Lab 11
Finite Difference Methods

Marine Modelling April 1, 2019

Katja Fennel
Oceanography
Dalhousie University

Outline

Plan for today:

- Finite Difference Approximation of N and P in a bottle using “Euler Forward” scheme (implicit vs. explicit)
- Finite Difference Approximation of tritium in a pipe using “FTCS” scheme
- Finite Difference Approximation of advection of a perturbation using “Upwind” scheme

N and P in a bottle

Phytoplankton culture, P [mmol N/m^3], in a bottle with nutrient, N [mmol N/m^3], and you know that uptake occurs according to Michaelis-Menten kinetics; you also know the uptake parameters approximately.

$$\frac{dP}{dt} = \mu_{max} \frac{N}{k_N + N} P - rP$$

$$\frac{dN}{dt} = -\mu_{max} \frac{N}{k_N + N} P + rP$$

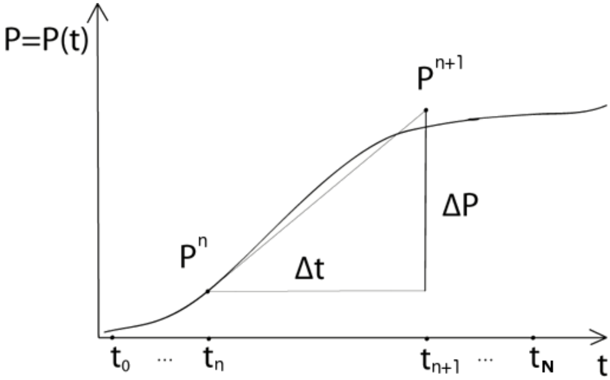
Recipe: replace $\frac{dP}{dt}$ by $\frac{\Delta P}{\Delta t}$ and $\frac{dN}{dt}$ by $\frac{\Delta N}{\Delta t}$

Conventions:

We want to solve for discrete time steps, t_i between t_0 and t_{end}:

$$t_n = t_0 + n \times \Delta t \quad (n = 0, \cdots, N)$$

Refer to N, P at t_i as N_i, P_i.



"Euler forward"

$$
\begin{aligned}
\frac{P^{n+1} - P^n}{\Delta t} &= \mu_{max} \frac{N^n}{k_N + N^n} P^n - rP^n \\
\frac{N^{n+1} - N^n}{\Delta t} &= -\mu_{max} \frac{N^n}{k_N + N^n} P^n + rP^n
\end{aligned}
$$

"Euler forward"

$$\frac{P^{n+1} - P^n}{\Delta t} = \mu_{max} \frac{N^n}{k_N + N^n} P^n - rP^n$$

$$\frac{N^{n+1} - N^n}{\Delta t} = -\mu_{max} \frac{N^n}{k_N + N^n} P^n + rP^n$$

and rearranging yields:

$$P^{n+1} = P^n + \Delta t \left(\mu_{max} \frac{N^n}{k_N + N^n} P^n - rP^n\right)$$

$$N^{n+1} = N^n + \Delta t \left(-\mu_{max} \frac{N^n}{k_N + N^n} P^n + rP^n\right)$$

N and P in a bottle

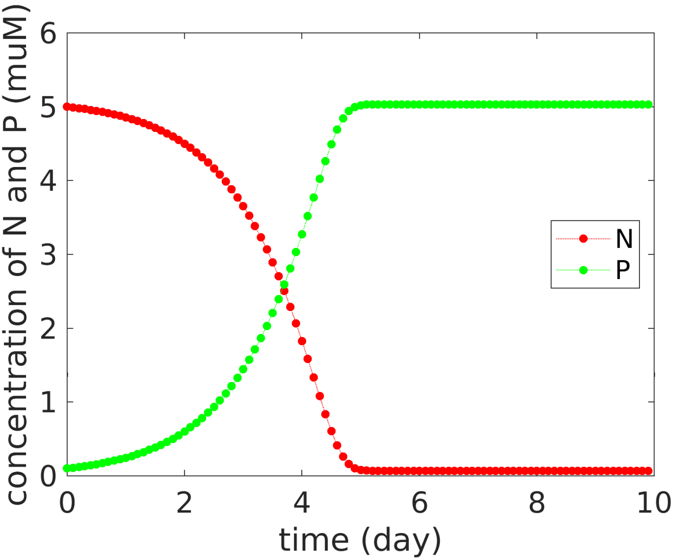
Follow along script `NP_bottle.m`.

```
% 1.) set constants
dt     = 0.1;    % time step (in days)
k_N    = 0.75;   % half sat. const. of N (in muM)
mu_max = 1.2;    % maximum growth rate (in days-1)
r      = 0.1;    % respiration rate (in days-1)
n_max  = 100;    % maximum number of timesteps

% 2.) initialize state variables N and P
% (and time -- only for plotting purposes)
N = zeros(1,n_max);
P = zeros(1,n_max);
t = zeros(1,n_max);
N(1) = 5.0;  % in muM; N at t0
P(1) = 0.1;  % in muM; P at t0
t(1) = 0;    % in days; t0
```

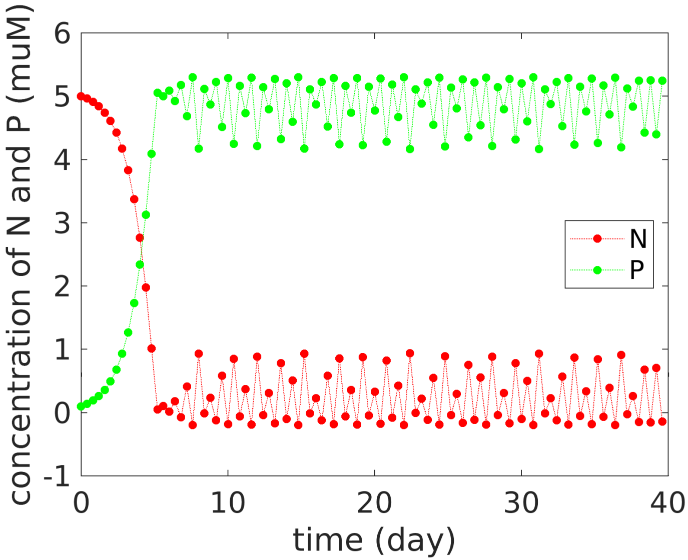

```
% 3.) calculate numerical solution from t1 to t_end
for n = 2:n_max
    t(n) = t(n-1)+dt;
    uptake = mu_max*N(n-1)/(k_N+N(n-1));
    P(n) = P(n-1) + dt*( uptake - r)*P(n-1);
    N(n) = N(n-1) + dt*(-uptake + r)*P(n-1);
end

% 4.) plot solution
hFig = figure(1);
hp = plot(t,N,'r.:',t,P,'g.:');
set(hp,'MarkerSize',16)
set(gca,'FontSize',16)
xlabel('time (day)')
ylabel('concentration of N and P (muM)')
legend('N','P', 'Location', 'best')
```


concentration of N and P (muM)
time (day)
0
1
2
3
4
5
6
0
2
4
6
8
10
N
P

So far we have used a time step of 0.1 d.

Now increase time step to 0.4 d and see what happens.



concentration of N and P (muM)
6
5
4
3
2
1
0
-1
0
10
20
30
40
time (day)
N
P

$$P^{n+1} = P^n + \Delta t \, (\mu_{max} \frac{N^n}{k_N + N^n} P^n - rP^n)$$

$$N^{n+1} = N^n + \Delta t \, (-\mu_{max} \frac{N^n}{k_N + N^n} P^n + rP^n)$$

Note that only concentrations from "previous" time point *n* are used to arrive at "next" time point $(n+1)$.

$C^{n+1} = f(C^n)$ **"explicit scheme"**

Think about the implications in terms of "tangent on the curve" or "control volume".

$$
\begin{aligned}
P^{n+1} &= P^n + \Delta t \, (\mu_{max} \frac{N^n}{k_N + N^n} P^n - rP^n) \\
N^{n+1} &= N^n + \Delta t \, (-\mu_{max} \frac{N^n}{k_N + N^n} P^n + rP^n)
\end{aligned}
$$

Note that only concentrations from "previous" time point n are used to arrive at "next" time point $(n+1)$.

$$C^{n+1} = f(C^n) \quad \textbf{"explicit scheme"}$$

Think about the implications in terms of "tangent on the curve" or "control volume".

Wouldn't it be more "accurate" to allow C to change over the time period Δt?

$$C^{n+1} = f(C^n, C^{n+1}) \quad \textbf{"implicit scheme"}$$

$$C^{n+1} = f(C^n, C^{n+1}) \quad \textbf{"implicit scheme"}$$

May seem tricky, but is tractable.

Leads to a set of simultaneous equations that need to be solved for each time step (not bad in one dimension, but gets expensive for higher spatial dimensions).

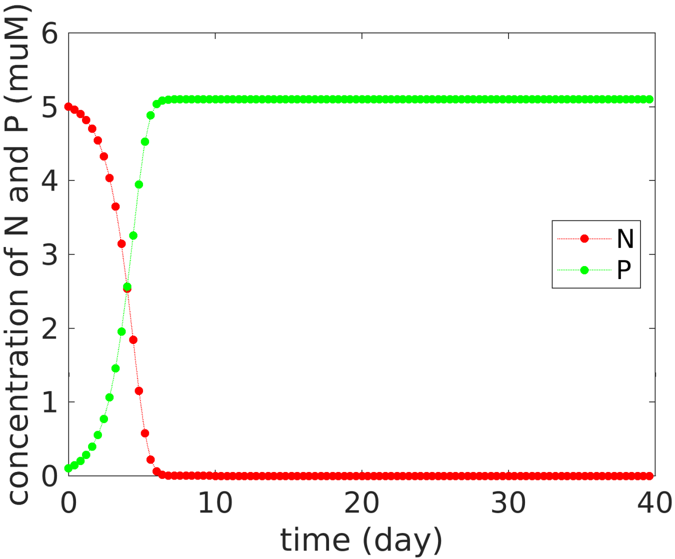
Example for nutrient uptake:

$$\begin{aligned} \frac{dN}{dt} &= -\mu \frac{N}{k+N} P \\ \frac{N^{n+1} - N^n}{\Delta t} &= -\mu \frac{N^{n+1}}{k+N^n} P^n \end{aligned}$$

$$N^{n+1} = \frac{N^n}{1 + \frac{\mu \Delta t P^n}{k + N^n}}$$

A non-negative number is divided by a positive number.
This scheme is positive definite.

See script `NP_bottle_impl.m`.



concentration of N and P (muM)
time (day)
0
1
2
3
4
5
6
0
10
20
30
40
N
P

Key change to make the scheme implicit:

Explicit:

```
for n = 2:n_max
    t(n) = t(n-1)+dt;
    uptake = mu_max*N(n-1)/(k_N+N(n-1));
    P(n) = P(n-1) + dt*uptake*P(n-1);
    N(n) = N(n-1) - dt*uptake*P(n-1);
end
```

Key change to make the scheme implicit:

Explicit:

```
for n = 2:n_max
    t(n) = t(n-1)+dt;
    uptake = mu_max*N(n-1)/(k_N+N(n-1));
    P(n) = P(n-1) + dt*uptake*P(n-1);
    N(n) = N(n-1) - dt*uptake*P(n-1);
end
```

Implicit:

```
for n = 2:n_max
    t(n) = t(n-1)+dt;
    cff = mu_max*dt*P(n-1)/(k_N+N(n-1));
    N(n) = N(n-1)/(1+cff);
    P(n) = P(n-1) + cff*N(n);
end
```

FTCS example: Tritium in a pipe

- Pipe with water flowing down and mixing.
- Water enters at the surface.
- Imagine pipe as streamline in subtropical gyre.

$$
\begin{aligned}
\frac{\partial C}{\partial t} &= -u\frac{\partial C}{\partial t} + K\frac{\partial^2 C}{\partial t^2} - \lambda C \\
u &= 0.01 \text{ ms}^{-1} \\
K &= 1000 \text{ m}^2\text{s}^{-1} \\
\lambda &= \frac{1}{18}\text{yr}^{-1} \\
\Delta x &= 50,000 \text{ m (assume total length 12,000 km)} \\
\Delta t &= 200,000 \text{ s (2.5 days)}
\end{aligned}
$$

In discrete terms:

$$C_i^{n+1} = C_i^n - \frac{u\Delta t}{2\Delta x}(C_{i+1}^n - C_{i-1}^n) + \frac{K\Delta t}{(\Delta x)^2}(C_{i-1}^n + C_{i+1}^n - 2C_i^n) - \Delta t \lambda C_i^n$$

or

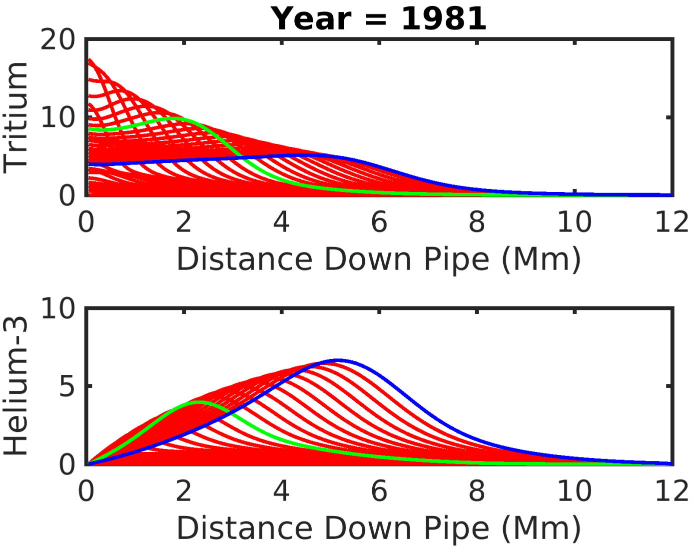
$$C_i^{n+1} = w_- C_{i-1}^n + w_0 C_i^n + w_+ C_{i+1}^n$$

where

$$\begin{aligned} w_- &= \frac{c}{2} + d \\ w_0 &= 1 - \Delta t \lambda - 2d \\ w_+ &= -\frac{c}{2} + d \end{aligned}$$

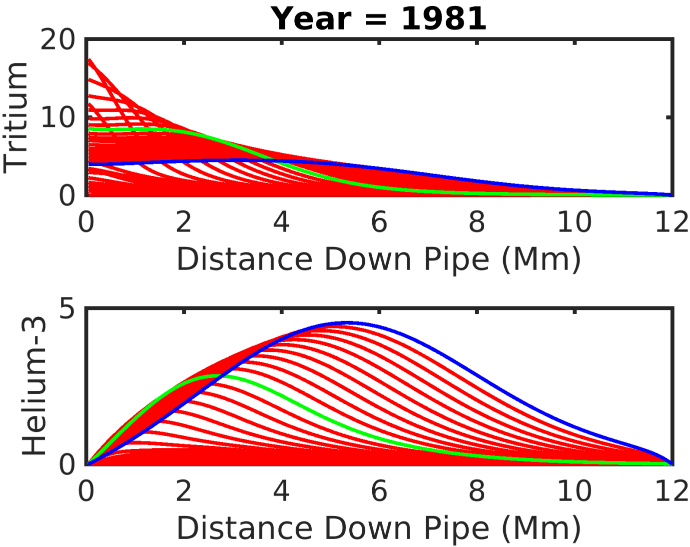
See script `pipe1.m`.

K = 1000 m^{2}s^{-1}

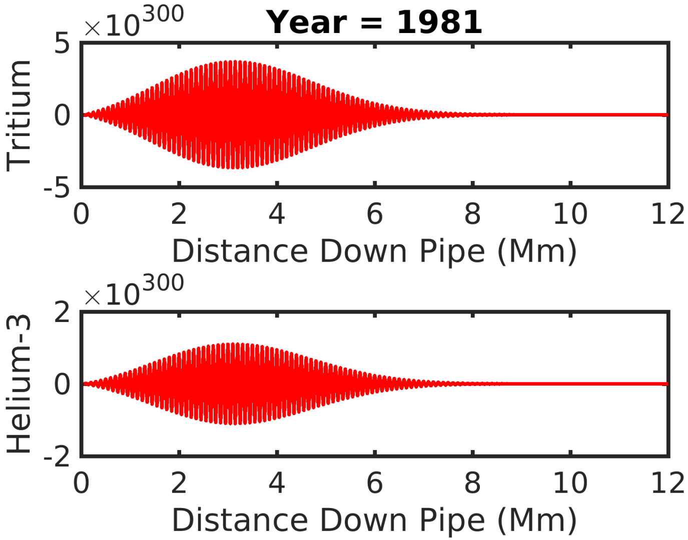


Year = 1981
Tritium
20
10
0
0 2 4 6 8 10 12
Distance Down Pipe (Mm)
Helium-3
10
5
0
0 2 4 6 8 10 12
Distance Down Pipe (Mm)

Adding more diffusion: K = 5000 m^2s^{-1}



Even more: K = 8000 m^2s^{-1} Now unstable!



“Upwind scheme”

Consider a purely advective, one-dimensional system:

$$\frac{\partial C}{\partial t} = -u\frac{\partial C}{\partial x}$$

"Upwind scheme"

Consider a purely advective, one-dimensional system:

$$\frac{\partial C}{\partial t} = -u\frac{\partial C}{\partial x}$$

Using FTCS would result in:

$$C_i^{n+1} = C_i^n - \frac{c}{2}\left(C_{i+1}^n - C_{i-1}^n\right) \quad \text{with: } c = \frac{u\Delta t}{\Delta x}$$

C_{i+1}^n implies that for $u > 0$, downstream information are used to calculate upstream results for next time step.

“Upwind scheme”

Consider a purely advective, one-dimensional system:

$$\frac{\partial C}{\partial t} = -u\frac{\partial C}{\partial x}$$

Using FTCS would result in:

$$C_i^{n+1} = C_i^n - \frac{c}{2}\left(C_{i+1}^n - C_{i-1}^n\right) \quad \text{with: } c = \frac{u\Delta t}{\Delta x}$$

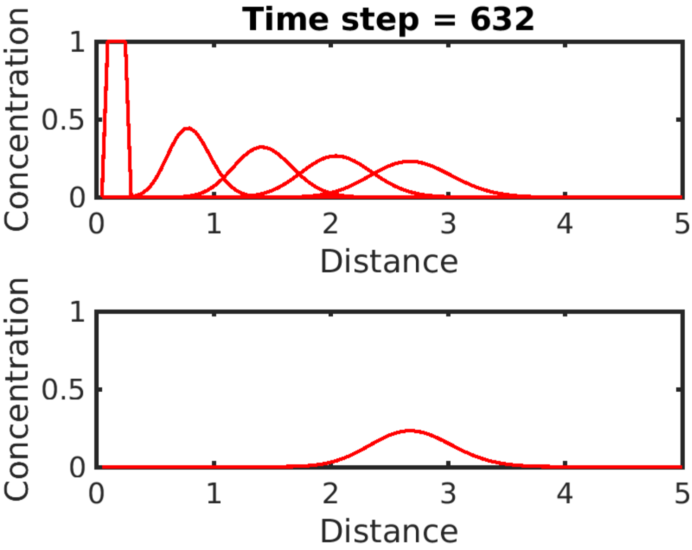
C_{i+1}^n implies that for $u > 0$, downstream information are used to calculate upstream results for next time step.
Therefore, we choose what information are used depending on the direction of the flow (i.e. the sign of u):

$$u > 0\text{: } -c\left(C_i^n - C_{i-1}^n\right), \qquad u < 0\text{: } -c\left(C_{i+1}^n - C_i^n\right)$$

See script `propagate_step.m`.

```
tyr = 24*3600*365.25; % seconds per year
dt = 200000;          % time step in seconds
ny = 5;               % number of years
dx = 50000;           % grid size in x-direction
nx = 500;             % # of cells in x-direction
u = 0.02;             % velocity in x-direction
nt = ny*tyr/dt;       % # of time steps
```

Numerical diffusion



Due to “numerical diffusion” the equation we actually solved looks like:

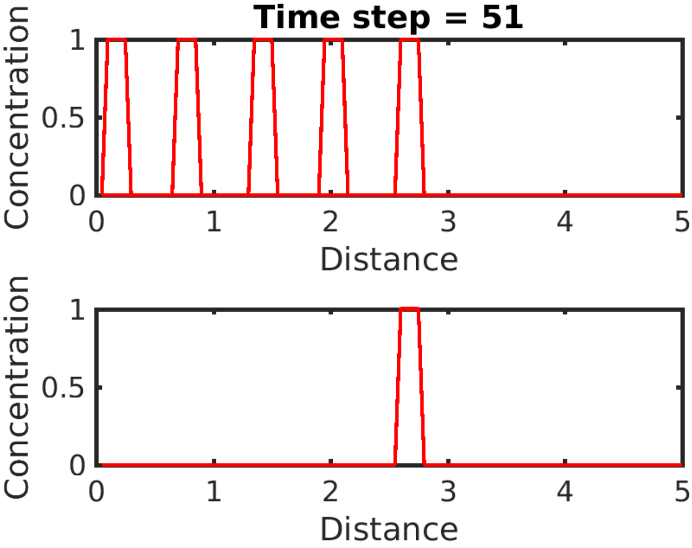
$$\frac{\partial C}{\partial t} = -u\frac{\partial C}{\partial x} + K_n\frac{\partial^2 C}{\partial x^2}, \quad \text{with: } K_n = \frac{u\Delta x}{2}(1-c)$$

If $c = u\frac{\Delta x}{\Delta t} = 1$, the numerical diffusion vanishes.

Back to `propagate_step.m`.

Now try: `dt = dx/u`

Numerical diffusion



Recommendations

- Keep it **simple**.
- **Don't reinvent** the wheel.
- **Know** your tools.
- Carry along **checksums/tests.**
- Test your model with idealized functions or well-known scenarios (**test cases**).
- Compute but recognize the scheme's limitations.