**Project Report**

**Pocket Tanks**



**Submitted by:**

**Meet Patel: 105116183**

**Manush Thaker: 110034919**

**Aum Patel: 105168607**

**Karan Adhaduk: 110029218**

**ABSTRACT:**

Pocket Tanks is the fast-paced artillery game that is simple to learn, and fun to master.  It is a very interactive game where the players try to hit a target on the other side of wall. In order to win game players uses some different methods like changing the height of the wall, the position of the wall, angles, wind speed as well as power. One can only win a game if the opposition gets bury in a mound of dirt or assail him with a barrage of bullets.

**INTRODUCTION:**

This model is based on the classic game Tank Wars. Game starts with two tanks, where one of the tanks fires on their target of another tank to smash them with fireball, but before reaching to the target fireball needs to over comes some obstacles like wall which is in between both tanks. So, for that player must determine the wall height and wall position. Gravity and wind also play an important role. Now, tank link with guns and when the fireball is shot from the tank, it leaves x and y origin along with velocity. The code prompts the fireball to die as soon as it is reached in the boundaries of the map.

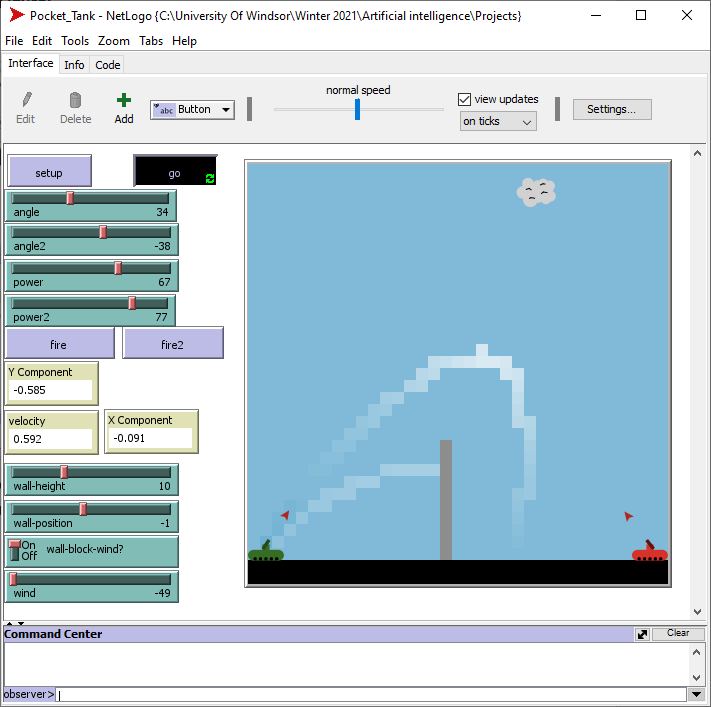
Pocket tank is the perfect quick game which can be play with friends and family, you will find yourself hooked for hours of play!! And from childhood I was curious to know how developers are making this game by creating such small details like the fireballs not going through the walls by identifying patch colors, how the position of the turtle changes when hit by fireballs. We just understand working and applications of tank1 created the game which can be played by two players. The agent Tank is act upon wall height, wall position, wind flow. So as a result, player has to set angle and power accordingly to hit the fireball at designated target.

**Illustration / Figure:**

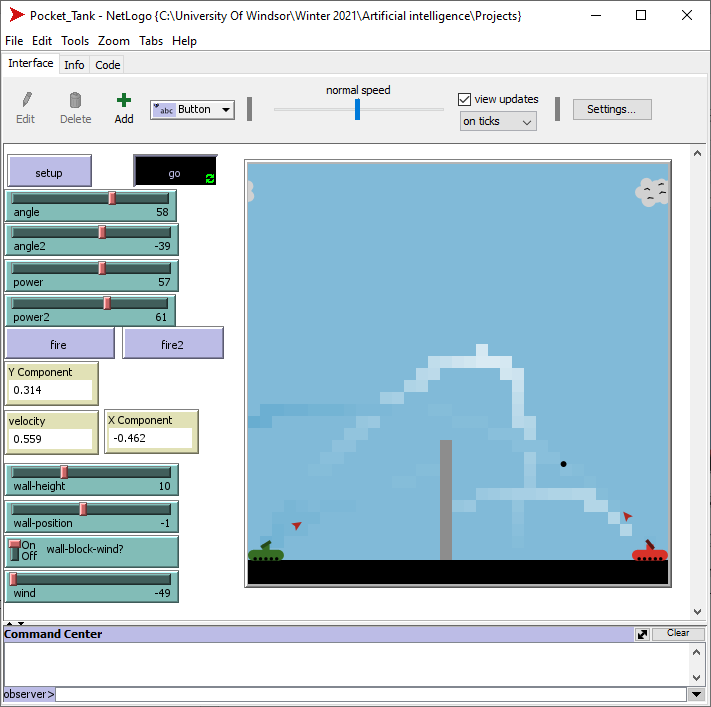
Graphical user interface

Description automatically generated

Figure above displays the Tank 1 firing on the Tanks 2 but unfortunately angle is not high enough for the ball to cross the wall.



This figure depicts the projectile of second fireball fired from Tank 1. This model explains the gravity and wind very accurately as seen in the figure.



This figure shows the firing of Tank 2 which would result in the ball to go upwards missing the Tank 1 because of the wind effect. Power is same for both tanks here, but projectile differentiates greatly for both tanks.

**Simulation Working and Algorithm Explanation:**

* Angle and Angle 2 are a type of slider that will adjust the angle of tank 1 and tank 2 respectively. You can set the angle so that the fireball will hit the other tank precisely.
* Power and Power 2 are slider that will adjust the power of the fireball that will leave from the tanks. So you can either increase or decrease the power of the fireball.
* Fire and Fire 2 are the button which when pressed will the fire the fireball ball from the tanks and it will send the fireball in the angle you set by angle slider and with the power you set in the power slider.
* Wall height is the simple slider which will increase or decrease the height of the wall.
* Wall positon is also a simple slider which move the wall on the x-axis and fix the wall positon before pressing the fire button
* Wall block wind is the on/off switch will when turned on will let the wall to block the wind and when turned off will let the wind to pass through the wall.
* Wind is slider from which the user can adjust in which direction the wind will flow. Which will allow the fireball to move in the direction of fireball.

The working of this model starts from the tanks being idle on both sides of the map. When the user clicks on setup and then go, simulation starts working and the cloud in sky starts moving towards the left with a specific speed indicating the intensity of wind flow in that direction set by user. After starting the simulation user has to set an angle for Tank 1 and the power intensity provided by sliders which would aim the fireball to eventually fall on Tank 2. After hitting the fire button, Tank 1 would fire the fireball from nozzle with each making each patch lighter in color displaying the path of the fireball. There are three more parameters which would show the velocity, x and y origins of any current fireball soaring in the sky. Now the user does the same things for Tank 2 in the process of attacking Tank 1. There is a feature of setting the wall position and height according to user in this simulation and also an option switch button to turn ON or OFF wall-block-wind. This wall block wind switch would rather allow the wind to pass wind through the wall or else wall would not let the wind pass through it.

The algorithm we used for this project was Netlogo supported Scala language. We chose this language because our simulation was in NetLogo software which supports Scala as programming language. We started our code with declaring breeds like cloud, tanks, fireballs (shells) and gun. After declaring the global variables like x, y origins, velocity and previous wall height and position we started the setup process. This is the part where everything except global variables is made to reset. Then we attached the gun nozzle and arrow with the tank so that arrow rotates with the changing angles. We had also inserted check shells which would check for the shells to stop when it hits the tank or whenever it reaches the end of map. We shaped every turtle by using the inbuilt library of shapes in Scala.

**Discussion:**

Our model met the standard set by our team at the start of the project. So, we would say that the model is performing well. We think that it is performing this well because firstly we felt to incorporate the things in the model which we could handle. We also worked tirelessly on the code so that it could work as we wanted it to without having to compromise on any of the parameters we added to project. There are two most surprising and interesting part of our model are the two parameters named Wind and the wall block wind feature we introduced in our projects. We learned during this project how the net logo software works and how to solve errors occurred during our projects.

**Limitation:**

Every projects have its limitation our project is no exception. Some of the limitation of our projects I would like to mention here are:

* The tanks in our project cannot move from its own place.
* Unlike the real Pocket tanks game our project has only one weapon system
* Our project does not have any points on successfully hitting the other tank

**References:**

This project was built in python language and is exactly same the original game

(https://github.com/sm3rta/PocketTanks-AI-Agent).

The below website is the official Wikipedia page for the original game. This is the first ever pocket tanks game built.

(https://en.wikipedia.org/wiki/Pocket\_Tanks).

This is the model named projectile attach in a software called net logo

(http://ccl.northwestern.edu/netlogo/models/ProjectileAttack).