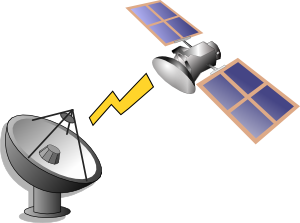
Ahmed Mostafa Shaheen

Mohamed Zahran El-Demerdash

Mini Project

Satellite (BETA 0.1) Program



**Abstract**

We have developed an easy gui software that help the user to compute Antenna alignment parameters and determine the angles for any satellite.

Program features:

1. Ask the user to input the following data: satellite name, type, position, user geographic coordinates, satellite orbital parameters, operating frequency and antenna dish size
2. Allows the user to get his position coordinates from google map.
3. Contains a database of all GeoStationary satellites’ names and positions to allow the user to choose from them.
4. Restrict the user from inputting invalid data format (allows only float numbers).
5. Calculate the satellite look angles, declination angle, distance and round trip time.

**Introduction**

we wrote this code with the help of combination of programming languages like: python2.7, javascript, css and html.

Why python?

It is an easy programming language with huge community that allows us to embed a lot of features to the program such as: gui, google map, 3D rendering and also converting our program into an Android App to use the GPS data of any mobile phone as input the the program.

**The code details**

Firstly, we wrote a python code (a function) that compute the satellite look angles, declination angle, distance and round trip time using the built in math library. Then, we created a sub-program using python to help us to build a dictionary (a data type in python … something like database) to get all satellite names and position, in order to append the names in a drop menu to let the user choose the satellite name and get the position automatically. After that, we created the gui interface for the user using Qt Designer, a program that provide a gui repository for C++, we draw the layout of the gui file and then we used pyuic library to convert the file to a python code that we can modify. We designed the program to restrict the user from inputting invalid data (out of range or string) and allow only float format by using float spin box. After that, we wrote an HTML code to create a web page with the help of javascript and CSS to call Google maps in our program. We programed the map to open by default on Tanta city. Finally, we combined the calculation code with the gui code and the output of the dictionary code and embedded the html file to the gui.

At the end, we converted the python code to an executable (exe) file.

How to use?

First, choose the satellite name that you want to align your dish antenna to, and its longitude will appear automatically.

Second, copy your location from the Google map and paste it to the user information section.

Third, Enter other optional data.

Finally, press calculate

**Conclusion**

The resultant program is a beta program (trial,proof of concept version) that can easily calculate elevation, azimuth and declination angle. And also can calculate the distance between the sat and earth station and the round trip.

Future development:

We can add some extra features like getting your position from gps and 3d model for the antenna direction and some enhancement for the gui shape.

**References:**

satellite database: <http://www.flysat.com/satlist.php>

python programing: <https://www.coursera.org/specializations/python>

Qt Designer: <https://www.youtube.com/playlist?list=PLF4575388795F2531>

google maps: <https://developers.google.com/maps/documentation/javascript/basics>

**Appendix**

Github link:<https://github.com/EngShaheen/Satellite-Antenna-Alignment>

The final code (all the program):

<https://github.com/EngShaheen/Satellite-Antenna-Alignment/blob/master/satellite.py>

The dictionary code:

<https://github.com/EngShaheen/Satellite-Antenna-Alignment/blob/master/Dictionary/dictionary.py>

google map HTML:

<https://github.com/EngShaheen/Satellite-Antenna-Alignment/blob/master/map.html>

