OBSERVING ULTRAVIOLET RADIATION IN THE STRATOSPHERE

Revision B: Final Analysis

KYLE PETTIT, FLETCHER COLLINS
PUEBLO COMMUNITY COLLEGE
8 December 2018
Revision Log

<table>
<thead>
<tr>
<th>Revision</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Immediate Analysis</td>
<td>12/8/2018</td>
</tr>
<tr>
<td>B</td>
<td>Final Analysis</td>
<td>1/13/2019</td>
</tr>
</tbody>
</table>

Table of Contents

Mission Overview: .................................................................................................................. 2
Mission Requirements: ........................................................................................................... 2
Design: .................................................................................................................................. 3
Management: .......................................................................................................................... 7
Budget: .................................................................................................................................... 7
Test Plan and Results: .............................................................................................................. 7
Expected Results: .................................................................................................................. 8
Launch and Recovery: .............................................................................................................. 9
Results and Conclusions: ......................................................................................................... 10
Mission Overview

In this experiment, we wanted to take a look at the difference in the intensity of the ultra-violet radiation between the ground and the stratosphere. We used a UV sensor with a bandwidth range of 200 nanometers (nm) to 370 nm, which was tied to a high altitude helium balloon provided by Edge of Space Sciences. We wanted to take these ultraviolet readings – received in the form of volts and read through an analog serial port – from both the launch site at 4,800 feet above sea level, and from 111,000 feet, where the balloon reached its maximum altitude. The readings from these different altitudes could have an impact on the way we see our atmosphere. The idea is that if we get the same particle count at both altitudes as historical values, then our atmosphere isn’t changing an extreme amount over time, but if we get fewer particles per square space as historically found, then that may indicate that the atmosphere is growing denser, which could have an implication in global warming.

Mission Requirements

To begin, we needed to do a lot of research. We started with trying to get an idea on what percentage of the atmosphere is between the ground near ocean level and the stratosphere at 100,000 feet up, because that is how far the balloon will travel in altitude. We know from the NASA database [1] that the atmosphere is about 60 miles or 96,561 meters thick. The pressure at sea level is 101,300 Pa from the Pearson textbook [2].

To begin the building of the payload, we had to find a sensor that detects a reasonable range of UV radiation. We chose the Arduino UV Sensor Module that detects wavelengths between 200 and 370 nm, which covers most of the UV spectrum, this is good because it’s right in the middle of the spectrum. The output of this will be an analog signal, so we will have to be able to turn this voltage into an actual particle count. We want also to tilt the box so that the camera is facing slightly upward at the angle the Sun will be over the horizon. At 8:30 A.M on November 10th, 2018, we determined that the sun will be at 17 degrees.

Another thing that we found might be useful is a camera in the payload with an ultraviolet filter on it to make sure we had a visual idea of what the sensor was looking at. We chose to use a Raspberry Pi 3 with a Pi cam to capture these images. With the Pi having its own Micro SD card slot, we took the data and photos on the card. Ideally, we wanted to take a section of 4k ultra video at 30 frames per second, but because of memory card requirements we had to take a photo three times every two seconds.

The biggest requirement was to have a balloon in place to get our payload up the altitude, which was done by the Colorado Space Grant Consortium and the Edge of Space Sciences.
Design

The design aspect of the mission was very simple. We started with a foam core template that was done from the summer balloon launch at PCC, the only thing that was challenging was the placement of the components so that we would have a 17 degree tilt (Figure 1). The tilt was necessary because the angle of the sun above the horizon in Eaton, Co. was between 12 and 20 degrees during the launch time on November 10\(^\text{th}\). To get the maximum possible exposure for the UV sensor, we wanted to make sure that the tilt angle aligned it as closely as possible with the sun, which at 8:30 a.m. was 16 degrees, 58 arcminutes, 59.2 arcseconds, according to the Sky Safari app [3] (Figure 2).

The components were simple, with the goal just being to take UV reading and pictures to go along with it. First we needed an ultra violet detector. We found one on Amazon that reads light wavelengths between 200 and 370 nm, and outputs its readings in the form of voltage values between 0 and 1023. To take the pictures, we decided to use a Raspberry Pi 3, along with a camera designed for that computer and a filter to accept only ultra violet light. Since an Arduino Uno has a better timing system for taking data, we decided to hook the UV sensor up to the Arduino, and hook the Arduino up to the Pi using the serial cable. The Pi also had a 64 GB SD card to store all the data and pictures. For the battery, we used a battery bank that output 5 volts at 2.1 amperes to run the whole system. The battery also provided the weight we needed to tilt the box in flight.

The UV sensor was attached to the outside of the box with hot glue, with the wires fed into the box then sealed shut with hot glue. For the camera, we 3D printed a case to hold the Pi Cam on the inside of the box, with a UV pass filter just in front of the camera, and then a lid to hold it all into place (Figure 3).
All code written for this project was in Python 3.6.6. The code shown in this paper are only selections given as examples. A snippet of the code used that dictated the program flow is shown in figure 4.

```python
def receiving(ser):
    global last_received

    buffer_string = ''
    while True:
        buffer_string = buffer_string + ser.read(ser.inWaiting())
        if '\n' in buffer_string:
            lines = buffer_string.split('\n')  # Guaranteed to have at least 2 entries
            last_received = lines[-2]
            buffer_string = lines[-1]

    def burst():
        for x in xrange(3):
            filename = str(time.time()) + "-burst" + str(x)
            camera.capture("/home/pi/data/" + filename + ".jpg", format='jpeg',
                quality=50)
            data = open("/home/pi/data/" + filename, "wb")
            currentdata = last_received
            data.write(currentdata[::1])
            data.close()
            print("took picture and saved under "+ filename + " data is " + currentdata)

Figure 3: the assembled payload, showing the camera (right) and the UV sensor (left) on the front of the box.

Figure 4: an example of the code written in Python for the SunSat project.

The function receiving ran on a separate thread and constantly stored the most recent line of data from the serial buffer. This was done to avoid data building up in the serial buffer and it being mismatched with the corresponding image and time. The function burst was ran once a second, and took a burst of 3 pictures and saving the data at the time the images were taken to a file.
To compile the data into an easy use format, a script was written to take the data from each file saved and store it into a csv file which could then be easily imported into our data visualization software of choice. Each record in the csv file contained the UNIX timestamp of when the data was collected, the filename of the data (so the image taken at that time could easily be found), the height in feet (taken from EOSS), and the UV intensity. A snippet of the code responsible for this is as follows in figure 5.

```python
for line in balloondata.readlines():
    heightindex = line.find("/A=")
    if heightindex != -1:
        datapoint = {}
        datapoint[0] = refballoontoascii(line[11:20])
        datapoint[1] = line[heightindex + 3:-14]
        heights.append(datapoint)

for i in filelist:
    currentfile = open(i, "r")
    for idx, entry in enumerate(heights[previndex:]):
        if entry[0] == refpitosec(currentfile.name[16:].split('-')[0]):
            height = entry[1]
            break
        if entry[0] > refpitosec(currentfile.name[16:].split('-')[0]):
            break
    ssvindex = idx
    previndex = previndex + ssvindex
    csvdatafile.write(str(refpitosec(currentfile.name[16:].split('-')[0]))) + ',' + currentfile.name[16:] + ',' + str(height) + ',' + line.split('?', 1)[0][1:] + '
' + currentfile.close()
```

Figure 5: Python code that helped turn data and pictures in a csv file format to be easily read.

The first for loop iterates through the general flight data from the balloon itself, pulling out every height reading, matching it to a time, and storing a list containing those two entries into another list. The second loop iterates through every datapoint taken by the payload, matched it with the height measurement closest in time, and packaged all that information into the csv record format described before.

In our data analysis, it will be important to be able to separate the max data points from the many others taken during the duration of the flight. Figure 6 shows a snippet attempts to isolate these points by iterating through every entry of the csv file previously generated and only pulling records whose UV intensities are higher that the two surrounding it. The script then does a second pass and removes any records whose UV intensity is 40% less than the previous intensity.
The UV sensor used in this project has a specific responsivity to the UV spectrum. Its graph is shown in figure 7 [4].

From this, a very large portion of the UV intensity measured is in the UVA and UVB spectrum, which together are from 280 to 400 nm, while UVC, which exists from 100 to 280 nm, is partially outside of the range measured [5]. This means that the sensor will detect very little UVC.
Management

With there being only two people on the team, all of the duties were split between us. We both came up with the original idea, although inspiration was drawn from the previous PCC’s Air Heads experiment, which also had to do with light physics. Once we got the idea down, Fletcher did all of the programming to get the readings from the UV sensor and the Arduino, and the video from the camera. Kyle was in charge of building the box, which came from a design template for PCC’s DemoSat Project.

Budget

Most of the items we used came from Amazon, although the Arduino Uno was left over from a previous project. All items are listed in Figure 7, except the Arduino Uno.

<table>
<thead>
<tr>
<th>Name</th>
<th>Price</th>
<th>Quantity</th>
<th>Shipping</th>
<th>Total Price</th>
<th>Seller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium UV Sensor</td>
<td>$11.99</td>
<td>2</td>
<td>??</td>
<td>$23.98</td>
<td>Waveshare-Module</td>
</tr>
<tr>
<td>Raspberry Pi 3</td>
<td>$42.99</td>
<td>1</td>
<td>??</td>
<td>$42.99</td>
<td>CanaKit</td>
</tr>
<tr>
<td>SD Card 64GB</td>
<td>$16.99</td>
<td>1</td>
<td>??</td>
<td>$16.99</td>
<td>SanDisk</td>
</tr>
<tr>
<td>LV Lightbulb</td>
<td>$11.99</td>
<td>1</td>
<td>??</td>
<td>$11.99</td>
<td>Solray</td>
</tr>
<tr>
<td>LV Pass Filter</td>
<td>$3.35</td>
<td>2</td>
<td>$0.00</td>
<td>$6.70</td>
<td>Hedcoser</td>
</tr>
<tr>
<td>Battery</td>
<td>$14.99</td>
<td>1</td>
<td>??</td>
<td>$14.99</td>
<td>Kmaxhi</td>
</tr>
<tr>
<td>Pi Cam BMP</td>
<td>$25.58</td>
<td>2</td>
<td>??</td>
<td>$51.16</td>
<td>Raspberry Pi</td>
</tr>
<tr>
<td><strong>TOTAL:</strong></td>
<td></td>
<td></td>
<td>??</td>
<td><strong>$168.60</strong></td>
<td></td>
</tr>
</tbody>
</table>

Test Plan and Results

To begin testing, we did a filter test. This consisted of simply taking a picture with any camera having the filter on the top of the camera lens. This produced a significant photo of Pueblo Community College, where the sky is bright with ultraviolet radiation, and the trees noticeably absorb the UV radiation (Figure 8).

Figure 7: the budget list for the SunSat project.

Figure 8: The trees noticeably absorb the UV radiation while the sky remains bright.
After that, we did a simple systems test. This was done with an LED ultraviolet lightbulb put close to the sensor and camera. We found that the LED was not near as powerful as we were expecting, and we had to be within centimeters of the bulb to get any reading on the UV sensor (Figure 9 & 10). Note that in figure 5 the distance at which we got a reading of 0 was at 10 cm, the highest reading was at 0 cm where the sensor was touching the glass of the bulb. This proved not to be a problem when we took the test outdoors, though (Figure 11). The next test to do was the cold testing. We placed the experiment in a cooler with two pounds of dry ice directly on top of it for three hours to find out if it could handle the stratospheric temperature of -110 degrees Fahrenheit. The biggest worry with the temperature was that the battery would fail, but in fact the battery worked superbly with only a 25% loss of battery life.

**Expected Results**

As our payload ascends through the atmosphere, a general trend of an increasing UV intensity is to be expected. In addition to this, the rate at which the UV intensity increases with respect to the height is expected to change as the device passes between the different layers of the atmosphere. The maximum UV reading received is expected to be taken at the maximum height of the flight, or right around the time the balloon pops and the payload begins to descend back down. Since the height depends on the time that the balloon has been in the air it will also be important to compare the time passed since launch with the UV intensity.

While the height is expected to be an important factor in the intensity of the UV readings, it important to note that the UV sensor is placed only at a single point on the box, so the rotation of the payload will cause the readings at a given height to vary greatly, as a UV reading taken when the sensor is facing away from the sun will be much lower than a reading taken when the sensor is facing directly towards the sun. To compensate for this, we will mostly be paying attention to the maximum UV readings taken at that height.
From the ground testing shows the UV sensor outputting around 200 units when pointed directly at the sun during the day. It is expected that the values measured will be noticeably higher than this by the time the payload reaches the apex of its flight.

Launch and Recovery

To ensure that our device is powered on and measuring data, An LED indicator is placed and viewable from the outside of the box, and will show whether the device is on. The Raspberry Pi has been programmed to automatically run the script responsible for the main control loop that measures data approximately 20 seconds after it has been powered on.

Fletcher Collins will be responsible for ensuring the device is powered on and is safely attached to the balloon during launch. Both Kyle Petit and Fletcher Collins will approach the payload during recovery and one will switch the device off, ending data collection. The data stored on the Micro SD card connected to the Raspberry Pi will be pulled and copied onto a computer and backed up on an external hard drive. This procedure has been tested by switching the device on and pointing it at the sun for a short period of time. The test data measured was then pulled off the Micro SD card, copied to a computer, and confirmed to be non-corrupt and readable. For longer term testing, the device was left powered on taking measurements in a room for a number of hours. The data was pulled and shown to all be readable.

During the launch, Fletcher Collins successfully ensured that the device was powered on. It is to be noted the flight cord of the balloon got caught on a different group’s payload, causing all payloads under it to be angled differently than expected. However, SunSat’s payload was above the caught cord and the angle it was facing remained unaffected. Therefore, the payload was safely launched and was recording data.

Recovery involved the group approaching the downed balloon and finding the payload, it was observed that the LED indicator showing whether the device was still powered was still on. When the group received the OK to interact with the payload, it was powered off and opened. Insides showed no signs of physical damage. The Micro SD card was removed from the Raspberry Pi and the data was shown to be intact and non-corrupt. It is to be noted that the power source used did not run out of power and the Micro SD card ran out of space approximately 15 minutes after landing.
Results, Analysis, and Conclusions

Time since liftoff, height, and UV intensity are the three variables whose relationships will be analyzed. It is to be noted that the general relationship between the data is that UV intensity depended on height, and height depended on time since liftoff. To give a general idea of how the UV intensity changed, time was graphed against UV intensity in chart 1 (figure 12). One must be careful in doing this, as relating UV intensity to time passed hides how height may have impacted UV intensity. However, chart 2 (figure 13) shows that as the balloon ascended over time, the height increased linearly, so no complex relationships are hidden when plotting UV intensity against time.

To understand how height impacted UV intensity, the two were plotted in chart 3 (figure 14). One may notice that there appear to be a number of different trendlines in the data. This is because figure 14 includes measurements taken after pop, in which the balloon was falling back down. To see a cleaner representation of the general trend, chart 4 (figure 15) plots height only up to the apex of the flight and UV intensity. One of the most important points on this chart occurs around 60000 feet, where the UV intensities begin to increase much slower. It is at this point, that the balloon appears to have completely entered the Stratosphere.
It will be important to know whether the amount of time that passed while the balloon affects the UV intensities measured. To test this, a filtered dataset is used which attempts to only take the maximum readings for each height. Chart 5 (figure 16) plots all values from this trimmed set where the device was in the Stratosphere. The device passed through all heights plotted twice as it ascended and then came back down. Because it is clear that chart 5 (figure 16) only depicts one trendline when each height was measured twice at a different time, it may be concluded that maximum UV readings taken during the day in the stratosphere are unaffected from approximately 8:00 AM to 9:20 AM. It is likely that readings taken from these heights during other times of day are also unaffected.

A function modeling the UV intensity through the entire Stratosphere may be easily constructed when one notices that the relationship between height and UV intensity is linear. By taking the data points (60814, 539) and (111029, 596), the function \( f(h) = 0.001135 \times h + 470 \) which passes through those two points is found, where \( h \) is the height in feet and \( f(h) \) is the UV intensity at that height. This function is only defined for heights within the stratosphere, which we define as between 60000 feet and 160000 feet. To find the total amount of UV that the stratosphere absorbs, we take \( f(160000 \text{ feet}) - f(60000 \text{ feet}) \), to find that the Stratosphere absorbs 113 units of UV.

Due to the responsivity of the UV sensor used, we were unable to measure much of the lower wavelength ultraviolet radiation that the Stratosphere is responsible for absorbing. This is unfortunate, as the impact the Stratosphere has on ultraviolet levels was one of the key goals of this project. This comes down to the choice of UV sensor used. If this experiment is ever to be repeated in the future, it would be beneficial to select an ultraviolet sensor better suited to measure higher energy ultraviolet radiation.

Moving onto the absorption on ultraviolet radiation by the Troposphere, ultraviolet intensity readings between the ground and a height right as the balloon transitions into the Stratosphere will be compared. While the balloon was in the air, ultraviolet readings were being done from the ground to compare against those taken by the payload. A maximum reading of 291 units of ultraviolet radiation was found at around 9:45 AM. As The balloon began to move from the Troposphere to the Stratosphere, a reading of 520 units was taken. With this information, we find that 229 units of ultraviolet radiation are absorbed by the Troposphere. We find that the Troposphere absorbed UV radiation much quicker than the Stratosphere. This results likely comes from the responsivity of the UV sensor used, which was, again, weighted towards the measurement of lower energy ultraviolet radiation.

In conclusion, we find that the Troposphere is responsible for the absorption of more low UVA and some UVB than the Stratosphere is. This may be attributed to the fact that the Troposphere has a much higher
density. If one were to perform a similar experiment, it would be beneficial to use a UV sensor that primarily responds to UVC to obtain a better understanding of how that Stratosphere interacts with that kind of ultraviolet radiation.
References


