**Team Number: 1021 or 80**
**Team Members: Ivan Rodriguez, Erica Ceniceros**

**Sponsoring Teacher: Dale Henderson.**
**School Name: Nuestros Valores Charter High**
**Area of Science: Oceanography**
**Project Title: Tsunami's**

## Table of Contents

Table of Figure

## Abstract

The program is about simulating a tsunami. In this paper we explain the basics of the model, how the model is created and the physics of the model. We have a step by step version of the model. How the program progressed throughout the process. We also explain the mathematical terms for the program. We explain each part of the code what each main point does and what we use it for.

## Introduction

The simulation our team will develop will form a tsunami. “Tsunami” is a Japanese word in which “tsu” means harbor and “nami” means wave. Thus the word means “harbor wave.” a tsunami is a series of waves created when a body of water, such as an ocean, is rapidly displaced. Our project will simulate how a tsunami is formed. Commonly a tsunami is formed when two plates shift forming a fault. The fault changes the volume is changed. This causes the water to move at the bottom of the ocean, and create a wave at the surface of the ocean. The average wave speed of a tsunami is 450 miles per hour. In the program we will center the fault in the middle of the ocean, Thus the ocean will go half left and half right from the center when the fault is activated.

## Model Introduction

We are using an agent based model, Net logo, where the agents are called turtles. Agents interact with other agents and interact with the environment which are called patches in the net logo program. For the tsunami program the turtles are the water. The land is patches. In most programs the patches do tend to be the back round.

## Model Basics

The simulation has agents that act as water molecules or “clumps” of water which form the body of water. That body of water might form a wave with a change of volume. A possibility for the simulation is to get a hole in the middle of the ocean. The turtle in the epicenter will give off its velocity to the turtle molecule directly above it and will also give off its velocity to the other turtle molecules. The transfer of energy through the ocean is lost slowly. This is done by the transfer between the turtles. In mathematical terms 90 and 99 percent of the energy is being retained as the turtle passes its energy to another turtle. In the program we are simulating that water is much easier to move than for example steel. This is because the bonds in the water molecules are much weaker then the bonds of steel. This type of bonding is covalent bond. Which is a form of chemical bonding that is characterized by sharing of pairs of electrons between atoms and other covalent bonds.

Figure 1 **This is the final version of the project.**

## Physics

Water molecules are formed when a Oxygen atom which is a negative joins two hydrogen atoms which are positive. When they join they are H2o this particular molecule is a polar molecule, which means there is a positive side and a negative side. Water molecules join together because they are polar molecules, which mean they are attracted to each other and they move together. That's why in our project when the fault is activated it spreads all over and rises the ocean together and the energy is spread to a dome form.

## Math

Figure 2 the Force

Tan 0 = opp/Adj

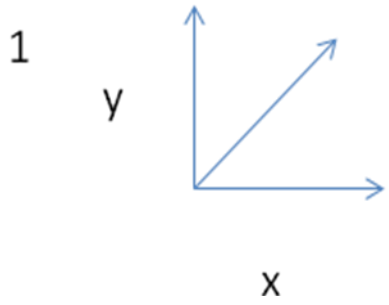
Tan 0 = vy/vx

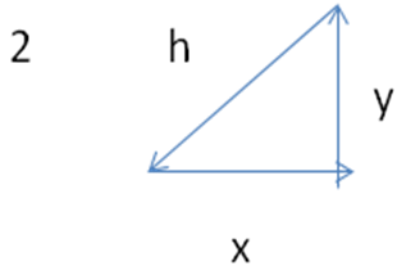
Arc tan (tan0) = arc tan (vy/vx)

0 = arc tan (vy/vx)

Heading = atan vy vx

**Figure 3** the heading goes between the x and y coordinates. Diagram 1 shows y on the left x on bottom. Diagram 2 shows y on the right which makes a triangle and helps solve for arctangent. (Atan)




**How it Works**

So far the simulation forms an ocean by using turtles as water “molecules.” The simulation also forms land under the ocean the same way. There is a fault on the land and when activated it pushes the adjacent turtles upward and eventually spreads all the way to the top. This was done by making the turtles give their velocity to each other until the turtles have reached the top. Then the top of the ocean builds up and simulates a wave. We also decided to put a patch of land on each side of the simulation to see if the tsunami has the water splash onto the land. In our current version the water does not splash onto the land. At the time of this paper this is our major focus of improvement.

The problem was the vy and vx velocity variables. Vy which goes up is working but vx which goes to the side is not working at all. That’s why the wave doesn’t move to the side and go up to the land, 1 cannot be used on vx and vy at the same time. The reason why the wave was just going on one side was that vy needs to be given to both vx and vy which was fixed. Now the wave goes to both sides and is now starting to go onto the land.

## Problem Statement]

What we are trying to create is a tsunami simulation model.

## Objective

Our objective is to make the tsunami look and react like a real wave. It has a fault that triggers the tsunami and has land on the sides so the tsunami can go onto the land such as a realistic tsunami.

## Expected Results

The expected results for the project is for the simulation to look as best as it could to a tsunami. The simulation of the tsunami should go over the land like a tsunami would. The tsunami should also form a realistic wave. A good model has to go back to its original form after it has been run for awhile. So it has to be a regular calm ocean.

## Mechanics of the model

Here are two diagrams of how the flat ocean and then the fault pushes up the water. Then the fault causes a dome shape by radiating the force in a perpendicular fashion. The water gets this dome shape because of the polar bonds of the water molecules. The polar bonds makes the water stick together because negative and positives attract to each other. When one water molecule moves it influences its neighbors to move in a similar direction.

Figure 4

This shows how the fault will make the ocean spread to all sides

## Model versions

In version one there was a patch of land at the bottom and there were ocean turtles wrapped around the screen. For this version the turtles were not moving at all and would stay inside the land instead of on top, and the turtles were lower resolution.

In version two the turtles were starting to stack on top of each other instead of not moving at all. The turtles were gapped so it wasn't very realistic because the ocean is always together not in pieces.

In version three we made the land into gray turtles. It was easier to put the land as a turtle because the turtles would stack on top of the land instead go in to the land. The problem with this version was that the land would not stay down at the bottom it would just go up in the air just like the other turtles. Putting the land as turtles made this work because the turtles were programmed to stack on top of each other.

In version four the fault was put in the program. A specific turtle was set to turn a different color for us to know the fault was working. The specific turtle would also let us know if the velocity was being given to other turtles.

In version five we made the land thicker. We wanted to make the land thicker because it was more realistic to how a land really is.

In version six the turtles turned white if it was given energy. If the turtle had velocity it was passed on to each turtle it would touch. What the problem was that not all turtles were being touched and hence were not getting the velocity.

For version seven the screen was put to be smaller which made the turtles smaller. When the first turtle passed the velocity to the other turtle it was passed on. It looked like the wind was blowing all of the turtles to the right. It did get all turtles to get velocity but the “wave” didn’t seem realistic.

In version eight the turtles started to bunch up instead of stack on to each one another. This made it easier to push the “water” up and make it spread out.

For version nine the fault was put on the left bottom side of the screen so it can push the “water” up and to the right. The problem was that it stopped half way so it wouldn’t go more than half the screen because that is what the program was telling it to do.

For version ten we were able to get the water to move more and give the velocity to most turtles, but the water that went into the air was unable to come down.

In version eleven we added a “land shift” button to simulate the plates moving. The fault was also moved to the middle of the screen and went to the right.

For the twelfth version the ocean was set to go up in a shape of a V and the turtles inside that v had velocity but the turtles outside of the v had no velocity. This worked out because the turtles were going up like water would but wouldn’t come down and if the program was run for a while the whole ocean would just go up and scatter.

The next version the ocean was under control. The fault caused the water to go up then to the right side and when the program was run for a while it would just stay at where it was at. This was a good thing but bad. We need the water to go up to form an ocean but not to go too crazy.

The next version we stretched the screen out and made the ocean bigger. The fault was left in the middle and the water was pushed and it only spread out at the top layer.

The fifteenth version we put land on the side so the water can go up over the land. The water would still go up in a v and the water would go up but it would go all over the screen.

For this version the water was able to be controlled but instead of going on the land it went in through the side of the land, but the water wasn’t going up to form a wave.

Our latest version has been a success so far. We have started experimenting with decreasing the layers of water. We wanted to see how much of a difference it

would make if the water was smaller and see if it would be easier to push up, but it was the same results it stays the same.

## Code Description

### To setup

It makes the land and the ocean. It uses a program which sets them up by transforming the red patches to turtles.

### Convert

Everything that is set up in one color turns into turtles. This also sets the patch of ocean which is half of the screen once it turns red the turtles is made. Once go is pushed the command is to set the patch blue and make the ocean**.**

### To go

Used so all turtles in the ocean can give each other their velocity which is how the wave moves, also adjusting the loss of energy.

### To fall

The fall program makes the turtles that are moving on top of the wave go back down so it can look like a real wave.

### To fault

Sends a turtle straight up and spreads the energy to cause the wave.

### Limit

Keeps all turtles from going above the velocity of one or below – 1. Limit also sets the velocity for each water molecule. The end result is to set the limit on the speed of the molecules.

## Simulation Results

What we have learned from our simulation is that it now acts like a real Tsunami would. When the fault at the bottom of the simulation is activated, it sends energy evenly up and on the sides of ocean. When the energy gets to the top it makes the ocean rise up forming the wave. Then the wave goes to the sides because of the force that is being delivered. The wave then splashes on to the land retreats back to the ocean and calms down again after awhile. This is what a real Tsunami will do in real life.

## Tower Team Project

Our Tsunami Project is similar to the Tower project because it's the same mathematical model, but the tower model has stronger bonds and the tsunami one has weaker bonds. They both have a fault that simulates force being unleashed into a body of turtles. The force move the turtles in both projects.

## Project Summary

We first started our project not knowing if we would even finish it. But our teacher Mr. Henderson helped us a lot. We didn't even know what net logo was or how we were gonna use it. He started by teaching us how to make simple simulations. Like learning what a turtle was, how to get them on the screen and how to make them move. Then it went to harder things like learning how to communicate with each other and use them together to build stuff. Then the projects just got more and more complicated until we learned how to use net logo pretty good. We also learned about our teammates a lot and helped each other even with the other people on different teams.

We started building our project by first getting a big body of water and filling it half the screen. Then we made changed the ocean from one giant body of water to hundreds of little turtles. This way they can communicate with each other and make the ocean move. Then we made a fault which is what pushed the water up and made the wave. We messed with the velocity so the power could be controlled. Then we changed how the force is spread, it is spread in both vx and vy directions. We then added land to the sides of the ocean so the wave could splash on to the ground.

## Code

**breed [ land ]**

**breed [ ocean ]**

**turtles-own**

**[vx**

```
vy
direction
dist]

to setup

ca
 create-land world-width * 3
   [
    set color grey
    set shape "square"
    set size 1.2
    ifelse who > world-width
    [
    setxy who world-height / 2 + 2
    ]
    [
    setxy who world-height / 2 + 1; command on how high the screen is
    ]
    if who > world-width * 2
    [
    setxy who world-height / 2 + 3
    ]
   ]
 ask patches[
```

```
    ;; if patches are between (0,0) to (0,edge)...

     if ( pxcor > 0 - world-width / 2 + 30 and  pxcor <= 70 and pycor >= 3.5 - world-height / 2 and pycor <= -20 )

      [set pcolor blue]   ; THIS IS THE OCEAN                          ;; ... draws left edge in red

      if ( pxcor >  70 and  pxcor < world-width / 2  and pycor >= 0 - world-height / 2 and pycor <= 2 );RIGHT SIDE Of LAND

      [set pcolor grey]

       if ( pxcor > 0 - world-width / 2 and pxcor < 0 - world-width / 2 + 30 and pycor >= -10 - world-height / 3 and pycor <= 2 )

      [set pcolor grey] ;LEFT SIDE OF LAND



    ]

end


to convert

ask patches

[

if pcolor = blue

[

sprout-ocean 1

set pcolor black

]

]

ask turtles

  [

  if breed != land
```

```
    [
    set color blue
    set shape "circle"
    set heading 0
    set size 1.5
    ]
  ]
end



to limit
   if vx >= 1 ; if vx is greater then or equal to 1 it sets vx to 1
   [          ; 1 is the max energy it can have
   set vx 1
   ]
   if vx <= -1 ; if vx is less then or equal to -1 sets vx to -1
   [
   set vx -1
   ]
   if vy >= 1 ;if vy is greater then or equal to it sets vy to 1
   [
   set vy 1
   ]
   if vy <= -1 ; if vy is less then or equal to -1 sets vy to -1
```

```
  [
   set vy -1
  ]
end

to go

;set heading 0
if any? ocean-on patch-left-and-ahead 45 1 ; if there is a trtle in front then
 [
  ask one-of ocean-on patch-left-and-ahead 45  1
   [
   set vx vx - ([vy] of myself) / 2 ; give it our velocity
   set vy vy + ([vy] of myself) / 2 ; give it our velocity
;    set ([vx] of myself) ([vx] of myself)
;    set ([vy] of myself) ([vy] of myself)
   ]
 ]
if any? ocean-on patch-right-and-ahead 45 1 ; if there is a trtle in front then
 [
  ask one-of ocean-on patch-right-and-ahead 45 1
   [
   set vx vx + ([vy] of myself) / 2 ; give it our velocity
   set vy vy + ([vy] of myself) / 2 ; give it our velocity
;    set ([vx] of myself) ([vx] of myself)
```

```
;    set ([vy] of myself) ([vy] of myself)
    ]
  ]
if any? ocean-on patch-ahead 1 ; if there is a trtle in front then
  [
  ask one-of ocean-on patch-ahead 1
    [
;    set vx vx + ([vx] of myself) ; give it our velocity
    set vy vy + ([vy] of myself) ; give it our velocity
    ]
  ]


ifelse (abs vy + abs vx) > 1
  [
  set color sky
  limit
  ]
  [
  set color blue
  ]
;limit


 set heading atan vx  (vy + .000000000000000000000001 );takes vy and adds a
very small numb
 set dist ((abs  vx + abs  vy ) / 3 )
```

```
 if dist < 1
   [
   if[pcolor] of patch-ahead dist = black ;if the color is not black move forward
     [
     set vx vx * loss / 100 ;takes a certain percentage of energy from vx trtl
     set vy vy * .99 ; takes a certain percentage of energy from vy trtl

       ifelse not any? turtles-on patch-ahead dist; if there is a turtle infront don't go
         [
         fd dist ; shows the absolute value of a number, if there is not a turtle go full
force
         ]
         [
         ;fd .1 * abs (( round vx + round vy ) / 2 ) ; shows the absolute value of a
number, if there is a turtle then loose 10 percent of energy.
         ]
       ]
    ]
;set heading atan vx   (vy + .0000000000000000001 ); after looking around tells
them to look back down
gravity
end


to gravity
if breed = ocean
[
```

```
set direction heading ; set of original heading heading
 ;set heading 90 + random 181
  set heading 180
if not any? turtles-on patch-ahead 1 ; if there is a trtle in front stop
    [
   if patch-ahead 1 = black
     [
      fd .1
     ]
    ]
set heading direction ; setting the heading back to the original heading
 ]
end


to fault
if who = 0 ; only for turtle 0
  [
  setxy 0  world-height / 2 + 6
  set color red
  set vx 0 ; sets velocity for x only for 0
  set vy 1 ;sets velocity for y only for 0
   ]
end
```