

2.2 3-box model of marine phosphorus cycle

In this section, we build a box model including the interactions between multiple boxes. This is a conceptual model developed by Robbie Toggweiler and Jorge Sarmiento (Sarmiento and Toggweiler, 1984; Toggweiler and Sarmiento 1985). Here, we focus on the phosphorus cycle in the ocean. Figure 2.3 is the schematic diagram of the model. The model consists of low latitude (L), high latitude (H) and deep ocean (D), and their volume are V_L , V_H and V_D (m^3) respectively. The black arrow indicates the ocean's thermohaline circulation (C) including the upwelling of deep water at low latitudes, poleward surface flow and sinking at high latitudes. There is also deep convection at high latitude that exchanges waters vertically (M).

Phosphorus is a necessary nutrient for marine food web. In addition to the ocean circulation, the photosynthesis and formation of sinking organic material can transport phosphorus vertically. As indicated by the green dash line, phosphorus is consumed at the surface, sinks down as organic particles, and returns to water as it decomposes at depths.

The rate of change of phosphorus can be expressed as the contrast between inputs and outputs for each boxes.

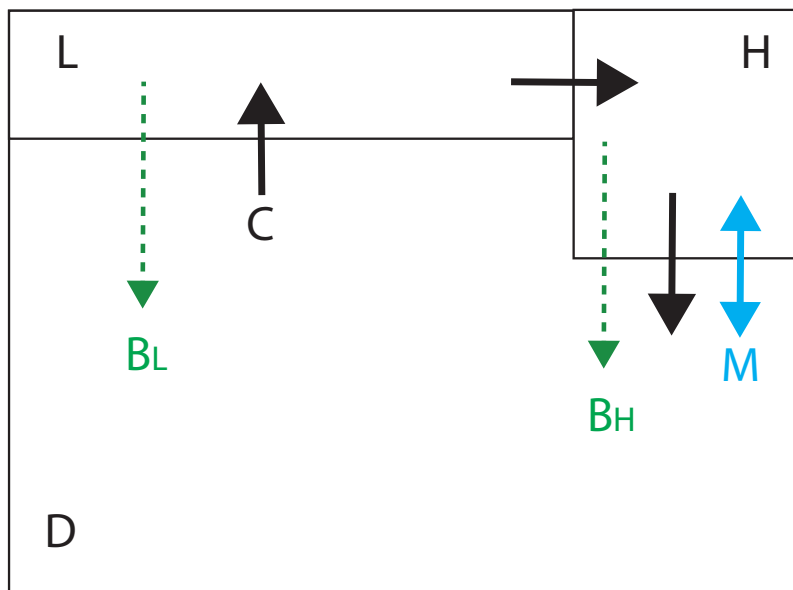


Figure 2.3: A box model for ocean phosphorus cycling.

$$V_L \frac{dP_L}{dt} = C(P_D - P_L) - B_L \quad (2.5)$$

$$V_H \frac{dP_H}{dt} = C(P_L - P_H) + M(P_D - P_H) - B_H \quad (2.6)$$

$$V_D \frac{dP_D}{dt} = (C + M)(P_H - P_D) + B_L + B_H \quad (2.7)$$

where P_L , P_H and P_D are phosphorus concentrations (molPm^{-3}). The circulations (C , M) are in the units of volume flux (m^3s^{-1}).

Numerical solutions to a system of ordinary differential equations

This is a system of coupled three ordinary differential equation. The biological uptake of phosphorus at low and high latitudes, B_L and B_H are in the units of $molPs^{-1}$. Biological uptake is a complex function of light, temperature, nutrients and ecosystem processes. In the simplest possible sense, it can be modeled as a simple function of local phosphorus concentration, $B_i = \lambda V_i P_i$ (where i can be L or H). λ measures the nutrient specific phosphorus uptake rate, and is a measure of biological efficiency.

We can express this system of equations in the vector-matrix form.

$$\mathbf{P} = \begin{pmatrix} P_L \\ P_H \\ P_D \end{pmatrix} \quad (2.8)$$

$$\mathbf{T} = \begin{pmatrix} -(c_L + \lambda) & 0 & c_L \\ c_H & -(c_H + m_H + \lambda) & m_H \\ \frac{\lambda V_L}{V_D} & c_D + m_D + \frac{\lambda V_H}{V_D} & -(c_D + m_D) \end{pmatrix} \quad (2.9)$$

$$\frac{d}{dt} \mathbf{P} = \mathbf{TP} \quad (2.10)$$

where $c_i = \frac{C}{V_i}$ and $m_i = \frac{M}{V_i}$ ($i=L, H$ or D). Similar to the case of the one box model, the system of equation can be integrated forward as follows.

$$\begin{aligned} \mathbf{P}_{n+1} &= \mathbf{P}_n + dt \mathbf{TP}_n \\ &= (\mathbf{I} + dt \mathbf{T}) \mathbf{P}_n \end{aligned} \quad (2.11)$$

This is called Euler forward scheme. Given an initial condition in \mathbf{P}_0 , the equation 2.11 can be used to predict the next time step, \mathbf{P}_1 . This is called time stepping. Given a value for \mathbf{P}_n , you can calculate the next time step \mathbf{P}_{n+1} . Then, using the value for \mathbf{P}_{n+1} , you can get \mathbf{P}_{n+2} . We can repeat the procedure to march forward in time. The limitation of this approach is that it becomes unstable if the time step (dt) is large.

Alternatively, the time stepping scheme can be formulated slightly differently as follows.

$$\begin{aligned} \mathbf{P}_{n+1} &= \mathbf{P}_n + dt\mathbf{T}\mathbf{P}_{n+1} \\ (\mathbf{I} - dt\mathbf{T})\mathbf{P}_{n+1} &= \mathbf{P}_n \\ \mathbf{P}_{n+1} &= (\mathbf{I} - dt\mathbf{T})^{-1}\mathbf{P}_n \end{aligned} \quad (2.12)$$

This is called Euler backward scheme. While this requires additional calculations for taking the inverse of $(\mathbf{I} - dt\mathbf{T})$, it is known that the Euler backward scheme is stable even with a long time step.

Figure 2.4 shows the numerical solution using the Euler forward scheme. This particular simulation is initialized with the uniform P concentration of $2 \cdot 10^{-3} \text{mol P m}^{-3}$. After 10 years of integration (3,600 days with 1 day time step), the model has reached a steady state solution.

Euler forward and backward scheme

- Two first order finite difference schemes for integrating $\frac{d}{dt}\mathbf{P} = \mathbf{T}\mathbf{P}$
- Euler forward: $\mathbf{P}_{n+1} = (\mathbf{I} + \Delta t\mathbf{T})\mathbf{P}_n$
- Euler backward: $\mathbf{P}_{n+1} = (\mathbf{I} - \Delta t\mathbf{T})^{-1}\mathbf{P}_n$
- They are similar in terms of accuracy. However, Euler forward scheme is unstable if a large time step is taken. Euler backward scheme is always stable.

Surface ocean is relatively low in the nutrient level, and the deep ocean is enriched as observed. This is due to the biological uptake of nutrient in the surface layers and the return of nutrients in the deep layer. Table 2.1 shows the set of parameters used for this calculation.

Eigenvalue and stability

This model is stable. How do we know that it is a stable system? One empirical way to check the model's stability is to numerically integrate the system and see whether or not the solution converges. It means that when the ocean phosphorus cycle goes out of balance, it has a stabilizing mechanism to bring it back to the steady state.

However, this empirical approach leaves some uncer-

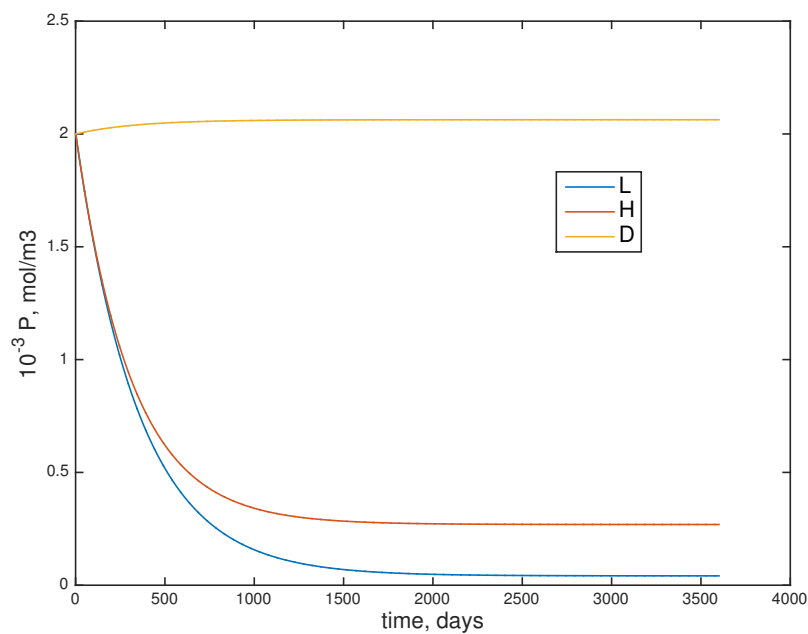


Figure 2.4: A box model for ocean phosphorus cycling.

| Symbol | Name | Value | Units |
|-----------|-----------------|---------------------|--------------|
| V_L | Low lat volume | $3 \cdot 10^{16}$ | m^3 |
| V_H | High lat volume | $1.6 \cdot 10^{16}$ | m^3 |
| V_D | Deep volume | $1.4 \cdot 10^{18}$ | m^3 |
| C | Circulation | $6 \cdot 10^7$ | $m^3 s^{-1}$ |
| M | Convection | $8 \cdot 10^7$ | $m^3 s^{-1}$ |
| λ | P uptake rate | $3.2 \cdot 10^{-8}$ | $m^3 s^{-1}$ |

Table 2.1: Model parameter used in the 3 box model of ocean phosphorus cycling

tainty because it depends on how the numerical solutions are derived. For example, the size of the time steps can influence the stability of the numerical model, but that does not come from the formulation of the original phosphorus cycle model. That means, the stability of the system of equations themselves can be different from that of the recursive relations derived from the numerical implementation of the model.

Consider a simple example, $\frac{dx}{dt} = kx$. It's analytic solution is $x(t) = x_0 e^{kx}$. In this example, the value of k determines the stability of $x(t)$. If $k > 0$, the solution for $x(t)$ grows exponentially, and it never reaches a steady state. If $k < 0$, it decays exponentially towards zero which is the steady state solution. If $k = 0$, the system is neutrally stable; $x = (\text{constant})$.

What controls the stability of the original model equation, $\frac{d}{dt}\mathbf{P} = \mathbf{TP}$? Eigenvalues of \mathbf{T} provides similar information to k in the previous paragraph,

$$\mathbf{T}\mathbf{q} = \lambda\mathbf{q}. \quad (2.13)$$

where λ and \mathbf{q} are the eigenvalue and corresponding eigenvector of \mathbf{T} . Since it is a 3×3 matrix, we get 3 sets of eigenvalues and eigenvectors. In MATLAB, you can calculate the eigenvalue of the matrix as follows.

```
>> [Q,L]=eig(T);
```

Imagine a solution with a form of $\mathbf{P} \sim e^{\lambda t} \mathbf{q}$, and substitute it into $\frac{d}{dt} \mathbf{P} = \mathbf{T} \mathbf{P}$. Using the definition of eigenvalue-eigenvector (2.13), we see that it satisfies the differential equation. So we can write the solution for \mathbf{P} ;

$$\mathbf{P}(t) = c_1 e^{\lambda_1 t} \mathbf{q}_1 + c_2 e^{\lambda_2 t} \mathbf{q}_2 + c_3 e^{\lambda_3 t} \mathbf{q}_3. \quad (2.14)$$

If one of the eigenvalue takes a positive value, the system is unstable. Small change will exponentially grow into a larger and larger values. The condition for stability for a system of ODE is that all of eigenvalues are negative or equal to zero.

Analytic solution: matrix exponential

The eigenvalue-eigenvector analysis provides additional benefit. It leads to the analytic solution. The constant coefficients, $c_i (i = 1, 2, 3)$ needs to be determined from the initial condition. At $t = 0$,

$$\mathbf{P}(0) = c_1 \mathbf{q}_1 + c_2 \mathbf{q}_2 + c_3 \mathbf{q}_3 = \mathbf{Q} \mathbf{c} \quad (2.15)$$

Then we get $\mathbf{c} = \mathbf{Q}^{-1} \mathbf{P}(0)$. The whole solution can then be written as follows;

$$\mathbf{P}(t) = \mathbf{Q} e^{\mathbf{A}t} \mathbf{c} \quad (2.16)$$

$$= \mathbf{Q} e^{\mathbf{A}t} \mathbf{Q}^{-1} \mathbf{P}(0) \quad (2.17)$$

$$= e^{\mathbf{T}t} \mathbf{P}(0) \quad (2.18)$$

Here the diagonal matrix $\mathbf{\Lambda}$ contains eigenvalues (λ) in its main diagonal, and $e^{\mathbf{\Lambda}t}$ is a diagonal matrix with the elements set to the exponential of eigenvalues times t . Each column vector of \mathbf{Q} contains the corresponding eigenvectors (\mathbf{q}). The last two steps defines the solution in the form of matrix exponential.

$$\mathbf{P}(t) = e^{\mathbf{T}t}\mathbf{P}(0) \quad (2.19)$$

where the matrix exponential $e^{\mathbf{T}t}$ is defined as

$$e^{\mathbf{T}t} = \mathbf{Q}e^{\mathbf{\Lambda}t}\mathbf{Q}^{-1} \quad (2.20)$$

In MATLAB, you can calculate the matrix exponential as follows.

```
>> P(:,n) = expm(T*time(n))*P0;
```

Steady state solution

In this particular problem, the examination of the three eigenvalues shows that there are two negative and one zero eigenvalue. Remember that the solution can be written following Eq (2.21). Let us call the two decaying eigenvectors \mathbf{q}_2 and \mathbf{q}_3 , and they will vanish with time. So, the steady state solution will be dictated by the first eigenvector whose corresponding eigenvalue is zero.

$$\bar{\mathbf{P}} = c_1\mathbf{q}_1 \quad (2.21)$$

For this type of problem, it is also common to simply seek for steady state solution ($\frac{d}{dt} \rightarrow 0$) without considering the time stepping. Setting the (RHS) of Eqs 2.5 - 2.7 to zero leads to three algebraic equations with three unknowns (P_L, P_H, P_D). However, this system of algebraic equation is not solvable because the matrix \mathbf{T} is not full rank. Pick two of the three equations, and the third equation can be expressed as a linear combination of the first two.

The sum of Eqs 2.5 - 2.7 results in the cancellation of all transport and biological terms.

$$\frac{d}{dt} \Sigma_i \{V_i P_i\} = 0 \quad (2.22)$$

This implies that the constant of integration is the total phosphorus inventory. This is the statement of the conservation of phosphorus in the ocean. Ocean circulation and biological processes do not generate nor destroy phosphorus in the system. It is only re-distributing the phosphorus without changing its inventory.

$$\Sigma_i \{V_i P_i\} = \text{constant} \quad (2.23)$$

This conservation statement provides the third equation that allows us to determine the steady solution. For example, we can replace the deep ocean equation with the integral constant.

$$0 = c_L(P_D - P_L) - \lambda P_L \quad (2.24)$$

$$0 = c_H(P_L - P_H) + m_H(P_D - P_H) - \lambda P_H \quad (2.25)$$

$$\bar{P} = v_L P_L + v_H P_H + v_D P_D \quad (2.26)$$

where v_i is the fractional volume ($\frac{V_i}{V}$ where V is the total volume), and the \bar{P} is the prescribed global mean phosphorus concentration, which must be supplied to determine the steady solution. In this case you will get a full rank matrix, which can be inverted to find the steady solution.

In the vector-matrix form,

$$\mathbf{s} = \begin{pmatrix} 0 \\ 0 \\ \bar{P} \end{pmatrix} \quad (2.27)$$

$$\mathbf{U} = \begin{pmatrix} -(c_L + \lambda) & 0 & c_L \\ c_H & -(c_H + m_H + \lambda) & m_H \\ v_L & v_H & v_D \end{pmatrix} \quad (2.28)$$

$$\mathbf{s} = \mathbf{U}\mathbf{P} \quad (2.29)$$

$$\mathbf{P} = \mathbf{U}^{-1}\mathbf{s}. \quad (2.30)$$

This way, we can also calculate the steady state solution without taking a time step.

References:

Sarmiento, Jorge L., and J R Toggweiler, 1984: New model for the role of the oceans in determining atmospheric PCO₂. *Nature*, 308(5960), 621-624. Toggweiler,

J R., and Jorge L Sarmiento, 1985: Glacial to interglacial changes in atmospheric carbon dioxide: The critical role of ocean surface water in high latitudes In *The Carbon Cycle and Atmospheric CO₂: Natural Variations Archean to Present*, Geophysical Monograph 32, Washington, DC, American Geophysical Union, 163-184.

Exercises:

1. Carbon-14 (^{14}C) is subject to radioactive decay. The concentration of ^{14}C is known to decay following the equation, $\frac{d[^{14}\text{C}]}{dt} = -\lambda[^{14}\text{C}]$. What is the analytic solution for $[^{14}\text{C}](t)$ with an initial condition of $[^{14}\text{C}](t) = C_0$ at $t = 0$?
2. The time period over which the half of $[^{14}\text{C}]$ decays is known as the half-life, and it is approximately 5,730 years. What is the value for k ? (include units in your answer)
3. We want to solve the above problem numerically using the Euler forward time stepping method. What is the mathematical relationship between $[^{14}\text{C}]$ of two consecutive time steps?
4. Now we want to solve the above problem numerically using the Euler backward time stepping method. What is the mathematical relationship between $[^{14}\text{C}]$ of two consecutive time steps?

5. Develop a MATLAB script to simulate the decay of $[^{14}\text{C}]$ over 20,000 years. Make sure that your numerical integration accurately reproduces the analytical solution.
6. (extra credit +10%) Predator-Prey equation. Consider rabbit (X) and fox (Y) populations. Their evolution can be written as;

$$\begin{aligned}\frac{dX}{dt} &= aX - bXY \\ \frac{dY}{dt} &= bXY - cY\end{aligned}$$

where a , b and c are arbitrary constants (they can be all set to 1, for example). What is the steady states of this system?

7. (extra credit +40%) Develop a MATLAB script to simulate the time evolution and X and Y using the Euler backward time stepping method. This is a non-linear problem. You cannot use the matrix approach.