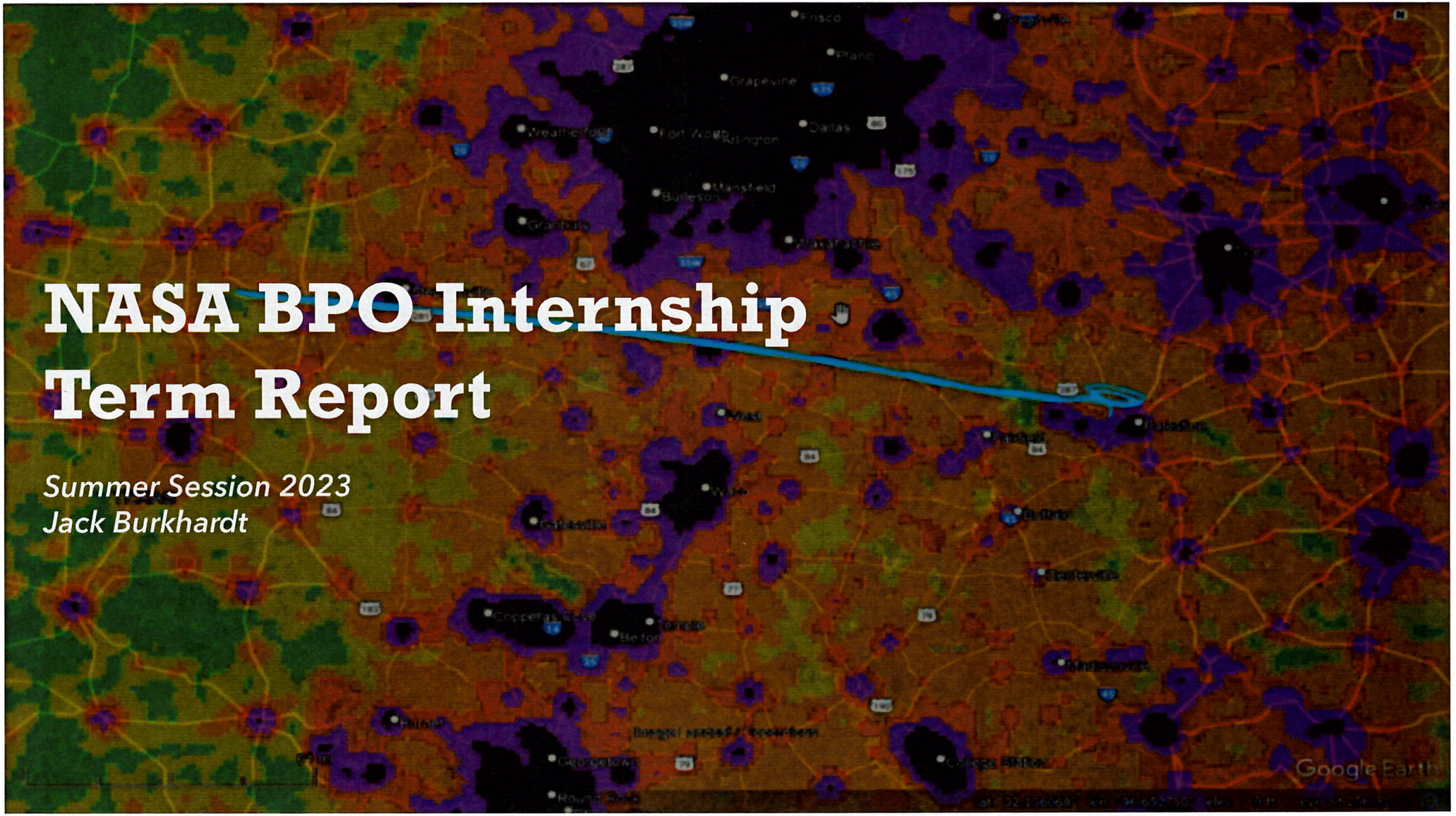


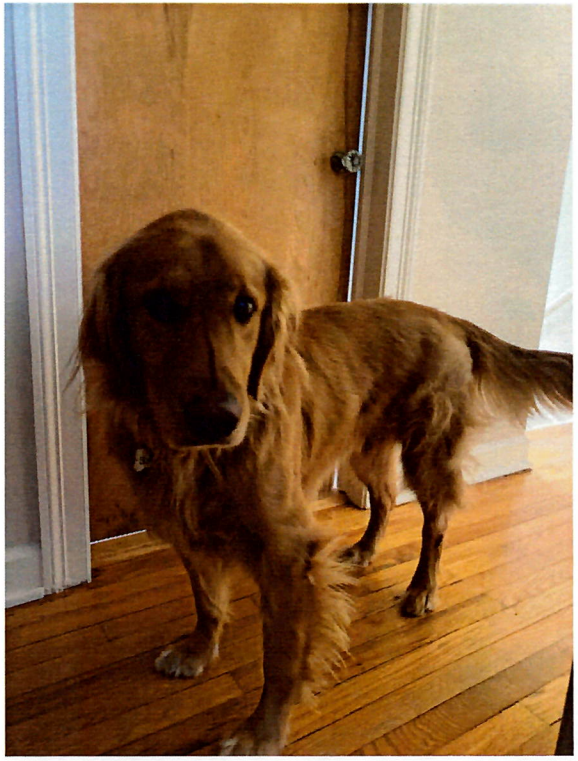
NASA BPO Internship Term Report

Summer Session 2023

Jack Burkhardt

About Me

- From Chicago, IL
- Rising senior at Northwestern University, studying CS
- Sam the Dog – a regular at the ER vet
- Work as a game developer and music festival producer



Project Overview

PortOSim (Portable Object Simulator)

"A versatile software tool for modeling and simulating a variety of launch-range systems and subsystems."

Launch Range Objects	Systems	Forces & Environments
Balloons	Navigation	Gravitational Fields
Satellites	CSBF Operators	Atmosphere
Thrusters	Sensors	Magnetic Fields
Radars	GPS	Springs
Missiles	External Signals	Solar Fluxes

The Problem (and Project)

PortOSim was pushed into balloon risk analysis use without thorough validation!

My job is to dig in, find issues, and build confidence in the tool.

Objectives

Increase Reliability

- Find and eliminate any unexpected points of failure.
- Develop a testing framework to ensure that the simulator is behaving correctly.

Prove Accuracy

- Verify that the math engine and number generators are accurate.
- Graph simulator data and analyze.
- Identify potential issues with the integrator or its inputs.

Understand Behavior

- Model logic and data flows.
- Write robust documentation for all processes.
- Analyze system implementation and compare to the official specifications.

Why is PortOSim useful?

a.k.a. Why is this project important?

Simulators provides a prediction of a system's behavior using provided input parameters and historical risk.

Engineers can model complex interactions between systems.

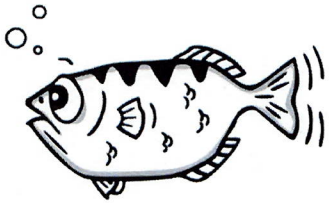
Flight safety teams can understand risk in a variety of scenarios.

Project managers can obtain quick and accurate data to make mission-related decisions.

Increasing Reliability

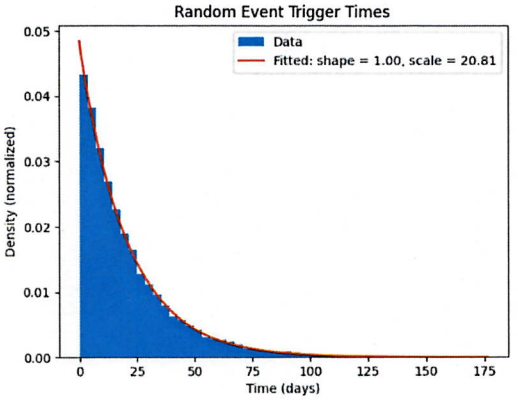
Profiling & debugging tools: Valgrind and GDB

- Great for analyzing software mid-operation
- Industry-standard and well-developed tools
- Used with PortOSim to find memory leaks, uninitialized values, and other errors



Developing a Test Framework

- Written in Python, can run simulations in parallel batches. Uses Matplotlib for graphing and SciPy for numerical analysis.
- After PortOSim is modified, tests can be run to verify correct behavior.
- Provides a basis for confidence in the simulator.

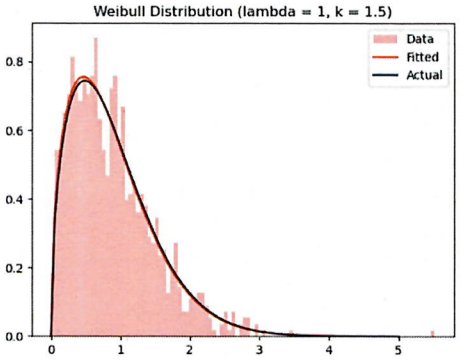


Controlled by: NASA

Proving Accuracy

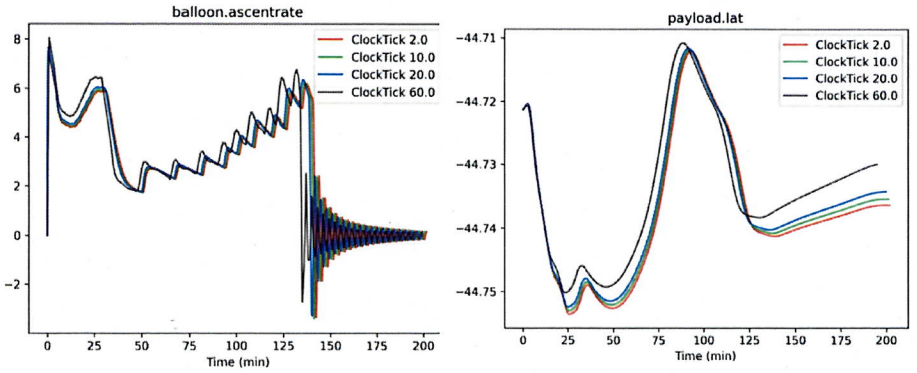
Number Generator Testing & Analysis

- Ran 15k simulations of the number generator with varying input parameters and plotted the resulting data
- Ensures that the data's actual distribution and parameters "fit" the original parameters



Integrator Testing

- The integrator module moves the simulation forward by "time steps" and ensures that the equations remain correct to a defined accuracy
- Ran PortOSim with a variety of "time steps" to measure integrator stability & accuracy



Understanding Behavior

Functional Analysis of the GRIB2 Decoder

- GRIB is a data format for forecast weather data
- Problem – it's all in bytes!
- Compared sim's decoder implementation against the official spec to ensure correctness

• Section 1 - Identification Section *(GribRecord::processSection1)*

Octet Number	Content
1-4	Length of the section in octets (21 or N)
5	Number of the section (1)
6-7	Identification of originating generating center (See Table C-1) (See note 1)
8-9	Identification of originating generating subcenter (See Table C)
10	GRIB master tables version number (currently 2) (See Table 1.0) (See note 1)
11	Version number of GRIB local tables used to augment Master Tables (See Table 1.1)
12	Significance of reference time (See Table 1.2)
13-14	Year (4 digits)
15	Month
16	Day
17	Hour

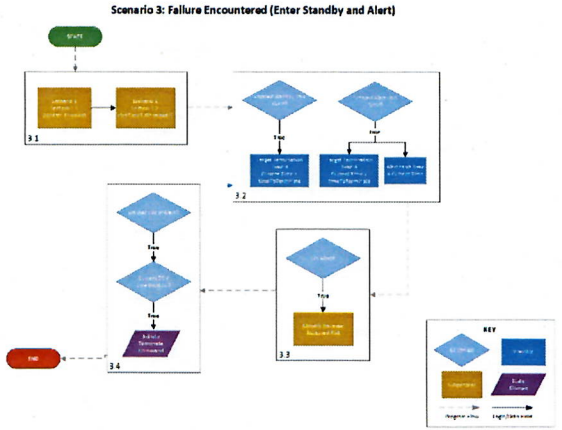
• Section 7 - Data Section *(GribRecord::processSection7)*

Octet Number	Content
1-4	Length of the section in octets (nn)
5	Number of the section (7)
Octet 6-nn	Data in a format described by data Template 7.X, where X is the data representation template number, given in octets 10-11 of Section 5

Notes: If the templateNo in section 5 is not supported (is not 0), then R, E, and D will be undefined. This will cause getDecodedValue to fail because it assumes those three variables are defined and uses them in the return statement.

Flow Charts for CSBF Operator Model

- Important for visualizing and understanding complex, branching logic in programs
- Mapped the flow of logic and data to examine the model's decision making and align it with operational standards



Challenges & Lessons

- **Identifying "correct/incorrect" is difficult**
  - Often hinges on relevant background knowledge (math, physics, etc)
- **Problems can manifest indirectly**
  - Something can look wrong but not be the problem itself
- **Document your work well**
  - Others need to understand it (and you too+ in a few years)
- **Don't be afraid to say "I don't know"!**
  - People here are more than willing to support you

Thank you, BPO & WFF!