GOING VERTICAL: THE INFLUENCE OF IRRADIANCE, MIXING, SINKING,

AND BENTHIC FLUX ON SPECIES SUPERSATURATED PHYTOPLANKTON

ASSEMBLAGES IN A ONE-DIMENSIONAL WATER COLUMN FRAMEWORK

A Thesis

By

FRANCES GRACE WITHROW

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,   Daniel L. Roelke
Committee Members,   William Grant
                       Antonietta Quigg
Head of Department,   Michael Masser

May 2017

Major Subject: Wildlife and Fisheries Sciences

ABSTRACT

Theoretical research discusses neutral, lumpy, and intransitive dynamics as mechanisms influencing species richness. Our research here focused on the effects on richness and diversity on both the   and   levels of disturbances (diminishing light availability throughout the water column, nutrient flux from the benthos, particle sinking, vertical water mixing, and eddy diffusion) in these supersaturated systems. Our simulations incorporated 30 supersaturated phytoplankton assemblages distributed into a one-dimensional, ten-layer vertical water column. We found that overall, there was little difference between   and   level sensitivity for all assemblage types and that assemblages following intransitive dynamics had the most richness and diversity sensitivity to all disturbance types on both the   and   level. This aligns with previous research discussing neutral, lumpy, and intransitive dynamics and their robustness under disturbances. This research opens up a path for future work that explores in more detail the effects of various oceanic disturbances on phytoplankton richness and diversity.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professors Daniel Roelke and William Grant of the Department of Wildlife and Fisheries Sciences and Professor Antonietta Quigg of the Department of Marine Biology.

All work for the thesis was completed by the student, in collaboration with Professor Daniel Roelke of the Department of Wildlife and Fisheries Sciences.

Funding Sources

TABLE OF CONTENTS

LIST OF FIGURES

Page

1. INTRODUCTION

Ecologists have long been fascinated with species rich ecosystems with few

limiting resources, an interest famously epitomized by the "Paradox of the Plankton"

(Hutchinson, 1961). Based on theory, species competition for resources will lead to a

number of co-existing species not greater than the number of limiting resources if the

system arrives at a steady state. Therefore, a solution to the Paradox of Plankton is

prevention of competitive exclusion by a fluctuating environment (Tilman, 1977, 1981,

1982; Sommer, 1984, 1985; Grover, 1989). Fluctuations in abiotic conditions caused by

external disturbances, in turn, prevent the completion of competitive exclusion

processes. This often results in periodic cycling of populations (Sommer, 1984, 1985).

But it can also result in protracted periods of transient population dynamics (Hastings,

2010) or chaotic population dynamics (May and Leonard, 1975). Periodic, transient and

chaotic population dynamics all facilitate species-richness.

There are, however, other mechanisms leading to high species richness that do

not involve external disturbances. A condition coined 'supersaturated coexistence'

(sensu Schippers et al., 2001) refers to species rich assemblages that maintain a number

of species higher than the number of limiting resources. Mechanisms or conditions

leading to species supersaturation include neutral, lumpy, and intransitive ('rock-paper-

scissors') coexistence.

Neutral coexistence refers to the co-occurrence of species that are similar in their

competitive abilities (Hubbell, 2001). Each species in question competes for the same or

similar resources in the same or a similar way. Examples in nature include various wasp

species in Panama (Saez and Lozano, 2005), fungi (Bickford et al., 2007), and trees from the Barro Colorado Island (McGill, 2003). The theory applies to competitive coexistence within a given trophic level. Neutral refers to the treatment of organisms as essentially the same since their competitive ability is significantly similar. This allows multiple species to occupy the same niche, therefore allowing higher biodiversity in a given system. This could be a possible mechanism underlying the coexistence of many phytoplankton species when there are few limiting resources.

Lumpy coexistence describes scenarios where there are co-occurring and competing species clusters. Within a cluster, coexisting species compete similarly (like that of neutral theory), but clusters are sufficiently different from each other in competitive ability to promote coexistence of multiple clusters (Scheffer and van Nes, 2006). Lumpy coexistence reconciles the opposing conceptual models of niche theory and neutral theory. This system is applicable to systems that follow Lotka-Volterra population dynamics and can account for the possible decrease in species richness in the neutral theory due to system instability. For example, it is suspected in systems such as modern quaternary plants in Mediterranean-type ecosystems as an evolutionary advantage over the more ancient Tertiary species (Valiente-Banuet et al., 2006) and in species with size distributions such as European aquatic beetles (Drost et al., 1992) and prairie birds (Holling, C.S., 1992). Furthermore, lumpy coexistence was shown in a Monod model (Roelke and Eldridge, 2008) which we used in this research. For highly diverse phytoplankton systems, this may explain groupings of species with similar

competitive advantages/disadvantages coexisting with significantly different groups of species.

'Rock-paper-scissors' (RPS) coexistence results when competition is non-hierarchical. In other words, no single species can displace all other species present. This results in oscillations in the population dynamics where the sequence of population maxima repeats (Huisman et al., 1999; Huisman and Weissing, 2001; Roelke and Eldridge, 2010). Populations of coral reef invertebrates (Jackson and Buss, 1975) and side-blotched lizards in California (Sinervo and Lively, 1996) are a few examples of biological systems that experience high richness based on various competitive advantages. The name 'rock-paper-scissors' originates from the game children play. For any given three species (A, B, and C for example) that follow 'rock-paper-scissors' dynamics, B will outcompete A, C will outcompete B, and A will outcompete C. When all three species are present, no species can efficiently drive any other species to extinction and a population succession results in the sequence A-B-C-A-B-C-A … Hence, all species coexist. This could be yet another possible mechanism allowing phytoplankton species to constantly outcompete one another, but still have viable population sizes to allow coexistence.

The effect of migration on these biodiversity mechanisms was studied by Roelke and Eldridge (2008). They explored a plankton system where the influence of migration was a function of hydraulic mixing. In particular, they looked at two patch systems where water exchange between adjacent patches occurred in a direction perpendicular to flow through the system (lentic or "lake-like"), and where water exchange between

adjacent patches occurred in a direction parallel to flow through the system (lotic or "river like"). They found that immigration generally led to homogenization of species and extinction events, even at very low levels. The populations that were characterized by lumpy coexistence were more resistant to homogenization from migration than those of neutral coexistence and 'rock-paper-scissors'. But the overall sensitivity of these assemblage states to immigration questions the prevalence of these mechanisms as biodiversity sustaining processes in natural environments. In Roelke and Eldridge (2008), the modeled hydrology was intentionally kept simple, thereby enabling analysis of biological processes.

As already mentioned, fluctuations in conditions caused by external disturbances can contribute to species richness and diversity. This work focused on five disturbances commonly found in aquatic environments. First, a prime abiotic condition that contributes to the heterogeneity in phytoplankton distribution in a water column is the light gradient. Models have been developed to investigate the effects of light gradients on competition and composition (Huisman and Weissing, 1994, 1995; Weissing and Huisman, 1994) as well as tests on these models in monocultures and combinations of those monocultures (Huisman et al., 1999) typically finding that the species that competes best for light typically outcompetes all other species in the long run. This does not explain how super saturated systems can exist in light gradients and, as would be expected, it has been found that competition is complex and a species' ability to compete for light can also dependent on the intensity of the light itself (Han et al., 1999) leaving more room for exploration of assemblage competition for light.

As with light gradients, nutrient gradients are also important in producing vertical heterogeneity (Cullen, 1982). Although this, at times, is related to mixing or lack thereof, phytoplankton themselves also contribute to the formation of a nutrient gradient as they consume resources throughout the water column (Klausmeier and Litchman, 2001). There is evidence that some nutrients distributed in the water column originate from the benthos (Rowe et al. 1975; Nowicki and Nixon, 1985). Previous works explored vertical distribution of phytoplankton as a function of both light and benthic nutrient flux (Mellard et al., 2011; Klausmeier and Litchman, 2001; Huisman et al., 2006; Beckman and Hense, 2007), but none on nutrient flux alone.

The only non abiotic disturbance we investigated was the effect of cell density. It is known that phytoplankton cells tend to have a different density than their aquatic environment causing them to sink or rise (Smayda, 1970; Reynolds, 1977; Huisman and Sommeijer, 2002). This movement is independent of mixing and changes the vertical distribution of cells and thus the richness and diversity at a given point in the column. Research exists investigating the effects of sinking on diversity in conjunction with other disturbances such as light and turbulence (e.g. Huisman et al., 2002; Hsu and Lou, 2010; Peng and Zhao 2016), but rarely of the effects of sinking alone on richness and diversity.

A more mechanical example of disturbance is variations in both vertical and horizontal mixing. Whether it is in the form of vertical mixing and pulsed inflows (Richerson et al., 1970; Roelke et al., 1999; Perruche et al., 2010), seasonal changes in mixing depth (Sommer et al., 1986; Reynolds, 1993; Scavia et al., 2002) or horizontal mixing (Richerson et al., 1970; Codeco and Grover, 2001; Petrovskii and Malchow,

5

2001; Mckiver et al., 2009; Adjou et al., 2012) it is evident that mixing rates contribute

to richness and diversity (Prowe et al., 2014; Adjou et al., 2012).

2. PURPOSE AND JUSTIFICATION

Using only a two-patch modeling framework, the likelihood of source and sink patches occurring is very small. Migration of populations from source locations into sink locations is another mechanism enabling high species richness (Mouquet and Loreau, 2003). We explored this here, as we expanded the previous work of Roelke and Eldridge (2008) with the aim to investigate the role of neutral and lumpy coexistence, and 'rock-paper-scissors' dynamics in a framework more representative of natural conditions. The two-patch modeling framework was modified into a water column framework that is a one-dimensional model represented by vertically stacked patches. This model incorporates the same assemblages used previously (Roelke and Eldridge, 2008), each characterized by supersaturated coexistence. Furthermore, we introduced parameters that govern diminishing light availability throughout the water column, nutrient flux from the bottom layer (benthos), particle sinking, vertical water mixing, and eddy diffusion (to represent mixing of waters with an adjacent open ocean). Then, similar to previous work, the effects of these interactions on species richness and diversity were investigated. The fragility of these various assemblage types to environmental disturbances is largely unknown. This particular project provides an important primary assessment to supersaturated behavior types.

3. METHODS

The model consisted of ten layers stacked vertically to represent a water column. Within each layer, species competed for three growth-limiting resources, where the relationship between resource concentration and growth rate was based on an established model (León and Tumpson, 1975; Tilman, 1982).

These original models were constructed using time dependent ODEs (ordinary differential equations). As mentioned above, Roelke and Eldridge (2008) added a spatial dimension to the model to include interactions in a two-patch system. They explored both a lentic and lotic configuration of the two-patch framework, with both frameworks having the patches horizontally configured. Here, the modification of the Roelke and Eldridge (2008) model involved casting the model in a vertical configuration which enabled greater realism relating to water column processes.

A conceptual diagram of the model structure is shown in Figure 1. Light enters the system from the surface and diminishes based on, in part, existing populations in upper layers blocking light to lower layers. Possible interactions between layers include vertical mixing and sinking of cells. There is advection from nearshore flows, Eddy diffusion from an adjacent open ocean source, and nutrient flux from the benthos. Figure 1 shows the system where the phytoplankton assemblage is composed of five species. However, this changes with each simulation. There are a total of 30 assemblages that are known to emerge as supersaturated systems from Roelke and Eldridge (2008). These same 30 assemblages were used in these simulations. Figure 2 shows the typical expected dynamics of a neutral (Figure 2a.), lumpy (Figure 2b.), and 'rock-paper-
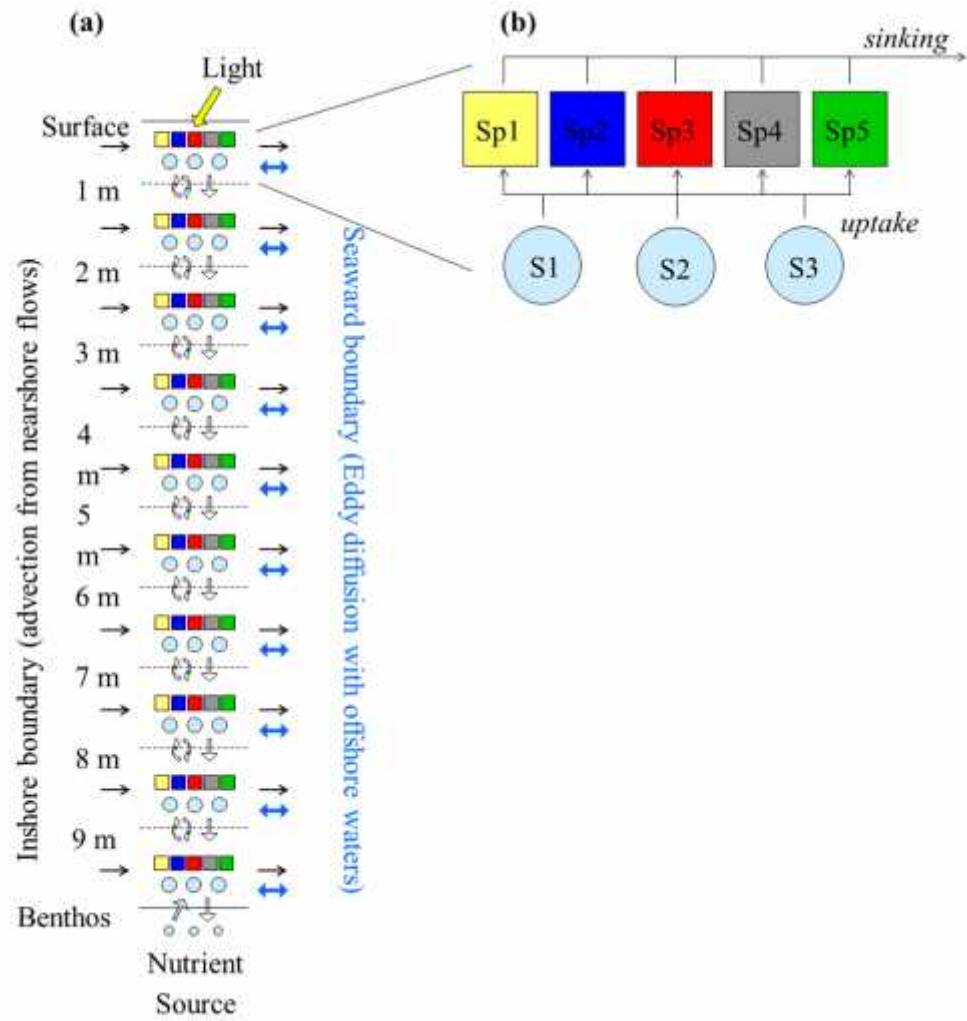
Figure 1. Graphical Representation of the Model. (a) depicts all ten layers and the interactions between the layer and its environment. (b) shows the graphical representation of the species (Sp1, Sp2, etc.) within a layer competing for three resources (S1, S2, S3). Note that within each layer, there isn't necessarily five species. There can be up to twenty species, but there are always three resources.
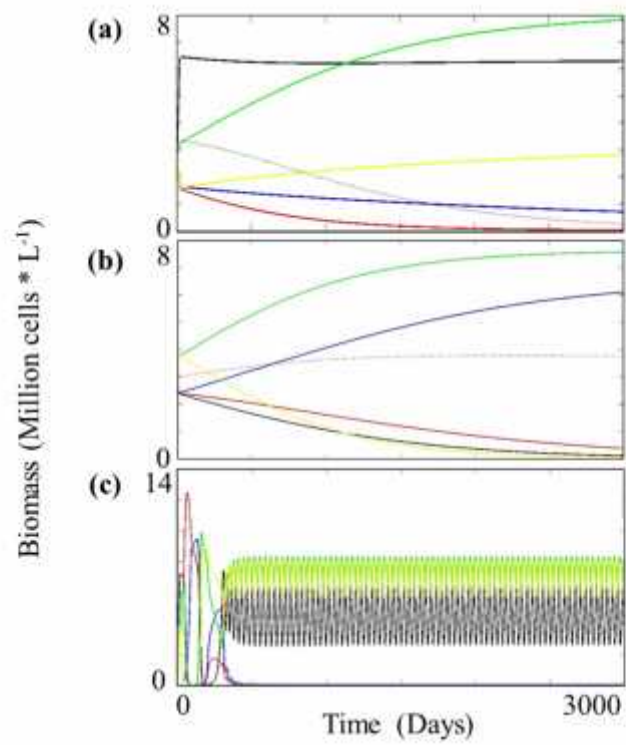
Figure 2. Typical Model Behavior. Model behavior given neutral (a), lumpy (b), or RPS (c) dynamics. The various colors represent different species over a time period of 3000 days. These simulations are simple single patch representations of the species composition with no additional interactions than inflowing nutrients and an outflow of cells and nutrients as demonstrated in Roelke and Eldridge 2008.

scissors', or RPS, (Figure 2c.) assemblage over time in a well-mixed patch with only the

influx of nutrients and flushing of nutrients and cells due to the influx into the system.

These examples represent typical behavior of the control simulations.

3.1. Description of Equations and Numerical Procedures

Note that a summarization of all parameters and units can be found in Appendix

A.

3.1.1. Phytoplankton

The differential equation for phytoplankton species i for layers two through nine

follow the form:

$$\frac{d\phi_i}{d} = \phi_i\left(\mu_i - s - v_u - v_D - \frac{f_o}{V} - D\right) + (s + v_u)\phi_u + v_D\phi_D + D\phi_O \qquad (1)$$

where ϕ_i refers to the population of species i in a given layer, ϕ_u refers to a population

source from the layer above, ϕ_D refers to a population source from the layer below, and

ϕ_O refers to a population source from Eddy diffusion with the adjacent open ocean water

column. The species specific growth rate (μ_i) refers to a per capital growth rate that is

discussed more below. Sinking (s) refers to the rate of cell settling from a layer to the

layer below. Vertical mixing (v_u) refers to the mixing rate between a given layer and the

layer below it. Similarly, vertical mixing (v_D) refers to the mixing rate between a given

layer and the layer above. For this research, the vertical mixing rate between any two

layers was constant between layers. The inflow (f_i) and outflow (f_o) refer to the

source from the nearshore flows and are always equal to each other. For this study, the

nearshore flow always supplied nutrients, but never new cells. The Eddy diffusion rate

(D) refers to the horizontal mixing between a given layer and the adjacent open ocean.

The open ocean source was assumed to be well mixed and introduced both phytoplankton and nutrients into the system.

The first and tenth layer (boundary layers) are similar with a few alterations. The equations for phytoplankton in the top and bottom layers, respectively, follow the form:

$$\frac{d\phi_i}{d} = \phi_i\left(\mu_i - s - v_D - \frac{J_o}{V} - D\right) + v_D\phi_D + D\phi_U \tag{2}$$

$$\frac{d\phi_i}{d} = \phi_i\left(\mu_i - s - v_u - \frac{J_o}{V} - D\right) + (s + v_w)\phi_u + D\phi_U \tag{3}$$

The specific growth rates (μ) for the populations were determined by using the Monod equation and Liebig's law of the minimum. The Monod equation (*sensu* Monod, 1949), is as follows:

$$\mu_i = \mu_{i,m}\left(\frac{S}{S + K_{i,S}}\right) \tag{4}$$

where $m_{i,max}$ is the maximum specific growth rate, S is an inorganic nutrient concentration (N, P or Si), and $K_{i,S}$ is the half-saturation coefficient for nutrient limited growth (again for N, P or Si). Note that in this research, the terms substrate, nutrient, and resource are used interchangeably and all refer to S.

In regards to estimating m_i as a function of light limitation, we employed a commonly used mathematical formula relating μ to irradiance (Platt and Jassby, 1976), as follows:

$$\mu_i = \mu_{i,m}\left(1 - e^{-A_i}\right) \tag{5}$$

where A_i is a scalar factor defined as:

$$A_i = \frac{a_i I_z}{\mu_{i,m}} \tag{6}$$

where α_i is the slope of the photosynthesis-irradiance curve for phytoplankton and I_z is the irradiance integrated over PAR (Platt, 1986), and other parameters are the same as previously defined.

Irradiance at a given depth was then determined with:

$$I_z = I_c e^{-K_T z} \tag{7}$$

where K_T is the extinction coefficient total, z is depth and I_0 is the scalar irradiance incident on the surface. Here, the extinction coefficient total is estimated using:

$$K_T = K_W + K_C \tag{8}$$

where K_W is the extinction of light based on pure water, tripton and CDOM, and K_C is the extinction of light based on the total concentration of phytoplankton (Olivieri and Chavez, 2000). Following this formula enables the phenomena of growth-limitation through self-shading (Klausmeier and Litchman, 2001).

Following these estimations, we then determined the realized specific growth rate for each species using Liebig's "Law of the Minimum":

$$\mu_{\phi_i} = \min(\mu_{i-N}, \mu_{i-P}, \mu_{i-S}, \mu_{i-li\ ht}) \tag{9}$$

where all parameters are the same as previously defined.

3.1.2. Resources

The differential equations for the three resources for layers two through nine follow the form:

$$\frac{d}{d} = \sum_{i=1}^{n} c_i \mu_i \phi_i + v_u(S_u - S) + v_D(S_D - S) - \frac{f_0}{V}S + \frac{f_{li}}{V}S_L - D + DS_0 \tag{10}$$

where S refers to a specific resource in a given layer, S_D refers to the resource source from the layer below, S_u refers to the resource source from the layer above, S_L refers to

the resource source from land inflow, and S_O refers to the resource source from Eddy

diffusion with the adjacent open ocean water column.

The equations for resources in the top and bottom layers, respectively, follow the

form:

$$\frac{d}{d} = \sum_{i=1}^{n} c_i \mu_i \phi_i + v_D(S_D - S) - \frac{I_o}{V} S + \frac{I_{fi}}{V} S_L - D + DS_O \tag{11}$$

$$\frac{d}{d} = \sum_{i=1}^{n} c_i \mu_i \phi_i + v_u(S_u - S) - \frac{I_o}{V} S + \frac{I_{fi}}{V} S_L - D + DS_O + S_B A \tag{12}$$

where S_B is the benthic flux rate and A is the area of the benthos under the water

column. In this case, we assumed that the volume is 1 m^3 of each layer and therefore $A =$

1 m^2. c_i is the fixed cellular content of a substrate in species i, and all other parameters

are the same as described above.

3.2. Simulation Analysis

The following are the scenarios employed to analyze the model described above.

Each scenario is independent of the others and incorporated one different aspect of the

model's structure while ignoring the others. For the simulations run under each scenario,

an assemblage of a given dynamic (neutral, lumpy, or RPS) was initially distributed in

the top five layers and another assemblage of the same dynamic was distributed in the

bottom 5 layers. Then this simulation was repeated, but with the distribution of the

assemblages switched. This process was repeated for assemblage pairings 1 and 2, 3 and

4, 5 and 6, etc. For each scenario, all 30 assemblages (see Appendix C) (15 pairings; 5

pairs of neutral assemblages, 5 pairs of lumpy assemblages, and 5 pairs of intransitive

assemblages) were used and the result on biodiversity was evaluated. The different

scenarios that were simulated involved light extinction (1), nutrient flux from the

14

benthos (2), vertical mixing (3), sinking (4), and eddy diffusion (5). Each scenario has an 'a', 'b', and 'c' rate for later comparison.

3.2.1. Scenario One: Effect of Light Limitation

In this first scenario, the model was explored by comparing a light-saturated simulation to one where light extinguishes with depth. In the light saturated simulations, only nutrients had the capacity to become growth-limiting. In the simulation where light extinguishes with depth, either nutrients or light became growth-limiting, depending on the population densities and competitive abilities of the species present. As mentioned above, α_i is the slope of the photosynthesis-irradiance curve for phytoplankton, and in this case we used $\alpha_i = 2.86 \ x \ 10^{-17}$, $4.59 \ x \ 10^{-17}$, and $6.32 \ x \ 10^{-17} \ cm^2 \ s \ quanta^{-1} \ day^{-1}$ (Platt et al., 1982; Platt, 1986; Moigis, 1999) to explore the sensitivity of the response variables (described below). For this first scenario, there was no nutrient flux from the benthos, no vertical mixing between layers, no sinking of cells from upper layers down to lower layers, and no mixing with offshore waters.

3.2.2. Scenario Two: Effect of Nutrient Flux from the Benthos

In this scenario, there was nutrient flux from the benthos. Nutrient flow from the benthos into the pelagic zone has been explored in previous work within coastal water and lagoon systems (Rowe et al., 1975; Nowicki and Nixon, 1985). For this model, we compared the inflow of nutrients at concentrations of 0.1, 10, and 100 $\mu M \ day^{-1}$ (Nowicki and Nixon, 1985). Light was saturated within the system, there was no vertical mixing between layers, no sinking of cells, and no mixing with offshore waters. Note that in this scenario, nutrients from the benthos were added only to the lowest patch

given the lack of vertical mixing. This alone does not represent a natural system, but as

mentioned above the goal of this experiment was to see how each process effects the

biodiversity in terms of the three classes of supersaturated systems. However, to check

that the lack of vertical mixing did not have a significant effect on the simulations with

nutrient flux alone, we included simulations with $v_u = v_b = 0.01$ day^{-1} simultaneously

with the various rates of nutrient flux. We concluded that the vertical mixing inclusion

with nutrient flux did not provide any different results; therefore, figures are included in

Appendix B, but not exclusively discussed further in this paper.

3.2.3. Scenario Three: Effect of Vertical Mixing

Values of $v_u = v_b = 0.001$, 0.01, and 0.1 day^{-1} were used for the rate of exchange

(both cells and nutrients) equally between each layer. These were the same rates used in

Roelke and Eldridge 2008 to evaluate the effect of horizontal mixing on a two patch

system. The values used here were chosen to remain consistent with that study.

Otherwise, light was saturated within the system, there was no nutrient flux from the

benthos, no sinking of cells, and no mixing with offshore waters.

3.2.4. Scenario Four: Effect of Sinking

In this scenario, we introduced sinking. It is not uncommon that phytoplankton

will move vertically within the water column based on their density rather than because

of the effects of mixing (Huisman and Sommeijer, 2002). Here, the simulations were run

using recorded rates of $s = 0.22$, 0.5, and 0.7 m day^{-1} (Huisman and Sommeijer, 2002).

Another complication in parameterizing sinking rates is the possibility for variability of

sinking rates between species and the possibility of buoyancy as well as sinking.

However, this exploration is beyond the scope of this study. For this scenario, light was saturated within the system, no flux of nutrients from the benthos, no vertical mixing between layers, and no mixing with offshore waters.

3.2.5. Scenario Five: Effect of Eddy Diffusion

Here, we added mixing with offshore waters. This introduced both nutrients and cells into the system. In order to decide which species was present in adjacent offshore patches, we used the initial species composition for the simulation. For example, if assemblage 1 was in patches 1-5 and assemblage 2 was in patches 6-10, then assemblage 1 was present in the 'offshore' waters for layers 1-5 and assemblage 2 was present for 'offshore' waters for layers 6-10. The species density was $\phi_0 = 8 * 10^6 c_i$ $/L$ based on literature specifying the mean chlorophyll concentration of open ocean waters during the summer time (van de Poll et al., 2013). In order to both keep the model simple and to not flush out the species within each patch, the Eddy diffusion mixing rates were $D = 0.001$, 0.01, and 0.1 day^{-1}, and the nutrient concentration was $S_0 = .01$ μM with the idea that offshore waters are more nutrient limited. These rates represent the lower range of recorded Eddy diffusion rates. Rather than explore the upper range, which would give us information about when mass effects from the open ocean sources occur, we found it more interesting to explore theoretical lower values and whether or not these low rates could still provide sufficient cells to affect species richness and biodiversity in the system. For this scenario, light was saturated within the system, there was no flux of nutrients from the benthos, no vertical mixing between layers, and no sinking of cells.

3.3. Parameters and Initial Conditions

For any given patch or species assemblage, the initial concentration of cells was 0.1 $\times 10^6$ cells liter^{-1} and all resources (S) were initially 10 µM. As mentioned above, a total of 30 assemblages were used in the analyses and were identical to those used in Roelke and Eldridge (2008). Therefore, all half saturation coefficients (K_S) and fixed cellular content for a given substrate (c_i) were used from their analysis to characterize the three different theoretical assemblages. The advection flow rate from land was a constant $f_L = 0.25$ day^{-1} and concentration of resources (S_L) from inflow was 10 µM throughout the simulations. The initial concentration of cells, the inshore nutrient concentration and hydraulic dilution rate were chosen to remain consistent with the Roelke and Eldridge 2008 paper. All simulations were run for a period of 3000 days. This duration was to ensure that the populations left any unstable initial behavior and reached a constant population (for neutral and lumpy assemblages) or a consistent periodic behavior (for RPS assemblages). Figure 3 depicts the various possibilities of the average total biomass versus the depth, which in this case was an assumed 10 meters (this was based on the earlier assumption that the volume equals 1 cubic meter). Depending on which scenario, there is a possibility of even distribution of cells throughout the water column (Figure 3a.), accumulation of cells in the lower layers (Figure 3b.), or a subsurface maximum (Figure 3c.).
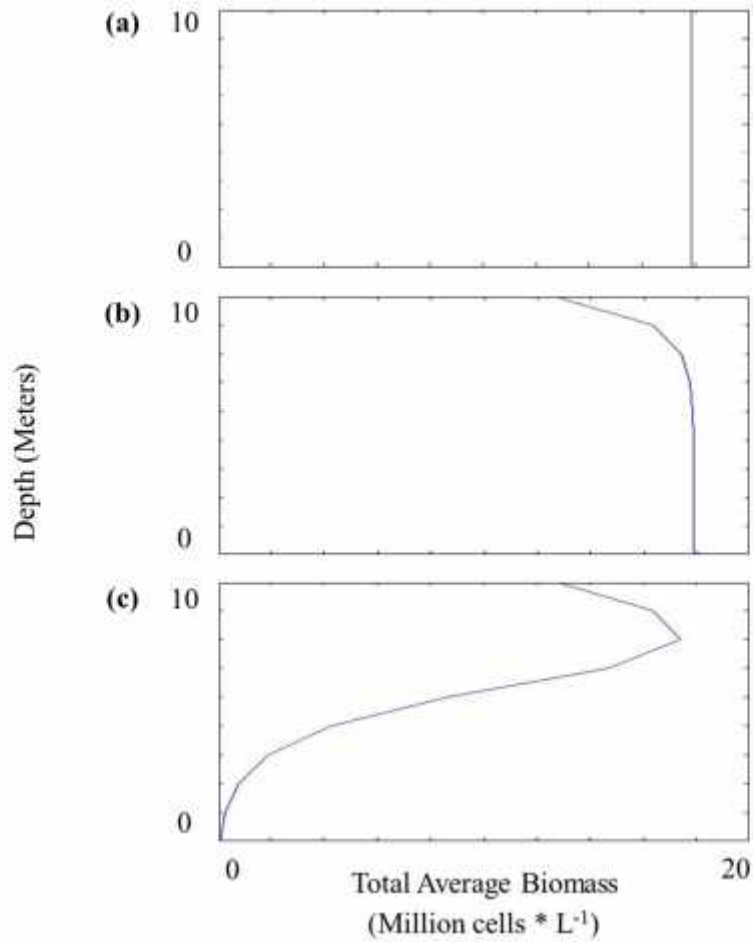
Figure 3. General Vertical Distributions of Cells. (a) depicts an even distribution characteristic of the control simulations with no mixing between layers, (b) depicts an accumulation of cells toward the bottom of the water column, and (c) depicts a subsurface maximum. These values depict the total biomass after the last 500 days were averaged and summed. This was to account for the oscillations characteristic in the RPS simulations.

3.4. Response Variables

In order to evaluate the effects of each of the previously discussed interactions, mean α richness, mean β diversity, γ richness, and α diversity were evaluated. As defined in Whittaker (1972), 'α' measures of biodiversity focus on a patch, 'γ' measures of biodiversity represent the entire system, and 'β' measures of biodiversity represent the differentiation of biodiversity between patches within the entire system. Because β diversity is the differentiation between local patches, its calculations are typically represented by the multiplicative format $\beta = \gamma / \alpha$ (Whittaker, 1972; Jost, 2007). Although the main analysis will only be looking at α and β values, γ level calculations are necessary part of the primary solutions.

We used the simplest calculation of α, β, and γ richness: the number of species in each patch and of the entire system respectively. Because of this, richness was simply the number of species remaining after completion of a simulation. If a population was higher than 0.1×10^6 cells liter^{-1}, it was considered present. This value was used because it was the value for the initial population size. Furthermore, so that different combinations of assemblages can be compared (the number of species in the 30 assemblages reported in Roelke and Eldridge (2008) ranges from 5 to 8), the richness was expressed as a proportion of the number of species present to the total number of species present initially throughout the water column. β richness then was equal to γ richness divided by the mean α richness.

The Shannon index is a common tool to calculate biodiversity (Shannon and Weaver, 1949):

$$H = - \sum_{1}^{N}(P_i \times \ln P_i) \tag{13}$$

N is the total number of species present initially throughout the entire water column (this changed based on what assemblages were being used) and P_i is the proportion of biomass (individuals) of species i to the total biomass. We modified the classic Shannon Weaver index so that diversity $= \exp(H)$ where H is the Shannon Weaver index calculation over a given patch, diversity $= \exp(H_{total})$ where H_{total} is the Shannon Weaver index calculated over the entire region, and diversity is diversity divided by average diversity (Jost, 2006; Jost, 2007). For all calculations, the biomass of populations were first averaged over the last 500 days. This accounted for the oscillations in the rock-paper-scissors assemblages. In the neutral and lumpy assemblages, populations typically never reach a true equilibrium. However, the increase or decrease in biomass was so small that we assumed the averaging over 500 days did not make a significant difference in the calculations.

After running all simulations, we calculated the percent difference between the scores for the a and b rates and then b and c rates within each scenario and then took the average of those two percent differences for both richness and diversity. These percent differences were used as a measure of sensitivity to a given scenario. That is, we did account for any changes from the control, but rather how the assemblages behaved in response to the various rates within a scenario. These plots are shown in Figure 4.
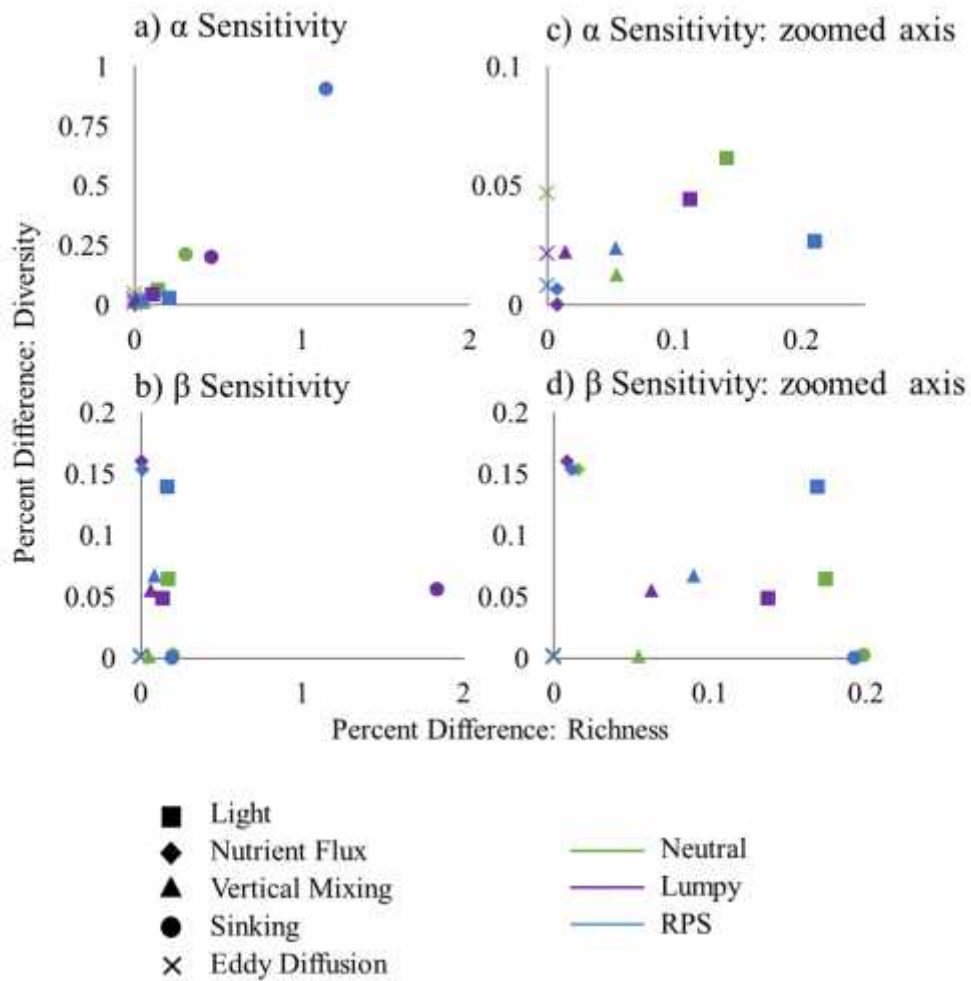
Figure 4. Assemblage Sensitivity. These were calculated by taking the percent differences between the various rates within a given scenario. (a) represents the sensitivities calculated on the   level while (b) represents the   level sensitivities.

4. RESULTS AND DISCUSSION

Figure 4 shows the sensitivity of each assemblage type in each scenario in regards to both richness and diversity. For both   and   levels, assemblages' sensitivity to sinking was significantly higher than any other scenario. For this reason Figure 4c. and Figure 4d provide a zoomed in range to observe patterns for the other scenarios. Excluding sensitivity to sinking, the overall sensitivity on the   level had a wider range in richness than that for the   level but a similar range in diversity.

4.1. Sensitivity to Incident Light

In regards to changes in the incident light, RPS assemblages showed the highest sensitivities in α richness (Figure 4c) and   diversity (Figure 4d). Conversely, lumpy assemblages showed the lowest sensitivities in α and   richness (Figure 4c, d), and the lowest sensitivity in   diversity (Figure 4d). Representative simulations using an RPS and a lumpy assemblage (Figure 5) show that light becomes growth limiting with depth, in the extreme with populations going extinct, and that the effect of this light limitation on the distribution of phytoplankton with depth is more pronounced with decreased incident light. All RPS assemblages showed aperiodic population dynamics when light effects were incorporated into the model, which were very different from the stable population oscillations observed previously for these assemblages (Roelke and Eldridge, 2008). While still being aperiodic, the timing of the population cycling in RPS assemblages became protracted when incident light was reduced. This resulted in fewer populations being counted as 'present' within the 500 day window used to quantify presence/absence in our analysis technique, thereby leading to decreased α richness and
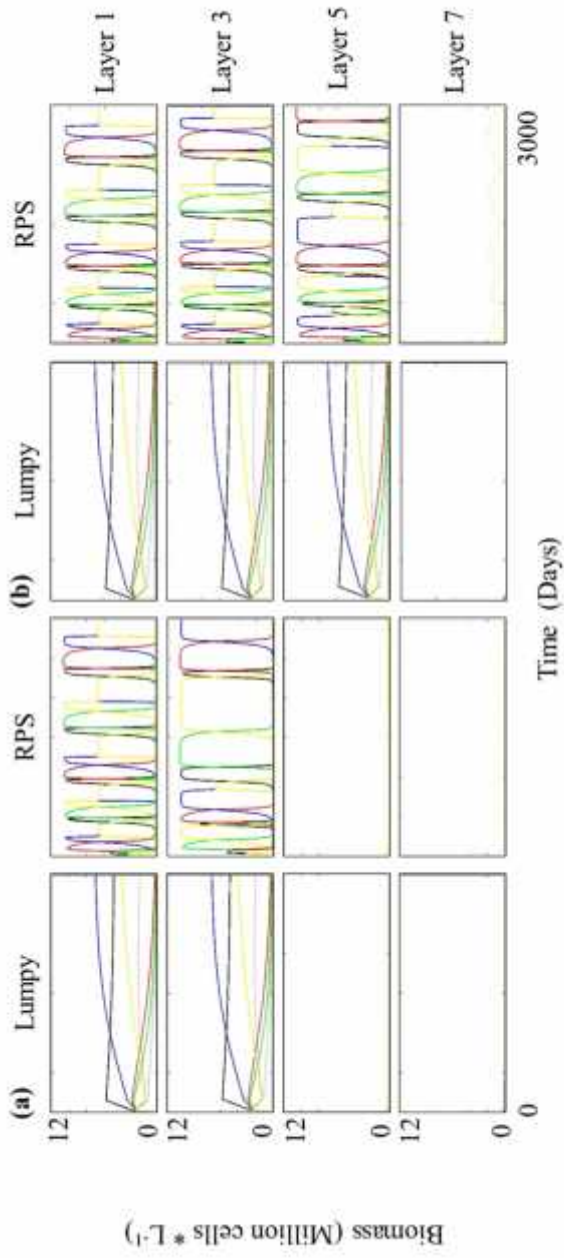
23

Figure 5. Assemblage Response to Light. Representation of population dynamics time series for a lumpy and RPS assemblage when (a) $a_i = 2.86 \times 10^{-17}$ and (b) $a_i = 6.32 \times 10^{-17}$ cm^2 s quanta^{-1} day^{-1} for layers 1, 3, 5, and 7. Light diminishes and eventually does not reach the lower layers resulting in the death of all cells.

diversity. Lumpy assemblages, on the other hand, had the fewest number of

populations that dropped below our 'presence/absence' threshold in the final 500 days of

the simulation.

Other research has demonstrated that light intensity influences phytoplankton

population dynamics in water columns (Litchman and Klausmeier, 2001; Flöder et al.,

2002; Flöder and Burns, 2005). Those papers included diel light fluctuations, but did not

address differences between assemblage types, i.e., neutral, lumpy and RPS. Future work

incorporating both diel light effects and assemblage type is merited, as it might show an

even greater sensitivity of RPS assemblages to diminished light.

4.2. Sensitivity to Nutrient Flux from the Benthos

Figure 6 shows layers nine and ten for two different nutrient flux rates (S_H = 0.1

and 10 µM day-1) for lumpy and RPS assemblages. Layer nine represents the

assemblage dynamics for all other layers as well. The flux of nutrients from the benthos

only affected the bottom most layer, hence the low   sensitivity for all assemblage types.

Given the higher availability of nutrients, layer ten has much higher biomasses (note the

differing scales in Figure 6).   diversity sensitivity is higher in all three assemblage types

compared to   richness. This is simply a result of the shifts in biomass in the bottom

layer given the different rates of nutrient flux.

We have a system that is characteristic of a coastal open ocean which allows a

"fresh water" source to bring in nutrients to all layers. Previous models integrating

phytoplankton vertical distribution and biomass as a function of light and benthic

nutrient flux exist (e.g. Mellard et al., 2011; Klausmeier and Litchman, 2001; Huisman
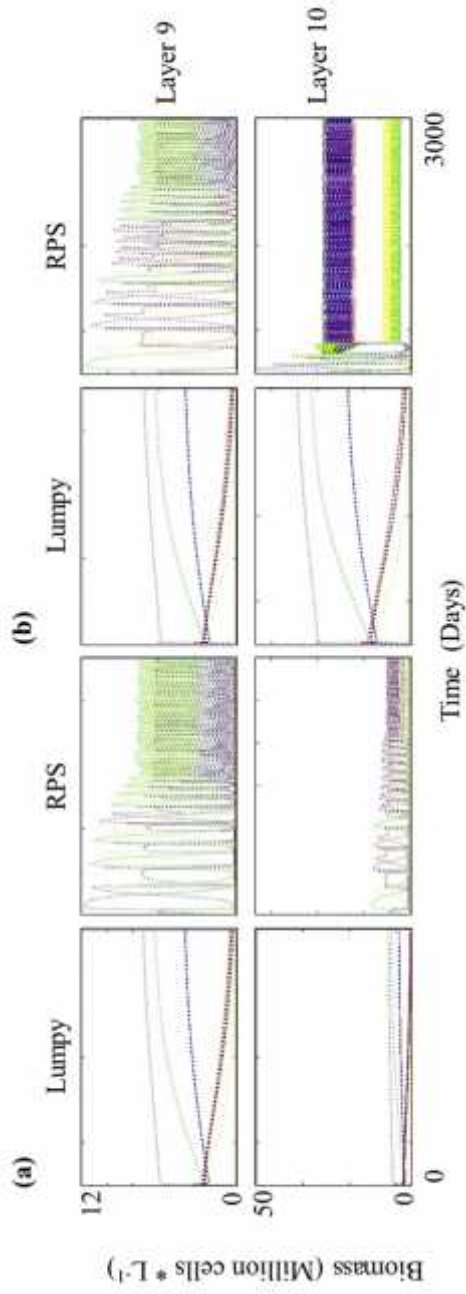
Figure 6. Assemblage Response to Nutrient Flux from the Benthos. Representation of population dynamics time series for a lumpy and RPS assemblage when (a) $S_{BF} = 0.1$ μM day^{-1} and (b) $S_{BF} = 10$ μM day^{-1} for layers 9 and 10. Given the control rates, the system is nutrient limited, but observe the large increase in biomass in the tenth layer when nutrient flux is introduced.
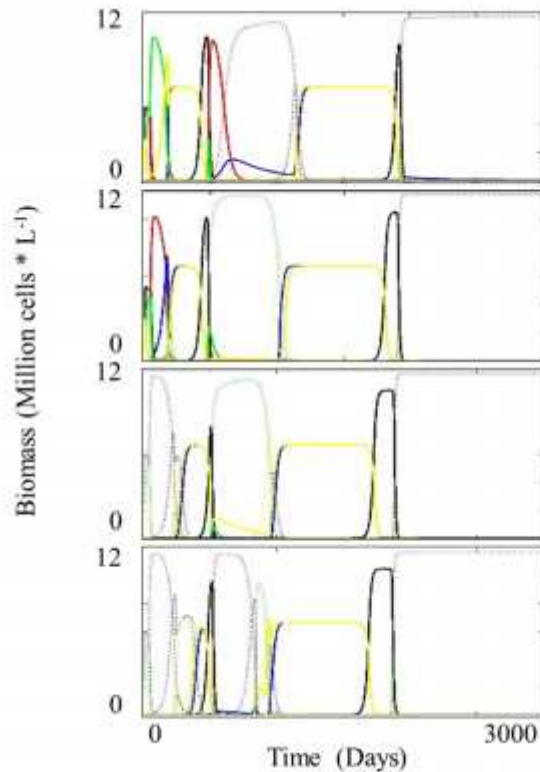
Figure 7. Homogenization of Assemblage with Vertical Mixing. Representative RPS assemblage time series when $v_u = v_b = 0.001$ day^{-1} for layers 1, 4, 7, and 10. Even at the lowest mixing rate, the patches have become homogenized by the end of the time frame. This was true for all assemblage types at all mixing rates.

et al., 2006; Beckman and Hense, 2007), but these models are closed systems making them even more sensitive and dependent on nutrient source.

4.3. Sensitivity to Vertical Mixing

Even with the lowest mixing rate, $v_u = v_b = 0.001$ day^{-1}, patches became uniform by the end of the given time range. This is shown in Figure 7 which depicts patches 1, 4, 7, and 10 for a representative RPS assemblage lowest mixing rate. The initial dynamics in a given patch may differ, but the outcome is similar for all patches within the column.

27

Figure 8. Assemblage Response to Vertical Mixing. Layer one of a representative RPS and neutral assemblage or mixing rates of $v_u = v_b =$ (a) 0.001, (b) 0.01, and (c) 0.1 day^{-1}. Note that for the neutral assemblage, the out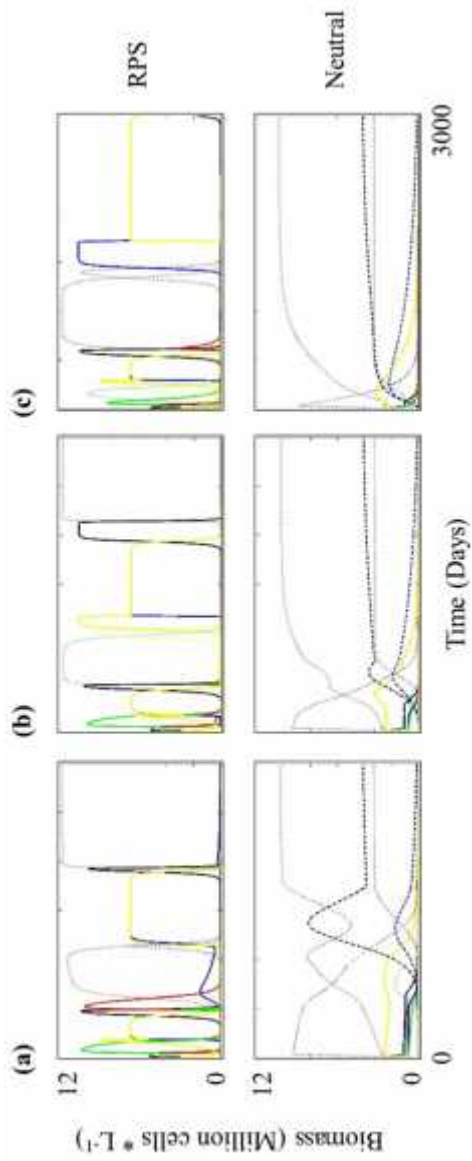come is the same for all mixing rates (lumpy was similar), but the outcome was not the same in all cases for the RPS assemblage.

This was true for all assemblage types and with all mixing rates. Figure 8 illustrates layer one of a representative RPS and neutral assemblage for all three mixing rates. Neutral assemblages ended with the same species and similar biomasses regardless of mixing rate (same is also true for lumpy) while RPS were not consistent. This explains the high richness and diversity sensitivity on both the   and   level for RPS assemblages while lumpy and neutral have low sensitivity on both scales. This is consistent with other research that found that RPS assemblages are sensitive and lose richness when mixing is high enough (Kerr et al., 2002; Reichenbach et al., 2007a, 2007b; Roelke and Eldridge, 2008).

4.4. Sensitivity to Sinking

All assemblage types are most sensitive to this mechanism on both the   and   level. Figure 9 illustrates layer nine for all three assemblage types when $s =$ (a) 0.22, (b) 0.5, and (c) 0.7 m day^{-1}. Similar to previous scenarios, changes in rates resulted in protract cycling in RPS assemblage types, and therefore very high   sensitivity.

Strangely, the   richness sensitivity of lumpy assemblage types is very high; however, this is a result of how the response variables were calculated. Recall that the threshold biomass for the richness calculations was set to 0.1 million cells L^{-1}. For   calculations, given the highest sinking rate, lumpy assemblages typically had very few species above the threshold and therefore low   scores. However, the accumulated biomass throughout the column added up to be higher than the threshold and resulted in high   richness scores. Given that   richness is   /   , this resulted in much higher   scores for the highest sinking rate and therefore much higher   sensitivity.
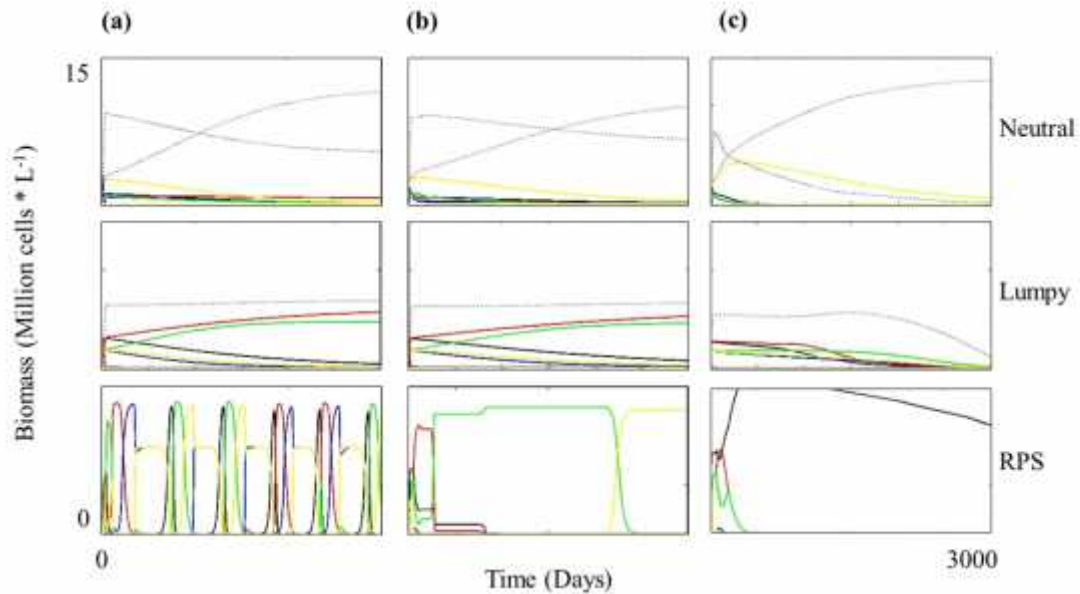
Figure 9. Assemblage Response to Sinking. Time series of representative neutral, lumpy, and RPS assemblages in patch nine when $s =$ (a) 0.22, (b) 0.5, and (c) 0.7 m day^{-1}.

It is also important to note that regardless of the orientation of the two initial assemblages throughout the water column, the upper assemblage provided the dominant species throughout the water column. In this scenario, the sinking rate was a press (ongoing) perturbation rather than a pulse perturbation. This constant pressure from invading species quickly dominated and outcompeted the lower assemblages. Furthermore, in both neutral and RPS assemblage types, particularly when the sinking rate was at its highest, one species dominated overall others.

Upon reference to the original development of these assemblages (Roelke and Eldridge, 2008), this dominate species happened to be one with a centrally located R* value in the resource trade off space. Lumpy assemblages typically did not have any species occupying this space at all hence, all species are decreasing over the extended

time frame. So why is being located centrally in the resource trade off space result in higher biomass given high sinking rates? Exploration of this is necessary in future work.

There is little research in regards to the effects of sinking alone, but plenty of work looking at one or few species responses in a water column with light, sinking, and turbulence (e.g. Huisman et al., 2002; Hsu and Lou, 2010; and Peng and Zhao, 2016).

4.5. Sensitivity to Eddy Diffusion

In this scenario, patches 1-5 all behave the same as do patches 6-10 because there are no interactions between patches and no influences vertically. Richness sensitivity was zero for all assemblage types for both   and   levels for all rates because there is a constant flow of new cells from the "marine" source, as defined in the Simulation Analysis section, resulting in a rescue effect. Therefore, we know that the differences in   and   diversity sensitivity for a given assemblage are due to the changes in biomass between the different diffusion rates.

Figure 10 shows layer one of a representative RPS and neutral assemblage for D = 0.001, 0.01, and 0.1 day^{-1}. RPS assemblages have a narrower range of biomass over the three rates while the neutral assemblages have the widest range of biomass distribution and a subsequent higher diversity sensitivity.

Given that this pattern is constant throughout the water column,   level scores were consistent between rates and resulted in very low sensitivity for all assemblage types. Even at these low rates, we can conclude that horizontal transport is an important factor when considering biodiversity which has also been found in other research (McKiver et al., 2009; Adjou et al., 2012). Here, the species introduced were of the same

31

existing assemblage. The outcome may be different if the species introduced were truly

"invaders" or even a mix of both species existing within the assemblage and invaders.
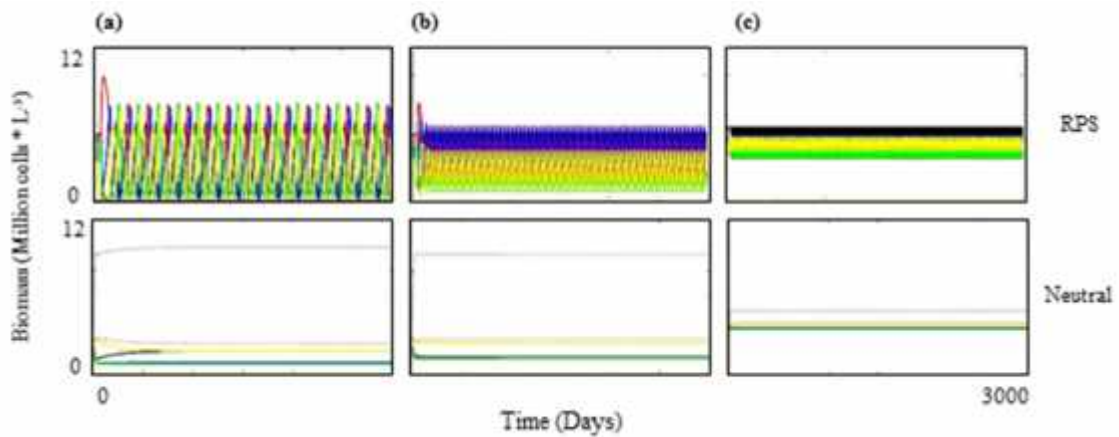

Figure 10. Assemblage Response to Eddy Diffusion. Layer one of a representative RPS and neutral assemblage for D = (a) 0.001, (b) 0.01, and (c) 0.1 day^{-1}. With increasing horizontal mixing rates, the range of biomasses of each species narrows and approaches an attraction basin. This was true for all assemblage types.

5. CONCLUSION

Previous work done by Roelke and Eldridge (2008) found that RPS and neutral assemblages lost super saturation qualities after introduced to immigration while lumpy assemblages maintained super saturation albeit lower richness. We found that RPS assemblages were the most sensitive to various environmental process in regards to assemblage richness and diversity while lumpy and neutral assemblages were the less sensitive on both the   and   scale. Overall, there were no consistent trends that differentiated richness and diversity between the   and   scales; however, the trends were still different. Few studies have addressed the fragility of these various assemblage types to environmental disturbances, such as those discussed in this paper, making this research an important primary assessment to supersaturated behavior types.

Although it is important to continuously assess which model better fits a given community, we agree with the sentiments of Leibold and McPeek (2006) that future work should focus more, as this paper does, on looking at the perspectives of these different assemblage types rather than only developing techniques to fit experimental data with neutral, non-neutral dynamics, or distributed along the continuum between neutral, lumpy, and niche theory (e.g. Chesson, 2000; Etienne, 2007; Pueyo et al., 2007) or even developing rules that unify all of the popular theories (McGill, 2010).

When looking at Figure 4, it is observed that the sensitivity of any given assemblage is affected more by the scenario than that of the assemblage type. Assemblage types were not grouped together regardless of the disturbance but rather grouped by disturbance type. This may not be the case if the sensitivity analysis has a

higher resolution and/or wider range. Future work should explore the sensitivity of these systems to these various mechanisms further and in greater detail. Our simplistic first step was only the initial view of how these super saturated systems may behave given more realistic environmental conditions. The more we know about how these systems behave in different environments, the more we can understand about observed phytoplankton assemblages and how they may be responding to every day disturbances.

WORKS CITED

Adjou, M., J. Bendtsen, and K. Richardson. 2012. "Modeling the influence from ocean transport, mixing and grazing on phytoplankton diversity." *Ecological Modelling* 225: 19-27.

Beckmann, A., and I. Hense. 2007. "Beneath the surface: Characteristics of oceanic ecosystems under weak mixing conditions - A theoretical investigation." *Progress in Oceanography* 75: 771-796.

Bickford, D., D. Lohman, N. Sodhi, P. Ng, R. Meier, K. Winker, K. Ingram, and I. Das. 2007. "Cryptic species as a window on diversity and conservation." *Trends In Ecology and Evolution* 22: 148-155.

Chesson, P. 2000. "Mechanisims of maintenance of species diversity." *Annual Review of Ecology and Systematics* 31: 343-366.

Codeco, C. T., and J. P. Grover. 2001. "Competition along a spatial gradient of resource supply: a microbial experimental model." *American Naturalist* 157: 300-315.

Cullen, J. J. 1982. "The deep chlorophyll maximum - comparing vertical profiles of chlorophyll a." *Canadian Journal of Fisheries and Aquatic Sciences* 39: 791-803.

Drost, M. B. H., H. P. J. S. Cuppen, E. J. Van Nieukerken, and M. Schreijer. 1992. *De Waterkevers van Nederland.* Utrecht: Uitgeverij.

Etieene, R. 2007. "A neutral sampling formula for multiple samples and an 'exact' test of neutrality." *Ecology Letters* 10: 608-618.

Flöder, S., and C. W. Burns. 2005. "The influence of fluctuating light on diversity and species number of nutrient-limited phytoplankton." *Journal of Phycology* 41: 950-956.

Flöder, S., J. Urabe, and Z. Kawabata. 2002. "The influence of fluctuating light intensities on species composition and diversity of natural phytoplankton communities." *Oecologia* 133: 395-401.

Grover, J.P. 1989. "Effects of Si:P supply ratio, supply variability, and selective grazing in the plankton: an experiment with a natural algal and protisan assemblage." *Limnology and Oceanography* 34 (2): 349-367.

Han, B., M. Virtanen, J. Koponen, and M. Straškraba. 1999. "Predictors of light-limited growth and competition of phytoplankton in a well-mixed water column." *Journal of Theoretical Biology* 197: 439-450.

Hastings, A. 2010. "Timescales, dynamics, and ecological understanding." *Ecological Society of America* 91 (12): 3471-3480.

Holling, C.S. 1992. "Cross-scale morphology, geometry, and dynamics of ecosystems." *Ecological Monographs* 62: 447-502.

Hsu, S., and Y. Lou. 2010. "Single phytoplankton species growth with light and advection in a water column." *SIAM Journal on Applied Mathematics* 70: 2942-2974.

Hubbell, S. P. 2001. *The Unified Neutral Theory of Biodiversity and Biogeography.* Princeton University Press.

Huisman, J, R. Jonker, C. Zonneveld, and F. Weissing. 1999. "Competition for light between phytoplankton species: Experimental tests of mechanistic theory." *Ecology* 80: 211-222.

Huisman, J., and B. Sommeijer. 2002. "Population dynamics of sinking phytoplankton in light-limited environments: simulation techniques and critical parameters." *Journal of Sea Research* 48: 83-96.

Huisman, J., and F. J. Weissing. 2001. "Biological conditions for oscillations and chaos generated by multispecies competition." *Ecology* 82: 2682-2695.

Huisman, J., and F. J. Weissing. 1995. "Competition for nutrients and light in a mixed water column: A theoretical analysis." *The American Naturalist* 146: 536-564.

Huisman, J., and F. J. Weissing. 1994. "Light-limited growth and competition for light in well-mixed aquatic environments: An elementary Model." *Ecology* 75: 507-520.

Huisman, J., M. Arrayás, U. Ebert, and B. Sommeijer. 2002. "How do sinking phytoplankton species manage to persist?" *The American Naturalist* 159: 245-254.

Huisman, J., N.N.P. Thi, D.M. Karl, and B. Sommeijer. 2006. "Reduced mixing generates oscillations and chaos in the oceanic deep chlorophyll maximum." *Nature* 439: 322-325.

Hutchinson, G.E. 1961. "The Paradox of the Plankton." *The American Naturalist* 95: 137-145.

Jackson, J.B., and L. Buss. 1975. "Allelopathy and spatial competition among coral reef Invertebrates." *Proceedings of the National Academy of Sciences U.S.A.* 72: 5160-5163.

Jost, L. 2006. "Entropy and diversity." *OIKOS* 113: 363-375.

Jost, L. 2007. "Partitioning diversity into independent alpha and beta components." *Ecology* 88: 2427-2439.

Kerr, B., M.A. Riley, M.W. Feldman, and B. J. M. Bohannan. 2002. "Local dispersal promotes biodiversity in a real-life game of rock-paper-scissors." *Nature* 418: 171-174.

Klausmeier, C.A., and E. Litchman. 2001. "Algal games: The vertical distribution of phytoplankton in poorly mixed water columns." *Limnology and Oceanography* 46: 1998-2007.

Leibold, M., and M. McPeek. 2006. "Coexistence of the niche and neutral perspectives in community ecology." *Ecology* 87: 1399-1410.

León, J. A., and D.B. Tumpson. 1975. "Competition between two species for complementary or substitutable resources." *Journal of Theoretical Biology* 50: 50:185-201.

Litchman, E., and C. A. Klausmeier. 2001. "Competition of phytoplankton under fluctuating light." *The American Naturalist* 157.

May, R., and W. Leonard. 1975. "Nonlinear aspects of competition between three species." *SIAM Journal on Applied Mathematics* 29: 243-253.

McGill, B. 2003. "A test of the unified neutral theory of biodiversity." *Letters to Nature* 422: 881-885.

McGill, B. 2010. "Towards a unification of unified theories of biodiversity." *Ecology Letters* 13: 627-642.

McKiver, W., Z. Neufeld, and I. Scheuring. 2009. "Plankton bloom controlled by horizontal stirring." *Nonlinear Processes in Geophysics* 16: 623-630.

Mellard, J., K. Yoshiyama, E. Litchman, and C. Klausmeier. 2011. "The vertical distribution of phytoplankton in stratified water columns." *Journal of Theoretical Biology* 269: 16-30.

Moigis, A. 1999. "Photosynthetic rates in the surface waters of the Red Sea: the radiocarbon versus the non-isotopic dilution method." *Journal of Plankton Research* 22: 713-727.

Monod, J. 1949. "The Growth of Bacterial Cultures." *Annual Review of Microbiology* 371-394.

Mouquet, N., and M. Loreau. 2003. "Community patterns in source-sink metacommunities." *The American Naturalist* 162: 544-557.

Nowicki, B.L., and S.W. Nixon. 1985. "Benthic nutrient remineralization in a coastal lagoon ecosystem." *Estuaries* 8: 182-190.

Olivieri, R.A., and F. P. Chavez. 2000. "A model of plankton dynamics for the coastal upwelling system of Monterey Bay, California." *Deep-Sea Research II* 47: 1077-1106.

Peng, R., and X. Zhao. 2016. "A nonlocal and periodic reaction-diffusion-advection model of a single phytoplankton species." *Journal of Mathematical Biology* 72: 755-791.

Perruche, C., P. Riviére, P. Pondaven, and X. Carton. 2010. "Phytoplankton competition and coexistence: Intrinsic ecosystem dynamcis and impact of vertical mixing." *Journal of Marine Systems* 81: 99-111.

Petrovskii, S. V., and H. Malchow. 2001. "Wave of chaos: new mechanism of pattern formation in spatio-temporal population dynamics." *Theoretical Population Biology* 59: 157-174.

Platt, T, W.G. Harrison, B Irwin, E. P. Horne, and C.L. Gallegos. 1982. "Photosynthesis and photoadaptation of marine phytoplankton in the Arctic." *Deep-Sea Research* 29: 1159-1170.

Platt, T. 1986. "Primary producation of the ocean water column as a function of surface light intensity: algorithms for remote sensing." *Deep-Sea Research* 33: 149-163.

Platt, T., and A. Jassby. 1976. "The relationship between photosynthesis and light for natural assemblages of coastal marine phytoplankton." *Journal of Phycology* 12: 421-430.

Prowe, A. E. F., M. Pahlow, S. Dutkiewicz, and A. Oschlies. 2014. "How important is diversity for capturing environmental-change responses in ecosystem models?" *Biogeosciences* 11: 3397-3407.

Pueyo, S., F. He, and T. Sillio. 2007. "The maximum entropy formalism and the idiosyncratic theory of biodiversity." *Ecology Letters* 10: 1017-1028.

Reichenback, T., M. Mobilia, and E. Frey. 2007a. "Mobility promotes and jeopardizes biodiversity in rock–paper–scissors games." *Nature* 448: 1046-1049.

Reichenback, T., M. Mobilia, and E. Frey. 2007b. "Noise and Correlations in a Spatial Population Model with Cyclic Competition." *Physical Review Letters* 99.

Reynolds, C. S. 1993. "Scales of disturbance and their role in plankton ecology." *Hydrobiologia* 249: 157-171.

Richerson, P. J., R. Armstrong, and C. R. Goldman. 1970. "Contemporaneous disequilibrium: a new hypothesis to explain the paradox of the plankton." *Proceedings of the National Academy of Sciences of the USA* 67: 1710-1714.

Roelke, D. L., and P. M. Eldridge. 2008. "Mixing of supersaturated assemblages and the precipitous loss of species." *The American Naturalist* 171: 162-175.

Roelke, D. L., and P.M. Eldridge. 2010. "Losers in the 'Rock-Paper-Scissors' game: The role of non-hierarchical competition and chaos as biodiversity sustaining agents in aquatic systems." *Ecological Modeling* 221: 1017-1027.

Roelke, D. L., P. M. Eldridge, and L. A. Cifuientes. 1999. "A model of phytoplankton competition for limiting and nonlimiting nutrients: implications for development of estuarine and nearshore management schemes." *Estuaries* 22: 92-104.

Rowe, G.T., C.H. Clifford, and K.L. Smith Jr. 1975. "Benthic nutrient regeneration and its coupling to primary productivity in coastal waters." *Nature* 215-217.

Saez, A., and E. Lozano. 2005. "Body doubles." *Nature* 433: 111.

Scavia, D., J. C. Field, D. F. Boesch, R. W. Buddemeier, V. Burkett, D. R. Cayan, and M. Fogarty. 2002. "Climate change impacts on U.S. coastal and marine ecosystems." *Estuaries* 25: 149-164.

Scheffer, M., and van Nes E.H. 2006. "Self-organized similarity, the evolutionary emergence of groups of similar species." *Proceedings of the National Academy of Sciences of the United States of America* 103: 6230-6235.

Schippers, P, A. M. Verschoor, M. Vos, and W. M. Mooij. 2001. "Does "supersaturated coexistence" resolve the "paradox of the plankton"?" *Ecology Letters* 4: 404-407.

Shannon, C.E., and W. Weaver. 1949. *The Mathematical Theory of Communication.* Urbana, IL.: University of Illinois Press.

Sinervo, B., and C.M. Lively. 1996. "The rock-paper-scissors game and the evolution of alternative male strategies." *Nature* 380: 240-243.

Sommer, U. 1984. "The paradox of the plankton: fluctuations of the phosphorus availability maintain diversity of phytoplankton in flow-through cultures." *Limnology and Oceanography* 29: 633-636.

Sommer, U. 1985. "Comparison between steady state and non-steady state competition: Experiments with natural phytoplankton." *Limnology and Oceanography* 30: 335-346.

Sommer, U., Z. M. Gliwicz, W. Lampert, and A. Duncan. 1986. "The PEG model of seasonal succession of plankton events in fresh waters." *Archiv für Hydrobiologie* 106: 436-440.

Tilman, D. 1982. *Resource competition and community structure.* Princeton, NJ: Princeton University Press.

Tilman, D. 1977. "Resource competition between planktonic algae: an experimental and theoretical approach." *Ecology* 58: 338-348.

Tilman, D. 1981. "Tests of resource competition theory using four species of Lake Michigan algae." *Ecology* 62: 802-815.

Valiente-Banuet, A., A.V. Rumebe, and V. Miguel. 2006. "Modern Quaternary plant lineages promote diversity through facilitation of ancient Tertiary lineages." *Proceedings of the National Academy of Sciences of the United States of America* 103: 16812-16817.

van de Poll, W.H., G. Kulk, K.R. Tmmermans, C.P.D. Brussard, H.J. van der Woerd, M.J. Kehoe, K.D.A. Mojica, R.J.W. Visser, P.D. Rozema, and A. G. J. Buma. 2013. "Phytoplankton chlorophyll a biomass, composition, and productivity

along a temperature and stratification gradient in the northeast Atlantic Ocean." *Biogeosciences* 10: 4227-4240.

Weissing, F. J., and J. Huisman. 1994. "Growth and competition in a light gradient." *Journal of Theoretical Biology* 168: 323-336.

Whittaker, R.H. 1972. "Evolution and measurement of species diversity." *Taxon* 21: 213-251.

APPENDIX A

The following is a summary table of symbols, what variable they represent, values used, and units.

Parameters		Value	Units
ϕ	Population density	0.1 (initial)	$10^6 c$ /L
ϕ_b	Population density in layer below	0.1 (initial)	$10^6 c$ /L
ϕ_u	Population density in layer above	0.1 (initial)	$10^6 c$ /L
ϕ_L	Population density from land inflow (advection)	0	$10^6 ce$ /L
ϕ_O	Population density off shore (Eddy diffusion)	8 (van de Poll et al., 2013)	$10^6 c$ /L
S	Nutrient concentration	10 (initial)	μ
S_b	Nutrient concentration in layer below	10 (initial)	μ
S_u	Nutrient concentration in layer above	10 (initial)	μ
S_L	Nutrient concentration from land inflow	10	μ
S_O	Nutrient concentration off shore	0.01	μ
S_B	Benthic flux rate	0, 0.1 10, 100 (Nowicki and Nixon, 1985)	µM day^{-1}
μ	Specific growth rate		da^{-1}
μ_m	Maximum specific growth rate	1	d^{-1}
s	Sinking rate	-1.992, 0, 0.1, 1.008 (Huisman and Sommeijer, 2002)	m/d

v_b	Vertical mixing with layer below	0, 0.001, 0.01, 0.1 (Roelke and Edlridge, 2008)	m/d
v_u	Vertical mixing with layer above	0, 0.001, 0.01, 0.1 (Roelke and Edlridge, 2008)	m/d
f_i /f_o	Flow rate	0.25 (Roelke and Edlridge, 2008)	m^3/d
V	Volume of layer	1	m^3
D	Horizontal mixing rate	0, 0.1, 0.01, 0.001	da^{-1}
c	Cell quota of resource	Varies	$\mu \times 10^6 c_i$
a_i	Phytoplankton group slope of the photosynthesis-irradiance curve	2.86 x 10^{-17}, $4.59x10^{-1}$, 6.32 x 10^{-17} (Platt et al., 1982; Platt, 1986)	c_i $^2 s$ q ^{-1}d $^{-1}$
I_0	scalar irradiance incident on the surface	$4.34x10^{16}$ (Moigis, 1999)	q c_i $^{-2}s^{-1}$
K_W	Extinction of light based on pure water, tripton, and CDOM	0.146 (Olivieri and Chavez, 2000)	m^{-1}
K_C	Extinction of light based on the total concentration of phytoplankton	0.024 (Olivieri and Chavez, 2000)	$m^2(m\ chl\ \alpha)^{-1}$
K_S	Half-saturation coefficient	Varies (Roelke and Eldridge, 2008) See Appendix C	
c_i	Fixed cellular content for a given substrate	Varies (Roelke and Eldridge, 2008) See Appendix C	
P_i	Proportion of biomass of species i to the total biomass		

APPENDIX B

The following box and whisker plots are the graphical representation of the
primary analysis for our results. There is a figure for   richness,   diversity,   richness,
and   diversity for all five scenarios plus the simulation added to test the significance of
adding vertical mixing to scenario two. These were run with the same nutrient flux rates
with a simultaneous vertical mixing rate of $v_u = v_D = 0.01$ day^{-1}.

In each figure, there are four box and whisker plots: one for the various
simulations all combined, one for the combined neutral simulations, one for the
combined lumpy simulations, and one for the combined RPS assemblages. These were
calculated by first running an ANOVA test with the data and then followed up with the
post hoc Tukey Kramer test to find significant differences. To understand which data is
significantly the same within each box and whisker plot, observe the letter labels above
the box and whisker for each rate in a given figure. For two columns that have the same
letter label, they are statistically the same.

Figure B-1.

Sensitivity of α richness to light

Figure B-2.

Sensitivity of α diversity to light

45

Figure B-3.

Sensitivity of β richness to light

Figure B-4.

Sensitivity of β diversity to light

46
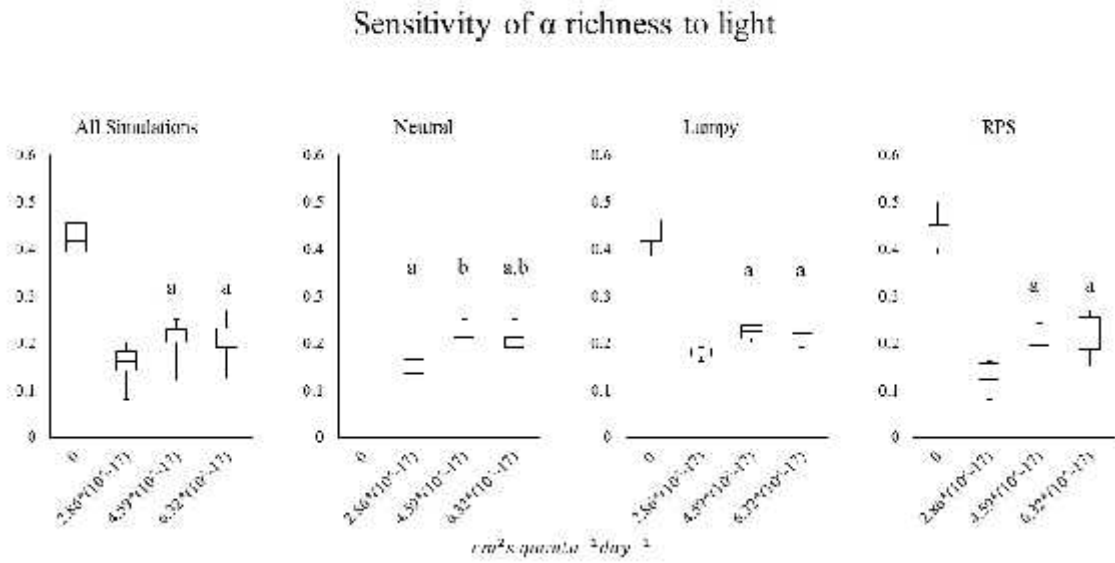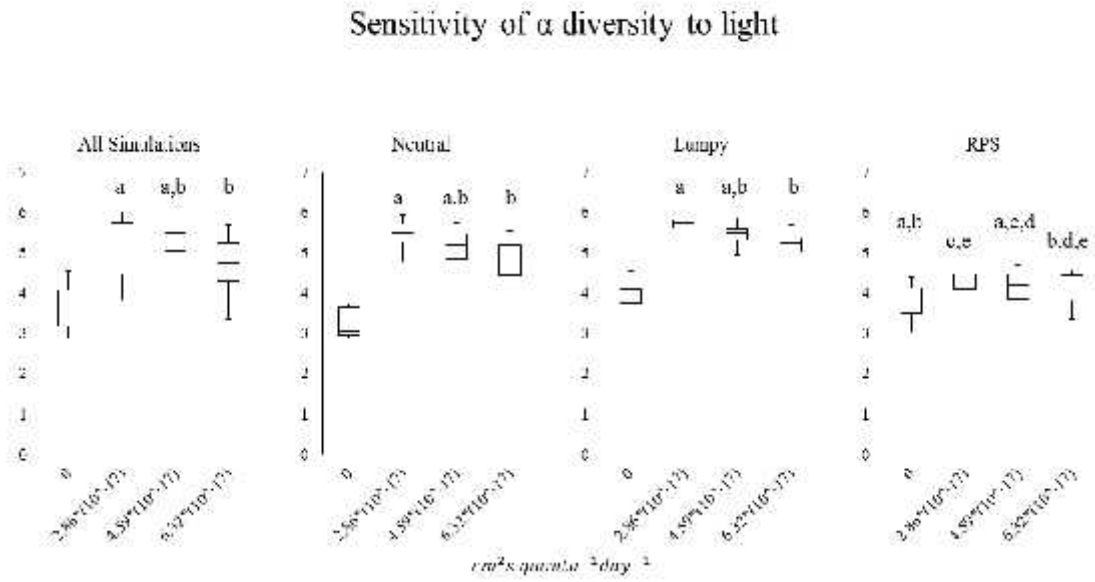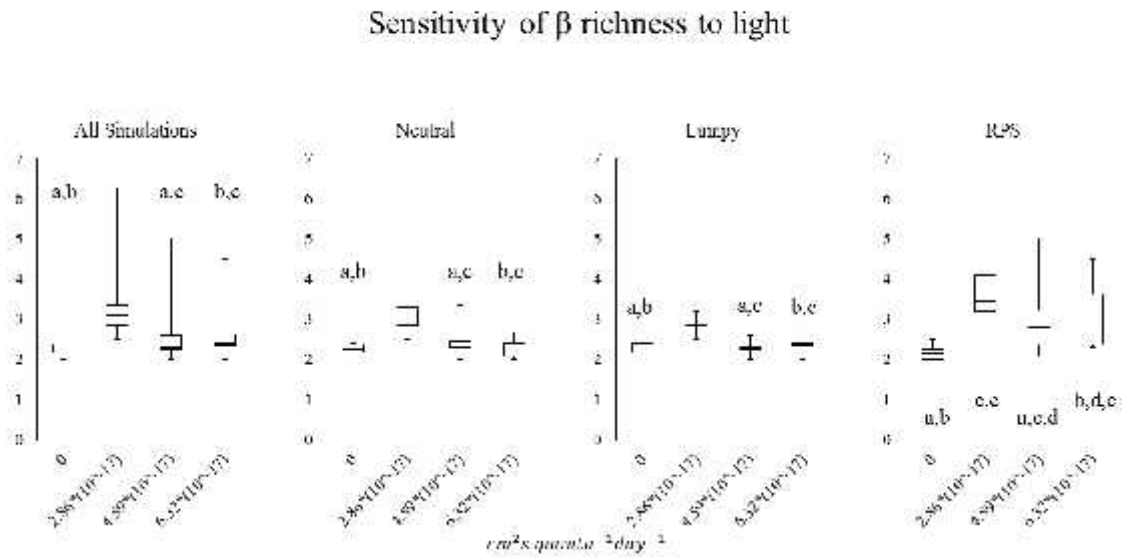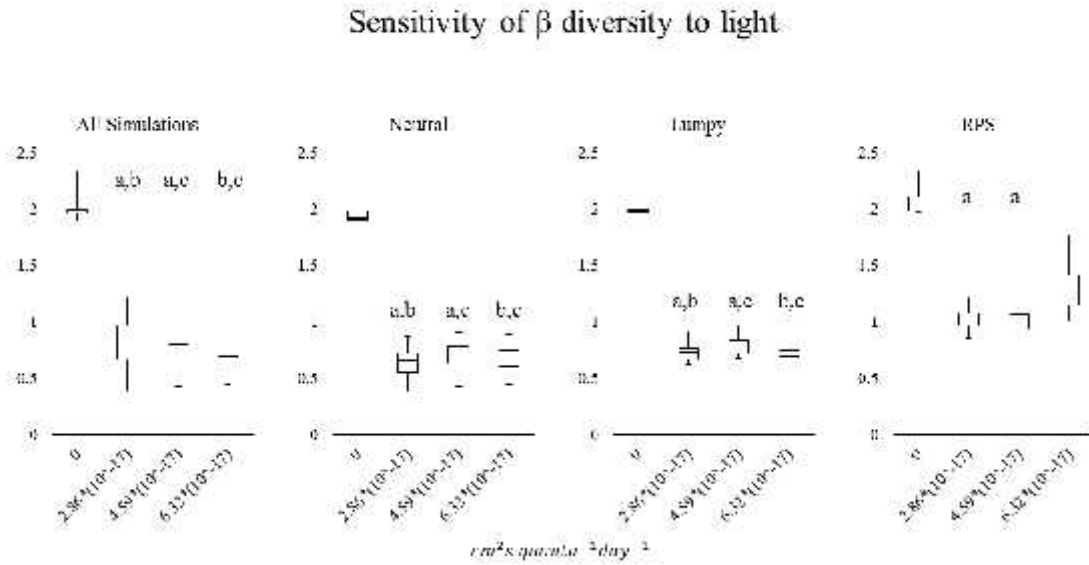
Figure B-5.

Sensitivity of α richness to nutrient flux
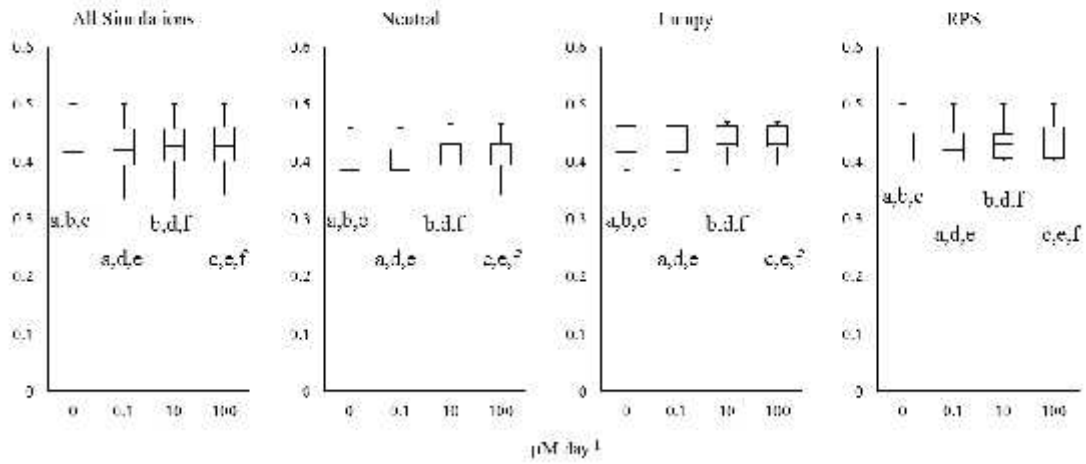
Figure B-6.

Sensitivity of α diversity to nutrient flux
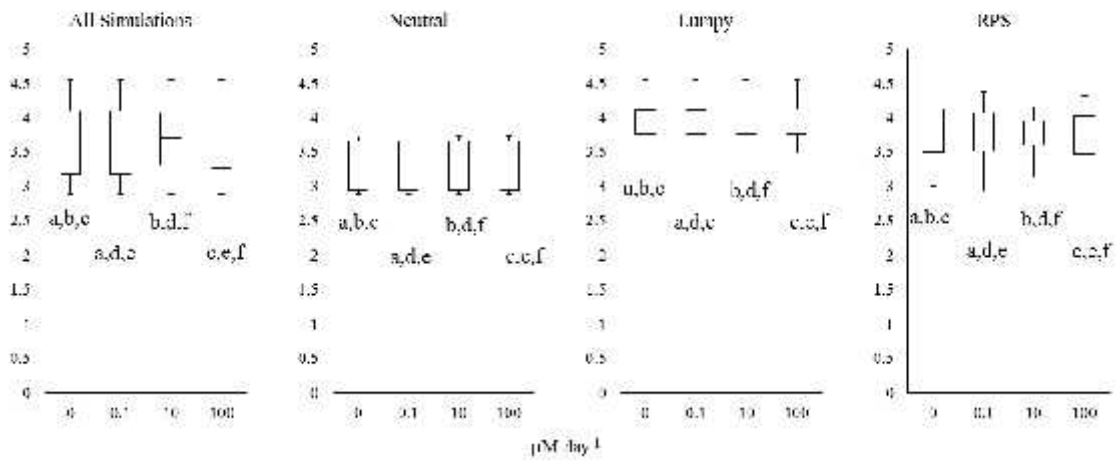
Figure B-7.

Sensitivity of β richness to nutrient flux
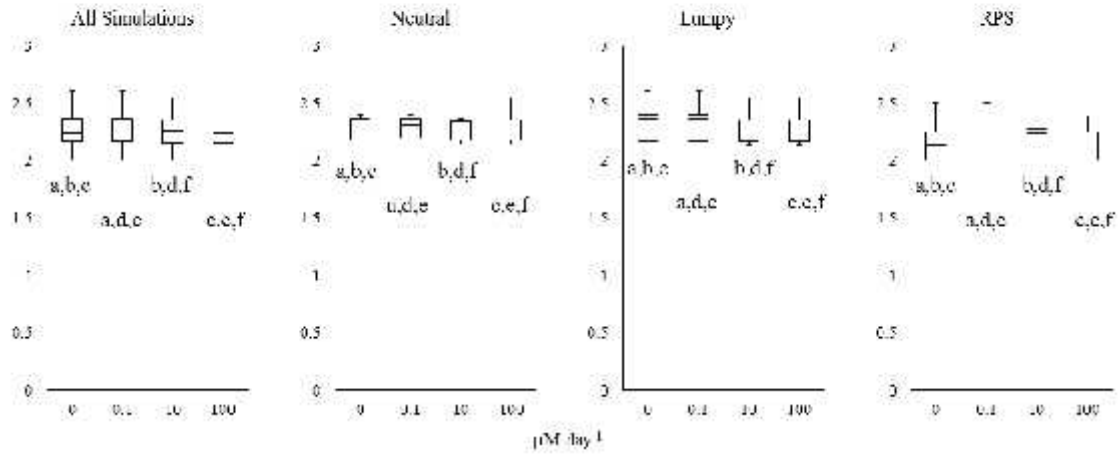
Figure B-8.

Sensitivity of β diversity to nutrient flux
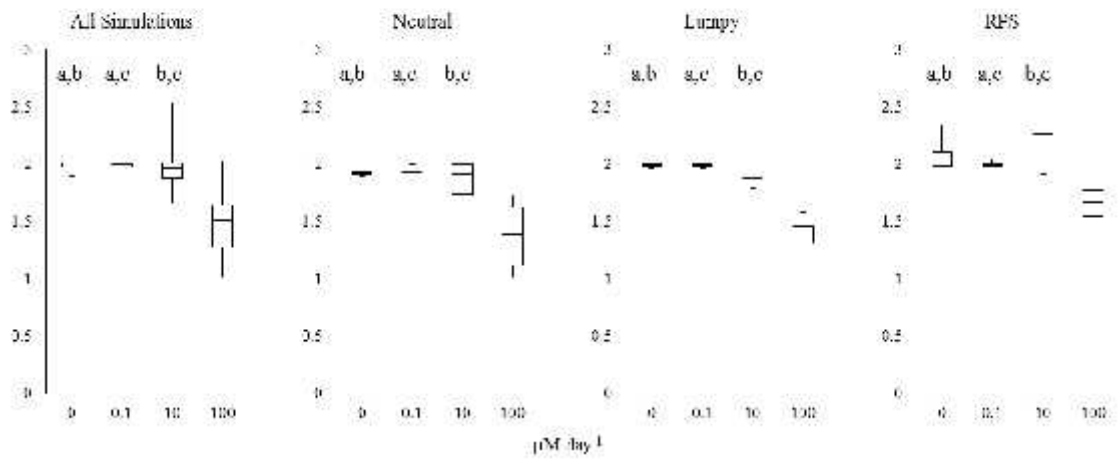
Figure B-9.

Sensitivity of α richness to nutrient flux with vertical mixing
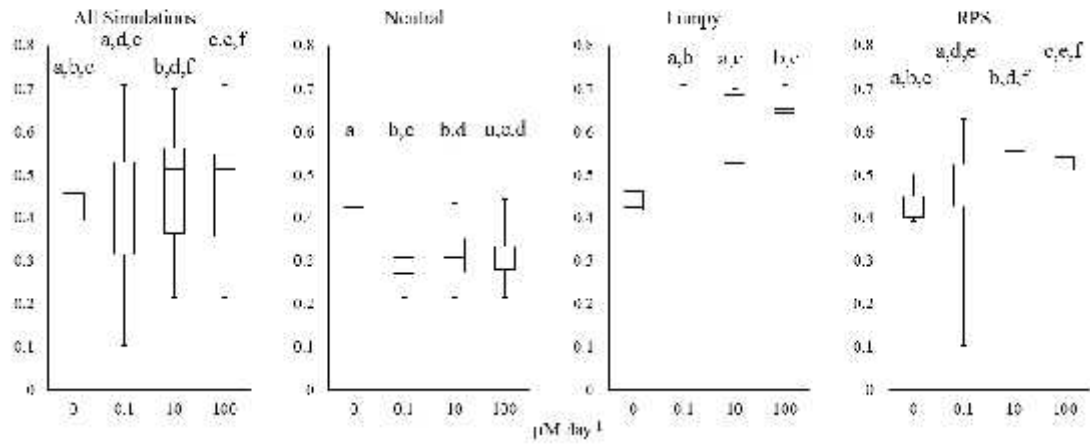
Figure B-10.

Sensitivity of α diversity to nutrient flux with vertical mixing

49
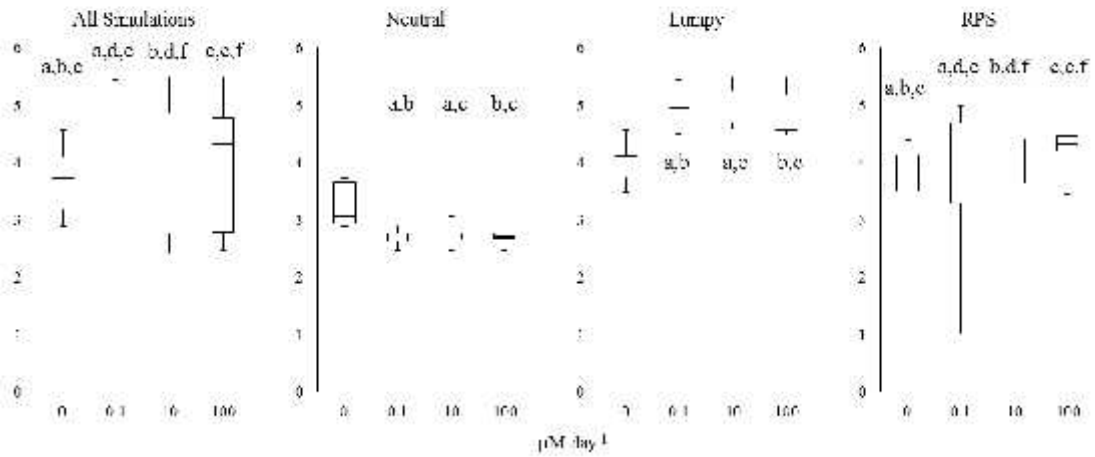
Figure B-11.

Sensitivity of β richness to nutrient flux with vertical mixing
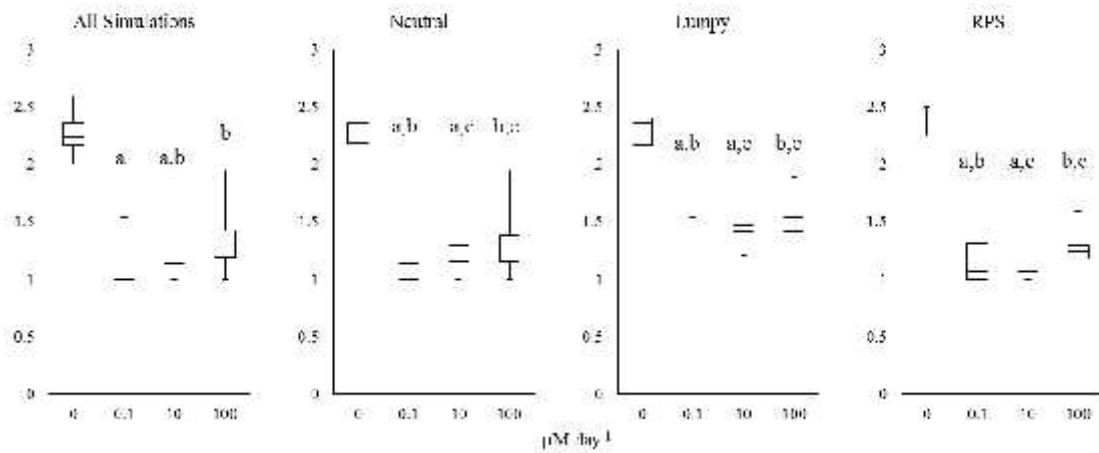
Figure B-12.

Sensitivity of β diversity to nutrient flux with vertical mixing
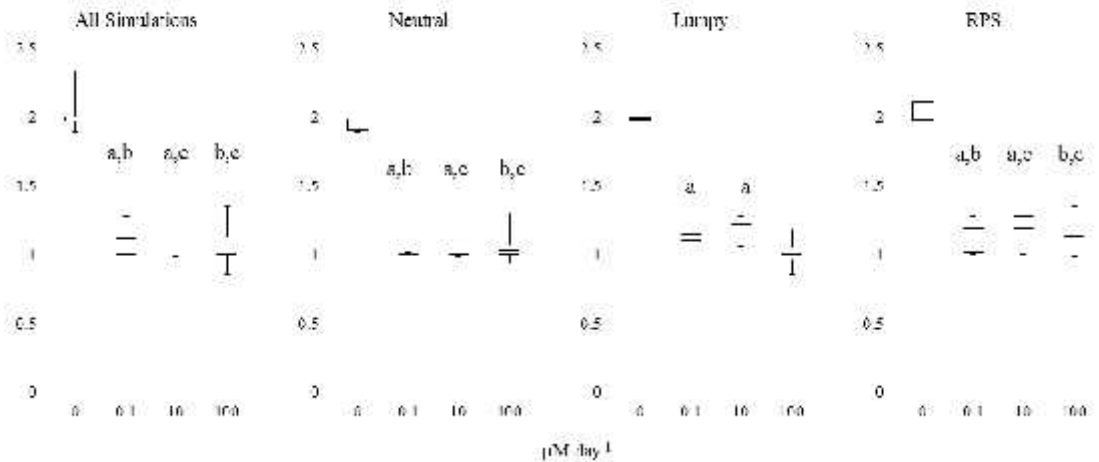
Figure B-13.

Sensitivity of α richness to vertical mixing
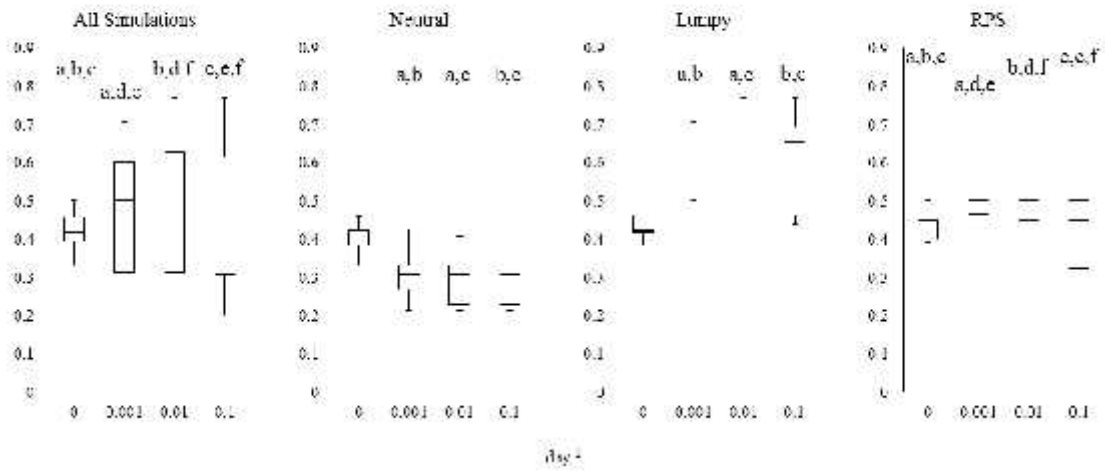
Figure B-14.

Sensitivity of α diversity to vertical mixing

51
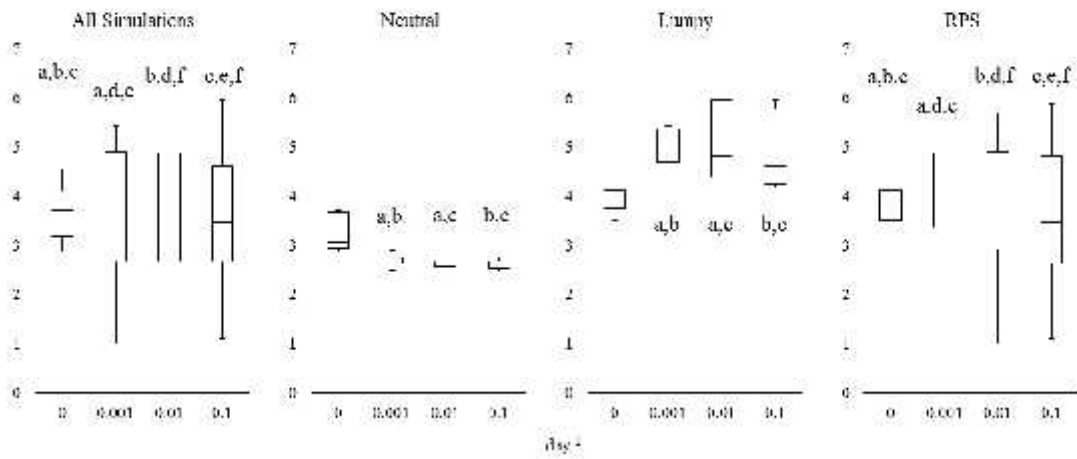
Figure B-15.

Sensitivity of β richness to vertical mixing
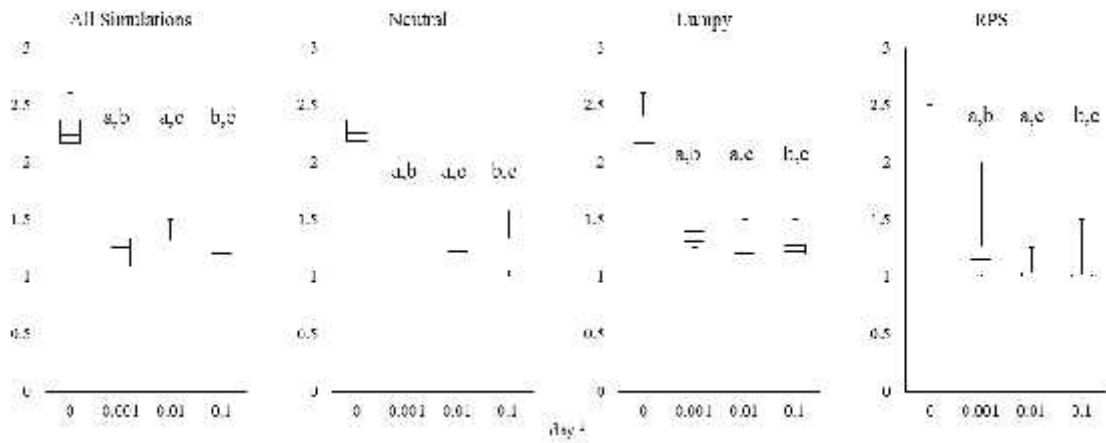
Figure B-16.

Sensitivity of β diversity to vertical mixing

52

Figure B-17.

Sensitivity of α richness to sinking

Figure B-18.

Sensitivity of α diversity to sinking

53
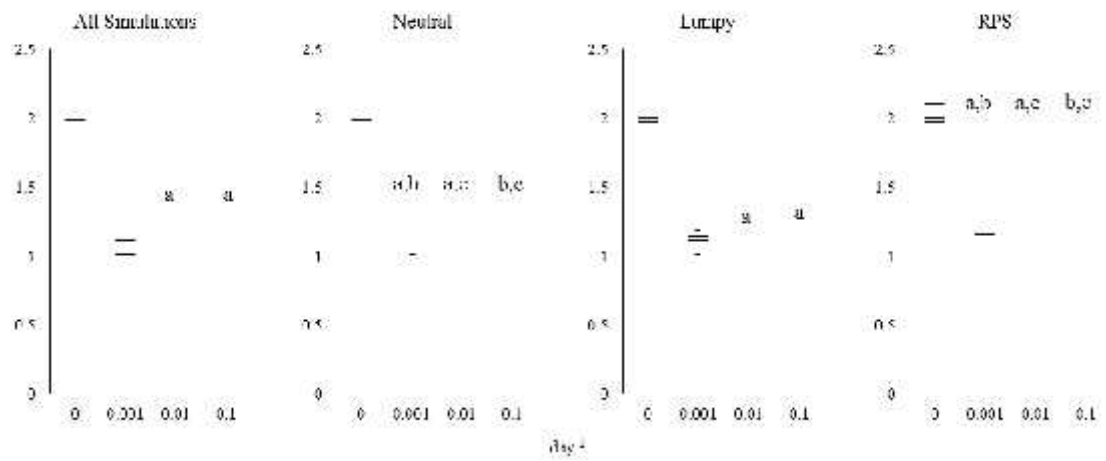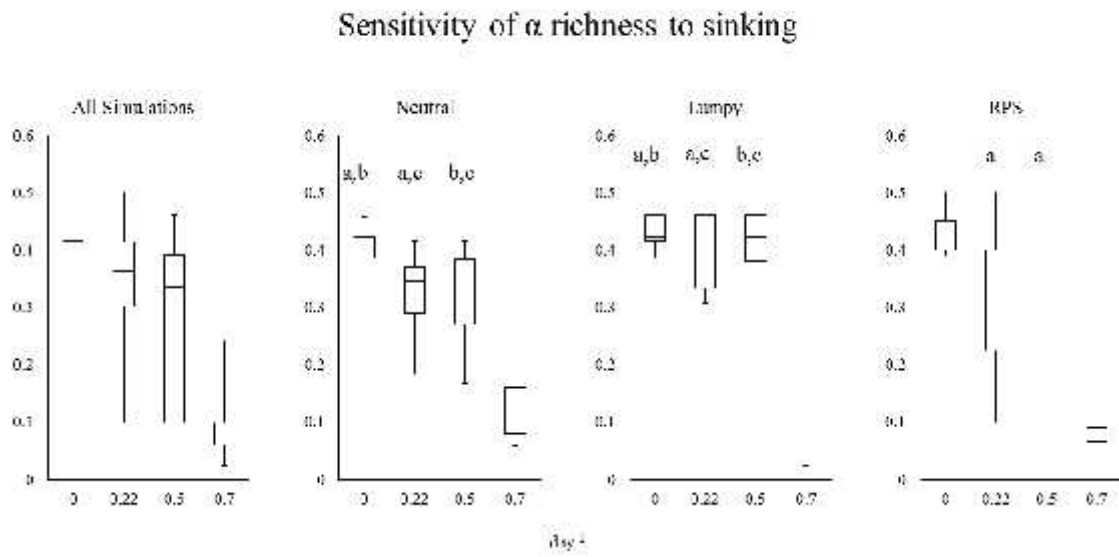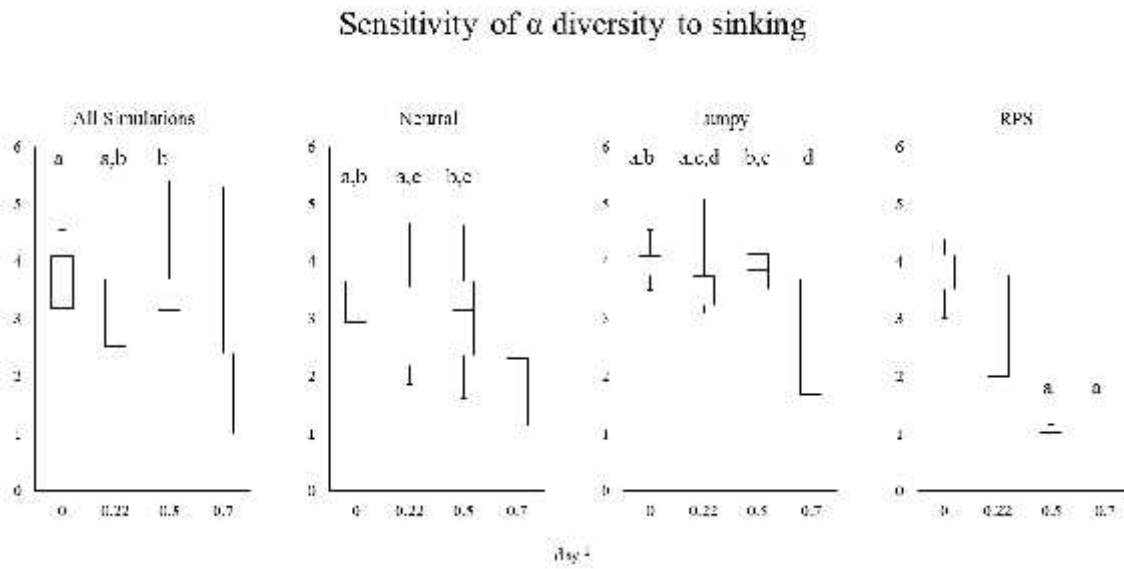
Figure B-19.

Sensitivity of β richness to sinking
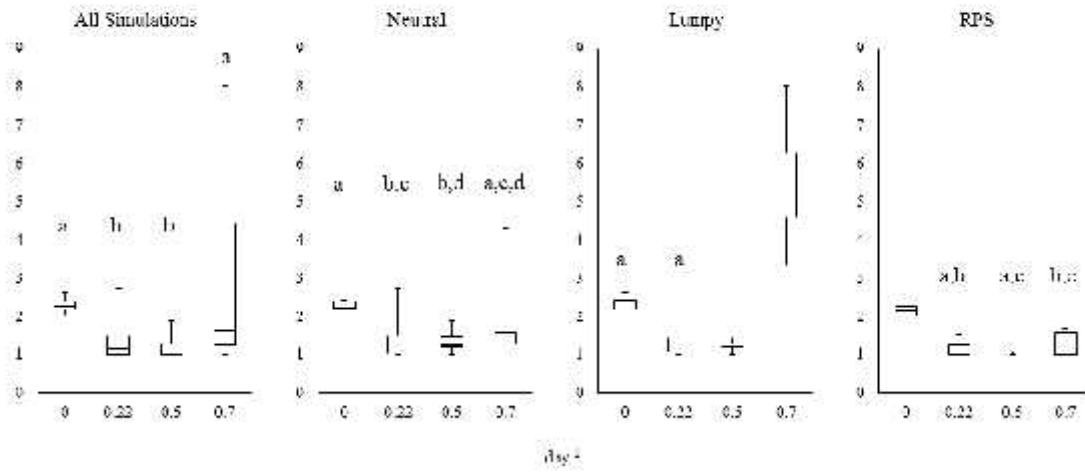
Figure B-20.

Sensitivity of β diversity to sinking

54
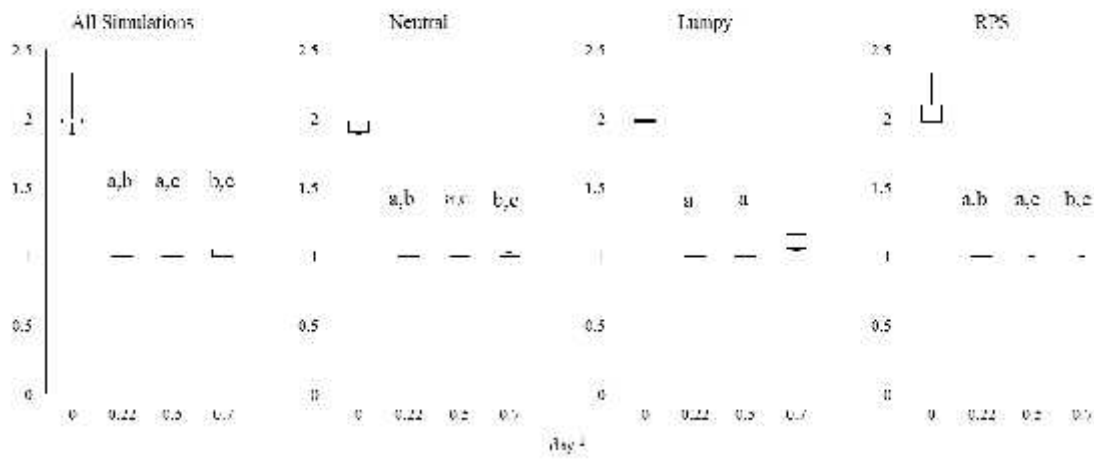
Figure B-21.

Sensitivity of α richness to eddy diffusion
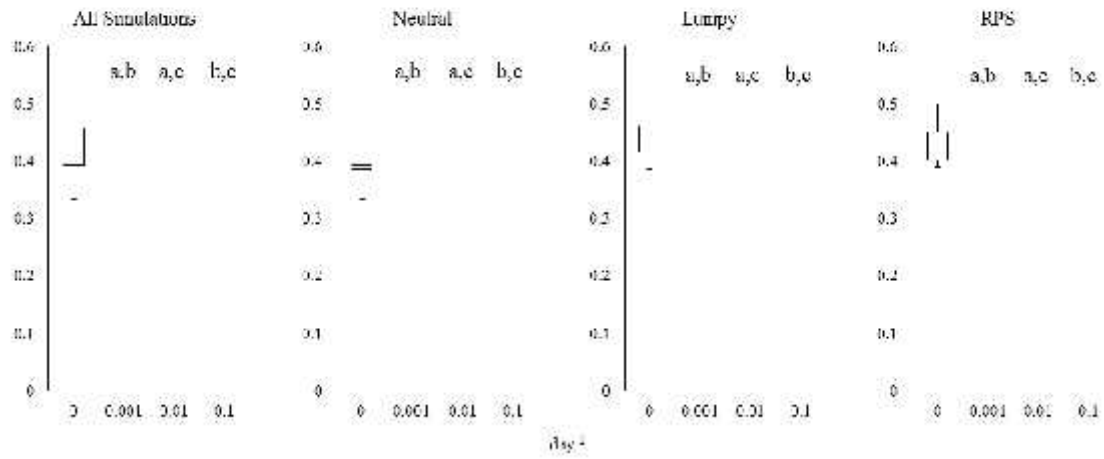
Figure B-22.

Sensitivity of α diversity to eddy diffusion

55
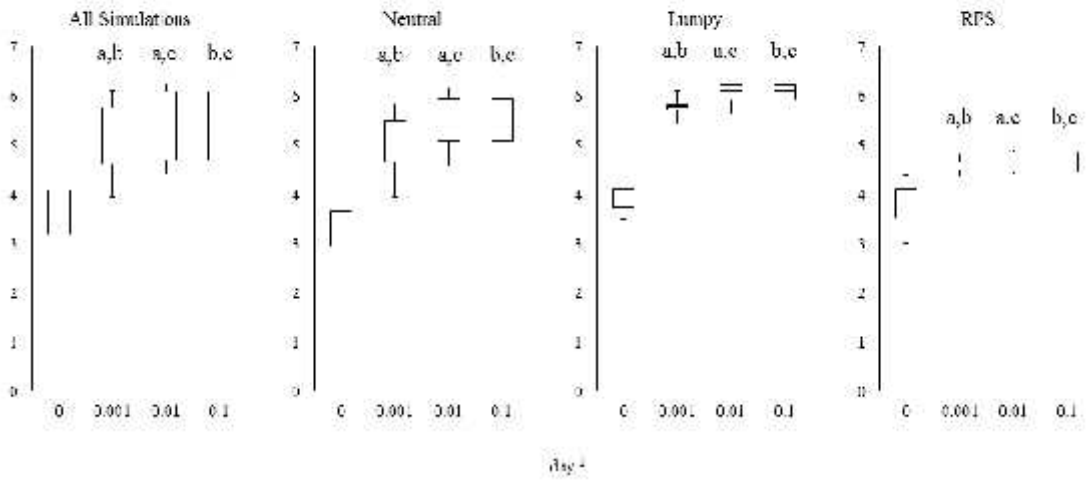
Figure B-23.

Sensitivity of β richness to eddy diffusion
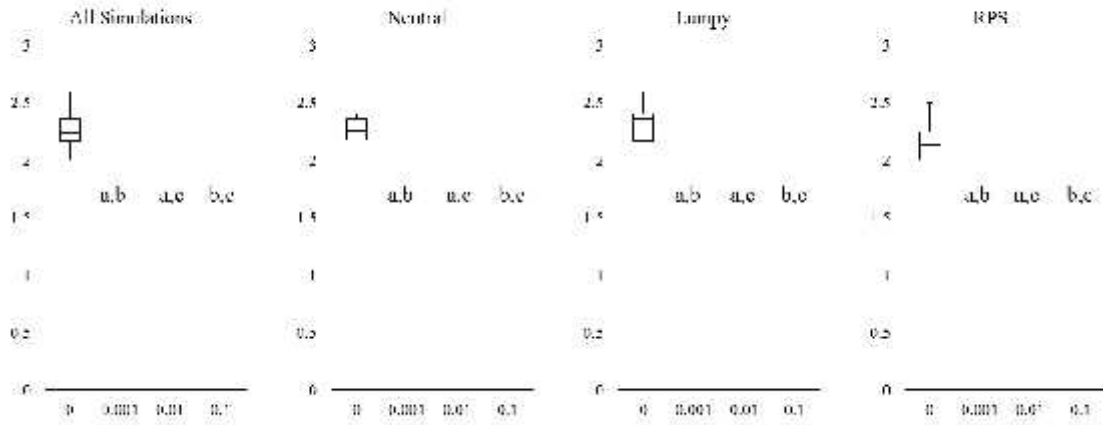
Figure B-24.

Sensitivity of β diversity to eddy diffusion

56
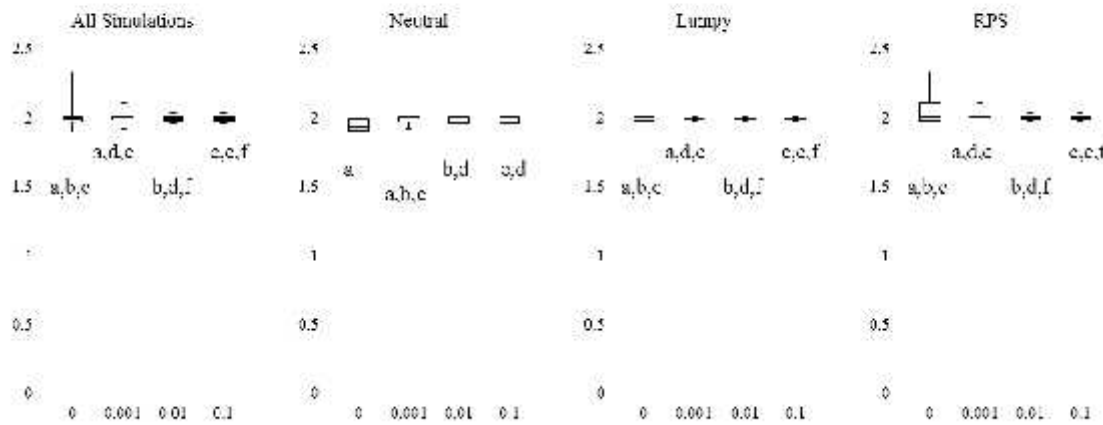
The following code was used to run all simulations. It will start with the main file followed by all of the additional functions used throughout the main file.

```matlab
%Written by Frances Withrow February 2014


% This program simulates multiple phytoplankton competing for multiple
% resources in a ten layer water column.  New nutrients enter the system
% from a "land source".  Cells and nutrients are lost from
% the system via turbulent mixing at the same rate as the inflow.
% Parameterization of phytoplankton groups is the same as Huisman's
% Am. Nat. paper, May 2001.


global r  K  C  mark  Sk Tb Kw  Kc irrinit CarCell CarChl lswitch  ...
    alphaset vect area V depth ntrflux flow sourcephyto ...
    sourcentr D oceanphyto oceanntr totlength lengthK1 lengthK2


set1 = [1 2 3 4 5 6 7 8 9 10];
set2 = [2 1 4 3 6 5 8 7 10 9];

totalpharich = [];
totalphaeven = [];
totalphadiv = [];
totgammarich = [];
totgammaeven = [];
totgammadiv = [];
totlayerbiomass = [];
justfortherecord = 1;
totspecies = [];


 for assemblage = 3:3
   for track = 9:9



%-----------------------------------------------------------------------
% PARAMETER VALUES
```

```matlab
%------------------------------------------------------------------------

%--------------------------------------------------------
%A few constants
%--------------------------------------------------------
%  Maximum specific growth rate, d-1
                        %Maximum specific growth rate in Roelke
                        %2008 r = 1
r    = 1;               % Maximum specific growth rate
depth = 1;               % depth of each interval
vect  = (1:10)*depth;       % These are the depths of the boxes
area  = (depth)^2;        % area of each layer's box
V    = (depth)^3;         % volume of each layer's box




%--------------------------------------------------------
%Vertical mixing
%--------------------------------------------------------
% Vertical mixing, must be in units of d-1
% These are options to create mixing at only certain
%     points in the column. This paper focused on
%     uniform mixing
% Tb(1)  = mix between Box 1 & Box 2
% Tb(2)  = mix between Box 2 & Box 3 etc
% Tb(10) = mix between Box 10 & sink

%Various possible vertical mixing rates
mix = 0;  % 0, .001, .01, .1

%Turning on/off the various options
uniform_switch   = 1;
twometer_switch  = 0;
fivemeter_switch = 0;

if uniform_switch == 1
   Tb = mix*ones(1,10);
end

if twometer_switch == 1
   Tb = mix*[1 0 1 1 1 1 1 1 1 1];
end

if fivemeter_switch == 1
```

```matlab
    Tb = mix*[1 1 1 1 0 1 1 1 1 1];
end


%---------------------------------------------------------
% Light
%---------------------------------------------------------
lswitch = 0;              % Self-shading switch (0=off, 1=on)
initcarcell = 5.5;           % carbon per cell; consistent with
                    % values in Mullin 1966
CarCell = initcarcell*12;   % convert to units of ug-C/(x10^6 cells)
CarChl  = 100;             % Carbon to Chlorophyll ratio (mg:mg),
                    % assuming ~25°N latitude
                    % Taylor 1997

%Need to do a sensitivity analysis for light sensitivity by adjusting
%alpha. Values taken from Nowicki_1985

%alpha = 4.59*(10^-17);
%alpha = 2.86*(10^-17);
%alpha = 6.32*(10^-17);


%Kw     = .038;        % Factors Contributing to Water-Column
                 % Light Attenuation SAV TECHNICAL SYNTHESIS II
Kw     = 0.146;        % Light attenuation based on water, CDOM, and
                 % tripton (/m), see Olivieri and Chavez 2000)
%Kw     = 0;           % turn off
Kc     = 0.024;        % Light attenuation based on chla (m2/mg-chla),
                 % see Olivieri and Chavez 2000)
%Kc     = 0.006;       % one fourth of literature value (results in
                 % K=0.41 when A=40)
                 % Roelke 2000  cm^2 s quanta^-1 day^-1
alpha = 2.86*(10^-17);
alphaset = alpha*ones(1,20);
irrinit = 4.34*10^16;   %Moigis 1999  quanta cm^-2 s^-1
                 %Had to convert micromol m^-2 s^-1.
                 %6*10^23 quanta= 1 mole
Kt     = Kw + Kc;      %Extinction coefficient


%---------------------------------------------------------
%LAND (Always turned on)
%---------------------------------------------------------
```

```matlab
%Land inflow rate
flow    = .25*ones(1,10);    % inflow m^3/day

%Concentration of phytoplankton from land. In this case it will always be
%zero
% Concentration of cells from inflow source cells/L
sourcephyto = zeros(1,20);

%Land source nurtrient concentration

%This is for easy adjustment if necessary
N      = 10;            % nitrogen source, microM
P      = 10;            % phosphorus source, microM
S      = 10;            % silica source, microM

% concentration of ntr from inflow source micromoles/L
sourcentr   = [N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S; ...
           N P S;];




%-----------------------------------------------------------
%Nutrient Flux
%-----------------------------------------------------------
%tenth layer benthos flux micromolar/day  N,P, S
%in the differential equations, this is multiplied by the area
%(in this case 1 m^2) to get the benthix flux rate
ntrflux_switch = 0;

%(a) fluxrate = .1;
%(b) fluxrate = 10;
%(c) fluxrate = 100;
%Information from Nowicki 1985

fluxrate = .1;
```

```matlab
if ntrflux_switch == 1
    ntrflux = fluxrate*ones(1,3);
end
if ntrflux_switch == 0
    ntrflux = 0*ones(1,3);
end


%--------------------------------------------------------
%Sinking
%--------------------------------------------------------

%  Sinking rate (each box is 1 meter in the vertical, so final units
%  for sinking rate are d^-1)

Sk_rate = ,22;
%.22, .5, .7 (Huisman and Sommeijer 2002)


sinking_switch =0;

if sinking_switch == 1
    Sk = Sk_rate*ones(1,20);
end
if sinking_switch == 0
    Sk = zeros(1,20);
end



%%----------------------------------------------------------------------
%  ASSEMBLAGES
%%----------------------------------------------------------------------

%In assemblage_func the inputs are the theoretical assemblage,
%and then the two groups within those subsets. Order matters for
%when we switch what species are found where

%1 is neutral
%2 is lumpy
%3 is Rock Paper Scissors (RPS)
]

[K1,K2,C1,C2] = assemblage_func(assemblage,set1(track),set2(track));
%Because of the capacity for 20 species, we need to make sure
%that there are 20 half saturation coefficients and 20 cell contents
```

```matlab
totlength = (length(K1) + length(K2));
leftover = 20 - totlength;
Kzeros = zeros(3,leftover);



K = [K1 K2 Kzeros];
C = [C1 C2 Kzeros];


%-------------------------------------------------------
%OCEAN MIXING
%-------------------------------------------------------

%If ocean mixing is on, ocean_switch = 1
%If ocean mixing is off, ocean_switch = 0
% a D_rate = .1
% b D_rate = .01
% c D_rate = .001


ocean_switch =0;
D_rate = .01;
if ocean_switch == 1

%Eddy diffusion rate 1/day
D = D_rate*ones(1,10);

% concentrations of phytoplankton in ocean will be variable based on
% species present without ocean mixing
oceanphytoones1 = ones(1,length(K1));
oceanphytozeros1 = zeros(1,20-length(K1));
oceanphyto1 = [oceanphytoones1 oceanphytozeros1];

oceanphytoones2 = ones(1,length(K2));
oceanphytozeros2 = zeros(1,length(K1));
oceanphytozeros3 = zeros(1,leftover);
oceanphyto2 = [oceanphytozeros2 oceanphytoones2 oceanphytozeros3];

oceanphyto = 8*[oceanphyto1; oceanphyto1; oceanphyto1; oceanphyto1; ...
    oceanphyto1; oceanphyto2; oceanphyto2; oceanphyto2; oceanphyto2;...
    oceanphyto2];

%Open ocean nutrient concentration micromoles/L
oceanntr = [.01 .01 .01; ...
        .01 .01 .01; ...
```

```matlab
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01; ...
        .01 .01 .01];
else if ocean_switch == 0
    D = zeros(1,10);
    oceanphyto = zeros(10,20);
    oceanntr = zeros(10,3);
  end
end
%-----------------------------------------------------------------------
% INITIAL CONDITIONS
%-----------------------------------------------------------------------

%  Box 1 through 10, initial resource concentration, microM
Rinit = [N, P, S];
Rin = [Rinit Rinit Rinit Rinit Rinit Rinit Rinit Rinit Rinit Rinit];

%Now going to use functions to distribute the initial populations
%'Block' refers to the first assemblage is in the top 5 patches and
%the second assemblage is in the bottom 5 patches

initconc = .1;

lengthK1 = length(K1);
lengthK2 = length(K2);

% Box 1 through 10 initial phytoplankton concentrations. units are 10^6
% cells/L and default condition is .1
%
Ain = initconc*phyto_init_func(lengthK1,lengthK2);




%-----------------------------------------------------------------------
% DO THE MODEL RUN
%-----------------------------------------------------------------------

t0    = 0;                % Julian day of start
```

```matlab
tfinal  = 3000;              % Julian day of finish
tspan = [t0:.1:tfinal];
mark    = t0;
zin     = [ Ain, Rin, 0 ];


[t,z]   = ode45('differential',tspan,zin);




%-------------------------------------------------------------------------
%Computing richness and evenness
%-------------------------------------------------------------------------
[alpharich, alphaeven, alphadiv] = alphacalc(assemblage,z);
[gammarich,gammaeven, gammadiv] = gammacalc(assemblage,z);
totbiomass = biomasscalc(assemblage, z);

totalpharich = [totalpharich alpharich];
totalphaeven = [totalphaeven alphaeven];
totalphadiv = [totalphadiv alphadiv];
totgammarich = [totgammarich gammarich];
totgammaeven = [totgammaeven gammaeven];
totgammadiv = [totgammadiv gammadiv];
totlayerbiomass = [totlayerbiomass; totbiomass];

justfortherecord = justfortherecord + 1;


ztotal = [ztotal; z];
```

```matlab
function [phyto_init]=phyto_init_func(x,y)

%Block formation

    lengthK1 = x;
    zerosK1 = 20 - x;
    valuesvectK1 = ones(1,lengthK1);
    zerosvectK1 = zeros(1,zerosK1);
    toplayer = [valuesvectK1 zerosvectK1];
    phyto_init_top = [toplayer toplayer toplayer toplayer toplayer];

    lengthK2 = y;
    zerosK2 = 20 - (x + y);
    valuesvectK2 = ones(1,lengthK2);
    beginK2 = zeros(1,lengthK1);
    zerosvectK2 = zeros(1,zerosK2);
    nextlayer = [beginK2 valuesvectK2 zerosvectK2];
    phyto_init_bottom = [nextlayer nextlayer nextlayer nextlayer nextlayer];

    phyto_init = [phyto_init_top phyto_init_bottom];
```

```
function [K1,K2,C1,C2]=assemblage_func(a,b,c)

if a == 1
  if b == 1
    K1 = [.5582 .5443 .3857 .5693 .3631 .5729;
        .4368 .6451 .6367 .5494 .6389 .5473;
        .6031 .3835 .6040 .4232 .5738 .4466];
    C1 = [.6051 .5524 .4016 .5789 .3786 .5757;
        .4378 .6647 .6667 .5895 .6495 .5714;
        .6416 .4095 .6174 .4483 .6113 .4788];
  else if b == 2
    K1 = [.3423 .5871 .3216 .4245 .4814 .5940 .4790;
        .6796 .7218 .7211 .7263 .5848 .4440 .5736;
        .5524 .2515 .5521 .3955 .5524 .5348 .5490];
    C1 = [.3890 .6226 .3644 .4306 .4904 .6122 .5287;
        .6907 .7686 .7548 .7294 .6080 .4595 .5780;
        .5932 .2883 .5730 .4117 .6005 .5609 .5692];
    else if b == 3
    K1 = [.6941 .4510 .6352 .4770 .5121 .5347 .6736;
        .4691 .5332 .5395 .4828 .5468 .5444 .5261;
        .3805 .6300 .3948 .6326 .5287 .4770 .3636];
    C1 = [.7167 .4721 .6539 .4907 .5387 .5803 .7111;
        .4953 .5625 .5668 .4941 .5585 .5621 .5546
        .4105 .6530 .4279 .6570 .5771 .4787 .3884];
      else if b == 4
    K1 = [.5283 .4373 .6427 .6578 .4759 .6631;
        .4745 .5624 .5617 .3450 .5529 .4184;
        .5680 .5573 .3655 .5608 .5667 .4480];
    C1 = [.5605 .4572 .6743 .6954 .5210 .6711;
        .5194 .5982 .5897 .3833 .5763 .4258;
        .5915 .5977 .3687 .5999 .5692 .4965];
        else if b == 5
    K1 = [.3772 .5977 .5759 .4364 .4289 .5959;
        .5685 .5441 .5647 .4389 .4918 .5417;
        .6160 .3962 .4108 .6719 .6775 .4712];
    C1 = [.4087 .6206 .6095 .4568 .4634 .6395;
        .6079 .5707 .6052 .4633 .5111 .5540;
        .6210 .4240 .4263 .6929 .7048 .5070];
            else if b == 6
```

66

```
K1 = [.4256 .4264 .6071 .6318 .5077 .6321;
      .5476 .5300 .5648 .4371 .5704 .4463;
      .6166 .6165 .4096 .5121 .4745 .5105];
C1 = [.4468 .4655 .6331 .6347 .5172 .6631;
      .5609 .5692 .6035 .4793 .6053 .4931;
      .6450 .6304 .4567 .5489 .4785 .5134];
          else if b == 7
K1 = [.3871 .5291 .5153 .5740 .5706 .5375;
      .5696 .3754 .5680 .4919 .5196 .3504;
      .6235 .6607 .4671 .5443 .4763 .6558];
C1 = [.3877 .5428 .5190 .6175 .5715 .5710;
      .6170 .4007 .5736 .5185 .5648 .3598;
      .6355 .6772 .4708 .5754 .5075 .6599];
              else if b == 8
K1 = [.5139 .5078 .4199 .3389 .4668 .5917 .5896 .5470;
      .5067 .5450 .6704 .6526 .6657 .5967 .4501 .4362;
      .5821 .5868 .4954 .5838 .4630 .4205 .5323 .5879];
C1 = [.5155 .5382 .4690 .3530 .4905 .6330 .6122 .5838;
      .5529 .5865 .7083 .6939 .6843 .6356 .4906 .4788;
      .5988 .6027 .5346 .6250 .5085 .4550 .5696 .6241];
              else if b == 9
K1 = [.7731 .6916 .6216 .4978 .7438 .5869;
      .3730 .3241 .5279 .5083 .3212 .3938;
      .4016 .5330 .4121 .5523 .4799 .5589];
C1 = [.8119 .7067 .6240 .5141 .7655 .6066;
      .3948 .3281 .5655 .5505 .3604 .4281;
      .4123 .5405 .4494 .5731 .5125 .5796];
                else if b == 10
K1 = [.3079 .5796 .5749 .5756 .4709 .3426;
      .6536 .4788 .5995 .4367 .4518 .6764;
      .6078 .4932 .4250 .5107 .6027 .5421];
C1 = [.3175 .5973 .6110 .5888 .4801 .3869;
      .6795 .4795 .6263 .4674 .4702 .7128;
      .6430 .5192 .4390 .5166 .6171 .5630];
                        end
                      end
                    end
                  end
                end
              end
            end
          end
        end
end
```

```
if c == 1
    K2 = [.5582 .5443 .3857 .5693 .3631 .5729;
          .4368 .6451 .6367 .5494 .6389 .5473;
          .6031 .3835 .6040 .4232 .5738 .4466];
    C2 = [.6051 .5524 .4016 .5789 .3786 .5757;
          .4378 .6647 .6667 .5895 .6495 .5714;
          .6416 .4095 .6174 .4483 .6113 .4788];
else if c == 2
    K2 = [.3423 .5871 .3216 .4245 .4814 .5940 .4790;
          .6796 .7218 .7211 .7263 .5848 .4440 .5736;
          .5524 .2515 .5521 .3955 .5524 .5348 .5490];
    C2 = [.3890 .6226 .3644 .4306 .4904 .6122 .5287;
          .6907 .7686 .7548 .7294 .6080 .4595 .5780;
          .5932 .2883 .5730 .4117 .6005 .5609 .5692];
    else if c == 3
    K2 = [.6941 .4510 .6352 .4770 .5121 .5347 .6736;
          .4691 .5332 .5395 .4828 .5468 .5444 .5261;
          .3805 .6300 .3948 .6326 .5287 .4770 .3636];
    C2 = [.7167 .4721 .6539 .4907 .5387 .5803 .7111;
          .4953 .5625 .5668 .4941 .5585 .5621 .5546
          .4105 .6530 .4279 .6570 .5771 .4787 .3884];
        else if c == 4
    K2 = [.5283 .4373 .6427 .6578 .4759 .6631;
          .4745 .5624 .5617 .3450 .5529 .4184;
          .5680 .5573 .3655 .5608 .5667 .4480];
    C2 = [.5605 .4572 .6743 .6954 .5210 .6711;
          .5194 .5982 .5897 .3833 .5763 .4258;
          .5915 .5977 .3687 .5999 .5692 .4965];
            else if c == 5
    K2 = [.3772 .5977 .5759 .4364 .4289 .5959;
          .5685 .5441 .5647 .4389 .4918 .5417;
          .6160 .3962 .4108 .6719 .6775 .4712];
    C2 = [.4087 .6206 .6095 .4568 .4634 .6395;
          .6079 .5707 .6052 .4633 .5111 .5540;
          .6210 .4240 .4263 .6929 .7048 .5070];
                else if c == 6
    K2 = [.4256 .4264 .6071 .6318 .5077 .6321;
          .5476 .5300 .5648 .4371 .5704 .4463;
          .6166 .6165 .4096 .5121 .4745 .5105];
    C2 = [.4468 .4655 .6331 .6347 .5172 .6631;
          .5609 .5692 .6035 .4793 .6053 .4931;
          .6450 .6304 .4567 .5489 .4785 .5134];
                    else if c == 7
    K2 = [.3871 .5291 .5153 .5740 .5706 .5375;
```

68

```matlab
            .5696 .3754 .5680 .4919 .5196 .3504;
            .6235 .6607 .4671 .5443 .4763 .6558];
        C2 = [.3877 .5428 .5190 .6175 .5715 .5710;
            .6170 .4007 .5736 .5185 .5648 .3598;
            .6355 .6772 .4708 .5754 .5075 .6599];
                    else if c == 8
        K2 = [.5139 .5078 .4199 .3389 .4668 .5917 .5896 .5470;
            .5067 .5450 .6704 .6526 .6657 .5967 .4501 .4362;
            .5821 .5868 .4954 .5838 .4630 .4205 .5323 .5879];
        C2 = [.5155 .5382 .4690 .3530 .4905 .6330 .6122 .5838;
            .5529 .5865 .7083 .6939 .6843 .6356 .4906 .4788;
            .5988 .6027 .5346 .6250 .5085 .4550 .5696 .6241];
                      else if c == 9
        K2 = [.7731 .6916 .6216 .4978 .7438 .5869;
            .3730 .3241 .5279 .5083 .3212 .3938;
            .4016 .5330 .4121 .5523 .4799 .5589];
        C2 = [.8119 .7067 .6240 .5141 .7655 .6066;
            .3948 .3281 .5655 .5505 .3604 .4281;
            .4123 .5405 .4494 .5731 .5125 .5796];
                        else if c == 10
        K2 = [.3079 .5796 .5749 .5756 .4709 .3426;
            .6536 .4788 .5995 .4367 .4518 .6764;
            .6078 .4932 .4250 .5107 .6027 .5421];
        C2 = [.3175 .5973 .6110 .5888 .4801 .3869;
            .6795 .4795 .6263 .4674 .4702 .7128;
            .6430 .5192 .4390 .5166 .6171 .5630];
                            end
                          end
                        end
                      end
                    end
                  end
                end
              end
            end
          end
end


%***************************************************
if a == 2
  if b == 1
    K1 = [.7100 .7084 .7049 .4673 .4553 .3130;
        .3860 .2779 .2788 .7089 .7156 .5249;
        .4869 .5532 .5706 .4041 .3804 .6889];
```

```
        C1 = [.7448 .7518 .7153 .4976 .4191 .3353;
              .4035 .3054 .3119 .7377 .7563 .5568;
              .5327 .6026 .5925 .4188 .5025 .7148];
    else if b == 2
      K1 = [.7126 .7099 .5585 .4859 .4349 .3749;
            .3391 .3952 .7032 .6974 .7017 .5079;
            .4951 .4379 .2985 .3728 .4211 .6872];
      C1 = [.7256 .7453 .5645 .5182 .4357 .4206;
            .3653 .4393 .7422 .7126 .7175 .5571;
            .5377 .4805 .3090 .4056 .4645 .7278];
      else if b == 3
      K1 = [.7246 .7241 .7205 .6988 .6340 .2711;
            .3896 .2768 .2137 .7042 .7013 .5935;
            .4577 .5586 .6479 .1494 .2036 .7172];
      C1 = [.7560 .7370 .7492 .7427 .6770 .2823;
            .4178 .3211 .2438 .7180 .7075 .6180;
            .4950 .5687 .6859 .1930 .2260 .7316];
        else if b == 4
      K1 = [.7334 .7397 .7348 .7223 .3880 .2080 .3125;
            .1758 .3813 .3400 .7377 .4376 .6577 .5226;
            .6921 .4017 .4973 .1232 .7171 .7153 .7159];
      C1 = [.7375 .7874 .7845 .7422 .4297 .2404 .3352;
            .1996 .4132 .3785 .7519 .4684 .6729 .5682;
            .6969 .4000 .4995 .1597 .7307 .7526 .7170];
          else if b == 5
      K1 = [.6972 .4144 .5651 .4119 .3252 .4030 .2868;
            .2411 .7153 .7107 .7170 .5389 .4344 .5700;
            .6575 .4042 .2665 .3953 .6966 .7025 .6975];
      C1 = [.7097 .4248 .6022 .4549 .3563 .4834 .3194;
            .2546 .7343 .7190 .7595 .5574 .4483 .5966;
            .6917 .4390 .3018 .4212 .7078 .7060 .7265];
            else if b == 6
      K1 = [.6957 .5440 .4721 .5267 .4054 .2079;
            .2992 .7182 .7250 .7224 .4296 .6366;
            .5927 .3069 .3905 .3300 .7258 .7225];
      C1 = [.7080 .5547 .4820 .5706 .4459 .2220;
            .3127 .7326 .7480 .7407 .4160 .6483;
            .5974 .3299 .4182 .3310 .7562 .7640];
              else if b == 7
      K1 = [.7160 .7158 .7139 .7095 .3386 .2579 .2788;
            .2637 .3077 .3144 .7140 .4861 .6108 .5827;
            .6027 .5676 .5245 .1632 .7203 .7274 .7235];
      C1 = [.7246 .7519 .7430 .7511 .3512 .2753 .2989;
            .2895 .3412 .3225 .7594 .5349 .6185 .6245;
```

```matlab
        .6138 .5988 .5711 .1845 .7667 .7704 .7299];
            else if b == 8
K1 = [.7180 .7151 .7184 .4358 .3365 .4093;
    .2999 .4117 .3945 .7015 .5418 .4587;
    .5805 .4497 .4865 .4307 .7226 .7259];
C1 = [.7575 .7625 .7423 .4427 .3785 .4245;
    .3322 .4425 .4110 .7391 .5548 .4683;
    .5865 .4653 .5306 .4326 .7452 .7321];
            else if b == 9
K1 = [.6964 .6019 .4676 .6067 .5108 .4041;
    .2994 .6899 .6913 .6871 .6847 .4383;
    .5454 .2580 .4257 .2502 .3323 .7199];
C1 = [.7325 .6123 .4903 .6555 .5472 .4443;
    .3388 .7340 .7013 .7370 .6905 .4519
    .5686 .2968 .4407 .2559 .3533 .7388];
            else if b == 10
K1 = [.7291 .7295 .7256 .5897 .6412 .3122 .2655;
    .2191 .1164 .3511 .6938 .6948 .5241 .5894;
    .6467 .7111 .5239 .2990 .2136 .7059 .7079];
C1 = [.7353 .7451 .7434 .6217 .6670 .3176 .2755;
    .2666 .1546 .3743 .7167 .7357 .5662 .5898;
    .6750 .7764 .5371 .3100 .2270 .7380 .7292];
                end
                end
              end
            end
          end
        end
      end
    end
  end
end
if c == 1
  K2 = [.7100 .7084 .7049 .4673 .4553 .3130;
    .3860 .2779 .2788 .7089 .7156 .5249;
    .4869 .5532 .5706 .4041 .3804 .6889];
  C2 = [.7448 .7518 .7153 .4976 .4191 .3353;
    .4035 .3054 .3119 .7377 .7563 .5568;
    .5327 .6026 .5925 .4188 .5025 .7148];
else if c == 2
  K2 = [.7126 .7099 .5585 .4859 .4349 .3749;
    .3391 .3952 .7032 .6974 .7017 .5079;
    .4951 .4379 .2985 .3728 .4211 .6872];
  C2 = [.7256 .7453 .5645 .5182 .4357 .4206;
```

```
    .3653 .4393 .7422 .7126 .7175 .5571;
    .5377 .4805 .3090 .4056 .4645 .7278];
else if c == 3
K2 = [.7246 .7241 .7205 .6988 .6340 .2711;
    .3896 .2768 .2137 .7042 .7013 .5935;
    .4577 .5586 .6479 .1494 .2036 .7172];
C2 = [.7560 .7370 .7492 .7427 .6770 .2823;
    .4178 .3211 .2438 .7180 .7075 .6180;
    .4950 .5687 .6859 .1930 .2260 .7316];
    else if c == 4
K2 = [.7334 .7397 .7348 .7223 .3880 .2080 .3125;
    .1758 .3813 .3400 .7377 .4376 .6577 .5226;
    .6921 .4017 .4973 .1232 .7171 .7153 .7159];
C2 = [.7375 .7874 .7845 .7422 .4297 .2404 .3352;
    .1996 .4132 .3785 .7519 .4684 .6729 .5682;
    .6969 .4000 .4995 .1597 .7307 .7526 .7170];
        else if c == 5
K2 = [.6972 .4144 .5651 .4119 .3252 .4030 .2868;
    .2411 .7153 .7107 .7170 .5389 .4344 .5700;
    .6575 .4042 .2665 .3953 .6966 .7025 .6975];
C2 = [.7097 .4248 .6022 .4549 .3563 .4834 .3194;
    .2546 .7343 .7190 .7595 .5574 .4483 .5966;
    .6917 .4390 .3018 .4212 .7078 .7060 .7265];
            else if c == 6
K2 = [.6957 .5440 .4721 .5267 .4054 .2079;
    .2992 .7182 .7250 .7224 .4296 .6366;
    .5927 .3069 .3905 .3300 .7258 .7225];
C2 = [.7080 .5547 .4820 .5706 .4459 .2220;
    .3127 .7326 .7480 .7407 .4160 .6483;
    .5974 .3299 .4182 .3310 .7562 .7640];
            else if c == 7
K2 = [.7160 .7158 .7139 .7095 .3386 .2579 .2788;
    .2637 .3077 .3144 .7140 .4861 .6108 .5827;
    .6027 .5676 .5245 .1632 .7203 .7274 .7235];
C2 = [.7246 .7519 .7430 .7511 .3512 .2753 .2989;
    .2895 .3412 .3225 .7594 .5349 .6185 .6245;
    .6138 .5988 .5711 .1845 .7667 .7704 .7299];
            else if c == 8
K2 = [.7180 .7151 .7184 .4358 .3365 .4093;
    .2999 .4117 .3945 .7015 .5418 .4587;
    .5805 .4497 .4865 .4307 .7226 .7259];
C2 = [.7575 .7625 .7423 .4427 .3785 .4245;
    .3322 .4425 .4110 .7391 .5548 .4683;
    .5865 .4653 .5306 .4326 .7452 .7321];
```

```matlab
                else if c == 9
    K2 = [.6964 .6019 .4676 .6067 .5108 .4041;
        .2994 .6899 .6913 .6871 .6847 .4383;
        .5454 .2580 .4257 .2502 .3323 .7199];
    C2 = [.7325 .6123 .4903 .6555 .5472 .4443;
        .3388 .7340 .7013 .7370 .6905 .4519
        .5686 .2968 .4407 .2559 .3533 .7388];
                else if c == 10
    K2 = [.7291 .7295 .7256 .5897 .6412 .3122 .2655;
        .2191 .1164 .3511 .6938 .6948 .5241 .5894;
        .6467 .7111 .5239 .2990 .2136 .7059 .7079];
    C2 = [.7353 .7451 .7434 .6217 .6670 .3176 .2755;
        .2666 .1546 .3743 .7167 .7357 .5662 .5898;
        .6750 .7764 .5371 .3100 .2270 .7380 .7292];
                        end
                    end
                end
                end
            end
        end
    end
end
end

%***********************************************************

if a == 3
  if b == 1
    K1 = [.4367 .1798 .8578 .1798 .8578;
        .6146 .2574 .1798 .8578 .8276;
        .6264 .8578 .8276 .2574 .1798];
    C1 = [.7203 .3601 .3601 .3601 .3601;
        .3601 .7203 .3601 .3601 .7203;
        .3601 .3601 .7203 .7203 .3601];
  else if b == 2
    K1 = [.6026 .1698 .9078 .1698 .9078;
        .5222 .2247 .1698 .9078 .7574;
        .6124 .9078 .7574 .2247 .1698];
    C1 = [.7402 .3710 .3710 .3710 .3710;
        .3710 .7402 .3710 .3710 .7402;
        .3710 .3710 .7402 .7402 .3710];
    else if b == 3
```

```
K1 = [.4716 .1756 .9338 .1756 .9338;
      .4100 .2463 .1756 .9338 .6465;
      .4702 .9338 .6465 .2463 .1756];
C1 = [.5790 .3034 .3034 .3034 .3034;
      .3034 .5790 .3034 .3034 .5790;
      .3034 .3034 .5790 .5790 .3034];
   else if b == 4
K1 = [.4339 .1912 .8409 .1912 .8409;
      .6276 .2835 .1912 .8409 .6717;
      .4612 .8409 .6717 .2835 .1912];
C1 = [.5279 .3221 .3221 .3221 .3221;
      .3221 .5279 .3221 .3221 .5279;
      .3221 .3221 .5279 .5279 .3221];
    else if b == 5
K1 = [.4918 .2441 .8207 .2441 .8207;
      .5735 .2709 .2441 .8207 .7685;
      .5299 .8207 .7685 .2709 .2441];
C1 = [.9023 .5432 .5432 .5432 .5432;
      .5432 .9023 .5432 .5432 .9023;
      .5432 .5432 .9023 .9023 .5432];
       else if b == 6
K1 = [.5428 .2258 .7951 .2258 .7951;
      .4386 .2954 .2258 .7951 .5926;
      .4804 .7951 .5926 .2954 .2258];
C1 = [.7776 .4665 .4665 .4665 .4665;
      .4665 .7776 .4665 .4665 .7776;
      .4665 .4665 .7776 .7776 .4665];
        else if b == 7
K1 = [.4379 .2401 .7922 .2401 .7922;
      .5151 .3383 .2401 .7922 .6370;
      .4740 .7922 .6370 .3383 .2401];
C1 = [.9332 .5358 .5358 .5358 .5358;
      .5358 .9332 .5358 .5358 .9332;
      .5358 .5358 .9332 .9332 .5358];
         else if b == 8
K1 = [.4971 .2290 .7575 .2290 .7575;
      .4760 .2372 .2290 .7575 .6692;
      .3967 .7575 .6692 .2372 .2290];
C1 = [.7557 .4443 .4443 .4443 .4443;
      .4443 .7557 .4443 .4443 .7557;
      .4443 .4443 .7557 .7557 .4443];
           else if b == 9
K1 = [.5423 .1282 .8977 .1282 .8977;
      .3567 .1522 .1282 .8977 .5579;
```

```
          .4192 .8977 .5579 .1522 .1282];
C1 = [.9313 .5397 .5397 .5397 .5397;
      .5397 .9313 .5397 .5397 .9313;
      .5397 .5397 .9313 .9313 .5397];
                    else if b == 10
K1 = [.3678 .1052 .9961 .1052 .9961;
      .4072 .1258 .1052 .9961 .8186;
      .5051 .9961 .8186 .1258 .1052];
C1 = [.8482 .4804 .4804 .4804 .4804;
      .4804 .8482 .4804 .4804 .8482;
      .4804 .4804 .8482 .8482 .4804];
                        end
                      end
                    end
                  end
                end
              end
            end
          end
        end
      end
if c == 1
  K2 = [.4367 .1798 .8578 .1798 .8578;
        .6146 .2574 .1798 .8578 .8276;
        .6264 .8578 .8276 .2574 .1798];
  C2 = [.7203 .3601 .3601 .3601 .3601;
        .3601 .7203 .3601 .3601 .7203;
        .3601 .3601 .7203 .7203 .3601];
else if c == 2
  K2 = [.6026 .1698 .9078 .1698 .9078;
        .5222 .2247 .1698 .9078 .7574;
        .6124 .9078 .7574 .2247 .1698];
  C2 = [.7402 .3710 .3710 .3710 .3710;
        .3710 .7402 .3710 .3710 .7402;
        .3710 .3710 .7402 .7402 .3710];
    else if c == 3
  K2 = [.4716 .1756 .9338 .1756 .9338;
        .4100 .2463 .1756 .9338 .6465;
        .4702 .9338 .6465 .2463 .1756];
  C2 = [.5790 .3034 .3034 .3034 .3034;
        .3034 .5790 .3034 .3034 .5790;
        .3034 .3034 .5790 .5790 .3034];
        else if c == 4
  K2 = [.4339 .1912 .8409 .1912 .8409;
```

```
       .6276 .2835 .1912 .8409 .6717;
       .4612 .8409 .6717 .2835 .1912];
C2 = [.5279 .3221 .3221 .3221 .3221;
      .3221 .5279 .3221 .3221 .5279;
      .3221 .3221 .5279 .5279 .3221];
       else if c == 5
K2 = [.4918 .2441 .8207 .2441 .8207;
      .5735 .2709 .2441 .8207 .7685;
      .5299 .8207 .7685 .2709 .2441];
C2 = [.9023 .5432 .5432 .5432 .5432;
      .5432 .9023 .5432 .5432 .9023;
      .5432 .5432 .9023 .9023 .5432];
         else if c == 6
K2 = [.5428 .2258 .7951 .2258 .7951;
      .4386 .2954 .2258 .7951 .5926;
      .4804 .7951 .5926 .2954 .2258];
C2 = [.7776 .4665 .4665 .4665 .4665;
      .4665 .7776 .4665 .4665 .7776;
      .4665 .4665 .7776 .7776 .4665];
          else if c == 7
K2 = [.4379 .2401 .7922 .2401 .7922;
      .5151 .3383 .2401 .7922 .6370;
      .4740 .7922 .6370 .3383 .2401];
C2 = [.9332 .5358 .5358 .5358 .5358;
      .5358 .9332 .5358 .5358 .9332;
      .5358 .5358 .9332 .9332 .5358];
           else if c == 8
K2 = [.4971 .2290 .7575 .2290 .7575;
      .4760 .2372 .2290 .7575 .6692;
      .3967 .7575 .6692 .2372 .2290];
C2 = [.7557 .4443 .4443 .4443 .4443;
      .4443 .7557 .4443 .4443 .7557;
      .4443 .4443 .7557 .7557 .4443];
            else if c == 9
K2 = [.5423 .1282 .8977 .1282 .8977;
      .3567 .1522 .1282 .8977 .5579;
      .4192 .8977 .5579 .1522 .1282];
C2 = [.9313 .5397 .5397 .5397 .5397;
      .5397 .9313 .5397 .5397 .9313;
      .5397 .5397 .9313 .9313 .5397];
             else if c == 10
K2 = [.3678 .1052 .9961 .1052 .9961;
      .4072 .1258 .1052 .9961 .8186;
      .5051 .9961 .8186 .1258 .1052];
```

```matlab
        C2 = [.8482 .4804 .4804 .4804 .4804;
            .4804 .8482 .4804 .4804 .8482;
            .4804 .4804 .8482 .8482 .4804];
                            end
                        end
                    end
                end
            end
          end
        end
      end
    end
  end
end
```

```matlab
%Written by Frances Withrow December 2015
%This function is to calculate which simulation for each assemblage type
%has the most biomass


function [totlayerbiomass]=biomasscalc(structure,z)

global totlength

%just need the species
box1 = z(:,1:20);
box2 = z(:,21:40);
box3 = z(:,41:60);
box4 = z(:,61:80);
box5 = z(:,81:100);
box6 = z(:,101:120);
box7 = z(:,121:140);
box8 = z(:,141:160);
box9 = z(:,161:180);
box10 = z(:,181:200);


av = 500;    %average days I want to calculate over

%Need to find the average biomass for each box


for i = 1:20
    meanbox1(i) = mean(box1(end-av:end,i));
end
for i = 1:20
    meanbox2(i) = mean(box2(end-av:end,i));
end
for i = 1:20
    meanbox3(i) = mean(box3(end-av:end,i));
end
for i = 1:20
    meanbox4(i) = mean(box4(end-av:end,i));
end
for i = 1:20
    meanbox5(i) = mean(box5(end-av:end,i));
end
for i = 1:20
    meanbox6(i) = mean(box6(end-av:end,i));
```

```matlab
end
for i = 1:20
    meanbox7(i) = mean(box7(end-av:end,i));
end
for i = 1:20
    meanbox8(i) = mean(box8(end-av:end,i));
end
for i = 1:20
    meanbox9(i) = mean(box9(end-av:end,i));
end
for i = 1:20
    meanbox10(i) = mean(box10(end-av:end,i));
end


%Need to find the total biomass of each box

totlayerbiomass = [sum(meanbox1) sum(meanbox2) sum(meanbox3)...
    sum(meanbox4) sum(meanbox5) sum(meanbox6) sum(meanbox7)...
    sum(meanbox8) sum(meanbox9) sum(meanbox10)];
```

```matlab
%Written by Frances Withrow March 2014
%This function is to calculate the alpha richness and evenness

function [rich,even, div]=gammacalc(structure,z)

global totlength

%first need to group biomass by SPECIES not BOX


av = 500;

counter = [0:3001:90030];
rich = [];
div = [];

set1 = [1 2 3 4 5 6 7 8 9 10];
set2 = [2 1 4 3 6 5 8 7 10 9];
trackpart = [1:10];
track = [trackpart trackpart trackpart];


for j = 1:30
    ztemp = [z((counter(j)+1):counter(j+1),:)];

assemblage1 = ones(1,10);
assemblage2 = 2*ones(1,10);
assemblage3 = 3*ones(1,10);
assemblagetotal = [assemblage1 assemblage2 assemblage3];
assemblage = assemblagetotal(j);

[K1,K2,C1,C2] = assemblage_func(assemblage,set1(track(j)),set2(track(j)));
totlength = (length(K1) + length(K2));

species1 = [];
for i = [1:20:200]
species1 = [species1 ztemp(:,i)];
end
sumspecies1 = sum(species1,2);

species2 = [];
for i = [2:20:200]
species2 = [species2 ztemp(:,i)];
end
```

```
sumspecies2 = sum(species2,2);

species3 = [];
for i = [3:20:200]
species3 = [species3 ztemp(:,i)];
end
sumspecies3 = sum(species3,2);

species4 = [];
for i = [4:20:200]
species4 = [species4 ztemp(:,i)];
end
sumspecies4 = sum(species4,2);

species5 = [];
for i = [5:20:200]
species5 = [species5 ztemp(:,i)];
end
sumspecies5 = sum(species5,2);

species6 = [];
for i = [6:20:200]
species6 = [species6 ztemp(:,i)];
end
sumspecies6 = sum(species6,2);

species7 = [];
for i = [7:20:200]
species7 = [species7 ztemp(:,i)];
end
sumspecies7 = sum(species7,2);

species8 = [];
for i = [8:20:200]
species8 = [species8 ztemp(:,i)];
end
sumspecies8 = sum(species8,2);

species9 = [];
for i = [9:20:200]
species9 = [species9 ztemp(:,i)];
end
sumspecies9 = sum(species9,2);
```

```
species10 = [];
for i = [10:20:200]
species10 = [species10 ztemp(:,i)];
end
sumspecies10 = sum(species10,2);

species11 = [];
for i = [11:20:200]
species11 = [species11 ztemp(:,i)];
end
sumspecies11 = sum(species11,2);

species12 = [];
for i = [12:20:200]
species12 = [species12 ztemp(:,i)];
end
sumspecies12 = sum(species12,2);

species13 = [];
for i = [13:20:200]
species13 = [species13 ztemp(:,i)];
end
sumspecies13 = sum(species13,2);

species14 = [];
for i = [14:20:200]
species14 = [species14 ztemp(:,i)];
end
sumspecies14 = sum(species14,2);

species15 = [];
for i = [15:20:200]
species15 = [species15 ztemp(:,i)];
end
sumspecies15 = sum(species15,2);

species16 = [];
for i = [16:20:200]
species16 = [species16 ztemp(:,i)];
end
sumspecies16 = sum(species16,2);

species17 = [];
for i = [17:20:200]
```

```matlab
species17 = [species17 ztemp(:,i)];
end
sumspecies17 = sum(species17,2);

species18 = [];
for i = [18:20:200]
species18 = [species18 ztemp(:,i)];
end
sumspecies18 = sum(species18,2);

species19 = [];
for i = [19:20:200]
species19 = [species19 ztemp(:,i)];
end
sumspecies19 = sum(species19,2);

species20 = [];
for i = [20:20:200]
species20 = [species20 ztemp(:,i)];
end
sumspecies20 = sum(species20,2);

sumspecies = [sumspecies1 sumspecies2 sumspecies3 sumspecies4 sumspecies5 ...
    sumspecies6 sumspecies7 sumspecies8 sumspecies9 sumspecies10...
    sumspecies11 sumspecies12 sumspecies13 sumspecies14 sumspecies15...
    sumspecies16 sumspecies17 sumspecies18 sumspecies19 sumspecies20];

for i = 1:20
    meantot(i) = mean(sumspecies(end-av:end,i));
end

[x,richtest] = size(find(meantot > .1));
richtemp = richtest./totlength;




%calculate diversity

totbiomass = sum(meantot);

for i = 1:20
    pr(i) = (meantot(i)/totbiomass);
    H(i) = pr(i)*log(pr(i));
```

```matlab
    if isnan(H(i)) == 1   %need this to get rid of NaN values
        H(i) = 0;
      end
end
divtemp = exp(-1*sum(H));

rich = [rich richtemp];
div = [div divtemp];

end


end
```

```
%Written by Frances Withrow March 2014
%This function is to calculate the alpha richness and evenness

function [rich,even,div]=alphacalc(structure,z)

global totlength


counter = [0:3001:90030];
rich = [];
div = [];

set1 = [1 2 3 4 5 6 7 8 9 10];
set2 = [2 1 4 3 6 5 8 7 10 9];
trackpart = [1:10];
track = [trackpart trackpart trackpart];


for i = 1:30
    ztemp = [z((counter(i)+1):counter(i+1),:)];

assemblage1 = ones(1,10);
assemblage2 = 2*ones(1,10);
assemblage3 = 3*ones(1,10);
assemblagetotal = [assemblage1 assemblage2 assemblage3];
assemblage = assemblagetotal(i);

[K1,K2,C1,C2] = assemblage_func(assemblage,set1(track(i)),set2(track(i)));
totlength = (length(K1) + length(K2));


%just need the species
box1 = ztemp(:,1:20);
box2 = ztemp(:,21:40);
box3 = ztemp(:,41:60);
box4 = ztemp(:,61:80);
box5 = ztemp(:,81:100);
box6 = ztemp(:,101:120);
box7 = ztemp(:,121:140);
box8 = ztemp(:,141:160);
box9 = ztemp(:,161:180);
box10 = ztemp(:,181:200);
```

```
av = 500;    %average days I want to calculate over

%Need to find the average biomass for each box


for i = 1:20
    meanbox1(i) = mean(box1(end-av:end,i));
end
for i = 1:20
    meanbox2(i) = mean(box2(end-av:end,i));
end
for i = 1:20
    meanbox3(i) = mean(box3(end-av:end,i));
end
for i = 1:20
    meanbox4(i) = mean(box4(end-av:end,i));
end
for i = 1:20
    meanbox5(i) = mean(box5(end-av:end,i));
end
for i = 1:20
    meanbox6(i) = mean(box6(end-av:end,i));
end
for i = 1:20
    meanbox7(i) = mean(box7(end-av:end,i));
end
for i = 1:20
    meanbox8(i) = mean(box8(end-av:end,i));
end
for i = 1:20
    meanbox9(i) = mean(box9(end-av:end,i));
end
for i = 1:20
    meanbox10(i) = mean(box10(end-av:end,i));
end

%first need to calculate richness for each box
%Find which species is present and then count how many
[~,rich1] = size(find(meanbox1 > .1));
[~,rich2] = size(find(meanbox2 > .1));
[~,rich3] = size(find(meanbox3 > .1));
[~,rich4] = size(find(meanbox4 > .1));
[~,rich5] = size(find(meanbox5 > .1));
```

```matlab
[~,rich6] = size(find(meanbox6 > .1));
[~,rich7] = size(find(meanbox7 > .1));
[~,rich8] = size(find(meanbox8 > .1));
[~,rich9] = size(find(meanbox9 > .1));
[~,rich10] = size(find(meanbox10 > .1));


richtemp = [rich1; rich2; rich3; rich4; rich5; rich6; rich7;...
    rich8; rich9; rich10]./totlength;




%second need to calculate diversity

N = totlength;


%  box1

%diversity
totbio1 = sum(meanbox1);
for i = 1:20
   pr1(i) = (meanbox1(i)/totbio1);
   H1(i) = pr1(i)*log(pr1(i));
   if isnan(H1(i)) == 1   %need this to get rid of NaN values
      H1(i) = 0;
   end
end


div1 = exp(-1*sum(H1));


%  box2

%diversity
totbio2 = sum(meanbox2);
for i = 1:20
   pr2(i) = (meanbox2(i)/totbio2);
   H2(i) = pr2(i)*log(pr2(i));
   if isnan(H2(i)) == 1   %need this to get rid of NaN values
      H2(i) = 0;
```

```matlab
        end
end
div2 = exp(-1*sum(H2));

%  box3

%diversity
totbio3 = sum(meanbox3);
for i = 1:20
    pr3(i) = (meanbox3(i)/totbio3);
    H3(i) = pr3(i)*log(pr3(i));
    if isnan(H3(i)) == 1   %need this to get rid of NaN values
        H3(i) = 0;
        end
end
div3 = exp(-1*sum(H3));

%  box4

%diversity
totbio4 = sum(meanbox4);
for i = 1:20
    pr4(i) = (meanbox4(i)/totbio4);
    H4(i) = pr4(i)*log(pr4(i));
    if isnan(H4(i)) == 1   %need this to get rid of NaN values
        H4(i) = 0;
        end
end
div4 = exp(-1*sum(H4));

%  box5

%diversity
totbio5 = sum(meanbox5);
for i = 1:20
    pr5(i) = (meanbox5(i)/totbio5);
    H5(i) = pr5(i)*log(pr5(i));
    if isnan(H5(i)) == 1   %need this to get rid of NaN values
        H5(i) = 0;
        end
end
div5 = exp(-1*sum(H5));

%  box6
```

```matlab
%diversity
totbio6 = sum(meanbox6);
for i = 1:20
    pr6(i) = (meanbox6(i)/totbio6);
    H6(i) = pr6(i)*log(pr6(i));
    if isnan(H6(i)) == 1   %need this to get rid of NaN values
        H6(i) = 0;
        end
end
div6 = exp(-1*sum(H6));

%  box7

%diversity
totbio7 = sum(meanbox7);
for i = 1:20
    pr7(i) = (meanbox7(i)/totbio7);
    H7(i) = pr7(i)*log(pr7(i));
    if isnan(H7(i)) == 1   %need this to get rid of NaN values
        H7(i) = 0;
        end
end
div7 = exp(-1*sum(H7));

%  box8

%diversity
totbio8 = sum(meanbox8);
for i = 1:20
    pr8(i) = (meanbox8(i)/totbio8);
    H8(i) = pr8(i)*log(pr8(i));
    if isnan(H8(i)) == 1   %need this to get rid of NaN values
        H8(i) = 0;
        end
end
div8 = exp(-1*sum(H8));

%  box9

%diversity
totbio9 = sum(meanbox9);
for i = 1:20
    pr9(i) = (meanbox9(i)/totbio9);
```

```matlab
        H9(i) = pr9(i)*log(pr9(i));
        if isnan(H9(i)) == 1   %need this to get rid of NaN values
            H9(i) = 0;
            end
    end
div9 = exp(-1*sum(H9));

%  box10

%diversity
totbio10 = sum(meanbox10);
for i = 1:20
    pr10(i) = (meanbox10(i)/totbio10);
    H10(i) = pr10(i)*log(pr10(i));
    if isnan(H10(i)) == 1   %need this to get rid of NaN values
        H10(i) = 0;
        end
end
div10 = exp(-1*sum(H10));


divtemp = [div1; div2; div3; div4; div5; div6; ...
    div7; div8; div9; div10];

rich = [rich richtemp];

div = [div divtemp];

end

end
```

```matlab
% This is a function called by the master routine to calculate the
% rates.
% Original programing by Daniel Roelke
% Editing by Frances Withrow
%------------------------------------------------------------------------

function [zdot]=differential(t,z)

global r  K  C  mark  Sk  Kw  Kc  Tb irrinit CarCell CarChl lswitch ...
    alphaset vect area V ntrflux flow sourcephyto sourcentr D ...
    oceanphyto oceanntr depth


%------------------------------------------------------------------------
% SPLIT UP INCOMING INFORMATION
%------------------------------------------------------------------------

A1  = z(1:20)';                    % Box 1 Algae
A2  = z(21:40)';                   % Box 2 Algae
A3  = z(41:60)';                   % Box 3 Algae
A4  = z(61:80)';                   % Box 4 Algae
A5  = z(81:100)';                  % Box 5 Algae
A6  = z(101:120)';                 % Box 6 Algae
A7  = z(121:140)';                 % Box 7 Algae
A8  = z(141:160)';                 % Box 8 Algae
A9  = z(161:180)';                 % Box 9 Algae
A10 = z(181:200)';                 % Box 10 Algae

R1  = z(201:203)';                 % Box 1 Resources
R2  = z(204:206)';                 % Box 2 Resources
R3  = z(207:209)';                 % Box 3 Resources
R4  = z(210:212)';                 % Box 4 Resources
R5  = z(213:215)';                 % Box 4 Resources
R6  = z(216:218)';                 % Box 4 Resources
R7  = z(219:221)';                 % Box 4 Resources
R8  = z(222:224)';                 % Box 4 Resources
R9  = z(225:227)';                 % Box 4 Resources
R10 = z(228:230)';                 % Box 4 Resources

LN1 = sourcentr(1,:);              % Box 1 Land Resources
LN2 = sourcentr(2,:);              % Box 2 Land Resources
LN3 = sourcentr(3,:);              % Box 3 Land Resources
LN4 = sourcentr(4,:);              % Box 4 Land Resources
LN5 = sourcentr(5,:);              % Box 5 Land Resources
LN6 = sourcentr(6,:);              % Box 6 Land Resources
```

```
LN7 = sourcentr(7,:);                % Box 7 Land Resources
LN8 = sourcentr(8,:);                % Box 8 Land Resources
LN9 = sourcentr(9,:);                % Box 9 Land Resources
LN10 = sourcentr(10,:);               % Box 10 Land Resources

OP1 = oceanphyto(1,:);                % Box 1 Open Ocean Phyto
OP2 = oceanphyto(2,:);                % Box 2 Open Ocean Phyto
OP3 = oceanphyto(3,:);                % Box 3 Open Ocean Phyto
OP4 = oceanphyto(4,:);                % Box 4 Open Ocean Phyto
OP5 = oceanphyto(5,:);                % Box 5 Open Ocean Phyto
OP6 = oceanphyto(6,:);                % Box 6 Open Ocean Phyto
OP7 = oceanphyto(7,:);                % Box 7 Open Ocean Phyto
OP8 = oceanphyto(8,:);                % Box 8 Open Ocean Phyto
OP9 = oceanphyto(9,:);                % Box 9 Open Ocean Phyto
OP10 = oceanphyto(10,:);              % Box 10 Open Ocean Phyto

ON1 = oceanntr(1,:);                 % Box 1 Open Ocean Resources
ON2 = oceanntr(2,:);                 % Box 2 Open Ocean Resources
ON3 = oceanntr(3,:);                 % Box 3 Open Ocean Resources
ON4 = oceanntr(4,:);                 % Box 4 Open Ocean Resources
ON5 = oceanntr(5,:);                 % Box 5 Open Ocean Resources
ON6 = oceanntr(6,:);                 % Box 6 Open Ocean Resources
ON7 = oceanntr(7,:);                 % Box 7 Open Ocean Resources
ON8 = oceanntr(8,:);                 % Box 8 Open Ocean Resources
ON9 = oceanntr(9,:);                 % Box 9 Open Ocean Resources
ON10 = oceanntr(10,:);                % Box 10 Open Ocean Resources


%------------------------------------------------------------------------
% FIND ALGAE GROWTH RATE
%------------------------------------------------------------------------

for i=1:length(R1)   % Keep this control check, sometimes Monod overshoots

   R1(i)  = max(R1(i), 0);
   R2(i)  = max(R2(i), 0);
   R3(i)  = max(R3(i), 0);
   R4(i)  = max(R4(i), 0);
   R5(i)  = max(R5(i), 0);
   R6(i)  = max(R6(i), 0);
   R7(i)  = max(R7(i), 0);
   R8(i)  = max(R8(i), 0);
   R9(i)  = max(R9(i), 0);
   R10(i) = max(R10(i),0);
```

```
end

%u is growth rate as a function of nutrients
%rL is the growth rate as a function of light

for i=1:length(K(:,1))

    for j=1:length(K(1,:))

        u1(i,j) = r.*(R1(i)./(K(i,j)+R1(i)));
        u2(i,j) = r.*(R2(i)./(K(i,j)+R2(i)));
        u3(i,j) = r.*(R3(i)./(K(i,j)+R3(i)));
        u4(i,j) = r.*(R4(i)./(K(i,j)+R4(i)));
        u5(i,j) = r.*(R5(i)./(K(i,j)+R5(i)));
        u6(i,j) = r.*(R6(i)./(K(i,j)+R6(i)));
        u7(i,j) = r.*(R7(i)./(K(i,j)+R7(i)));
        u8(i,j) = r.*(R8(i)./(K(i,j)+R8(i)));
        u9(i,j) = r.*(R9(i)./(K(i,j)+R9(i)));
        u10(i,j)= r.*(R10(i)./(K(i,j)+R10(i)));

    end

end


if lswitch == 1

for i = 1:10

    Atot   = sum(z(1:(20*i)))/(depth*i) ;        % Find the average
                                         %  A for all layers
                                         %  considered



    Ktot   = Kw + Kc.*(Atot.*CarCell./CarChl);  % Dynamic K

    irrz   = irrinit*exp(-1*i*Ktot);        % extinction

    A      = alphaset.*irrz./r;             % scalar factor
    A(1:10);
    ulight(i,1:20)  = r.*(1-exp(-A));          % growth based on light
```

```matlab
%     if t>100

    end

    clear Atot Ktot A


end
if lswitch == 0
    ulight = r*ones(1,10);
end


% Use "nested" Law of the Minimum
umin1   = min(min(u1),ulight(1));
umin2   = min(min(u2),ulight(2));
umin3   = min(min(u3),ulight(3));
umin4   = min(min(u4),ulight(4));
umin5   = min(min(u5),ulight(5));
umin6   = min(min(u6),ulight(6));
umin7   = min(min(u7),ulight(7));
umin8   = min(min(u8),ulight(8));
umin9   = min(min(u9),ulight(9));
umin10  = min(min(u10),ulight(10));



%-------------------------------------------------------------------------
% DIFFERENTIAL EQUATIONS
%-------------------------------------------------------------------------
% Note: First equation acounts for 5 phytoplankton populations
% Next three account for the three nutrient resources

%This if else accounts for the buoyant option for sinking scenario
if Sk < 0
    % Box 1 (top)
z1dot = A1.*(umin1 - Tb(1)-(flow(1)/V) -D(1)) + A2.*Tb(1) +
sourcephyto.*(flow(1)/V) + D(1)*OP1 - Sk.*A2;
z2dot = - sum(C(1,:).*umin1.*A1) - R1(1)*Tb(1) + R2(1)*Tb(1) + (flow(1)/V)*LN1(1)
- (flow(1)/V)*R1(1) - D(1)*(R1(1) - ON1(1));
z3dot = - sum(C(2,:).*umin1.*A1) - R1(2)*Tb(1) + R2(2)*Tb(1) + (flow(1)/V)*LN1(2)
- (flow(1)/V)*R1(2) - D(1)*(R1(2) - ON1(2));
z4dot = - sum(C(3,:).*umin1.*A1) - R1(3)*Tb(1) + R2(3)*Tb(1) + (flow(1)/V)*LN1(3)
- (flow(1)/V)*R1(3) - D(1)*(R1(3) - ON1(3));
```

```
% Box 2 (bookended)
z5dot = A2.*(umin2 + Sk - Tb(1) - Tb(2)-(flow(2)/V)-D(2)) + A1.*(Tb(1)) + A3.*Tb(2)
+ sourcephyto.*(flow(2)/V) + D(2)*OP2 - Sk.*A3;
z6dot = - sum(C(1,:).*umin2.*A2) - R2(1)*(Tb(1) + Tb(2)) + R1(1)*Tb(1) +
R3(1)*Tb(2) + (flow(2)/V)*LN2(1) - (flow(2)/V)*R2(1) - D(2)*(R2(1) - ON2(1));
z7dot = - sum(C(2,:).*umin2.*A2) - R2(2)*(Tb(1) + Tb(2)) + R1(2)*Tb(1) +
R3(2)*Tb(2) + (flow(2)/V)*LN2(2) - (flow(2)/V)*R2(2) - D(2)*(R2(2) - ON2(2));
z8dot = - sum(C(3,:).*umin2.*A2) - R2(3)*(Tb(1) + Tb(2)) + R1(3)*Tb(1) +
R3(3)*Tb(2) + (flow(2)/V)*LN2(3) - (flow(2)/V)*R2(3) - D(2)*(R2(3) - ON2(3));

% Box 3 (bookended)
z9dot = A3.*(umin3 + Sk - Tb(2) - Tb(3)-(flow(3)/V)-D(3)) + A2.*(Tb(2)) + A4.*Tb(3)
+ sourcephyto.*(flow(3)/V) + D(3)*OP3 - Sk.*A4;
z10dot = - sum(C(1,:).*umin3.*A3) - R3(1)*(Tb(2) + Tb(3)) + R2(1)*Tb(2) +
R4(1)*Tb(3) + (flow(3)/V)*LN3(1) - (flow(3)/V)*R3(1) - D(3)*(R3(1) - ON3(1));
z11dot = - sum(C(2,:).*umin3.*A3) - R3(2)*(Tb(2) + Tb(3)) + R2(2)*Tb(2) +
R4(2)*Tb(3) + (flow(3)/V)*LN3(2) - (flow(3)/V)*R3(2) - D(3)*(R3(2) - ON3(2));
z12dot = - sum(C(3,:).*umin3.*A3) - R3(3)*(Tb(2) + Tb(3)) + R2(3)*Tb(2) +
R4(3)*Tb(3) + (flow(3)/V)*LN3(3) - (flow(3)/V)*R3(3) - D(3)*(R3(3) - ON3(3));

% Box 4 (bookended)
z13dot = A4.*(umin4 + Sk - Tb(3) - Tb(4)-(flow(4)/V)-D(4)) + A3.*(Tb(3)) +
A5.*Tb(4) + sourcephyto.*(flow(4)/V) + D(4)*OP4 - Sk.*A5;
z14dot = - sum(C(1,:).*umin4.*A4) - R4(1)*(Tb(3) + Tb(4)) + R3(1)*Tb(3) +
R5(1)*Tb(4) + (flow(4)/V)*LN4(1) - (flow(4)/V)*R4(1) - D(4)*(R4(1) - ON4(1));
z15dot = - sum(C(2,:).*umin4.*A4) - R4(2)*(Tb(3) + Tb(4)) + R3(2)*Tb(3) +
R5(2)*Tb(4) + (flow(4)/V)*LN4(2) - (flow(4)/V)*R4(2) - D(4)*(R4(2) - ON4(2));
z16dot = - sum(C(3,:).*umin4.*A4) - R4(3)*(Tb(3) + Tb(4)) + R3(3)*Tb(3) +
R5(3)*Tb(4) + (flow(4)/V)*LN4(3) - (flow(4)/V)*R4(3) - D(4)*(R4(3) - ON4(3));

% Box 5 (bookended)
z17dot = A5.*(umin5 + Sk - Tb(4) - Tb(5)-(flow(5)/V)-D(5)) + A4.*(Tb(4)) +
A6.*Tb(5) + sourcephyto.*(flow(5)/V) + D(5)*OP5 - Sk.*A6;
z18dot = - sum(C(1,:).*umin5.*A5) - R5(1)*(Tb(4) + Tb(5)) + R4(1)*Tb(4) +
R6(1)*Tb(5) + (flow(5)/V)*LN5(1) - (flow(5)/V)*R5(1) - D(5)*(R5(1) - ON5(1));
z19dot = - sum(C(2,:).*umin5.*A5) - R5(2)*(Tb(4) + Tb(5)) + R4(2)*Tb(4) +
R6(2)*Tb(5) + (flow(5)/V)*LN5(2) - (flow(5)/V)*R5(2) - D(5)*(R5(2) - ON5(2));
z20dot = - sum(C(3,:).*umin5.*A5) - R5(3)*(Tb(4) + Tb(5)) + R4(3)*Tb(4) +
R6(3)*Tb(5) + (flow(5)/V)*LN5(3) - (flow(5)/V)*R5(3) - D(5)*(R5(3) - ON5(3));

% Box 6 (bookended)
z21dot = A6.*(umin6 + Sk - Tb(5) - Tb(6)-(flow(6)/V)-D(6)) + A5.*(Tb(5)) +
A7.*Tb(6) + sourcephyto.*(flow(6)/V) + D(6)*OP6 - Sk.*A7;
```

z22dot = - sum(C(1,:).*umin6.*A6) - R6(1)*(Tb(5) + Tb(6)) + R5(1)*Tb(5) + R7(1)*Tb(6) + (flow(6)/V)*LN6(1) - (flow(6)/V)*R6(1) - D(6)*(R6(1) - ON6(1));
z23dot = - sum(C(2,:).*umin6.*A6) - R6(2)*(Tb(5) + Tb(6)) + R5(2)*Tb(5) + R7(2)*Tb(6) + (flow(6)/V)*LN6(2) - (flow(6)/V)*R6(2) - D(6)*(R6(2) - ON6(2));
z24dot = - sum(C(3,:).*umin6.*A6) - R6(3)*(Tb(5) + Tb(6)) + R5(3)*Tb(5) + R7(3)*Tb(6) + (flow(6)/V)*LN6(3) - (flow(6)/V)*R6(3) - D(6)*(R6(3) - ON6(3));

% Box 7 (bookended)
z25dot = A7.*(umin7 + Sk - Tb(6) - Tb(7)-(flow(7)/V)-D(7)) + A6.*(Tb(6)) + A8.*Tb(7) + sourcephyto.*(flow(7)/V) + D(7)*OP7 - Sk.*A8;
z26dot = - sum(C(1,:).*umin7.*A7) - R7(1)*(Tb(6) + Tb(7)) + R6(1)*Tb(6) + R8(1)*Tb(7) + (flow(7)/V)*LN7(1) - (flow(7)/V)*R7(1) - D(7)*(R7(1) - ON7(1));
z27dot = - sum(C(2,:).*umin7.*A7) - R7(2)*(Tb(6) + Tb(7)) + R6(2)*Tb(6) + R8(2)*Tb(7) + (flow(7)/V)*LN7(2) - (flow(7)/V)*R7(2) - D(7)*(R7(2) - ON7(2));
z28dot = - sum(C(3,:).*umin7.*A7) - R7(3)*(Tb(6) + Tb(7)) + R6(3)*Tb(6) + R8(3)*Tb(7) + (flow(7)/V)*LN7(3) - (flow(7)/V)*R7(3) - D(7)*(R7(3) - ON7(3));

% Box 8 (bookended)
z29dot = A8.*(umin8 + Sk - Tb(7) - Tb(8)-(flow(8)/V)-D(8)) + A7.*(Tb(7)) + A9.*Tb(8) + sourcephyto.*(flow(8)/V) + D(8)*OP8 - Sk.*A9;
z30dot = - sum(C(1,:).*umin8.*A8) - R8(1)*(Tb(7) + Tb(8)) + R7(1)*Tb(7) + R9(1)*Tb(8) + (flow(8)/V)*LN8(1) - (flow(8)/V)*R8(1) - D(8)*(R8(1) - ON8(1));
z31dot = - sum(C(2,:).*umin8.*A8) - R8(2)*(Tb(7) + Tb(8)) + R7(2)*Tb(7) + R9(2)*Tb(8) + (flow(8)/V)*LN8(2) - (flow(8)/V)*R8(2) - D(8)*(R8(2) - ON8(2));
z32dot = - sum(C(3,:).*umin8.*A8) - R8(3)*(Tb(7) + Tb(8)) + R7(3)*Tb(7) + R9(3)*Tb(8) + (flow(8)/V)*LN8(3) - (flow(8)/V)*R8(3) - D(8)*(R8(3) - ON8(3));

% Box 9 (bookended)
z33dot = A9.*(umin9 + Sk - Tb(8) - Tb(9)-(flow(9)/V)-D(9)) + A8.*(Tb(8)) + A10.*Tb(9) + sourcephyto.*(flow(9)/V) + D(9)*OP9 - Sk.*A10;
z34dot = - sum(C(1,:).*umin9.*A9) - R9(1)*(Tb(8) + Tb(9)) + R8(1)*Tb(8) + R10(1)*Tb(9) + (flow(9)/V)*LN9(1) - (flow(9)/V)*R9(1) - D(9)*(R9(1) - ON9(1));
z35dot = - sum(C(2,:).*umin9.*A9) - R9(2)*(Tb(8) + Tb(9)) + R8(2)*Tb(8) + R10(2)*Tb(9) + (flow(9)/V)*LN9(2) - (flow(9)/V)*R9(2) - D(9)*(R9(2) - ON9(2));
z36dot = - sum(C(3,:).*umin9.*A9) - R9(3)*(Tb(8) + Tb(9)) + R8(3)*Tb(8) + R10(3)*Tb(9) + (flow(9)/V)*LN9(3) - (flow(9)/V)*R9(3) - D(9)*(R9(3) - ON9(3));

% Box 10 (bottom)
z37dot = A10.*(umin10 + Sk - Tb(9) - Tb(10)-(flow(10)/V)-D(10)) + A9.*(Sk + Tb(9)) + sourcephyto.*(flow(10)/V) + D(10)*OP10;
z38dot = - sum(C(1,:).*umin10.*A10) - R10(1)*(Tb(9) + Tb(10)) + R9(1)*Tb(9) + ntrflux(1)*area + (flow(10)/V)*LN10(1) - (flow(10)/V)*R10(1) - D(10)*(R10(1) - ON10(1));

```matlab
z39dot = - sum(C(2,:).*umin10.*A10) - R10(2)*(Tb(9) + Tb(10)) + R9(2)*Tb(9) +
ntrflux(2)*area + (flow(10)/V)*LN10(2) - (flow(10)/V)*R10(2) - D(10)*(R10(2) -
ON10(2));
z40dot = - sum(C(3,:).*umin10.*A10) - R10(3)*(Tb(9) + Tb(10)) + R9(3)*Tb(9) +
ntrflux(3)*area + (flow(10)/V)*LN10(3) - (flow(10)/V)*R10(3) - D(10)*(R10(3) -
ON10(3));

slowdown = sin(10*t);

else


% Box 1 (top)
z1dot = A1.*(umin1 - Sk - Tb(1)-(flow(1)/V) -D(1)) + A2.*Tb(1) +
sourcephyto.*(flow(1)/V) + D(1)*OP1;
z2dot = - sum(C(1,:).*umin1.*A1) - R1(1)*Tb(1) + R2(1)*Tb(1) + (flow(1)/V)*LN1(1)
- (flow(1)/V)*R1(1) - D(1)*(R1(1) - ON1(1));
z3dot = - sum(C(2,:).*umin1.*A1) - R1(2)*Tb(1) + R2(2)*Tb(1) + (flow(1)/V)*LN1(2)
- (flow(1)/V)*R1(2) - D(1)*(R1(2) - ON1(2));
z4dot = - sum(C(3,:).*umin1.*A1) - R1(3)*Tb(1) + R2(3)*Tb(1) + (flow(1)/V)*LN1(3)
- (flow(1)/V)*R1(3) - D(1)*(R1(3) - ON1(3));


% Box 2 (bookended)
z5dot = A2.*(umin2 - Sk - Tb(1) - Tb(2)-(flow(2)/V)-D(2)) + A1.*(Sk + Tb(1)) +
A3.*Tb(2) + sourcephyto.*(flow(2)/V) + D(2)*OP2;
z6dot = - sum(C(1,:).*umin2.*A2) - R2(1)*(Tb(1) + Tb(2)) + R1(1)*Tb(1) +
R3(1)*Tb(2) + (flow(2)/V)*LN2(1) - (flow(2)/V)*R2(1) - D(2)*(R2(1) - ON2(1));
z7dot = - sum(C(2,:).*umin2.*A2) - R2(2)*(Tb(1) + Tb(2)) + R1(2)*Tb(1) +
R3(2)*Tb(2) + (flow(2)/V)*LN2(2) - (flow(2)/V)*R2(2) - D(2)*(R2(2) - ON2(2));
z8dot = - sum(C(3,:).*umin2.*A2) - R2(3)*(Tb(1) + Tb(2)) + R1(3)*Tb(1) +
R3(3)*Tb(2) + (flow(2)/V)*LN2(3) - (flow(2)/V)*R2(3) - D(2)*(R2(3) - ON2(3));

% Box 3 (bookended)
z9dot = A3.*(umin3 - Sk - Tb(2) - Tb(3)-(flow(3)/V)-D(3)) + A2.*(Sk + Tb(2)) +
A4.*Tb(3) + sourcephyto.*(flow(3)/V) + D(3)*OP3;
z10dot = - sum(C(1,:).*umin3.*A3) - R3(1)*(Tb(2) + Tb(3)) + R2(1)*Tb(2) +
R4(1)*Tb(3) + (flow(3)/V)*LN3(1) - (flow(3)/V)*R3(1) - D(3)*(R3(1) - ON3(1));
z11dot = - sum(C(2,:).*umin3.*A3) - R3(2)*(Tb(2) + Tb(3)) + R2(2)*Tb(2) +
R4(2)*Tb(3) + (flow(3)/V)*LN3(2) - (flow(3)/V)*R3(2) - D(3)*(R3(2) - ON3(2));
z12dot = - sum(C(3,:).*umin3.*A3) - R3(3)*(Tb(2) + Tb(3)) + R2(3)*Tb(2) +
R4(3)*Tb(3) + (flow(3)/V)*LN3(3) - (flow(3)/V)*R3(3) - D(3)*(R3(3) - ON3(3));

% Box 4 (bookended)
```

z13dot = A4.*(umin4 - Sk - Tb(3) - Tb(4)-(flow(4)/V)-D(4)) + A3.*(Sk + Tb(3)) + A5.*Tb(4) + sourcephyto.*(flow(4)/V) + D(4)*OP4;
z14dot = - sum(C(1,:).*umin4.*A4) - R4(1)*(Tb(3) + Tb(4)) + R3(1)*Tb(3) + R5(1)*Tb(4) + (flow(4)/V)*LN4(1) - (flow(4)/V)*R4(1) - D(4)*(R4(1) - ON4(1));
z15dot = - sum(C(2,:).*umin4.*A4) - R4(2)*(Tb(3) + Tb(4)) + R3(2)*Tb(3) + R5(2)*Tb(4) + (flow(4)/V)*LN4(2) - (flow(4)/V)*R4(2) - D(4)*(R4(2) - ON4(2));
z16dot = - sum(C(3,:).*umin4.*A4) - R4(3)*(Tb(3) + Tb(4)) + R3(3)*Tb(3) + R5(3)*Tb(4) + (flow(4)/V)*LN4(3) - (flow(4)/V)*R4(3) - D(4)*(R4(3) - ON4(3));

% Box 5 (bookended)
z17dot = A5.*(umin5 - Sk - Tb(4) - Tb(5)-(flow(5)/V)-D(5)) + A4.*(Sk + Tb(4)) + A6.*Tb(5) + sourcephyto.*(flow(5)/V) + D(5)*OP5;
z18dot = - sum(C(1,:).*umin5.*A5) - R5(1)*(Tb(4) + Tb(5)) + R4(1)*Tb(4) + R6(1)*Tb(5) + (flow(5)/V)*LN5(1) - (flow(5)/V)*R5(1) - D(5)*(R5(1) - ON5(1));
z19dot = - sum(C(2,:).*umin5.*A5) - R5(2)*(Tb(4) + Tb(5)) + R4(2)*Tb(4) + R6(2)*Tb(5) + (flow(5)/V)*LN5(2) - (flow(5)/V)*R5(2) - D(5)*(R5(2) - ON5(2));
z20dot = - sum(C(3,:).*umin5.*A5) - R5(3)*(Tb(4) + Tb(5)) + R4(3)*Tb(4) + R6(3)*Tb(5) + (flow(5)/V)*LN5(3) - (flow(5)/V)*R5(3) - D(5)*(R5(3) - ON5(3));

% Box 6 (bookended)
z21dot = A6.*(umin6 - Sk - Tb(5) - Tb(6)-(flow(6)/V)-D(6)) + A5.*(Sk + Tb(5)) + A7.*Tb(6) + sourcephyto.*(flow(6)/V) + D(6)*OP6;
z22dot = - sum(C(1,:).*umin6.*A6) - R6(1)*(Tb(5) + Tb(6)) + R5(1)*Tb(5) + R7(1)*Tb(6) + (flow(6)/V)*LN6(1) - (flow(6)/V)*R6(1) - D(6)*(R6(1) - ON6(1));
z23dot = - sum(C(2,:).*umin6.*A6) - R6(2)*(Tb(5) + Tb(6)) + R5(2)*Tb(5) + R7(2)*Tb(6) + (flow(6)/V)*LN6(2) - (flow(6)/V)*R6(2) - D(6)*(R6(2) - ON6(2));
z24dot = - sum(C(3,:).*umin6.*A6) - R6(3)*(Tb(5) + Tb(6)) + R5(3)*Tb(5) + R7(3)*Tb(6) + (flow(6)/V)*LN6(3) - (flow(6)/V)*R6(3) - D(6)*(R6(3) - ON6(3));

% Box 7 (bookended)
z25dot = A7.*(umin7 - Sk - Tb(6) - Tb(7)-(flow(7)/V)-D(7)) + A6.*(Sk + Tb(6)) + A8.*Tb(7) + sourcephyto.*(flow(7)/V) + D(7)*OP7;
z26dot = - sum(C(1,:).*umin7.*A7) - R7(1)*(Tb(6) + Tb(7)) + R6(1)*Tb(6) + R8(1)*Tb(7) + (flow(7)/V)*LN7(1) - (flow(7)/V)*R7(1) - D(7)*(R7(1) - ON7(1));
z27dot = - sum(C(2,:).*umin7.*A7) - R7(2)*(Tb(6) + Tb(7)) + R6(2)*Tb(6) + R8(2)*Tb(7) + (flow(7)/V)*LN7(2) - (flow(7)/V)*R7(2) - D(7)*(R7(2) - ON7(2));
z28dot = - sum(C(3,:).*umin7.*A7) - R7(3)*(Tb(6) + Tb(7)) + R6(3)*Tb(6) + R8(3)*Tb(7) + (flow(7)/V)*LN7(3) - (flow(7)/V)*R7(3) - D(7)*(R7(3) - ON7(3));

% Box 8 (bookended)
z29dot = A8.*(umin8 - Sk - Tb(7) - Tb(8)-(flow(8)/V)-D(8)) + A7.*(Sk + Tb(7)) + A9.*Tb(8) + sourcephyto.*(flow(8)/V) + D(8)*OP8;
z30dot = - sum(C(1,:).*umin8.*A8) - R8(1)*(Tb(7) + Tb(8)) + R7(1)*Tb(7) + R9(1)*Tb(8) + (flow(8)/V)*LN8(1) - (flow(8)/V)*R8(1) - D(8)*(R8(1) - ON8(1));

```matlab
z31dot = - sum(C(2,:).*umin8.*A8) - R8(2)*(Tb(7) + Tb(8)) + R7(2)*Tb(7) +
R9(2)*Tb(8) + (flow(8)/V)*LN8(2) - (flow(8)/V)*R8(2) - D(8)*(R8(2) - ON8(2));
z32dot = - sum(C(3,:).*umin8.*A8) - R8(3)*(Tb(7) + Tb(8)) + R7(3)*Tb(7) +
R9(3)*Tb(8) + (flow(8)/V)*LN8(3) - (flow(8)/V)*R8(3) - D(8)*(R8(3) - ON8(3));

% Box 9 (bookended)
z33dot = A9.*(umin9 - Sk - Tb(8) - Tb(9)-(flow(9)/V)-D(9)) + A8.*(Sk + Tb(8)) +
A10.*Tb(9) + sourcephyto.*(flow(9)/V) + D(9)*OP9;
z34dot = - sum(C(1,:).*umin9.*A9) - R9(1)*(Tb(8) + Tb(9)) + R8(1)*Tb(8) +
R10(1)*Tb(9) + (flow(9)/V)*LN9(1) - (flow(9)/V)*R9(1) - D(9)*(R9(1) - ON9(1));
z35dot = - sum(C(2,:).*umin9.*A9) - R9(2)*(Tb(8) + Tb(9)) + R8(2)*Tb(8) +
R10(2)*Tb(9) + (flow(9)/V)*LN9(2) - (flow(9)/V)*R9(2) - D(9)*(R9(2) - ON9(2));
z36dot = - sum(C(3,:).*umin9.*A9) - R9(3)*(Tb(8) + Tb(9)) + R8(3)*Tb(8) +
R10(3)*Tb(9) + (flow(9)/V)*LN9(3) - (flow(9)/V)*R9(3) - D(9)*(R9(3) - ON9(3));

% Box 10 (bottom)
z37dot = A10.*(umin10 - Sk - Tb(9) - Tb(10)-(flow(10)/V)-D(10)) + A9.*(Sk + Tb(9))
+ sourcephyto.*(flow(10)/V) + D(10)*OP10;
z38dot = - sum(C(1,:).*umin10.*A10) - R10(1)*(Tb(9) + Tb(10)) + R9(1)*Tb(9) +
ntrflux(1)*area + (flow(10)/V)*LN10(1) - (flow(10)/V)*R10(1) - D(10)*(R10(1) -
ON10(1));
z39dot = - sum(C(2,:).*umin10.*A10) - R10(2)*(Tb(9) + Tb(10)) + R9(2)*Tb(9) +
ntrflux(2)*area + (flow(10)/V)*LN10(2) - (flow(10)/V)*R10(2) - D(10)*(R10(2) -
ON10(2));
z40dot = - sum(C(3,:).*umin10.*A10) - R10(3)*(Tb(9) + Tb(10)) + R9(3)*Tb(9) +
ntrflux(3)*area + (flow(10)/V)*LN10(3) - (flow(10)/V)*R10(3) - D(10)*(R10(3) -
ON10(3));

slowdown = sin(10*t);
end


%------------------------------------------------------------------------
% COUNTER (for impatient modelers)
%------------------------------------------------------------------------

if t>mark
  format compact
  t
  mark=mark+100;
end


%------------------------------------------------------------------------
% OUTPUT
%------------------------------------------------------------------------
```

```
zdot=[z1dot,z5dot,z9dot,z13dot,z17dot,z21dot,z25dot,z29dot,z33dot,z37dot, ...
    z2dot,z3dot,z4dot,...
    z6dot,z7dot,z8dot,...
    z10dot,z11dot,z12dot,...
    z14dot,z15dot,z16dot,...
    z18dot,z19dot,z20dot,...
    z22dot,z23dot,z24dot,...
    z26dot,z27dot,z28dot,...
    z30dot,z31dot,z32dot,...
    z34dot,z35dot,z36dot,...
    z38dot,z39dot,z40dot,...
    slowdown]';
```

```matlab
%Written by Frances Withrow December 2016
%Reading raw output to calculate response variables
%Redundant to info saved in main file to ensure correct calcs

%Reading a specific text file
z = textread('control.txt');


t = [0:1:3000];

[alpharich, alphadiv] = alphacalc(z);
[gammarich, gammadiv] = gammacalc(z);


ass1alpharich = [alpharich(:,1:10)];
ass1alphadiv  = [alphadiv(:,1:10)];
avgass1rich = mean(ass1alpharich);
avgass1div = mean(ass1alphadiv);

totavgass1rich = mean(avgass1rich);
totavgass1div = mean(avgass1div);

ass2alpharich = [alpharich(:,11:20)];
ass2alphadiv  = [alphadiv(:,11:20)];
avgass2rich = mean(ass2alpharich);
avgass2div = mean(ass2alphadiv);

totavgass2rich = mean(avgass2rich);
totavgass2div = mean(avgass2div);

ass3alpharich = [alpharich(:,21:30)];
ass3alphadiv  = [alphadiv(:,21:30)];
avgass3rich = mean(ass3alpharich);
avgass3div = mean(ass3alphadiv);

totavgass3rich = mean(avgass3rich);
totavgass3div = mean(avgass3div);

writealpharich = [totavgass1rich; totavgass2rich; totavgass3rich];
writealphadiv = [totavgass1div; totavgass2div; totavgass3div];

ass1gammarich = [gammarich(1:10)];
ass1gammadiv  = [gammadiv(1:10)];
```

101

```
ass2gammarich = [gammarich(11:20)];
ass2gammadiv  = [gammadiv(11:20)];

ass3gammarich = [gammarich(21:30)];
ass3gammadiv  = [gammadiv(21:30)];


ass1betarich = ass1gammarich./avgass1rich;
ass1betadiv = ass1gammadiv./avgass1div;

totavgass1betarich = mean(ass1betarich);
totavgass1betadiv = mean(ass1betadiv);

ass2betarich = ass2gammarich./avgass2rich;
ass2betadiv = ass2gammadiv./avgass2div;

totavgass2betarich = mean(ass2betarich);
totavgass2betadiv = mean(ass2betadiv);

ass3betarich = ass3gammarich./avgass3rich;
ass3betadiv = ass3gammadiv./avgass3div;

totavgass3betarich = mean(ass3betarich);
totavgass3betadiv = mean(ass3betadiv);

writebetarich = [totavgass1betarich; totavgass2betarich; totavgass3betarich];
writebetadiv = [totavgass1betadiv; totavgass2betadiv; totavgass3betadiv];
```