We're Conor Myhrvold and Soren Larson, Harvard graduate students in a new Computational Science & Engineering master's program. We study how to use computers, to solve math and statistics problems that arise in science, in engineering, and now in an increasing extent to business as well. Often this involves setting up simulations -- making a computer model to replicate a key aspect of what you are trying to find out about.

When we read about the disappearance of MH370 this spring, we thought it would be a great real-life example of how to use computer modeling to find out: where is MH370 most *likely* located?

We don't know where it is for sure, but we ought to be able to tell where MH370 is more likely to be, and less likely to be. We can do this because we know several key specifics.

We know where MH370 deviated from its flight path – at the Igari Waypoint. We know where it was last visible to radar – near Pulau Perak – and from the two locations, the direction it was headed during that time period – to the west, and slightly south.

Then we said, let's model where MH370 could be, using tools and techniques we learned this spring in AM207, a Monte Carlo modeling and statistics course taught in Python by the Scientific Program Director of Harvard's Institute for Applied Computational Science himself -- Pavlos Protopapas.

Our project was divided into 3 main method applications. We now describe the first one, a Monte Carlo Model. Here's how it works.

Every hour, MH370 will want to head toward the ping arcs which constrain the plane's position relative to a geostationary satellite over the Indian Ocean – the other key piece of information known to us -- which put the plane along a large arc, at (roughly) hourly intervals after its disappearance from radar. We model these arcs as probabilities, not fixed absolute lines -- error allowance as we don't know the arcs precisely.

At the same time, planes tend to go in more or less the same direction as they're headed if they want to get somewhere. They can turn, but rarely if ever double back on themselves.

So, MH370's heading at each hourly interval will be determined by a combination of its tendency to move toward the satellite ping arcs -- our known geography constraints -- *and* not doing so in a way that would be unrealistic for a plane's flight path, so no quickly doubling back on itself -- our likely heading constraints.

We multiply these probabilities together to create a joint probability distribution and get weighted likelihoods of the plane possibilities in a full 360 degree circle from its current location.

The radius of the circle is the distance MH370 travels, Boeing 777 cruising speed for an hour.

A heading is drawn from these likelihoods, the plane moves, and the process starts over again for the next hour.

We ran 10000 simulations using several different heading constraints and scenarios, which all pointed to the same conclusion: MH370 is much more likely to have gone to the south than to the north – over 80% of the time, in fact.

Also, given the possible satellite ping arcs spacings, MH370 did not *ever* fly in a Great Circle route from its last location, to its most likely final locations off Australia.

For more on why this is important, and Parts 2 and 3 of our project which involve implementing a Hidden Markov Model and a Kalman Filter, check out the IPython Notebooks and writeup on our website.