id(5)*(id(4)-2)+i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, j=1)
end
z(id(5)*id(4))=id(1)*(2*y(2*id(5))ˆn-2*(1+id(2)*hx)*y(id(5))ˆn)/hxˆ2+...+id(1)*(2*y(id(5)-1)ˆn-2*y(id(5))ˆn)/hyˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
end

% function z=fparabolik2(t,y);
global id hx hy n
% id=[D alpha epsilon N M]; % input data
z(1)=id(1)*(2*y(id(4)+1)ˆn-2*(1+id(2)*hx)*y(1)ˆn)/hxˆ2+id(1)*(2*y(2)ˆn-2*y(1)ˆn)/hyˆ2;% x=0, y=0; (i=1, j=1)
for i=1:id(4)-2
    z(i*id(5)+1)=id(1)*(y((i+1)*id(5)+1)ˆn-2*y(i*id(5)+1)ˆn+...+y((i-1)*id(5)+1)ˆn)/hxˆ2
    % 0<x<1, y=0; (j=1, 1<= i<= N-2)
end
z(id(5)*(id(4)-1)+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*(id(4)-1)+i)ˆn+y(id(5)*(id(4)-2)+i)ˆn)/hxˆ2
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(id(5)*id(4)-1+i)=id(1)*((1-id(3)*cos(2*pi*(i-1)*hy))ˆn-2*y(id(5)*id(4)-1+i)ˆn+y(id(5)*id(4)-2-i)ˆn)/hxˆ2
    % x=1-hx, 0<y<1; (i=N-1, 1<j<M)
end
z(id(5)*id(4))=id(1)*(2*y(id(5)*id(4)-1)ˆn-y(id(5)*id(4)-1)ˆn)/hxˆ2;
for i=2:id(5)-1
    z(j*id(5)+i)=id(1)*(y((j+1)*id(5)+i)^n-2*y(j*id(5)+i)^n+y((j-1)*id(5)+i)^n)/hx^2+
    id(1)*(y(j*id(5)+i+1)^n-2*y(j*id(5)+i)^n+y(j*id(5)+i-1)^n)/hy^2;
    if 0<x<1, 0<y<1, (1<i<N, 1<j<M)
end
end
for i=2:id(4)-1
    z(id(5)*i)=id(1)*(y((i+1)*id(5))^n-2*y(i*id(5))^n+y((i-1)*id(5))^n)/hx^2+
    id(1)*(2*y(i*id(5)-1)^n-2*y(i*id(5))^n)/hy^2;
end
z=z';

4. Conclusions

In this paper we have presented the MOL for the numerical solution of a mathematical model for capillary formation in two space dimensions. As seen from the Chapter 2, this method can be applied to higher dimensions, and it does not require large computer memory, and avoids linearization and physically unrealistic assumptions. We have run our computer program for $n = 2$ (the nonlinear case) for numerical purposes. This porosity constant $n$ can even be taken as a positive rational number in which case it slows down the computation time. One can see from the comparison of Figs. 3 and 4 that in the nonlinear case the TAF in the ECM reaches to the steady state earlier than in the linear case, as expected. In conclusion, as mentioned in [4], MOL provides very accurate and stable numerical solutions for linear or nonlinear PDE’s in comparison with other existing methods.

References