Homework set 2

Christian Huber (christian.huber@eas.gatech.edu), Carlos Cardelino (carlos.cardelino@eas.gatech.edu)

July 15, 2011

1 Problem 1 - The Carbon cycle

(a) An interesting example of a biogeochemical cycle is the exchange of carbon dioxide between the ocean surface water and the atmosphere, and between the atmosphere and the crust. Figure 1 shows a diagram for a simplified Carbon cycle box model, with 4 boxes. The two ocean boxes represent all forms of dissolved carbon CO_2, H_2CO_3, HCO_3^- and CO_2^{--}. The other two boxes represent all forms of carbon in the terrestrial system and atmosphere. The carbon inventory and fluxes in figure 1 are in units of 10^{15} grams. Write a Matlab script **CarbonC.m** to follow the evolution of the carbon content in each of the four reservoirs for 100 years. Calculate the lifetimw of carbon for each reservoir at time t=0. Obtain the approximate analytical expressions for the time evolution of each reservoir.

NOTE: The input files should be in the form of a matrix $\mathbf{F}$, with all positive off-diagonal fluxes, and a vector $\mathbf{M_0}$ containing the initial condition for each reservoir. **CarbonC.m** should build a matrix $\mathbf{K}$ based on the inputs $\mathbf{F}$ and $\mathbf{M_0}$ (as indicated in class).

(b) We can observe the non-linear response of the carbon cycle to perturbations. Let us first assume that the terrestrial system starts to increase its output to the atmosphere (e.g. increase in fossil fuel combustion) from 103×10^{15} to 120×10^{15} g/yr. Repeat the calculations performed in (a) for 100 years. In a second time assume another perturbation: introduce a flux from the terrestrial to the ocean-surface layer reservoir of 60×10^{15} g/yr, all other fluxes being identical to (a). Follow the evolution of each reservoir for 100 yrs.

Turn in a copy of **CarbonC.m** and 12 plots (4 from (a), i.e. 1 per reservoir, and 8 from (b)). Use the command *subplot* to include the plots for the 4 reservoirs in a single page (3 pages total). Include - **ONLY FOR (a)** - the four lifetimes and the approximate analytical expressions for the time evolution of each reservoir.

2 Problem 2 - Ca isotopes in sea sediments, erosion and weathering rates

Stable (i.e. non-radioactive) isotopes, like ^{44}Ca and ^{40}Ca behave identically chemically, but their different masses induces a possible fractionation between the two isotopes (as seen briefly in class). The first major process involving isotopic fractionations is equilibrium fractionation, which is a quantum mechanical effect associated with the mass-dependent bonding of atoms in a crystalline structure (like in the calcite forming carbonate sediments). In this case, a common rule is that, under equilibrium conditions between sea water and sediments (calcite), the heavier isotope (^{44}Ca) is preferentially

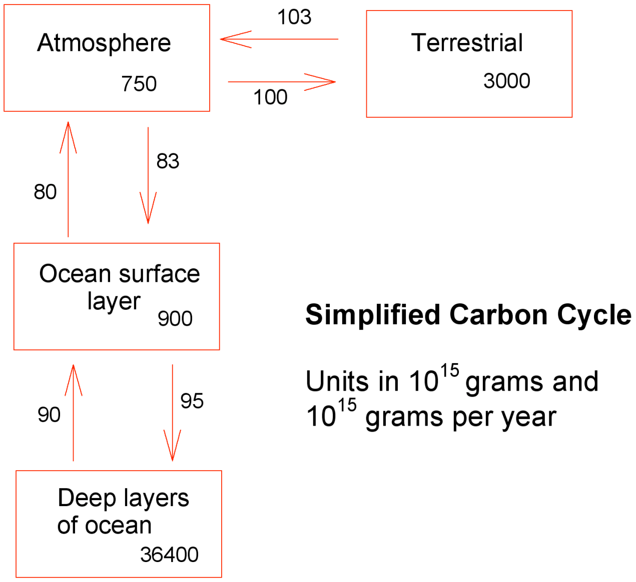

Figure 1 Four-box model of the carbon cycle

Figure 1: Carbon box (4) model.

incorporated into the solid phase, yielding a heavy isotopic composition (larger fraction of ${}^{44}Ca$) in the solid and a lighter composition in the fluid (relatively speaking more ${}^{40}Ca$). The precipitation of calcite from seawater is an exception, the calcite incorporates more readily the light isotope and leaves the seawater depleted and heavier isotopically. We introduce the delta notation (common in geochemistry)

$$\delta Ca = \left(\frac{R_s - R_{ref}}{R_{ref}}\right) 1000, \tag{1}$$

where R is the ratio (proportion) of heavy (44) to light (40) Ca, the subscripts s and ref refer respectively to a sample (e.g. seawater, sediments...) and a reference standard. In that notation sediments with light Ca composition have a negative δCa. We will assume that the difference in isotopic composition between the seawater and sediments, defined as Δ_{sed}, is fixed with a value of -1.3 (generally consistent with biogenic calcite precipitation)

$$\Delta_{sed} = \delta_{sed} - \delta_{sea} = -1.3, \tag{2}$$

where δ_{sed} and δ_{sea} are the isotopic compositions of the sediments and the seawater respectively.

(a) The calcium ocean cycle can be idealized with a single box with incoming fluxes from weathering and sub-oceanic volcanic sources. We will merge these two contributions into a general weathering flux F_w. Similarly we can construct a single sink (flux out) associated with the precipitation of calcite from the seawater F_p. Write a set of two coupled differential equations to describe the evolution of δ_{sea} and δ_{sed} assuming that the weathering flux F_w carries Ca with a fixed (not time dependent) isotopic composition δ_w. Write down a third equation to describe the evolution the total amount of Ca dissolved in the seawater N_{Ca}.

(b) Assume that $N_{Ca}(t_0) = 1.4 \times 10^{19}$ grams, that $\delta_{sea}(t_0) = \delta_{sed}(t_0) = 0$ and that $\delta_w = -0.44$. Is the seawater-sediment system at equilibrium isotopically at $t = t_0$? Write two function files **Flux_weathering.m** and **Flux_p.m** that accept a single input variable (time) and send back the respective flux at that time. Assume that $F_p = N_{Ca}(t_0)/10$ and $F_w = 0.7 \times F_p$. Write a Matlab script **runCa.m** that solves for the coupled set of ODE written in (a) using a Forward-Euler marching scheme. The script should ask you for the run time ($t_{end} - t_0$ in millions of years) and number of timesteps (N). Using the equations you wrote under (a), estimate the time it takes for the system to reach a steady-state, compare it to a calculation (runtime 100 Ma with 100 timesteps). How do they compare ? Make a single plot for both time series $\delta_{sed}(t)$ and $\delta_{sea}(t)$ use labels and legends.

(c) Repeat the same calculations, this time with $F_p(t) = N_{Ca}(t)/10$ and $F_w(t) = N_{Ca}(t_0)/10$. Make a single plot with both time-series. If you find a steady-state, how different are δ_{sea} and δ_{sed} at steady-state ? How long does it take to reach the steady-state this time ? Explain the difference if any.

Turn in a copy of **Flux_weathering.m** and **Flux_p.m** as well as the script to run the coupled ODE and plot the results. Include the system of equations and the analytical estimates for steady state (question (b)) with the calculated time to reach steady-state for the two cases (b-c).

3 Problem 3 - Marching equation, the road to Chaos

This problem is often referred to as the logistic map. A very simple non-linear marching equation can yield some very interesting features. One of which is period doubling, i.e. the solution oscillates between a number of exact solutions that increases (in general doubles at certain critical values) as a free parameter of the function is increased. The equation we will solve for is

$$\frac{dx}{dt} = Rx(1 - x), \tag{3}$$

where R is the phase parameter that controls the period doubling of the solution (see figure 2). This phenomena is a famous example of a chaotique problem.

(a) Write a Matlab script **logistic1.m** that uses a Forward-Euler time marching scheme to solve the logistic equation above for any choice of R. We will use $x_0 = 0.5$ as an initial condition and run the code for 10'000 iterations. Look up the Matlab functions **unique**, **roundn** and **size** on the on-line help (type help xxx in the command window). You will use both of these functions to sort the values of the array x. Once all the iterations are done and the results of the marching equation, the array x, stored, (1) plot x vs iterations (over the last 1000 iterations only), (2) plot a histogram (see command **histo**) of the same last 100 iterations using 1000 bins - use the subplot command to use only one page for both plots, (3) discuss what the command *size(unique(roundn(x(9000:10000),-3)))* does ?

(b) Use the script and the new command to write a second script **logistic2.m** that performs a loop over different values of R from $R = 2.9$ to 3.6 with 1000 equally-spaced values of R (hint: use the command *linspace*). For each R values, solve the logistic equation identically as in (a) and plot the number of different solutions you find as function of R. If we define the distance between each bifurcation as

$$d_i = R_i - R_{i-1}, \tag{4}$$

where the subscript i is related to the number of solutions of the logistic equation which is equal 2^i, and R_i is defined to be the value of R at which new bifurcations are found (new doubling of the number of solutions). Measure approximately (from the data you generated) d_8, d_4 and d_2. Compute the two ratios d_4/d_8 and d_2/d_4, what value did you find ? This number is often called the Feigenbaum constant.

Turn in all the plots, values and discussions requested for Problem 3.

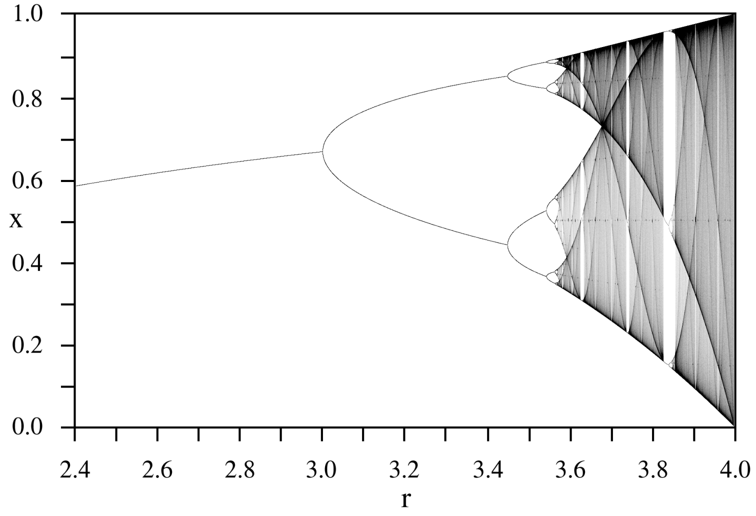

Figure 2: Bifurcation map for the logistic equation.