C:\Patent\Satellite-Simulation\2018-10-19-LightningFlashes-A

Lightning Flashes

One of the potential problems with our computer simulation is an easy way to display colorful radio connections and the transmission of data. This and the excellent maps (West Africa, and Egypt) already present will be the heart of the simulation.

Steve, you are the math major, and this is probably already obvious to you, but I have had some thoughts on the process and offer them to you. Here goes as they might be of value. (Please, challenge me on this as I may be in error.)

1) Treat satellites and ground stations as nodes. Write a subroutine that creates a flash between nodes. In future simulations where we will have satellite to satellite links and routes, this will be of prime importance.

2) Node 1 (X1, Y1)

Node 2 (X2, Y2)

Length of flash = (X1-X2) squared plus (Y1-Y2) Squared

And then take the square root of above - Pythagorean Theorem

3) Now, we have a choice of how to display this for maximum impact. This should be fun, as the display can be original and flashy here. It will help in a presentation where we try to raise money for the project.

I am going to suggest a series of dots that move sequentially from Cairo to a satellite and then back earth to one of ten different cities in the general geographic area.

If we for our first presentation of the simulation pretend we have 10 messages arriving at Cairo each second, for one of 10 randomly chosen destinations, we display a message movement during a period of 1/20 of a second. That should give us the ability to keep up, plus work on reducing the queue of unsent messages.

4) Divide the link in 5 (?) different intermediate points. Conceptionally, however, we know the link length, so we then know the distance between the different points.

5) We know delta Y and we know delta X. This comes from step 2 above

6) We now know the length of all three of the line segments that describe the little right angle triangle. We then can derive an ArcTangent or similar method of going from a ratio to an angle.

7) Thus, we can take a step from a transmitting node to a receiving node.

8) Now, the fun part, should each dot last for 1/100 of a second and then display the next dot?

Probably, not, as the user might not see anything. Perhaps, the dot should last for a much longer time, and during that time the next dot appears…………………… etc. etc. etc. and then the first dot dis-appears, etc. etc. etc.

A message should be going to a random location every second.

The program will randomly pick the destination to which the message is to be sent to be sent.