On RTG Design Hoppe, Sheneman 2015

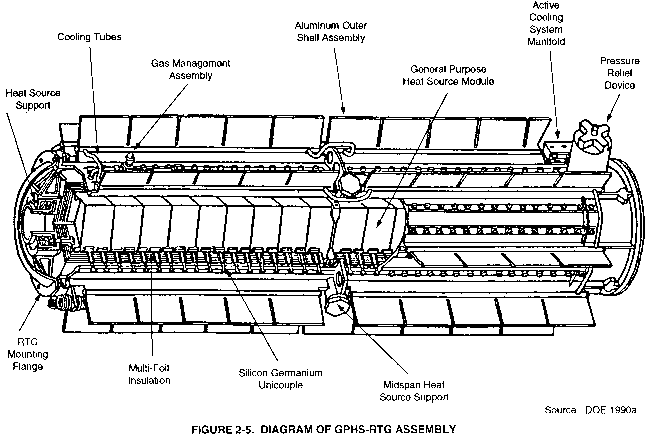
# The Issue

Currently, NASA uses what’s known as the MMRTG (Multi-Mission RTG) to power deep-space missions. These are designed to fit the parameters of multiple missions, which, in a world where it costs over 8 million dollars and most of a year to manufacture a kilogram of 238 Pu, is simply not a cost efficient endeavor. Especially when all of the natural reserves of 238 Pu have been completely depleted by earlier RTG missions[[1]](#footnote-1).

# Our Question

How little plutonium can we use for a given power and mission length?

# The System



This is an RTG. It generates power by conducting heat generated by radioactive decay of plutonium across thermocouples.

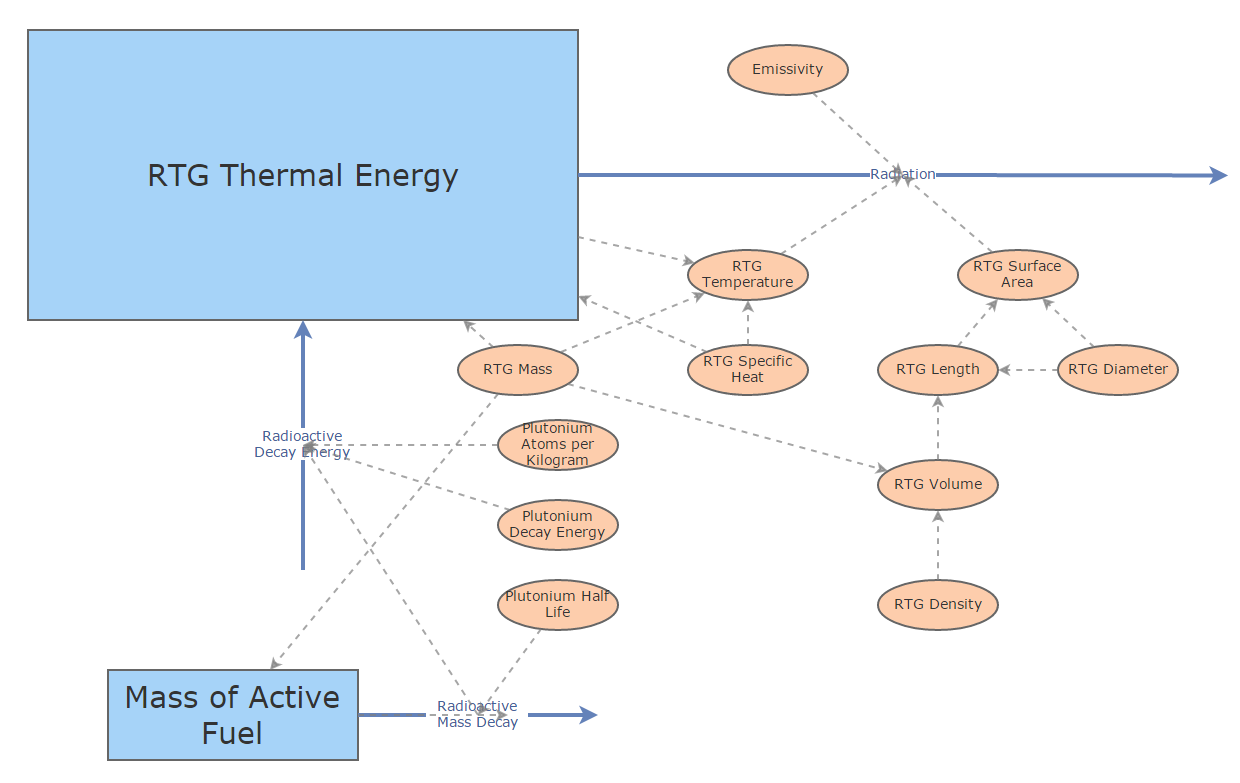
# Our Assumptions

To make our model, we had to make a few assumptions about the situation.

1. The RTG can be modeled as a cylinder of plutonium
2. The only way energy enters the system is by radioactive decay
3. The only ways energy leaves the system are by radiation or as electrical power
4. Electrical power is 6% of the radiation flow
5. The temperature of space is a constant 2 Kelvin

# Our Model

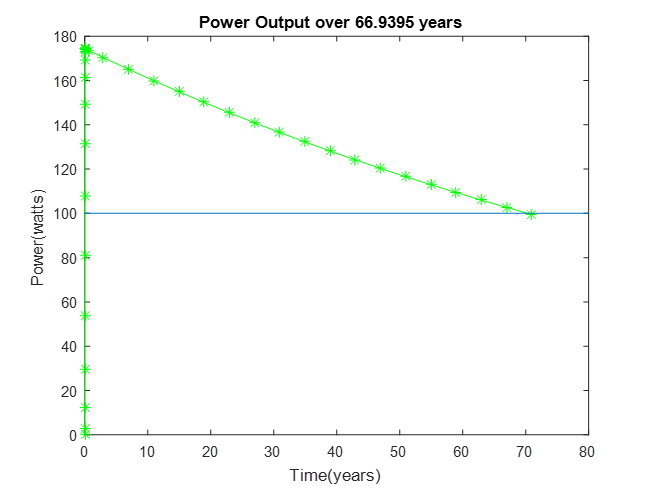
The following is our abstraction of the system. Since all of the heat flows out of the system by radiation or electricity, and we know that about 6% of the heat is captured as electricity, we’ve used a one-stock model for the hot fuel core.



Our stock and flow model. [[2]](#footnote-2)

# Our Equations

# Our Simulation Results

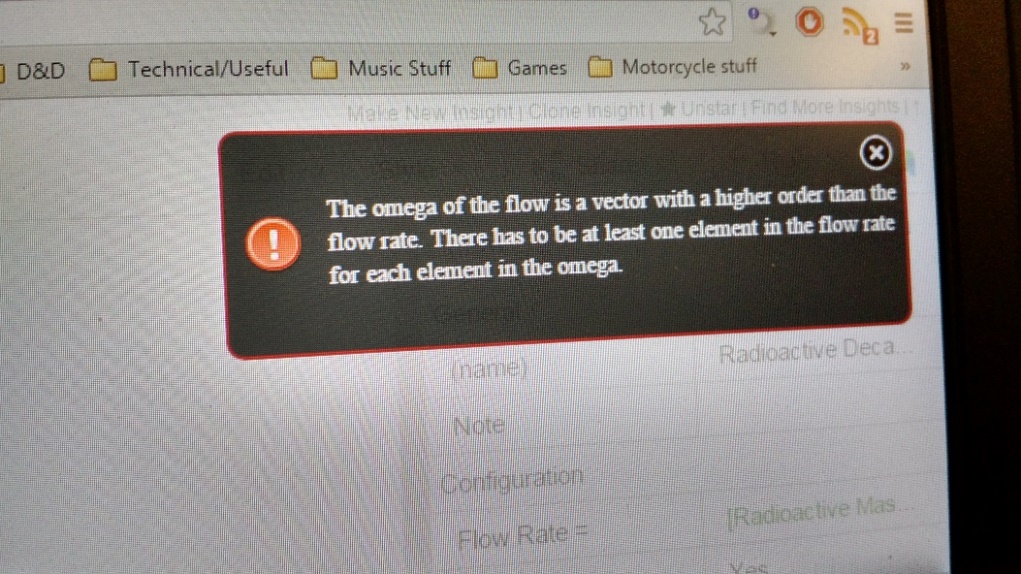


Power Threshold = 100W

The above image shows time series results of one call to our simulation function, which plots the electrical power output of the RTG over time until it reaches the specified threshold. This simulation is called with 4.8 kg of plutonium and a power threshold of 100W.

# Our Failures, and What We Learned

We ran into a wide array of issues with our modeling and simulation of the physical world[[3]](#footnote-3). At first, we ran into a wide variety of problems with Insight Maker, a particularly confounding example of which can be found in the image below.

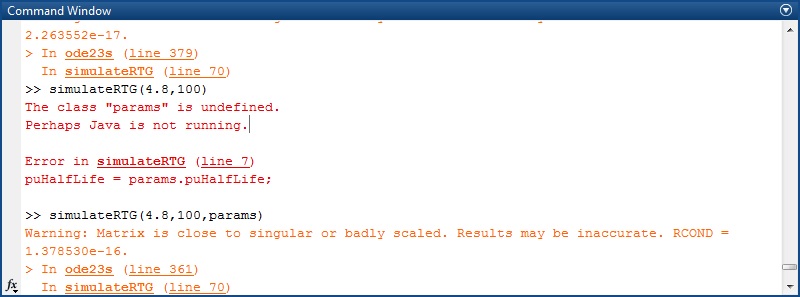


After we abandoned Insight Maker, we built an iteration of our model where the temperature of the RTG would, over the course of three thousandths of a second, plummet to -4 \* 1090 Kelvin, and then onwards to negative infinity. This was a recurring issue, until we resolved it with help from about three ninjas, multiple classmates and Mimi [[4]](#footnote-4).

Once our simulation was behaving somewhat normally, we faced the daunting problem of ode45 deciding that our equations were too stiff[[5]](#footnote-5). This was another major setback, but then we abandoned it for ode23s, which is designed to handle exactly whatever stiffness actually is.

At this point, we were faced with the even more formidable prospect of making an ode23s end case dependent on the results of a flow, *not a stock*. We eventually made it work by abusing a convoluted output function and making some conservative estimates.

Once we had surpassed that obstacle, we faced a completely bewildering error:



So, there’s that.

It was at this point that we realized that our simulation needed to be a function of mass and threshold power, but that our design recommendation would require a function of power and mission duration that would return a mass. 40,000 simulations later, we had a lookup table and a dedicated function to do exactly that: find simulation results that matched the input arguments. Fortunately, it made a *really* cool graph.

We learned a few key concepts from this entire process. First, Insight Maker is a tool that isn’t very useful for more complicated applications; it’s simply too finicky[[6]](#footnote-6). In addition, we learned (the hard way) that certain ODE solvers work much better than others for certain applications. In the future, it will likely be helpful to try many of them, before trying more difficult options. We learned a LOT about how to use ODE solvers, and how to abuse them. Another useful skill we learned from this project is pair programming, which is surprisingly robust in the face of debilitating bugs. All of these are going to be very useful as we move into our next project.

# Our Design Recommendation

As an example of the ramifications that our model could have, we compared the actual plutonium payload of the New Horizons probe to the required payload predicted by our model. New Horizons, which required 200W of power over its 10 year mission, was launched with 8.6 kilograms of 238 Pu. In contrast, our model predicts that an RTG need only contain 7 kilograms of 238 Pu to put out 200W of power after 10 years. This translates to a savings of 1.6 kilos of plutonium, which is worth about 12.8 million dollars. Next time, NASA should definitely consult us first[[7]](#footnote-7).

1. The Cassini-Huygens mission alone used approximately 23.4 kg of 238 Pu. [↑](#footnote-ref-1)
2. NB: electricity flow is not shown (it broke Insight Maker). Our MATLAB model handled this *slightly* more gracefully [↑](#footnote-ref-2)
3. See what we did there? [↑](#footnote-ref-3)
4. Credit where credit is due. [↑](#footnote-ref-4)
5. https://en.wikipedia.org/wiki/Stiff\_equation [↑](#footnote-ref-5)
6. At least for flow-dependent flows. [↑](#footnote-ref-6)
7. We’ll let them know that their RTGs are actually cylinders of plutonium floating in space. [↑](#footnote-ref-7)