Laboratory for Atmospheric and Space Physics
University of Colorado Boulder

Data Systems / Science Data Processing
Scientific Programmer / Software Engineer
Interview Coding Challenge
Revised Sept 2023

Congratulations on making it to this point in our interview process! This coding exercise is an opportunity to demonstrate that you can solve the kind of problems we often encounter and to demonstrate your skills with some of the core technologies we use in our data pipelines.

Please read the entire document before you begin.

Table of Contents

Expectations

We have many roles on our teams and this test addresses the skills for more than one role. As a result, *this coding test may be a significant challenge.* As in scientific research, there is no definitive "correct answer". You are not required to have previous familiarity with Docker, this data, or even spectroscopy; show us how you reason and adapt to unfamiliar problems. If some aspects are unfamiliar, ask questions and focus on the parts you know best.

- We are not trying to hide information and we understand you are not an expert with this dataset.
- Consulting external resources is acceptable.
- All your submitted material must be your own work.
- We expect a summary/report of your results, including plot figures where appropriate.
- Getting the "right answer" is not the focus of this exercise. **Use this as an opportunity to showcase your skills.**

Asking clarifying questions via email is expected but not required.

With your submission, we are trying to get a sense of:

- Presentation/documentation/communication skills
- Coding style and code usability
- Programming language best practices
- Familiarity with foundational Docker principles
- Physical/instrumentation knowledge
- Ability to learn on the fly and apply background knowledge

Scientific Background

The data provided is similar to the SORCE SOLSTICE instrument measurements (though it is a small subset of the wavelength range). The SORCE mission ended in February 2020 after 17 years of on-orbit operations. The SORCE mission measured Solar Irradiance variability over almost 2 complete 11-years Solar cycle.

The SOLSTICE instrument measured the daily Solar Irradiance spectra (Watts/m^2/nm) in the ultra-violet (from 115nm to 300nm). Light entered the aperture, followed the optical path, was dispersed by a diffraction grating, then landed on a photomultiplier tube (PMT) detector which counted the number of photon events detected per integration time (count rate). As the grating rotated, the wavelength of light hitting the PMT changed, allowing for the photon counts as a function of wavelength.

The SORCE spacecraft was in an orbit with roughly a ninety-minute period in Low-Earth Orbit (LEO). Later in the mission, the instrument saw larger than designed temperature swing within each orbit. As the PMT's efficiency in detecting photons changes with temperature, a temperature correction needs to be applied.

Also, during that time period, the grating position was reset with each orbit and never returned to the exact fiducial. This created a small grating offset from orbit to orbit, from what was actually reported, resulting in a wavelength shift. The provided reference spectra (with well-known wavelength and expected irradiances) could be used to align each dataset.

The SOLSTICE instrument ran various experiments on a daily basis throughout the mission. The included data covers about five orbits, with different experiments on each orbit. The various grating scans in this dataset correspond to actual Solar measurements around the Silicon-2 emission lines (~180nm). All the data provided is time-tagged and a subset of the activities are defined in the file plans.txt.

Your Tasks

In no particular order

Containerization

Your data processing algorithm needs to someday run in a serverless cloud environment and must be containerized with Docker. Design your calibration/correction code to run in a Docker container. You may begin by developing your algorithm and then proceed with containerizing it.
Suggestions:

- Use a Dockerfile to automatically install the necessary code and dependencies during image build.
- Implement your algorithm to run “headless” via the container ENTRYPOINT.
- Bind mount your data directory into the algorithm container for both input and output. (This is a simplification for this exercise. In a real setting you would likely read/write data from/to a cloud object store such as S3).
- You may hard-code the location of your input files into your processing code. Remember the location in the container may be different than the location on your filesystem. (This is another simplification).

Data Processing and Analysis

Use any programming language you are comfortable with.

1. Calculate the irradiance in Watts/m^2/nm for the UpScan and DownScan and compare the results.
2. Provide plots of your results along with your code.
    a. Specifically, plot the Irradiance as a function of Wavelength for the UpScan, DownScan and the Reference spectrum around the two emission lines at ~180nm [180 to 183nm].
    b. Calculate and plot the ratio of the Irradiances at each wavelength for each scan with respect to the reference spectrum [180 to 183nm] and provide comments/thoughts about your results?

Brief Report of Results

- Please provide a brief report containing plots and description of your process/results/interpretation.
- Include some technical discussion of what worked, what didn’t, and how you approached the problem.
- Also include a brief section on usage of your Docker image.

Data Files Included

detectorTemp.txt

Detector temperature, in degrees Celsius. It is roughly sampled at 1Hz.

distanceAndDoppler.txt

These are the corrections used to adjust for the changing distance and velocity of the spacecraft relative to the sun. The Doppler correction provided in this file can be ignored for this exercise.

instrumentTelemetry.txt

Includes grating position and measured detector counts. The detector counts correspond to the number of photons detected within the currently set integration time (in milliseconds).

integrationTime.txt

This is a readout of the currently set integration time (milliseconds) of the instrument. These are sampled at a different cadence than the instrument telemetry. Assume the value is constant until there is a new value.

plans.txt

This file includes the experiment names with start/end times. You can find the time ranges of the plans of interest here. [startTime, endTime]

referenceSpectrum.txt

This is a reference spectrum with accurate wavelengths. The current irradiance measurements will be within 15% of this spectrum.

Useful Equations and Parameters

The microsecondsSinceGpsEpoch column is the default time stamp for the telemetry coming from the spacecraft and corresponds to the number of microseconds since 1980-01-06T00:00:00.000.

Wavelength (the grating equation)

```
offset = 239532.38
stepSize = 2.4237772022101214E-6  # [rad/step]
d = 277.77777777777777  # [nm]
phiGInRads = 0.08503244115716374  # [rad]
ang1 = (offset - gratingPosition) * stepSize  # [rad]
wavelength = 2 * d * sin(ang1) * cos(phiGInRads / 2.0)  # [nm]
```

Photon Count Rate (counts/s/m^2/nm)

The sensitivity of the detector changes with temperature and a small correction is needed. This is a correction to the count rate as it deviates from the nominal temperature of 20.0°C at the time of the measurement. The median temperature-corrected dark count rate must then be subtracted from the base count rate for UpScan and DownScan.

Hints:

- Convert integration time from milliseconds to seconds.
- The units for count rate are [counts/s/nm]. The per nm is from sampling at a specific wavelength (via grating position).

```
tempCorrFactor = 0.0061628  # [counts/degC]
count_rate_corr = count_rate * (1.0 + tempCorrFactor * (20.0 - detectorTemp))
dark_count_rate = dark_counts / dark_integrationTime
dark_count_rate_corr = dark_count_rate * (1.0 + tempCorrFactor * (20.0 - detectorTemp))
median_dark_count_rate = median(dark_count_rate_corr)
apertureArea = .01 / (1E2 * 1E2)  # [m^2] (aperature area from cm^2 to m^2)
photonsPerSecondPerM2 = (count_rate_corr - median_dark_count_rate) / apertureArea
[photons/sec/m^2/nm]
```

Spectral Irradiance (W/m^2/nm)

Hints:

- Convert wavelengthInMeters from nm to m so units work out

```
h = 6.62606957E-34  # [J*s]
```

```
c = 299792458.0  # [m/s]
energyPerPhoton = h * c / wavelengthInMeters  # [J]
wattsPerM2 = photonsPerSecondPerM2 * energyPerPhoton  # [watts/m^2/nm]
```

1-AU Correction

It is typical to provide the Solar Irradiances as it would be measured from a standard distance from the SUN (remember these are [W/nm/m^2] of surface area). The standard distance is the average distance between the Earth and the SUN over a full orbit (1 year) known as one Astronomical Unit (AU).

```
wattsPerM2_1AU = wattsPerM2 / sunObserverDistanceCorrection  # [watts/m^2/nm]
```