A GULF OF OIL

The Effectiveness of Chemical Dispersants for Oil Spill Cleanup

TEAM #6
NEW MEXICO SUPERCOMPUTING CHALLENGE FINAL REPORT
ALBUQUERQUE ACADEMY

Team Members:
Joseph GRIEGO 10th
Geelon SO 10th
Eric LI 9th

Team Sponsor:
Mr. Jim MIMS

April 6, 2011

Abstract

Purpose The purpose of this model is to find the effects of oil dispersants on surface waters and the ecological impacts of their use. We will use our model to determine how the application of chemical dispersants can be used to eliminate large surface oil slicks and when they should be applied. The drawback to oil dispersants is the adverse effect on coral reefs by dispersed oil. With our model, we hope to determine the benefits and harms of the use of dispersants.

Procedures A model that factors in the effects of gravity, buoyancy, and density into the three-dimensional vector field to show the movement of a particle from position A to position B will be run twice. First, the model will determine the movement of water without additional forces. This models the oil with chemical dispersants because chemical dispersants make oil molecules act like water molecules. Second, the model will determine the movement of water with modified intermolecular forces and buoyancy. This models the oil without dispersants.

Results and Conclusions When we compare our results from a test with high concentrations of dispersants and the results from a test with low concentrations of dispersants, we found that the effects on surface slicks was not very dramatic but still substantial. While the comparison between the model's output images does not indicate a large change, the use of dispersants is still valuable insofar as it still substantially reduces the size of apparent oil slicks. In this sense, dispersants are valuable in reducing the ecological impact of oil slicks by minimizing damages to surface dwelling ocean life. However, no conclusions could be made about the effect of dispersed oil on subsurface ecosystems, as no data was collected for the toxicity of chemical dispersants to other life, such as corals. But, with our assumption that a concentration level below 0.1ppm is an acceptable level, we determined whether chemical dispersants should be used.

Contents

1 Executive Summary

Oil spills cause much damage to the environment and economy. Cleanup of oil spills are also costly, time-consuming, and will not completely remove the problem. In fact, at times, the method of cleanup can also negatively affect the environment. One of the most controversial methods is the use chemical dispersants to eliminate large oil slicks. Dispersants themselves are not only toxic to aquatic life but the oil they disperse also can cause damage over a larger area. However, chemical dispersants spread the oil over a larger area, lowering the concentration of oil. If the concentration of oil can be reduced enough to levels that do not affect valuable ecosystems such as coral reefs much, then the use of chemical dispersants can be advised. This model will deal with a generalized ocean area.

A model of ocean currents simulated the movement of oil with and without chemical dispersants. A model of such can help oil companies and the government determine the benefits of using chemical dispersants based on the quantity of oil, the current of the ocean, and the distance from coral reefs. This model will primarily focus on surface level oil slicks and the efficacy of dispersants in dispelling the slicks. By comparing oil slick size with and without chemical dispersants, we will evaluate the ability of chemical dispersants to reduce or dispel oil slicks, thus aiding the cleanup of surface level oil spills.

This project unites the many branches of science such as chemistry, incorporating intermolecular interaction into a macroscopic world, biology, using studies in marine biology to determine toxicity levels, and mathematics and physics to model fluid flow with factors such as gravity and buoyancy. All of this is combined in a visualization model written in C. This model along with future research will help scientists and decision-makers to weigh the outcome of using chemical dispersants.

2 Introduction

Our modern world is a busy place with great energy consumption. A large source of the energy we use come from crude oil, which, according to the Institute for Energy Research (IEA), accounts for 37% of energy demand in the United States. [1] Unfortunately, the world has seen many oil spills, and even with newer technology in recent decades, many oils spills have caused great damage because of either a fault in oil spill prevention, or an ineffective method of oil spill cleanup. Most still remember the spills, Amoco Cadiz, Argo Merchant, Burmah Agate, Exxon Valdez, Ixtoc I, Cosco Busan, and, of course, the Deepwater Horizon Oil Spill.

2.1 Problem Statement

How can data describing the ocean currents be efficiently applied to find the direction and magnitude of oil flow from source in order to find how much oil ends up where? How can this model be used so that decisions on the use of chemical dispersants are appropriate, effective, and beneficial?

2.2 Objective

The purpose of this project is to create a visualization model to demonstrate the movement of oil through a body of water with and without chemical dispersants. This model can be used to determine whether the use of chemical dispersants would be advisable. This will be determined based on the concentration of the oil in the water, whether the amount of oil is above or below acceptable levels and where the oil moves. This program should reduce costly effects of chemical dispersants used by weighing the benefits and disadvantages.

2.3 Topic Introduction

2.3.1 Environmental Impacts of Oil Spills

The major effects of oil on life and the environment stems from the toxicity of oil. Many birds and marine life can die from complications caused by the contact of oil in addition to the inherent

---

[1] http://www.instituteforenergyresearch.org/energy-overview/petroleum-oil/

toxic effects of oil. These complications include hypothermia, asphyxiation, weak immune system, and the inability to sense or escape from predators. As one part of the food chain is disrupted, a balance between predator and prey is broken, and, in extreme cases, can lead to extinction. Oil spills also impact coral reefs. Coral reefs are not motile, and can easily become contaminated. Oil, apart from being toxic to the animal life in coral reefs, can cause obscure the water, preventing the photosynthetic processes of symbiotic algae. This leads to the infamous coral bleaching, causing whole regions of coral to die off.

2.3.2 Economic Impacts of Oil Spills

Not only are trillions of dollars spent in oil spill cleanup, companies can also lose billions of dollars from spilled oil. Many fishing communities depend on fish in a region that might be contaminated with oil. This too can lead to economic stagnation. Coral reefs shelter many species of fish, and the fishing industry also rely on fishing in regions in and around coral reefs. Loss of fish, once again, can impact the economy negatively. Oil and fishing industries are not the only ones affected. Tourism is another industry affected adversely.

2.3.3 Oil Spill Mitigation

There are currently several methods of clean up. These include:

Bioremediation is the use of biological agents to break down oil in water.

Controlled Burning uses fires to reduce the amount oil in water.

Skimming is done by a machine that separates oil from water because of density differences.

Vacuum and centrifugation , which collects the oil by centrifuging the mixture of oil and sea water.

There are both pros and cons to these solutions. These include:

Bioremediation cannot remedy all contamination, especially heavy metals.

Controlled burning can only be done in calm weather and it releases much toxic pollutants into the air.

Skimming needs calm water in order to be effective.

Vacuum and centrifugation releases some traces of oil along with the released seawater.

There is another common method to deal with oil spills: chemical dispersants. They are detergents added to the water, reducing oil into smaller sizes and spreading oil out more. They are used to improve the aesthetics of the water because the oil is *dispersed* throughout a greater volume, yet, chemical dispersants and the oil they carry can cause harm to marine life. An assessment must be made. One must compare the adverse effects from highly concentrated oil in a relatively small region, or the adverse effects from lesser concentration of oil in a larger region.

2.3.4 Dispersants

The concept of dispersants is illustrated well through oil and water. Oil is composed of nonpolar molecules. Water, on the other hand, is composed of polar molecules. Therefore, when oil and water combine, they do not mix, hence, "like dissolves like." When two liquids do not mix, they are called *immiscible*. A dispersant is a molecule which exhibits both polar and nonpolar properties. When the dispersant is mixed with the oil and water, the oil binds to the nonpolar end and the water binds to the polar end. Therefore, a dispersant is like the bridge that connects two riverbanks; it allows intermolecular interaction between the oil and water. The dispersant, or its more general name, surfactant, surrounds a molecule of oil with its hydrophobic tail, while leaving an outside layer of polar heads to interact with the surrounding water. This formation is called a *micelle*, and the micelle can interact with water exactly like any other polar molecules. Dispersants are used in automobiles, material sciences, detergents, and many other manufacturing industries. They are used as well to deal with oil spills. Dispersants used in oil spills break up the oil and the oil mixes in water, so that the oil is carried out through the different currents found in the ocean and the concentration of oil is reduced.

2.3.5 Dispersants Used in Oil Spills

It can seem counterproductive to use dispersants to cleanup an oil spill. It seems that adding dispersants to cause the oil to spread out throughout the ocean does not help clean up the mess. However, there are other factors that play a role in the decision to use chemical dispersants. When the environmental damages can be reduced when chemical dispersants are used, they will be. A large part of the decision made is by looking at the effects of the surface oil slick. Animals with fur and feather are particularly susceptible to adverse impacts from oil spills. The oil, interfering with homeostatic regulation, can often cause these animal to die of hypothermia. Chemical dispersants are also used when close to the beach. Where many mangrove habitats, estuaries, and important coastal site exist nearby, chemical dispersants can prevent the oil slick from reaching the shore. Additionally, in rough waters, where controlled burning, skimming, vacuuming and centrifuging are not effective, chemical dispersants can provide some cleanup.

2.3.6 Coral Reefs

Coral is composed of small animals called polyps. These polyps have a mutualistic symbiotic relationship with a group of protozoans, zooxanthellae. The polyps create the familiar coral structure by excreting calcium carbonate, $CaCO_3$, into a skeleton. This protects not only the polyps, but the zooxanthellae as well. In return for the shelter, zooxanthellae use photosynthesis to create food for both symbionts. In a healthy environment, both the polyps and zooxanthellae flourish, but when the coral is stressed, it will either expel the zooxanthellae or digest it. In doing so, however, because the zooxanthellae provides the coral with food, both polyp and zooxanthellae then die. This event is what is commonly known as "coral bleaching," named due to the loss of coral color due to the loss of the pigmented zooxanthellae. These circumstances occur with even slight changes in pH levels, temperature, salinity, nutrient levels, concentration of chemicals, and the clarity of the water. The impacts of coral bleaching are wide-reaching, affecting some of the world's richest ocean ecosystems.

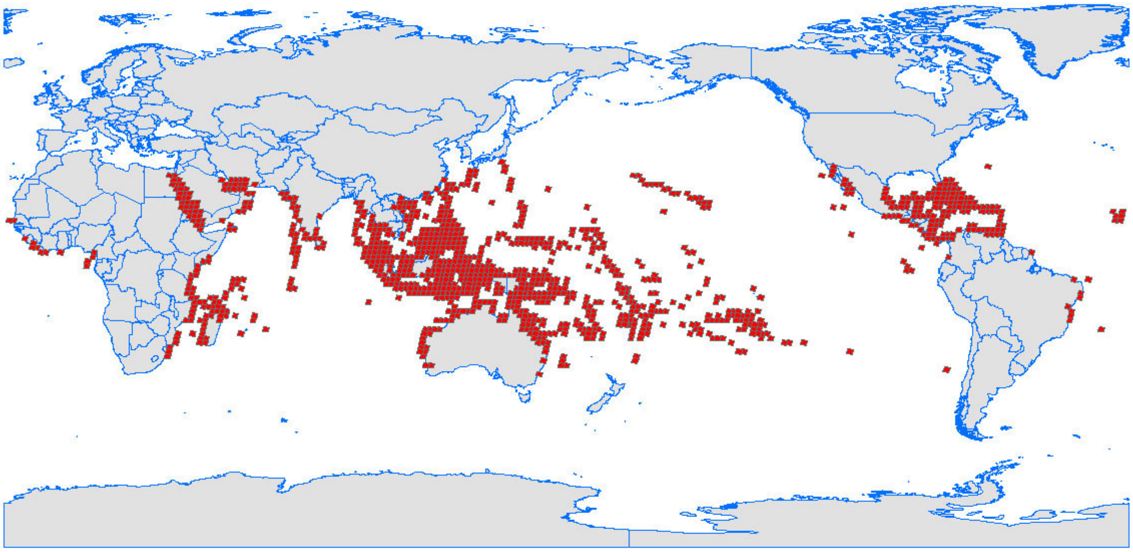
Other potential impacts of bleaching is the loss of coral reefs as a source of food, as coral reefs support many coastal economies. Coral reefs also protect the coastlines from hurricanes, tropical storms, and heavy waves. Additionally, corals are being increasingly used as a source for biomedical chemicals.[2]

---

[2] http://response.restoration.noaa.gov

2.3.7 The Focus on Coral Reefs

Because coral reefs are so important to so many parts of the environment and economy, we focus on the effects of dispersed oil on coral reefs. Using coral reefs as a criterion to base our decision of whether chemical dispersants should be use eliminate many other factors that would complicate this precursory study. As our study progresses, more criteria can be added to provide an even broader picture. In addition to simplifying our problem, because coral reefs exist in so many parts of the world, this model is designed to be able to be able to give rough estimates of the effects of dispersed oil for any part of the world provided that current data is given. Below is a map that shows regions with coral reefs.



Locations of Coral Reefs in the World

http://www.coralreefinfo.com/images/coralreef_map_large.jpg

2.4 Mathematics Introduction

2.4.1 Stochastic Modeling

A stochastic model is one, where, like a normal numeric model, some step is applied repeatedly in order to approximate some physical phenomenon. In a stochastic model, however, the same result is not reached with identical runs of the program; that is, an element of randomness is introduced. Stochastic models are useful where there does not exist a good analytic method for achieving the desired result, or where that method would be unwieldy, too slow, or unstable. This was the case for two steps in our simulation (as discussed below.)

While the general definition of stochastic models encompasses a broad topic, the specific type used in our simulation computes a certain probability, which we will call a *transition probability* to express the probability of a movement of oil, and then repeatedly moved the oil according the those computed probabilities.

2.4.2 Diffusion

Diffusion is a process by which a certain quantity, over time, 'distributes' itself throughout its containing space until, after an infinite time has passed, the quantity is equal at all points in space. This is a highly general definition, but in our project, this is the movement of oil from more concentrated spaces to less concentrated ones. Diffusion is observed in a number of places, from physical diffusion like we are discussing, to diffusion of heat through a substance, to diffusion of electric charge through a conductor. All of these forms of diffusion, however, are governed by the same equation, the diffusion equation.

$$\frac{\partial \psi}{\partial t} = D \nabla^2 \psi \tag{1}$$

Here, ψ represents any scalar field, a numerically-valued function of location in space and time, D represents the diffusion coefficient (with dimensions of $\frac{\text{length}^2}{\text{time}}$) and ∇^2 represents the Laplacian operator. Using subscripts to denote derivatives, the equation could be rewritten as follows. (Notice that the right-hand side contains only space derivatives of ψ)

$$\psi_t = D\left(\psi_{xx} + \psi_{yy} + \psi_{zz}\right)$$

Despite its apparent simplicity, this equation can be notoriously difficult to solve, and is likely impossible to solve analytically in more than one space dimension.

2.4.3 Advection

Advection, is, like diffusion, a movement of a given scalar field through a containing space. However, advection is not caused by a gradient in the distribution of the scalar field, but rather, by a velocity field within the space. In the context of fluid mechanics, advection is the process by which a moving fluid will carry a substance added to it. While not generally a major factor in the short term of the oil's lifespan, over time, its effects generally add up and can displace a large amount of oil. It, like diffusion, is also governed by a nasty partial differential equation, but the equation for advection is first order (only contains first derivatives,) and is as follows.

$$\frac{\partial \psi}{\partial t} = \vec{u} \cdot (-\nabla \psi) \tag{2}$$

Again, ψ represents a scalar field. $\vec{u}$ represents the velocity field of the containing fluid, $\cdot$ represents the vector dot product, and ∇ represents the gradient operator. Again, using subscripts to denote derivatives, the equation is as follows.

$$\psi_t = -\vec{u}_x \psi_x - \vec{u}_y \psi_y - \vec{u}_z \psi_z$$

3 Procedures

3.1 Overview

Data from Hybrid Coordinate Ocean Model (HYCOM) regarding the currents in the Gulf of Mexico gives the movement of water at different depths at different intervals. Using this data, a simple model of circulation can be run. Also factored into this are gravity, buoyancy, and density. This model will be run twice. First, the model ran without oil represents the circulation of oil with chemical dispersants. Because dispersants cause the oil to form micelles and act as polar molecules, they will be mixed into the surrounding water, virtually indistinguishable. Therefore, we are able to compare the results ran from our model without oil to that of oil with chemical dispersants. Second, the model ran with oil represents its flow without any effects of dispersants. For both trials, we will "follow" the movement of the volume of fluid comparable to the volume released in the recent Deepwater Horizon Oil Spill, 779,000,000 liters of oil. [3]

We can then compare the two results and find the concentration of oil at point x. We can define an arbitrary region to be a coral reef, and if point x in this region finds two quantity of oil, one higher, one lower, then, we are able to tell whether or not chemical dispersants are beneficial. Additionally, we can also find the distance from the source of oil so that the amount of oil are below the accepted levels.

3.2 General Structure

At the lowest level, our simulation is an agent-based simulation. Each 'agent' represents a single parcel of oil with a given mass. In each time step, agents (parcels of oil) are released from a given point underwater and allowed to move in the virtual ocean. The two major forces (besides buoyancy, the greatest force when oil is not at the surface) acting upon the oil are advection and diffusion. Both of these are modeled with a stochastic method, as discussed below. In each time step, all three forces are considered, and the boundary conditions (quantity of oil and oil location) are applied. This procedure is repeated as many times as needed.

We use the NetCDF library from Unidata to read the data from the HYCOM project.

3.3 General Stochastic Method

We used a method, which, instead of considering each parcel (agent) of oil, considered the entire ocean as a scalar field, where a density ρ is assigned to each chunk of ocean in the NetCDF dataset (which we'll call a *voxel* in this report.) We compute a quantity proportional to the theoretical time derivative and call it the probability that a given parcel of oil will leave the current voxel. We then move the agents according to these computed probabilities. We can verify that this probability is theoretically accurate by computing the expected value for the change in oil concentration in the

---

[3] http://www.pbs.org/newshour/rundown/2010/08/new-estimate-puts-oil-leak-at-49-million-barrels.html

given voxel, and verifying that it gives an answer that will approximate the derivative as given by the equation.

3.4 Computing Diffusion

Equation 1 shows the equation that governs diffusion when ρ is substituted for ψ and the Laplacian operator is expanded:

$$\frac{\partial\rho}{\partial t} = D\left(\frac{\partial^2\rho}{\partial x^2} + \frac{\partial^2\rho}{\partial y^2} + \frac{\partial^2\rho}{\partial z^2}\right)$$

For the purposes of this model, we took a symmetric difference to find the second space derivatives. Since $d\rho$ is approximately equal to the difference between two adjacent values of ρ. So,

$$d\rho = \rho_{+x} - \rho_0$$

The symmetric difference for the second derivative ($d^2\rho$) would be equal to the difference between two adjacent values of $d\rho$. So,

$$d^2\rho = (\rho_{+x} - \rho_0) - (\rho_0 - \rho_{-x}) = \rho_{+x} + \rho_{-x} - 2\rho_0$$

If we apply this substitution to Equation 1, we can get a real mess, but if we make the definition

$$(d^2\rho)_\delta = \rho_\delta - \rho_0$$

then we can split up the equation considerably. (A bit on notation: We denote the value of ρ at the neighboring cell in the $+x$ direction as ρ_{+x}, and in the previous equation, δ represents any one of $\pm x, \pm y, \pm z$.)

Note at this point that

$$\frac{\partial^2 \rho}{\partial x^2} \approx \frac{\left(d^2\rho\right)_{+x} + \left(d^2\rho\right)_{-x}}{\left(\Delta x\right)^2}$$

$$\frac{\partial \rho}{\partial t} \approx D\left(\frac{\left(d^2\rho\right)_{+x} + \left(d^2\rho\right)_{-x}}{\left(\Delta x\right)^2} + \frac{\left(d^2\rho\right)_{+y} + \left(d^2\rho\right)_{-y}}{\left(\Delta y\right)^2} + \frac{\left(d^2\rho\right)_{+z} + \left(d^2\rho\right)_{-z}}{\left(\Delta z\right)^2}\right) \tag{3}$$

Because we wish to compute the left hand side using a stochastic process as described above, we want to split it up as follows.

$$r_\delta = D\frac{\left(d^2\rho\right)_\delta}{\left(\Delta l\right)^2}$$

Where Δl is the space step in the appropriate direction. Because we want to approximate the actual change in density $\Delta\rho$ instead of $\frac{\partial \rho}{\partial t}$, we need to multiply by Δt.

We now want to nudge this value of r so that the expected value of the simulation ends up being $\Delta\rho$. The value gained in density from one agent moving away from a cell is $-\frac{m}{V}$ where m is the mass of a single agent, and V is the volume of the voxel. In addition, for each cell, a given transition (movement of an agent from cell to another) may be tested multiple times. This number happens to be (aptly) the number of agents in the cell, which is equal to $\frac{\rho V}{m}$. These are the adjustments to the probability necessary to make the expected value equal $\Delta\rho$.

$$\begin{aligned} P\left(\rho_0 \to \rho_\delta\right) &= r_\delta \left(\frac{V}{m}\right)\left(\frac{m}{\rho V}\right)\Delta t \\ P\left(\rho_0 \to \rho_\delta\right) &= D\frac{\left(\rho_\delta - \rho_0\right)\Delta t}{\rho\left(\Delta l\right)^2} \end{aligned} \tag{4}$$

Where $P\left(\rho_0 \to \rho_\delta\right)$ denotes the transition probability from the cell for ρ_0 to the cell for ρ_δ

We can verify this formula by computing the expected value of the stochastic process.

$$
\begin{aligned}
E = \sum v_i p_i &= \sum -\frac{m}{V} r_\delta \frac{V}{m} \frac{m}{\rho V} \Delta t \\
&= -\sum r_\delta \frac{m}{\rho V} \Delta t \\
&= -\sum_\delta r_\delta \Delta t \\
&\approx -\frac{\partial \rho}{\partial t} \Delta t \\
&\approx -\Delta \rho
\end{aligned}
$$

Note that the negative sign is correct, because the expected value is the amount of oil *leaving* the voxel.

3.5 Computing Advection

We may derive a formula for the transition probabilities for advection by an almost identical process, except that the first derivative is computed as the average for the derivatives of neighboring cells. The resulting formula is as follows.

$$P\left(\rho_0 \to \rho_\delta\right) = \frac{\vec{u}_l \left(\rho_\delta - \rho_0\right) \Delta t}{\Delta l}$$

3.6 Special Considerations for Δt and Δl

One must be extremely careful when choosing time and space steps, Δt, and Δl. Because the stochastic simulation only considers the cells *immediately* adjacent to the given cell, one must ensure that the new distribution for the oil lies completely within the given cell and its immediate neighbors. This keeps the transition probabilities below 1. (If the agents are supposed to be moved several cells in one direction, the probability of them transitioning out of the current cell will be *greater than 1*, which obviously can't be used.)

To choose values for our time steps, we must make them obey these approximate equalities.

For diffusion:

$$1 \approx \frac{D\Delta t}{(\Delta l)^2}$$

For advection:

$$1 \approx \frac{U\Delta t}{\Delta l}$$

Where U equals the approximate maximum magnitude for the velocity field.

Only under these conditions will the stochastic model give reasonably accurate results. Ideally, Δt will be smaller than the value computed here.

3.7 Computing Buoyancy

Buoyancy is computing using a simple formula for the upward acceleration of the particle, given as follows.

$$a = \frac{g\,(m - m_d)}{m + m_d} \tag{5}$$

Where m is the mass of the oil, m_d is the displaced mass of water, and g is standard gravitational acceleration.[4]

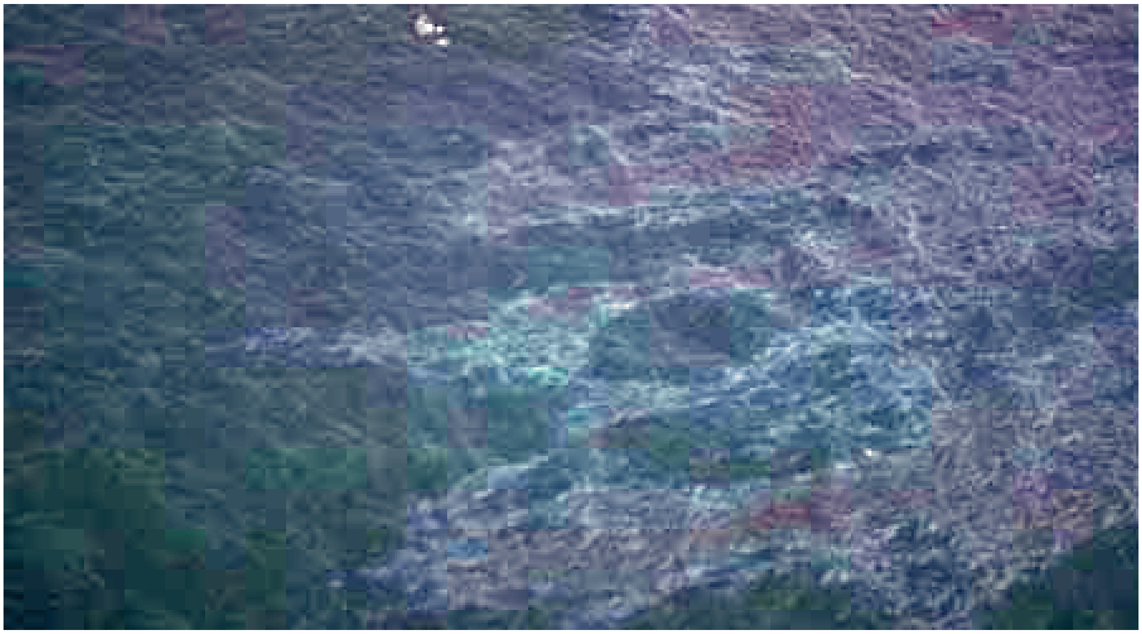
4 Discussion

4.1 Coral Reefs and Oil

It is very easy to demonstrate the appeal of dispersants to apply to oil spills. As these two photographs show, dispersants greatly improve the surface aesthetics of the water. In addition, using dispersants reduce the number of animals who contact oil slick, which can cause hypothermia, poor immune system, poisoning, and death.

---

[4] http://www.physicsmyths.org.uk/buoyancy.htm



Photograph of oil in water before chemical dispersants was applied.

http://response.restoration.noaa.gov/photo_album/photos/173_slick1.jpg

Photograph of oil in water one hour after chemical dispersants was applied.

http://response.restoration.noaa.gov/photo_album/photos/174_slick2.jpg

Yet, there are no doubts of disadvantages of using chemical dispersants. For now, we will look at the effects of oil only in coral reefs.

There are several ways that oil affect coral reefs. The oil itself can cause the polyp to atrophy or die. The polyp is weakened, unable to sustain itself in otherwise normal situations. For example, the polyp is less efficient at feeding itself and building up the calcium carbonate skeleton. Unable to withstand the changes in the delicate balance in the environment's the coral can die. The oil also diminishes the photosynthetic capacity of zooxanthellae; therefore, a major component of the energy source for coral is removed.

Of course, there are many more factors that play into how oil spills affect the health of the coral reef. For example, if the oil spill occurs when the coral is reproducing, the health of a coral reef may be affected for years to come. The duration and intensity of an oil spill also impacts the damage. A low concentration of oil for a long period of time may damage the coral reef just as much as a high concentration of oil for a shorter period of time.

Because of the complexity of all the factors that affect the well-being of coral reefs, we will focus on a basic and broad criterion. We will define acceptable toxicity levels as under 0.1 ppm. [5] This statistic is the line below which we define as amount of oil that will not noticeably damage a coral reef. Our results will compare to this and will determine the benefits of using chemical dispersants.

4.2 Analysis of Results

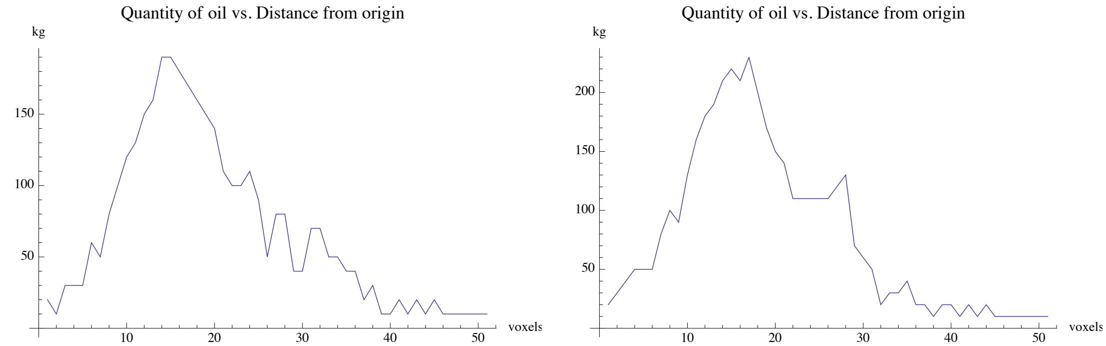
Images for the simulation run with dispersants (left) and without, with the time span set to 2 days.

---

[5] http://www.jstor.org/pss/4312725

Images for the simulation run with dispersants (left) and without, with the time span set to 10 days.

Images for the simulation run with dispersants (left) and without with the time span set to 20 days.



Graphs showing the falloff of oil quantity around the initial release of oil with dispersants (left) and without.

Despite their obvious potential to do so, we found in our simulation that an increased dispersion rate actually did little to change the overall distribution of oil at the surface. Our simulation did not account for changes in subsurface dynamics caused by dispersants, and so it is difficult to say precisely, without knowing exactly how dispersants alter the fluid dynamical properties of water-oil mixtures, how much dispersants would affect distributions even a few hundred meters below the surface. Because of this, our results are somewhat limited in scope, especially when taken in the light that the effects of dispersants are not directly observable on such a large scale.

4.3 Real World Relationship

Through the use of this model, it has been determined that dispersants still play a controversial role in oil spill cleanup. This model supports the efficacy of dispersants in reducing the size of surface oil slicks—a key advantage. The use of chemical dispersants then conceivably allows humans to reduce the ecological impacts of an oil spill at the surface level. By dispelling large surface oil slicks, mortality rates of birds, turtles and various other sea creatures caught in slicks can then be reduced. The next step is to evaluate the ecological impacts of chemical dispersants and dispersed oil on subsurface ecosystems and animals. This would provide another valuable piece of information in evaluating the utility of chemical dispersants by giving more information on the toxic effects on chemical dispersant and oil micelles on animals that exist far below the surface, such as corals.

Therefore, to determine whether chemical dispersants should be used, we converted our above data from kilograms of oil per liter of water into parts per million, and compared it to our "cut-off" point at which all levels above is toxic to coral reefs. The result for the maximum amount of concentration was 2.083ppm for our test with chemical dispersants. The result for oil by itself was 2.501ppm. In both cases, the level was dangerously high. Therefore, because we have not considered the toxic effects of chemical dispersants itself, we will draw the conclusion that benefits in using chemical dispersants outweigh the harms. As we can see from the above data, the use of chemical dispersants reduce the maximum concentration of oil by 20%. Even through 2.083ppm exceeds our earlier "cut-off" point of 0.1ppm, it seems that chemical dispersants do benefit coral reefs. Yet on the other hand, since we have not considered subsurface mixing much, we cannot compare the effects of the chemical dispersant on coral reefs that are significantly lower than the surface of the water. As we add more factors to this with our further research, we can draw better conclusions which will be increasingly more "life-like."

5 Further Research

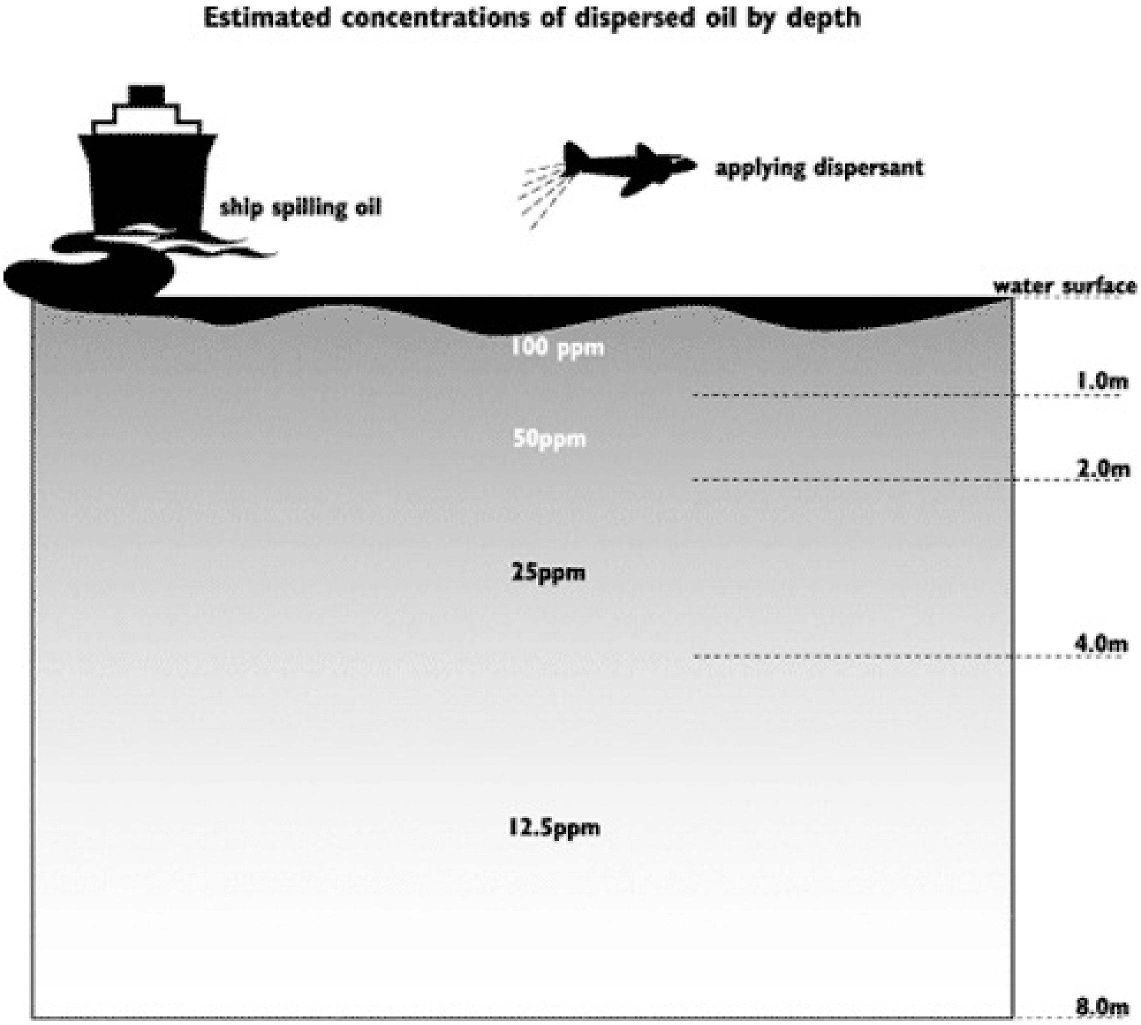
5.1 Analysis of Coding Method

While potentially not the *best* means of solving this problem when placed in the context of differential equations, a stochastic method offers a great deal of flexibility and stability when applied many times. However, in our implementation, the time and space steps had to be delicately balanced to yield transition probabilities in a correct range, and when not balanced this way, led to some numeric instability and physical inaccuracy. We also observed in the data produced, a bias towards the east-west axis, probably occurring in the dispersion step, since it spreads oil both with and against the current. This bias is stronger with a larger simulation time, and probably occurs because of an imbalance between the time and space steps and other parameters.

Also, because this method often requires larger time and space steps than can accurately represent smaller-scale ocean phenomena, it may also be difficult to adapt to include other factors such subsurface mixing and various other medium-scale phenomena. Our code also runs in quadratic time (Linear time with respect to number of agents) and linear space, making it difficult to run on a personal computer.

5.2 Analysis of Scientific Applications

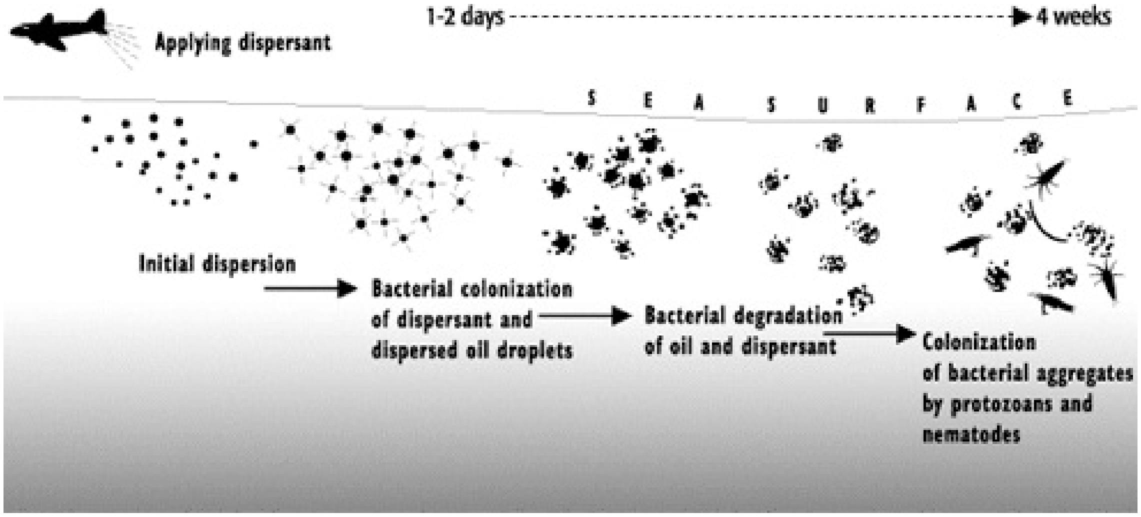
Our model has the potential to become more precise and effective to determine the costs and benefits to using chemical dispersants. Currently, we account for mainly surface currents with some depth, but no subsurface mixing. This is a valid assumption for this first model. As our work progresses, we can begin to apply increasingly more factors. There are many factors before the applications of our model can be used precisely in the real world. A main one is subsurface mixing. Because dispersants cause oil molecules to split, there is little differences in density between the oil and seawater mixture and seawater. Therefore, the oil not only diffuses out longitudinally and latitudinally, but also perpendicularly to these two axes. This will effectively cause the concentration of the oil to be lower than our current results because there is more dimension to the results. Below is a diagram to show the amount of of oil at different depths near an oil spill:



The concentration of oil with depth

http://response.restoration.noaa.gov/art_gallery/fig8a.gif

In addition, the oil biodegrades as it travels through the water over time. To improve our model, we can include a factor of amount of biodegration that occurs over time. Therefore, as time progresses, the concentration of oil is reduced increasingly, and the concentration of oil at x distance will be less than our current value. Below is a diagram that shows the extent of biodegradation of oil over time:

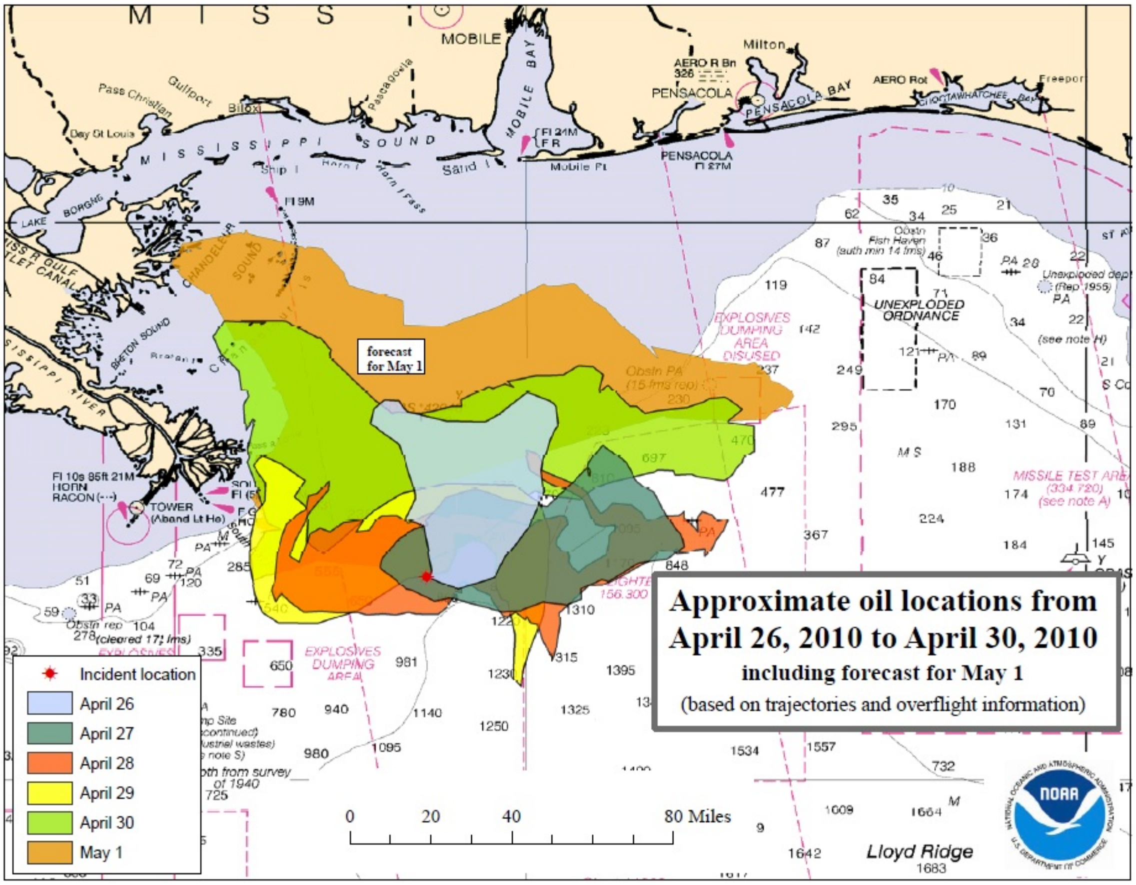


The biodegradation of oil over time

http://response.restoration.noaa.gov/art_gallery/fig9.gif

The time scale is within the general range of oil spill cleanup, therefore, it is important to look at these values, because a substantial amount of oil may have been degraded and the effects of the oil are less on coral reefs.

In our further research, we can also compare our data with these factors with NOAA maps of oil spread from the Deepwater Horizon Oil Spill:



An example of a NOAA oil trajectory map showing the movement of oil over time
http://response.restoration.noaa.gov/book_shelf/1891_TM-2010-04-30-1453.pdf

In the end, we may actually see our current projection of maximum oil concentration to be lowered. The oil will biodegrade, the oil will mix subsurfacely, and as we consider more variables, our results will become better and better.

6 Conclusion

The analysis of the results allows us to determine many of the significant advantages and disadvantages of using oil dispersants. Comparing the images of oil levels in the high dispersant concentration scenario and the low dispersant concentration scenario, we see that although the dispersants may seem to play a relatively minor role in dispersing surface-level oil, it in fact can substantially reduce surface level oil slicks. This major advantage will benefit many different types of surface-dwelling marine life such as seabirds. However, accompanying this advantage is the fact that the dispersion of oil in water causes a large sub-surface homogenous mixture of water and oil. This increase in sub-surface oil concentrations and adversely affect coral through the addition of another environmental stressor. Had dispersants not been used, oil would have remained mostly confined to the upper layers of the ocean, preventing direct contact with the coral. Therefore, we conclude that chemical dispersants are beneficial to coral reefs if the coral reefs are close to the surface, but this suggests that using chemical dispersants for coral reefs lower than the surface of the water will actually harm the reefs.

One problem was encountered with our simulation was the x-axis bias of the constructed algorithm. A trend can be noted through examining the images—areas of concentrated surface oil seem to be oriented more along the x-axis of the images as opposed to the y-axis. We have also determined that the skewed distribution of the oil is not a result of oceanic currents simply because none of the currents we were dealing with had sufficient force to pull out the oil slicks in such a shape. We can further improve the project by eliminating this bias from the constructed algorithms. Also, a possible improvement is to take into account subsurface mixing that takes place as oil slicks spread. This adds to the accuracy of the model and also takes into account another variable that has the potential to affect the general spread of oil through an area.

In the end, however, based on our 'black-and-white' analysis of the effects on chemical dispersants based on the concenctration, we have determined that chemical dispersants should be used if the coral reefs are at or close to the surface of the water, as chemical dispersants can actually decrease the maximum concentration of oil by 20%. We cannot give any judgement one way or another if the coral reefs are significantly below the water. However, this conclusion does not factor in the toxic effects of the chemical dispersants itself. As mentioned in further research, we will work to incorporate more or the major factors that also look at the effects of *not* adding chemical dispersants in order to draw a conclusion that is closer to the real-world than our current model.

References

[1] Buchheim, Jason. “Coral Reef Bleaching” *Odyssey Expeditions* http://www.marinebiology.org/coralbleaching.htm. Web.

[2] Jernelov, Arne “Ixtoc I: A Case Study of the World’s Largest Oil Spill” *Royal Swedish Academy of Sciences* http://www.jstor.org/pss/4312725. Web

[3] NOAA. “Trajectory Products by Date” *National Oceanic and Atmospheric Administration* http://response.restoration.noaa.gov. Web.

[4] NOAA. “Toxicity of Oil to Reef-Building Corals: A Spill Response Perspective” *National Oceanic and Atmospheric Administration* http://response.restoration.noaa.gov/book_shelf/1_coral_tox.pdf. Web

[5] Pacanowski, R. C. and Philander, S.G.H “Paramateriation of Vertical Mixing in Numerical Models of Tropical Oceans” *Journal of Physical Oceanography* Vol. 11, November 1981. Web.

[6] Yender, Ruth A. “Oil Spills in Coral Reefs: Planning and Considerations” *National Oceanic and Atmospheric Administration* http://response.restoration.noaa.gov/book_shelf/70_coral_full_report.pdf. Web.

[7] Zavala-Hidalgo, Morey, Stephen L. and James J. O’Brien “Seasonal circulation on the western shelf of the Gulf of Mexico using a high-resolution numerical model” *Journal of Geophysical Research* Vol. 108 No. C12, 2003. Web.

Acknowledgments

We would like to thank our sponsor, Mr. Jim Mims for his guidance in our project.

Data for the current in the Gulf of Mexico was from HYCOM.

This report was typeset in LaTeX.

7 Appenxicies

7.1 Appendix A: Code

```
// main.c
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#include "ncdf.h"
#include <stdlib.h>
#include "agent.h"
#include <stdio.h>
#include "constants.h"


int  main(){

    db_initialize();
    srandomdev();
    struct agent* agents=malloc(sizeof(struct agent)*100000);
    int agentcount=0;
    boundary(agents, &agentcount);

    double * densities = malloc(sizeof(double) * latxlon*no_depth);
    for(int i=1;i<5;i++){
        db_open_day(i);
        compute_densities(agents, densities, agentcount);

        for(int j=0;j<agentcount;j++){
            for(int k=0;k<10;k++){
                disperse(agents+j, densities, 100, seconds_day/10);
```

```
                compute_densities(agents, densities, agentcount);
                advect(agents+j, densities, seconds_day/10);
            }

        }
        db_closeday();
        buoy(agents, seconds_day, agentcount);
        boundary(agents, &agentcount);
        printf("%d\n",i);
    }
    printf("%d", agentcount);
    compute_densities(agents, densities, agentcount);
    db_openoutput();
    db_write_density(0, densities);
    db_closeoutput();

    for(int i=0;i<agentcount;i++){
        struct voxel v = agents[i].position;
        printf("%d,␣%f,␣%f,␣%f\n", i, v.lat, v.lon, v.depth);
    }
}
```

```
// ncdf.h
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#ifndef _NCDF_H
#define _NCDF_H

#include "datatypes.h"
#include "constants.h"
```

```
void db_initialize();

struct vel db_getvel(struct voxel position, float t);
struct vel db_getvel_ind(struct ncdfcoord, float t);

void db_open_day(int day);

void db_openoutput();
void db_writevoxel(struct voxel position, struct vel v, float t);
void db_write_density(int t, double * d);
void db_closeoutput();

void db_get_whole_vel(float * u, float * v, float * w);

struct ncdfcoord get_true_coords(struct voxel p);
struct voxel get_physical_coords(struct ncdfcoord c);

void db_closeday();

double db_get_ddepth(int depth);

float depthar[40];

char bool[latxlon*no_depth];
int depthindex;

int returns;

int isthere(struct ncdfcoord c);

```

```

#endif
```

```c
// ncdf.c
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#include "ncdf.h"

#include "constants.h"
#include <netcdf.h>
#include <stdio.h>
#include <math.h>
#include <string.h>

int ncid;
int depth_id;
int status, outputid=0;

#define handle_error(stat) printf("HOSHIT:%s\n",nc_strerror(stat))

void db_initialize(){
    status=nc_open("/Users/joe/sccdata/uvel/001.nc", 0, &ncid);
    status=nc_inq_varid(ncid, "Depth", &depth_id);
    status=nc_get_var_float(ncid, depth_id, depthar);
}

int uidf, vidf, widf;           int uidc, vidc, widc;
int uvaridf, vvaridf, wvaridf;  int uvaridc, vvaridc, wvaridc;
int lat_var_id, lon_var_id;

```

```
void db_open_day(int t){
    char upathf[70], vpathf[70], wpathf[70];
    sprintf(upathf, ”/Users/joe/sccdata/uvel/%03d.nc”, (int)floorf(t));
    sprintf(vpathf, ”/Users/joe/sccdata/vvel/%03d.nc”, (int)floorf(t));
    sprintf(wpathf, ”/Users/joe/sccdata/wvel/%03d.nc”, (int)floorf(t));

    char upathc[70], vpathc[70], wpathc[70];
    sprintf(upathc, ”/Users/joe/sccdata/uvel/%03d.nc”, (int)ceilf(t));
    sprintf(vpathc, ”/Users/joe/sccdata/vvel/%03d.nc”, (int)ceilf(t));
    sprintf(wpathc, ”/Users/joe/sccdata/wvel/%03d.nc”, (int)ceilf(t));

    status&=nc_open(upathf, 0, &uidf);
    status&=nc_open(vpathf, 0, &vidf);
    status&=nc_open(wpathf, 0, &widf);

    status&=nc_open(upathc, 0, &uidc);
    status&=nc_open(vpathc, 0, &vidc);
    status&=nc_open(wpathc, 0, &widc);

    status&=nc_inq_varid(uidf, ”u”, &uvaridf);
    status&=nc_inq_varid(vidf, ”v”, &vvaridf);
    status&=nc_inq_varid(widf, ”w_velocity”, &wvaridf);

    status&=nc_inq_varid(uidc, ”u”, &uvaridc);
    status&=nc_inq_varid(vidc, ”v”, &vvaridc);
    status&=nc_inq_varid(widc, ”w_velocity”, &wvaridc);

    nc_inq_varid(uidc, ”Latitude”, &lat_var_id);
    nc_inq_varid(uidc, ”Longitude”, &lon_var_id);

    if(status){
```

```
        handle_error(status);

    }
}

int outputid;
int lat_id, lon_id, depth_id, uvel_id, vvel_id, wvel_id, vt_id, density_id;
int tindex=0;
int outfileno=0;

void db_openoutput(){


    tindex=0;
    db_closeoutput();
    char s[15];
    sprintf(s, ”out/%03d.nc”, outfileno++);
    printf(”%s\n”,s);

    status=nc_create(s, 0, &outputid);//check;
    int tid, latdid, londid, depthdid;

    nc_def_dim(outputid, ”t”, NC_UNLIMITED, &tid);//check;
    nc_def_dim(outputid, ”lat”, no_lat, &latdid);
    nc_def_dim(outputid, ”lon”, no_lon, &londid);
    nc_def_dim(outputid, ”depth”, no_depth, &depthdid);

    int dimids[] = {tid, depthdid, latdid, londid};
    int tdim[] = {tid};
    int ladim[] = {latdid};
```

```
    int lodim[] = {londid};
    int dedim[] = {depthdid};

    status=nc_def_var(outputid, ”density”, NC_DOUBLE, 4, dimids, &density_id);
    handle_error(status);


    nc_enddef(outputid);
   // }
}

void db_writevoxel(struct voxel p, struct vel v, float t){
    const size_t locus[] = {tindex++};
    status=nc_put_var1_double(outputid, lat_id, locus, &p.lat);
    status=nc_put_var1_double(outputid, lon_id, locus, &p.lon);
    status=nc_put_var1_double(outputid, depth_id, locus, &p.depth);
    status=nc_put_var1_double(outputid, uvel_id, locus, &v.uvel);
    status=nc_put_var1_double(outputid, vvel_id, locus, &v.vvel);
    status=nc_put_var1_double(outputid, wvel_id, locus, &v.wvel);
    double tt=t;
    status=nc_put_var1_double(outputid, vt_id, locus, &tt);
}

void db_write_density(int t, double * d){
    const size_t start[]={0, 0, 0, 0};
    const size_t counts[]={1, no_depth, no_lat, no_lon};
    status=nc_put_vara_float(outputid, density_id, start, counts, d);
    handle_error(status);
}

```

```
void db_closeoutput(){

    if(outputid){
        nc_sync(outputid);
        nc_close(outputid);
    }
    outputid=0;

}

void db_closeday(){
    nc_close(uidf);
    nc_close(uidc);
    nc_close(vidf);
    nc_close(vidc);
    nc_close(widf);
    nc_close(widc);
}

struct vel db_getvel(struct voxel p, float t){

    struct ncdfcoord c = get_true_coords(p);
    printf("%d,␣%d,␣%d\n", c.lat, c.lon, c.depth);
    struct vel v = db_getvel_ind(c, t);
    return v;
}

struct vel db_getvel_ind(struct ncdfcoord c, float t){

    if((!(c.lat>=0&&c.lat<no_lat))||(!(c.lon>=0&&c.lon<no_lon))||(!(c.depth>=0&&c.depth<no_depth))){
```

```
        returns = 29;
        return vel_create(0, 0, 0);
    }


    float uvelf, vvelf, wvelf;
    float uvelc, vvelc, wvelc;

    const size_t coord[] = {0, c.depth, c.lat, c.lon};

    status=nc_get_var1_float(uidf, uvaridf, coord, &uvelf);
    status=nc_get_var1_float(vidf, vvaridf, coord, &vvelf);
    status=nc_get_var1_float(widf, wvaridf, coord, &wvelf);

    status=nc_get_var1_float(uidc, uvaridc, coord, &uvelc);
    status=nc_get_var1_float(vidc, vvaridc, coord, &vvelc);
    status=nc_get_var1_float(widc, wvaridc, coord, &wvelc);



    if(status){
        handle_error(status);
        returns=status;
        return vel_create(0, 0, 0);
    }


    if(uvelf>1.25e30f||uvelc>1.25e30f||vvelf>1.25e30f||vvelc>1.25e30f||wvelf>1.25e30f||wvelc>1.25e30f){
        returns = 24;
        return vel_create(0, 0, 0);
```

```
    }

    float ratio = t-floorf(t);

    double uvelwater, vvelwater, wvelwater; // m/s
    uvelwater = fmin(uvelf,uvelc)+ratio*fabs(uvelf-uvelc);
    vvelwater = fmin(vvelf,vvelc)+ratio*fabs(vvelf-vvelc);
    wvelwater = fmin(wvelf,wvelc)+ratio*fabs(wvelf-wvelc);

    returns=0;
    return vel_create(uvelwater, vvelwater, wvelwater);
}

struct ncdfcoord get_true_coords(struct voxel p){

    float lonind = (p.lon - lon_i) / dlon;
    float latind = (p.lat - lat_i) / dlat;
    int lonif = floorf(lonind);
    int latif = floorf(latind);

    depthindex=0;
    while(depthar[depthindex]<p.depth){
        depthindex++;
    }
    depthindex--;

    return ncdfc_create(latif, lonif, depthindex);
}

void db_get_whole_vel(float * u, float * v, float * w){
    nc_get_var_float(uidf, uvaridf, u);
```

```
    nc_get_var_float(vidf, vvaridf, v);
    nc_get_var_float(widf, wvaridf, w);
}

struct voxel get_physical_coords(struct ncdfcoord c){
    float lat, lon;
    size_t latc[] = {c.lat};
    size_t lonc[] = {c.lon};
    nc_get_var1_float(uidc, lat_var_id, latc, &lat);
    nc_get_var1_float(uidc, lon_var_id, lonc, &lon);

    struct voxel v = voxel_create(lat, lon, depthar[c.depth]);
    return v;
}

double db_get_ddepth(int depth){
    if(depth!=no_depth-1)
        return depthar[depth+1]-depthar[depth];
    return 100;
}

int isthere(struct ncdfcoord c){
    if((!(c.lat>=0&&c.lat<no_lat))||(!(c.lon>=0&&c.lon<no_lon))||(!(c.
       depth>=0&&c.depth<no_depth))){
        return 0;
    }

    if(bool[i3(c.lat,c.lon,c.depth)]==0){
        db_getvel_ind(c,0);
        if(status){
            bool[i3(c.lat,c.lon,c.depth)]=1;
```

```
        }
        else {
            bool[i3(c.lat,c.lon,c.depth)]=2;
        }

    }
    return (bool[i3(c.lat,c.lon,c.depth)]==2?1:0);
}
```

```
// datatypes.h
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#ifndef DATATYPES_H
#define DATATYPES_H

struct voxel{
    double lat;
    double lon;
    double depth;
};

struct vel{
    double uvel;
    double vvel;
    double wvel;
};

struct ncdfcoord{
    int lat;
    int lon;
```

```
    int depth;
};

struct region{
    struct ncdfcoord start;
    struct ncdfcoord end;
};

struct voxel voxel_create(double lat, double lon, double depth);
struct vel vel_create(double u, double v, double w);
struct ncdfcoord ncdfc_create(int i, int j, int k);

#endif
```

```
// datatypes.c
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#include "datatypes.h"

struct voxel voxel_create(double lat, double lon, double depth){
    struct voxel v;
    v.lat=lat;
    v.lon=lon;
    v.depth=depth;
    return v;
}

struct vel vel_create(double u, double v, double w){
    struct vel ve;
    ve.uvel=u;
```

```c
    ve.vvel=v;
    ve.wvel=w;
    return ve;
}

struct ncdfcoord ncdfc_create(int i, int j, int k){
    struct ncdfcoord c;
    c.lat= i;
    c.lon=j;
    c.depth=k;
    return c;
}
```

```c
// agent.h
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#include "datatypes.h"

struct agent{
    struct voxel position;
    struct vel velocity;
};

void compute_densities(struct agent *all_agents, double * densities,
    int count);
void disperse(struct agent *a, double *densities, double d_coeff,
    double dt);
void advect(struct agent *a, double *densities, double dt);

void buoy(struct agent *all_agents, double dt, int count);
```

```

void boundary(struct agent* all_agents, int *count);
```

```
// agent.c
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#include "agent.h"
#include "constants.h"
#include "ncdf.h"
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>

void compute_densities(struct agent *all_agents, double * densities,
    int counts){
    memset(densities, 0, sizeof(double)*latxlon*no_depth);

    for(int i=0;i<counts;i++){

        struct agent a=all_agents[i];
        struct voxel p=a.position;
        struct ncdfcoord c= get_true_coords(p);
        if(!isthere(c))
            continue;

        double dw = db_get_ddepth(c.depth);
        double dvol = dv*dw*du;

        densities[i3(c.lat, c.lon, c.depth)]+=stdmass;
```

```
    }


}

#define transition(i,j,k,dl,delta,dc) if(isthere(ncdfc_create((i),(j),(k))))\
{double p=(d-densities[i3((i),(j),(k))])/(dl/dt * dl/d_coeff * d) * (steps++/neighbors);\
if(p> rreal)\
{delta+=dc;return;}\
}

void disperse(struct agent *a, double * densities, double d_coeff, double dt){
    struct voxel p = a->position;
    struct ncdfcoord c =get_true_coords(p);
    if(!isthere(c))
        return;
    double d = densities[i3(c.lat,c.lon,c.depth)];

    double rreal = (random()%10000000L)/10000000.0;

    int i,j,k;
    i=c.lat; j=c.lon; k=c.depth;

    double neighbors=0;
    neighbors+=isthere(ncdfc_create(i+1,j,k));
    neighbors+=isthere(ncdfc_create(i-1,j,k));
    neighbors+=isthere(ncdfc_create(i,j+1,k));
    neighbors+=isthere(ncdfc_create(i,j-1,k));
```

```
    neighbors+=isthere(ncdfc_create(i,j,k+1));
    neighbors+=isthere(ncdfc_create(i,j,k-1));

    double steps=1;

    double dw = db_get_ddepth(k);
    double dvol = dv*du*dw;


    transition(i+1, j, k, dv/5, a->position.lat, dv/earth_radius*180/
        M_PI/5);
    transition(i-1, j, k, dv/5, a->position.lat, -dv/earth_radius*180/
        M_PI/5);
    transition(i, j+1, k, du/5, a->position.lon, du/earth_radius*180/
        M_PI/5);
    transition(i, j-1, k, du/5, a->position.lon, -du/earth_radius*180/
        M_PI/5);
    transition(i, j, k+1, dw/5, a->position.depth, dw/5);
    transition(i, j, k-1, dw/5, a->position.depth, -dw/5);
}

#define transition2(i,j,k,u,dl,delta,dc) if(isthere(ncdfc_create((i),(j
    ),(k))))\
{double p=(d-densities[i3((i),(j),(k))])*u/(dl/dt * d) * (steps++/
    neighbors);\
if(p> rreal)\
{delta+=dc;return;}\
}

void advect(struct agent *a, double *densities, double dt){
    struct voxel p = a->position;
```

```
    struct ncdfcoord c =get_true_coords(p);
    if(!isthere(c))
        return;
    double d = densities[i3(c.lat,c.lon,c.depth)];

    double rreal = (random()%10000000L)/10000000.0;

    int i,j,k;
    i=c.lat; j=c.lon; k=c.depth;

    double neighbors=0;
    neighbors+=isthere(ncdfc_create(i+1,j,k));
    neighbors+=isthere(ncdfc_create(i-1,j,k));
    neighbors+=isthere(ncdfc_create(i,j+1,k));
    neighbors+=isthere(ncdfc_create(i,j-1,k));
    neighbors+=isthere(ncdfc_create(i,j,k+1));
    neighbors+=isthere(ncdfc_create(i,j,k-1));

    struct vel v = db_getvel_ind(c, 0);

    double steps=1;

    double dw = db_get_ddepth(k);
    double dvol = dv*du*dw;

    transition2(i+1, j, k, v.vvel, dv, a->position.lat, dv/earth_radius*180/M_PI);
    transition2(i-1, j, k, v.vvel, dv, a->position.lat, -dv/earth_radius*180/M_PI);
    transition2(i, j+1, k, v.uvel, du, a->position.lon, du/earth_radius*180/M_PI);
```

```c
    transition2(i, j-1, k, v.uvel, du, a->position.lon, -du/earth_radius*180/M_PI);
    transition2(i, j, k+1, -v.wvel, dw, a->position.depth, dw);
    transition2(i, j, k-1, -v.wvel, dw, a->position.depth, -dw);
}

void buoy(struct agent * all_agents, double dt, int count){
    // a = g(rho V - m) / (m + rho v)
    for(int i=0;i<count;i++){
        if(all_agents[i].position.depth>5.0){
            all_agents[i].position.depth-=all_agents[i].velocity.wvel*dt;
            all_agents[i].velocity.wvel+=(grav_accel*(rho_water*stdmass/rho_oil - stdmass))/(stdmass + rho_water*stdmass/rho_oil)*dt;
        }
    }
}

void boundary(struct agent* all_agents, int *count){
    for(int i=0;i<*count;i++){
        if(all_agents[i].position.depth<5.0){
            all_agents[i].position.depth=1.0;
        }
    }

    if(*count < 100000){
        for(int i=0;i<10&&*count<100000;i++){
            all_agents[*count].position=voxel_create(break_lat, break_lon, break_depth);
            all_agents[*count].velocity=vel_create(0, 0, 10);
```

```
                (*count)++;
            }
        }


}
```

```
// constants.h
// Effectiveness of Dispersants for Oil Spill Cleanup
// Team 6

#ifndef CONSTANTS_H
#define CONSTANTS_H

#define lat_i 18.09165
#define lon_i -98

#define dlat 0.03401 // deg
#define dlon 0.03998 // deg

#define du 4445.57316725 //m
#define dv 3781.73945518 //m

#define no_lat 385
#define no_lon 541
#define no_depth 40

#define latxlon 208285

#define i3(i, j, k) ((j) + (i)*no_lon + (k)*latxlon)

```

```
#define dlatdlon 55190538543 // m^2

#define seconds_day 86400
#define tstep seconds_day/10

#define earth_radius 6371000 // m
#define rho_oil .9 // kg/L
#define rho_water 1 // kg/L
#define grav_accel 9.81 //m/s^2
#define elasticity 10000000 // m/m * kg/kg

#define avg_oil_density .001 // L/L

#define break_lat 26.08
#define break_lon -92.9
#define break_depth 500

//{{-92.9, 26.08}}

#define stdmass 10

#endif
```