Use of Temperature Compensation for Single Dish Total Power Observations

Dave Cohen

August 2, 2021

Contents

1 Introduction

Many of the components used in a radio telescope are very sensitive to temperature. This is especially true of the LNA (Low Noise Amplifier), where adding temperature compensation can result in the degradation of noise figure of the LNA (any evidence to support this?). This is one of the major challenges to total power measurements, as the change in output of the receiver due to temperature is typically much greater than the change due to an actual source.

Serveral attempts have been made by the author to stabilize the temperature of the front-end components of the telescope, with varying degrees of success. These usually involve an insulated chamber that houses the components, and a temperature control system involving a heating or cooling device that usually consists of a Peltier junction. If cooling/heating is done in conjunction with a temperature sensor in a closed-loop control system, temperature regualation within +/- 0.1 degree Celcius has been achieved.

The problem with temperature control is mainly with the complexity of the system, and the number of components. The following devices as a minimum would be required:

- Temperature sensor for feedback
- Processor to determine error and provide drive to amplifier
- Amplifier to drive the Peltier junction
- A fair amount of power supply to drive the Peltier amplifier
- The Peltier junction
- Insulated enclosure to reduce demand on the control system

In addition to the complexity of the temperature control, problems can arise from improper control of temperature. Crossing the dew point when using cooling can cause condensation, which can be harmful to electronics, as well as excessive heat. An LNA generates heat on its own, which causes further complications.

In order to avoid a lot of the problems previously discussed, an attempt will be made to remove the effects of temperature without any temperature control other than an insulated encloure to reduce temperature fluctuations. This should be possible with an accurate temperature sensor and a mathematical model of the temperature response of the entire receiver system.

2 Determining Temperature Response of the System

Before the temperature response of the system can be determined, it is necessary to have accurate temperature measurements of the system. It will also be necessary to have the ability to subject the system to the temperature ranges that it will undergo during normal operation. To that end, some equipment will need to be acquired.

2.1 Temperature Measurement Equipment

There are some excellent low-cost options for temperature measurement available.

- Platinum PT100 RTD three-wire temperature sensor.
  Platinum sensors are supposed to have higher accuracy and stability than thermocouples. They have a 100-ohm resistance at 0 degrees C, and vary reisistance at the rate of 0.385 ohms per degree C. The third wire is used as a voltage sense wire to compensate for lead length. The Adafruit 3290 will be used for this purpose.

- Temperature sensor amplifier.
  Adafruit 3328, a MAX31865 RTD amplifier will be used to diigtize the signal. It uses SPI (Serial Peripheral Interface) to transmit temperature data to a controller.

- Arduino Nano.
  A tiny, low-cost controller that will communicate with the MAX31865 via SPI. The Nano will communicate with a computer over USB, which will also provide power to the Nano. A library using a high-accuracy lookup table is available at
  https://github.com/drhaney/pt100rtd

2.2 Thermal Testing

An insulated enclosure must be provided to both minimize temperature fluctuations, and to provide an environment where temperature can be changed in a controlled manner. The enclosure must be of sufficient size to house all of the receiver components, from front end to computer. Weatherproof coolers are being evaluated for this purpose.

Thermal testing will be done as follows:

- Ice (or dry ice?) in a tray will be bring the temperature down the low extreme, where the test will be started. The ice will be allowed to melt (or sunlimate).

- A source of heat will be placed in the cooler. An incandescent bulb and fixture controlled by a dimmer are being considered for this purpose.

3 Journal

Parts Arrived 2021-08-11

Received parts and wired MAX31865 RTD amplifier and RTD. Had trouble with tiny solder jumpers on eval. board. Also had a lot crashed in Arduino IDE on Raspberry Pi, especially when connected to serial monitor. Ended up moving to NUCRA mini computer to get hardware working. Got Adafruit software working, but I moved on to the more precise software offered on (github ref), which uses a lookup table which supposedly provides the most accurate temperature readings. I got that software working as well. I am now just starting to modify the Arduino code to listen to the serial port. A python test application has been created. My next problem is to properly parse the usb devices to locate the serial port for the Nano. I think that I am doing this wrong for the USB relay in tp_soapee.py, but it happens to be working because it occurs as the first device, which I have hardcoded. That needs to get fixed when I bring in the new code for temperature measurement.

Plotting Temperature 2021-08-13

Got software temperature logging working on Arduino and plotting working on Python. Graph looks good and I have a few extra features, like subplots with varying height ratios.

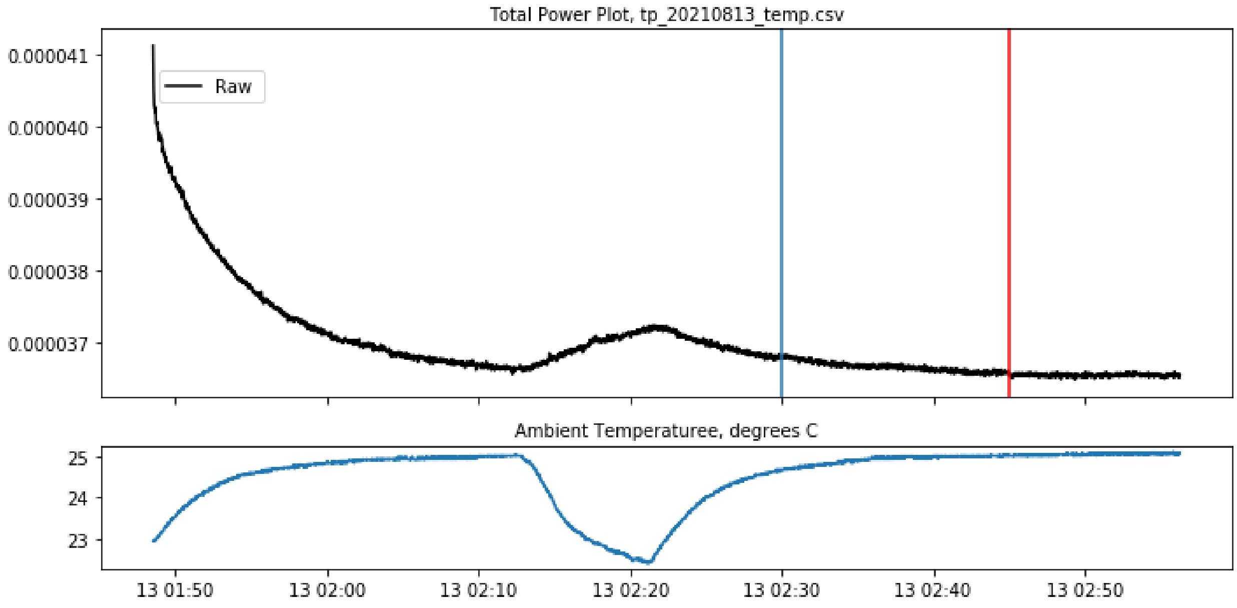

Figure 1: Total power with temperature plot.

There is a direct and obvious correlation between temperature and output visible when plotting this way. I hope that the resolution of the RTD and amplifier is sufficient to provide smooth temperature compensation.

The next step is to develop a temperature compensation method. The current idea is to write a separate script that guides the user through the process. It could go something like this:

- Electronics warmup
  Pause for a minimum warmup time, running the hardware but ignoring data. This should be for a minimum of 90 minutes.

- Cooldown
  Place a cold substance (dry ice preferred) in the container with the hardware, taking necessary precautions.

- Warmup
  An incandescent bulb or other heat souce is placed in the container.

Are these separate recordings or one continuous one? How will data be marked? It's probably a bad idea to use rapid temperature changes, as thermal lag may be a factor. So the best method might be to house the electronics in a well insulated cooler with dry ice, and place it outside (sun or shade?). Allow the temperature to slowly climb to around 40 °C, and stop recording.

Post-processing would entail creating temperature and output arrays that are on even temperature intervals, distributed through the entire temperature range. A curve fit would be done through the data, and each quantity plotted. This should result in an equation that could be fed back into the total power plotting program (how?). An additional command-line option would be added to allow for temperature compensation.

Temperature Calibration, Take One 2021-08-15

Rough temperature calibration developed, using the following steps:

- CSV file loaded into pandas dataframe.

- Time, total power, and temperature columns extracted and converted to Numpy arrays time_array, tp_array, and temperature_array.

- Based on the starting and ending offsets, existing arrays sliced to create new arrays time_array_window, tp_array_window, and temperature-array-window. These arrays will be used to calculate the temperature compensation curve.

- Temperature window array is sorted by ascending temperature. Total power window array is sorted by temperature to maintain the one- to-one correspondence between the temperature and total power.

- New temperature array temperature_array_uniq created from unique values of temperature from the sorted array. Correspondig total power array tp_array_uniq created.

- A curve-fit polynomial of order 3 is created from the unique temperature and total power arrays. This is assigned to the variable f.

- The temperature compensated total power array is created by iterting through the original total power array tp_array, and subtracting the returned value of f for the frequency corresponding to the total power value.

A three-hour run was made with the air conditioning cycling several times. A Noise calibrator signal was injected between 1700 and 1730 UTC. The results can be seen in figures 3, 3, and 3:

It can be seen that there is a reduction of the change in signal by the contribution of temperature. The contribution in signal by the noise source seems to be unaffected. Unfortunately, this is not effective enough to be called a success. A radio source could easily be masked by the curves in Figure 3; stronger sources being distorted by their presence.

The failure of this process to completely remove the termperature contribution is thought to be caused by the fact that for multiple measurements of the same temperature, the

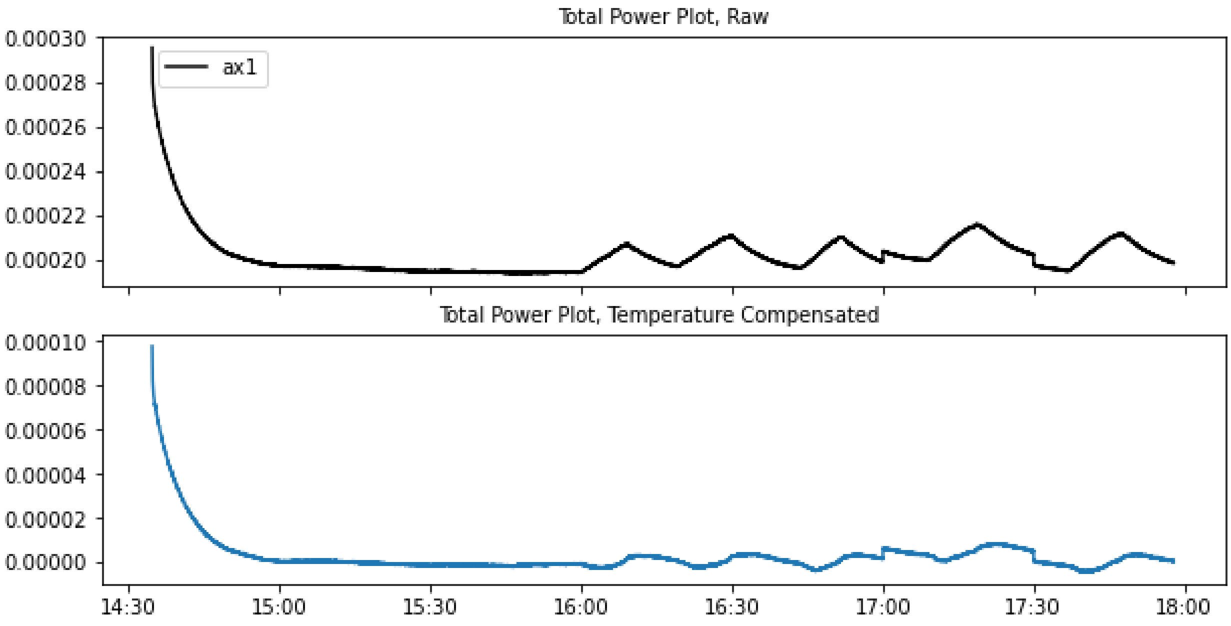

Figure 2: Total power, raw and temperature compensated.

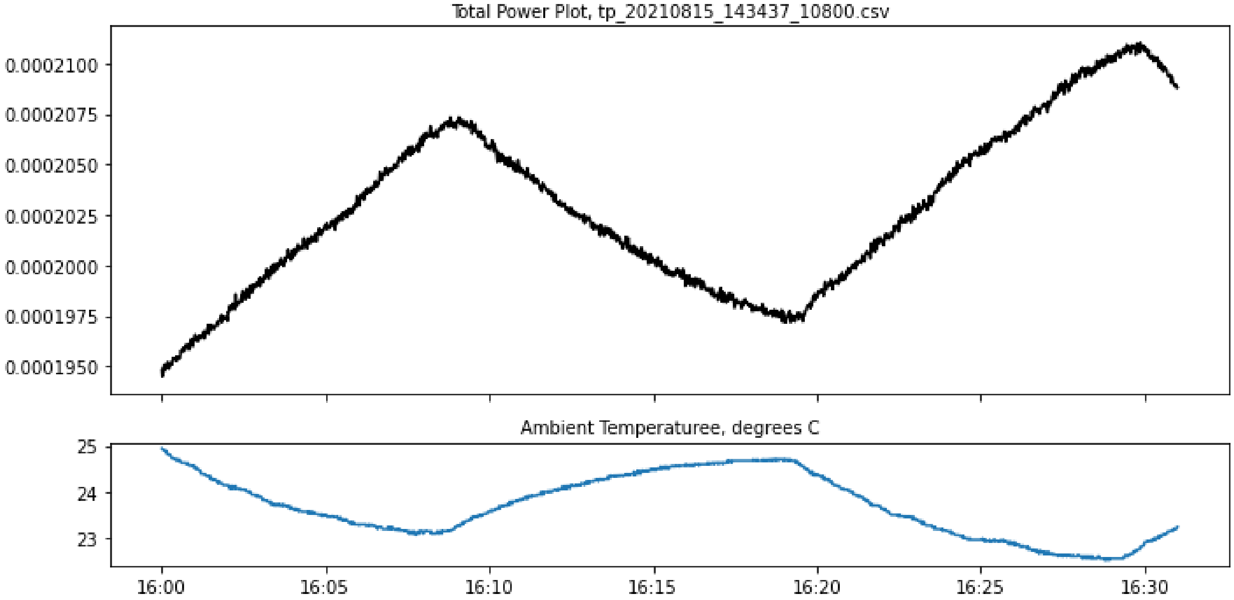

Figure 3: The segment of the graph defined as the evaluation window, which is used to determine the temperauture/power correlation.

total power corresponding to each measurement can differ. Table 1 shows the relationship between multiple measurements at a constant temperature, and the corresponding total power readings. These values are from the first six elements of temperature_array_sorted and

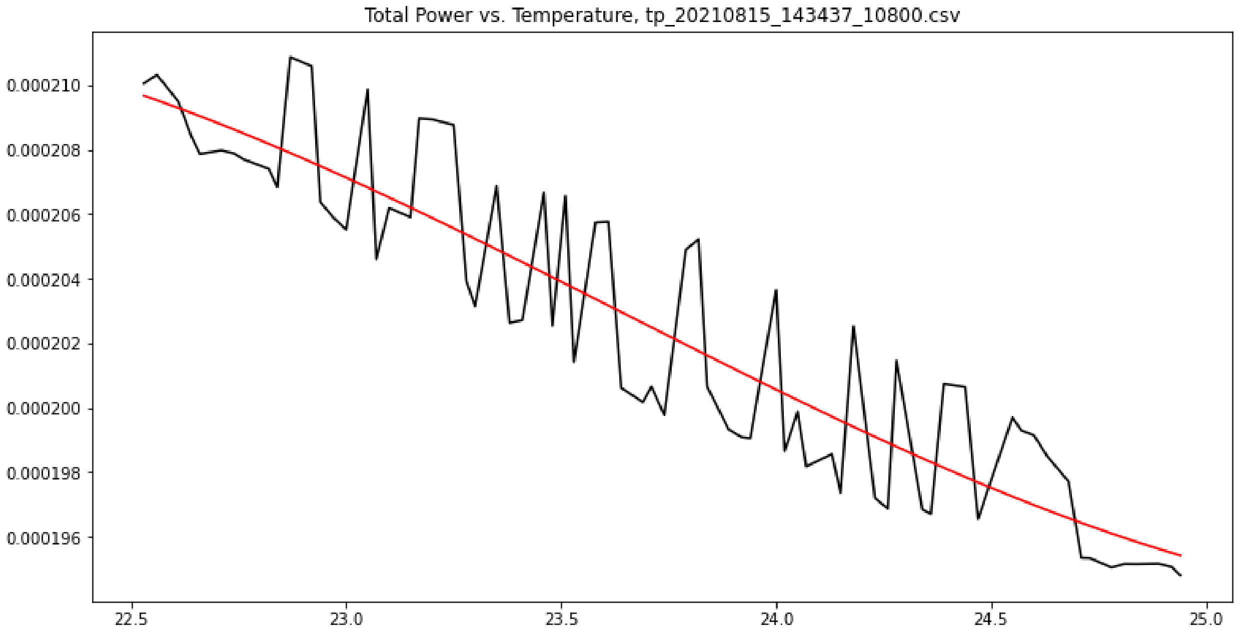

Figure 4: The relationship between temperature and power. The curve fit function is shown in red.

tp_array_sorted. Some quick math on these values indicate a 0.22% change from minimum to maximum values in this range. That's a pretty small change, but that might not be typical for all temperature values. Looking at Figure 3 on the left side of the graph, the power values are very close to the polynomial curve. That is not the case for the values in the middle of the curve.

Temperature	Total Power
22.53	0.000210056
22.53	0.000209956
22.53	0.00021007
22.53	0.000210417
22.53	0.000210384
22.53	0.00021006

Table 1: Temperature and corresponding total measurements for multiple occurences of a single temperature.

So what to do? A general statement could be made that the curve fit function isn't "strong" enough - it consistently doesn't remove enough of the temperature contribution, which might indicate that polynomial has the right slope, but the wrong offset in the y direction. But looking at Figure 3, the power values vary on both sides of the polynomial curve. Applying smoothing to the total power array does not noticable improve the outcome. Applying smoothing to the temperature array causes more distortion in the total power output. I am considering averaging all the total power readings for a given temperature, and constructing the unique arrays from that. At least the total power values won't be at a minimum or maximum.