Time synchronization under temperature and distance variations

M. Susin *L. F. Wanner*

Relatório Técnico - IC-PFG-17-04
Projeto Final de Graduação
2017 - Julho

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO

The contents of this report are the sole responsibility of the authors.
O conteúdo deste relatório é de única responsabilidade dos autores.

Time synchronization under temperature and distance variations

Matheus Susin[1], Lucas Wanner[2]

[1] Instituto de Computação Universidade Estadual de Campinas (UNICAMP)
13083-970 Campinas, SP, Brasil

ra147396@students.ic.unicamp.br

[2] Instituto de Computação Universidade Estadual de Campinas (UNICAMP)
13083-970 Campinas, SP, Brasil

lucas@ic.unicamp.br

Abstract. The *Trustful Space-Time Protocol* (TSTP) allows for time synchronization to be performed upon receiving any message from another node in a sensor network, removing the need for explicit messages. Previous work has shown that TSTP performs well under controlled experimental environments. In this work, we analyze how the quality of synchronization in TSTP is affected when nodes are communicating over varying distances, and across a temperature gradient. The results show that, while distance and packet loss has only a small effect on the quality of the synchronization, high temperatures of 80 Celsius and up can negatively affect it. Snippets of the results are presented, along with charts, statistics, and an analysis. A repository that contains all results, as well as the tools to analyze them, is available at https://gitlab.com/mathy/PFG.

Keywords: time synchronization; clock synchronization; TSTP; Internet of Things.

1. Introduction

Many wireless sensor network applications require time synchronization. Failing to coordinate action may lead to security risks, sensor inaccuracy, and loss of efficiency in power usage, given that duty cycles will be out of sync. The *Trustful Space-Time Protocol* (TSTP)[TP0][TP1] provides time synchronization by including a timestamp on the header of each message exchange by its nodes. The sensors are slaves to the gateway, and always adjust their clock accordingly.

Many factors can negatively, or even positively, as we will see, influence the quality of time synchronization. We focused on two independent variables, and analyzed the behavior of the protocol several different conditions. First, we kept the sensor on a fixed position, and performed tests with the gateway being put in different spots around the room, mimicking differences in latency and possible interference from other electronic devices. We then placed the gateway in a box, the sensor right next to the box, and with the help of a lamp and a digital thermometer, heated it up to different temperatures and took the same measures.

For the remainder of this report, we describe the experiments in more detail, provide snippets and statistical analyses of the results, and discuss what can be learned from it. The experiments with distance were performed on a room with many electronic devices, several of which were communicating via wireless, and showed that not only does distance play a small role in the quality of the synchronization, but longer distances sometimes resulted in slightly better averages. The experiments with temperature showed that, for this particular pair of crystals, the temperature of 70 Celsius applied to the usually slowest one minimized the average offset that was perceived by the protocol.

2. Materials

This report describes the results obtained from sets of spatial experiments, where distance between nodes was varied, and sets of temperature experiments, where one of the nodes was confined in a small box with a halogen lamp, and allowed to reach certain values of temperatures. The description of the experiments should allow for them to be easily understood and replicated, should one decide to verify the results, or perhaps experiment on different hardware and protocols. The experiments were performed using two EPOSMote 3 (eMote 3). There is no physical distinction between sink (gateway) and source (sensor). Since the gateway's clock is never adjusted to the network, the USB cable is being used strictly to power it. The sensor provides the output that's saved by minicom to a text file. The CPU used in each is an ARM Cortex M3.

For the temperature experiments, a small polystyrene box was assembled with a small hole in the lid for the wires that power the halogen lamp. Two holes were made on the side, close to the bottom, one for the thermometer probe, and another one for the cable that would power the eMote inside. A significant difference in how stable the temperature was inside the box with the lamp turned on was perceived when the holes were sealed tighter, possibly because the colder air from the room wouldn't cool the metallic probe. The lamp has a dimmer attached to its wiring, allowing for the operator to indirectly control how much it is heating the air inside the box.

For the spatial experiments, the Computer System Laboratory was used. It is a single 1300x1109cm rectangular room, with multiple electronic devices in use, and several people moving around all day. Data collected during the experiments include an estimation on how many packets may have been lost.

3. Methods

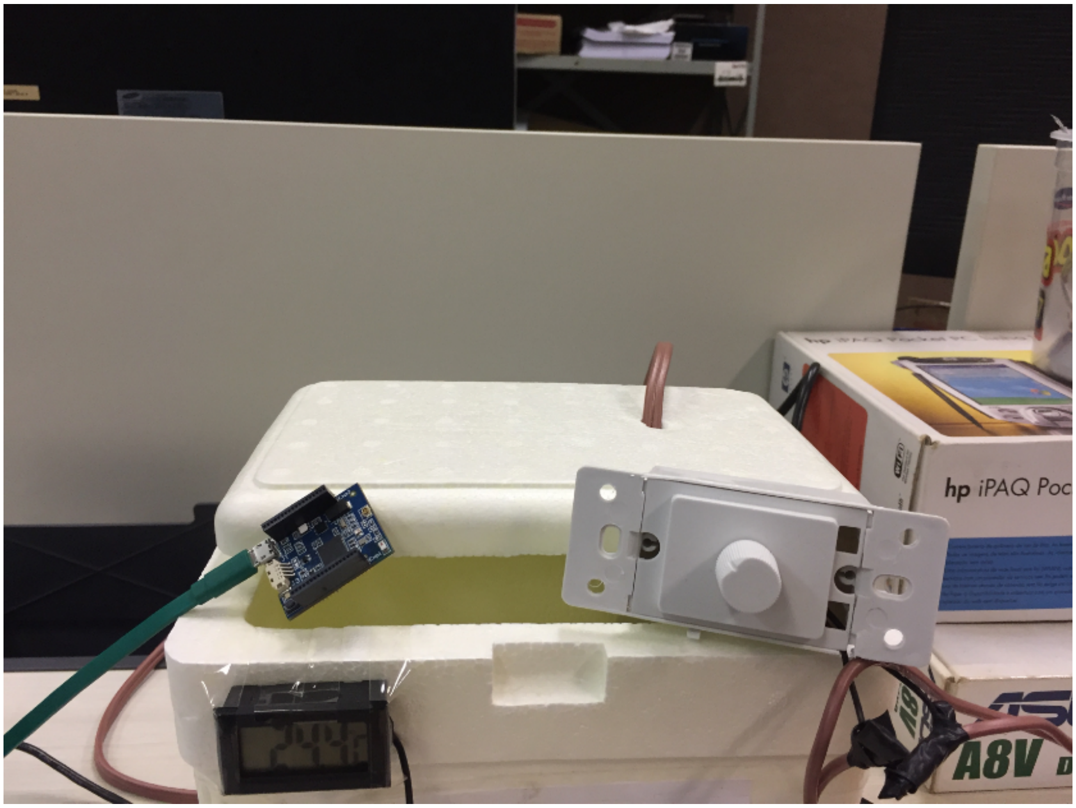
In order to analyze the time synchronization accuracy of TSTP, we log synchronization messages received and transmitted in the system. One iteration of the clock synchronization is shown below:

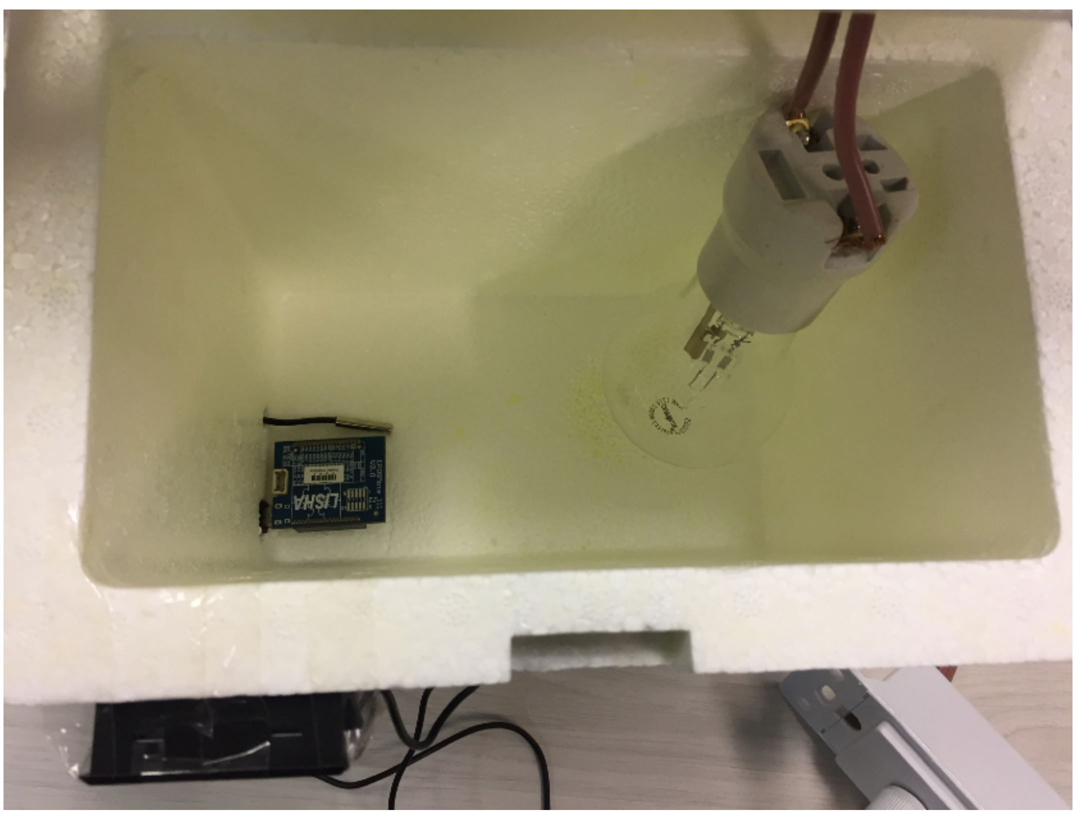
```
Line 1: TSTP::keep_alive():keep_alive = 0x20003bf4
=>{v=0,t=T,co=100,tr=1,s=0,ot=133846696,o=(10,10,0),lt=4200498129,l=(10,10,0)},st
=5
Line 2: TSTP::update: Keep_Alive:
{v=0,t=T,co=100,tr=0,s=0,ot=131266724,o=(0,0,0),lt=4210129171,l=(0,0,0)},st=5
Line 3: TSTP::Timekeeper::update: adjusted timer offset by -444
Line 4: TSTP::Timekeeper::update: synchronizing with (0,0,0)
Line 5: now() = 134813375
```

The first line shows a message from the sensor to the gateway, at time 4200498129, explicitly requesting a timestamp to synchronize its clock. Line 2 shows the gateway's response, with the corresponding high-accuracy time stamp of 4210129171 extracted from TSTP's message header. The sensor node uses this time stamp, as well as a precise measurement of the local clock when the radio interface receives the message, to adjusts its clock via the Timekeeper class (the precise description of TSTP's time synchronization mechanism is shown in previous work [TP0], and is out of scope for this work). Line 3 shows that the sensor node detected, and corrected, a clock offset of -444 ticks.

The fourth line simply informs the geolocation of the node that the clock is being adjusted to, which is used as a way to uniquely identify it.

The final line corresponds to the time reported by the system, in units of clock tick count. We know that, since the piezoelectrical crystal has a frequency of 32MHz, each clock tick corresponds to 31.25ns if the crystal is perfectly cut and kept at ideal temperature. IEEE 802.15.4 defines an acceptable divergence of the clock as being of, at most, 0.004% of the frequency, or 40 parts per million (ppm). Given that this clock synchronization process happens every 3 seconds, we establish 120 microseconds, or 3840 ticks, as the target maximum deviation.


hp iPAQ Poc
A8V


LISHA

The parser, available in the repository, is able to generate a CSV file that contains the “now” value paired with the offset, negative or positive. Since we are interested only in the value of the deviation, and not on which crystal has the highest frequency, the sign of the offset is removed, and internally, statistical calculations always take its absolute value.

Lost packets are registered when two or more “keep_alive” packets are sent in sequence, and no reply is given to them.

There are two noteworthy cases that may cause great impact on the standard deviation, but not so much on the average offset.

The first one, show below, is a complete reset on the connection, evidence by the huge setback on the offset, which follows a sequence of lost packets, indicating that the gateway died and recovered at some point, a hardware failure. Since the experiments are not concerned with major hardware faults, the results in which this happened were discarded. These happened at temperatures as low as 25°C, and as high as 95°C.

```
TSTP::update: Keep_Alive: {v=0,t=T,co=100,tr=0,s=0,ot=785205350,o=(0,0,0),lt=251
36165318,l=(0,0,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=787768
502,o=(10,10,0),lt=25126534205,l=(10,10,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=791117
598,o=(10,10,0),lt=25230427325,l=(10,10,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=794274
696,o=(10,10,0),lt=25333866813,l=(10,10,0)},st=5
TTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=797648
802,o=(10,10,0),lt=25442008202,l=(10,10,0)},st=5
TSTP::Timekeeper:TSTTSTP::Timekeeper::update: adjusted timer offset by 1
TSTP::update:interest=0x20006754 => {v=0,t=I,co=100,tr=0,s=0,ot=3808,o=(0,0,0),l
t=42365526,l=(0,0,0)},u={SI.I32:m^1s^-2},m=S,e=0,x=2000000,re={{c=(10,10,0),r=0}
,t0=900,t1=18446744073709551615},p=1000000
P::Timekeeper::update: adjusted timer offset by -111
TSTP::update:interest=0x20006d58 => {v=0,t=I,co=100,tr=0,s=0,ot=3808,o=(0,0,0),l
t=28220327,l=(0,0,0)},u={SI.I32:m^1s^-2},m=S,e=0,x=2000000,re={{c=(10,10,0),r=0}
,t0=900,t1=18446744073709551615},p=1000000
:update: adjusted timer offset by -25575590371
TSTP::Timekeeper::update: synchronizing with (0,0,0)
now() = 2407259
```

The other case happens for unknown reason, and can be observed when there is a huge negative offset, meaning that the sensor had to set its timer back, followed by a huge positive offset, meaning that the sensor realized the mistake from the next packet that arrived and corrected it. Adding both values up typically results in an offset value that isn't much larger than the other values around it. An example is shown below.

```
TSTP::update: Keep_Alive: {v=0,t=T,co=100,tr=0,s=0,ot=134366448,o=(0,0,0),lt=430
9320367,l=(0,0,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=137010
780,o=(10,10,0),lt=4299689421,l=(10,10,0)},st=5
TSTP::Timekeeper::update: adjusted timer offset by -445
TSTP::Timekeeper::update: synchronizing with (0,0,0)
now() = 138068914
TSTP::update: Keep_Alive: {v=0,t=T,co=100,tr=0,s=0,ot=137613112,o=(0,0,0),lt=441
3213769,l=(0,0,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=140177
865,o=(10,10,0),lt=4403582625,l=(10,10,0)},st=5
TSTP::Timekeeper::update: adjusted timer offset by -6764
TSTP::Timekeeper::update: synchronizing with (0,0,0)
now() = 141171792
TSTP::update: Keep_Alive: {v=0,t=T,co=100,tr=0,s=0,ot=140712844,o=(0,0,0),lt=451
2405231,l=(0,0,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=143338
752,o=(10,10,0),lt=4502774110,l=(10,10,0)},st=5
TSTP::Timekeeper::update: adjusted timer offset by 5901
TSTP::Timekeeper::update: synchronizing with (0,0,0)
now() = 144265888
TSTP::update: Keep_Alive: {v=0,t=T,co=100,tr=0,s=0,ot=143812585,o=(0,0,0),lt=461
1596933,l=(0,0,0)},st=5
TSTP::keep_alive():keep_alive = 0x20003bf4 => {v=0,t=T,co=100,tr=1,s=0,ot=146503
035,o=(10,10,0),lt=4601959429,l=(10,10,0)},st=5
TSTP::Timekeeper::update: adjusted timer offset by -458
TSTP::Timekeeper::update: synchronizing with (0,0,0)
now() = 147517738
```

Each experiment consists of logging the adjusted offset every 3 seconds, and takes between 15 and 20 minutes. Part of the experiments are grouped in sets of experiments which share the same location, and the rest are grouped in sets that share the same temperature, depending on what condition is being tested.

On the next page, we provide an example of the data available in a set, and how it should be interpreted.

```
In group : close
    Average: 287.589
    Std dev: 566.313
    MAD: 34.0
    Loss: 0.347%
    Over 40ppm: 0.262%
```

All statistics are computed over the magnitude of the offset.

The average is simply the average of all values.

The standard deviation is provided for completeness, but due to the outliers mentioned previously, they are not a good measure of the deviation.

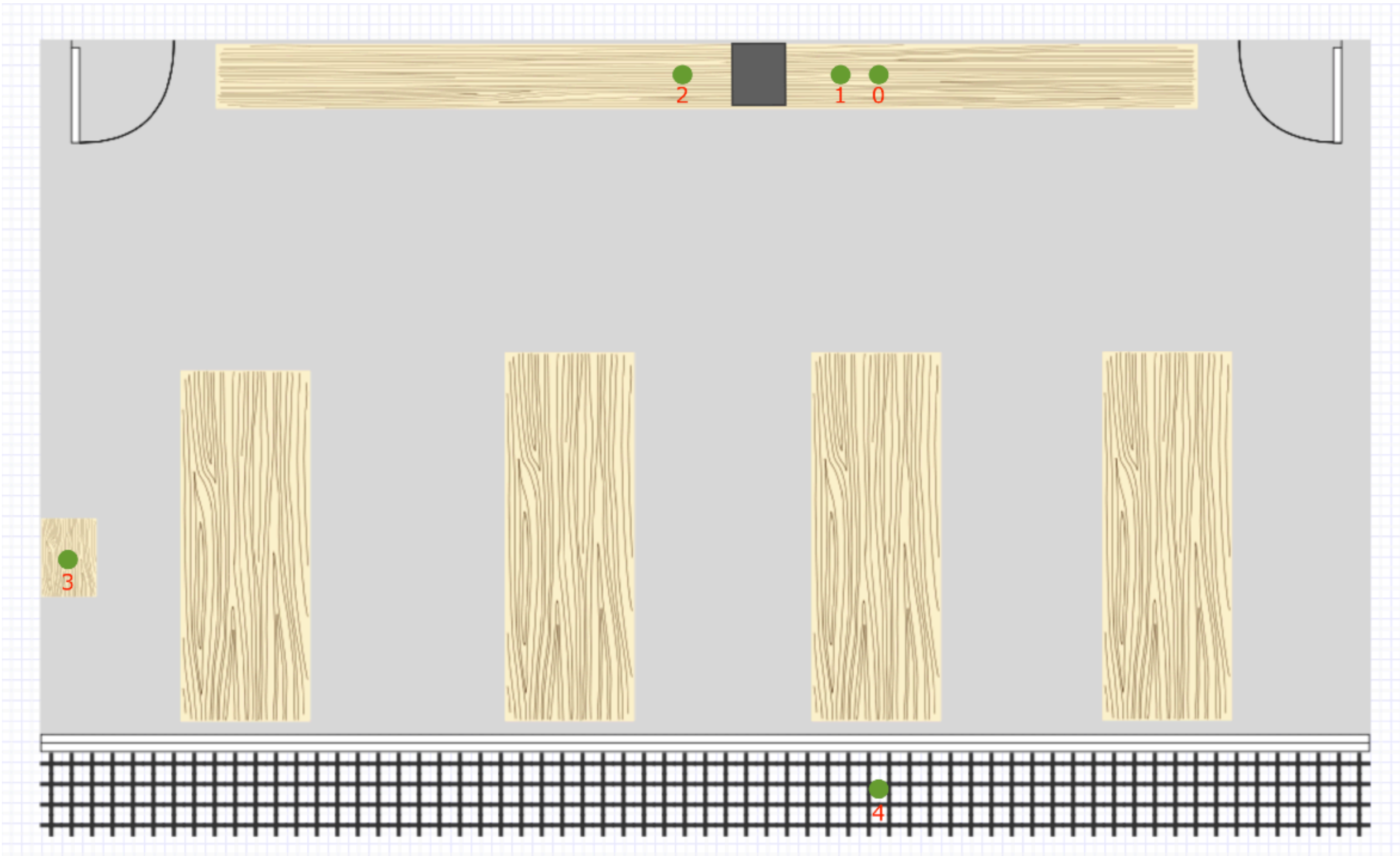
MAD is the median absolute deviation, a robust measure of variability.

The loss is the ratio of keep alive packets that were sent in succession without receiving a response.

The final line is the percentage of readings that were above the chosen standard of 40 ppm.

4. Experimental Results: Distance Variation

In this experiment, the sensor was kept on the position marked as 0, and the gateway was placed, for each step, on one of the other numbered positions.



The positions for the gateway were:

1 - "close" - about 10cm from the sensor, with no obstructions;

2 - "around" - about 1m from the sensor, with a 45cm wide desktop between the nodes;

3 - "distantcorner" - about 12.1m from the sensor;

4 - "outside" - about 11.2m from the sensor, outside the room, and running about 8°C hotter than the sensor.

```
In group : close
    Average: 287.589
    Std dev: 566.313
    MAD: 34.0
    Loss: 0.347%
    Over 40ppm: 0.262%

In group : around
    Average: 246.159
    Std dev: 58.296
    MAD: 34.0
    Loss: 0.262%
    Over 40ppm: 0.000%

In group : distantcorner
    Average: 292.011
    Std dev: 574.523
    MAD: 42.0
    Loss: 6.195%
    Over 40ppm: 0.661%

In group : outside
    Average: 187.030
    Std dev: 829.047
    MAD: 67.0
    Loss: 3.212%
    Over 40ppm: 0.716%
```

It's interesting to note that the second experiment had a better average than the first one, with no offset going over the standard, even with a metal box and a running computer between the nodes between. The third experiment has a significantly high loss ratio, but even then, the average offset wasn't much higher than the previous tests.

The final experiment shows that, despite the number of outliers growing, the average offset was the best one among the distance tests, and even when compared to the temperature tests, it remains the second best average.

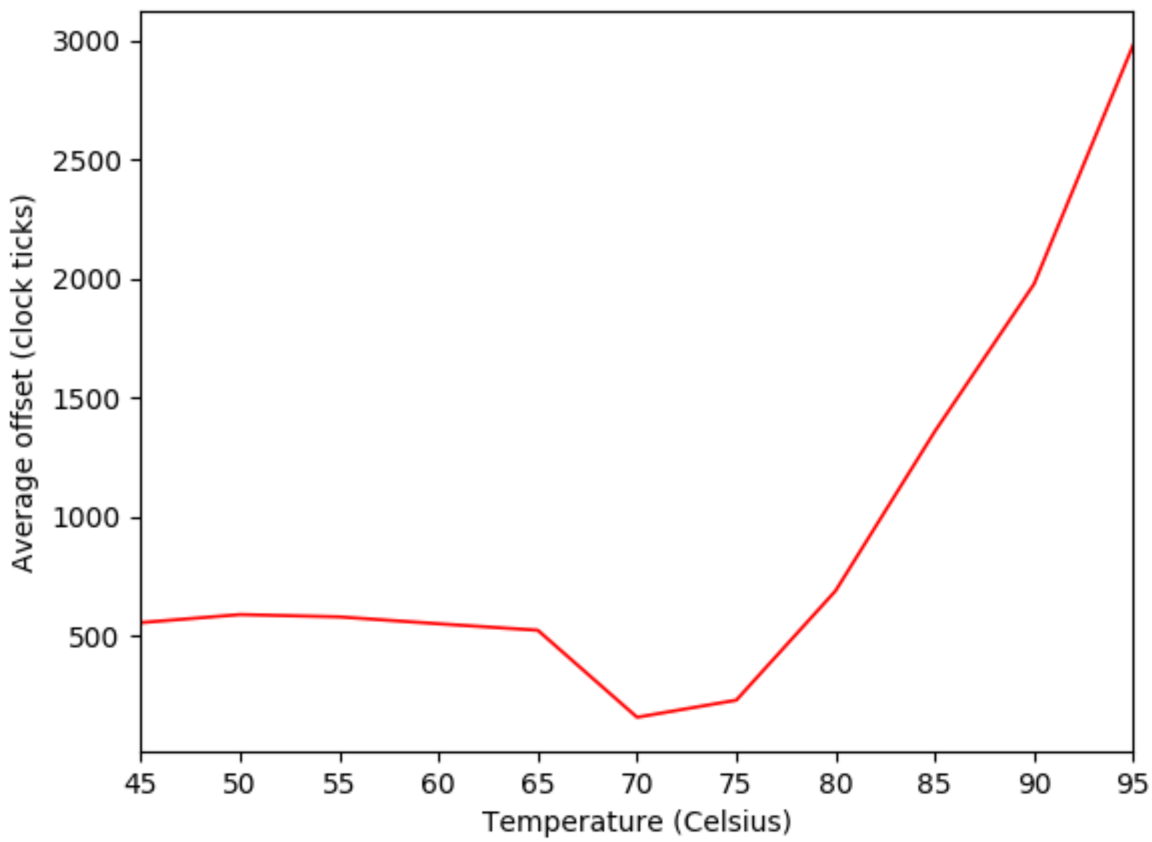
5. Experimental Results: Temperature Variation

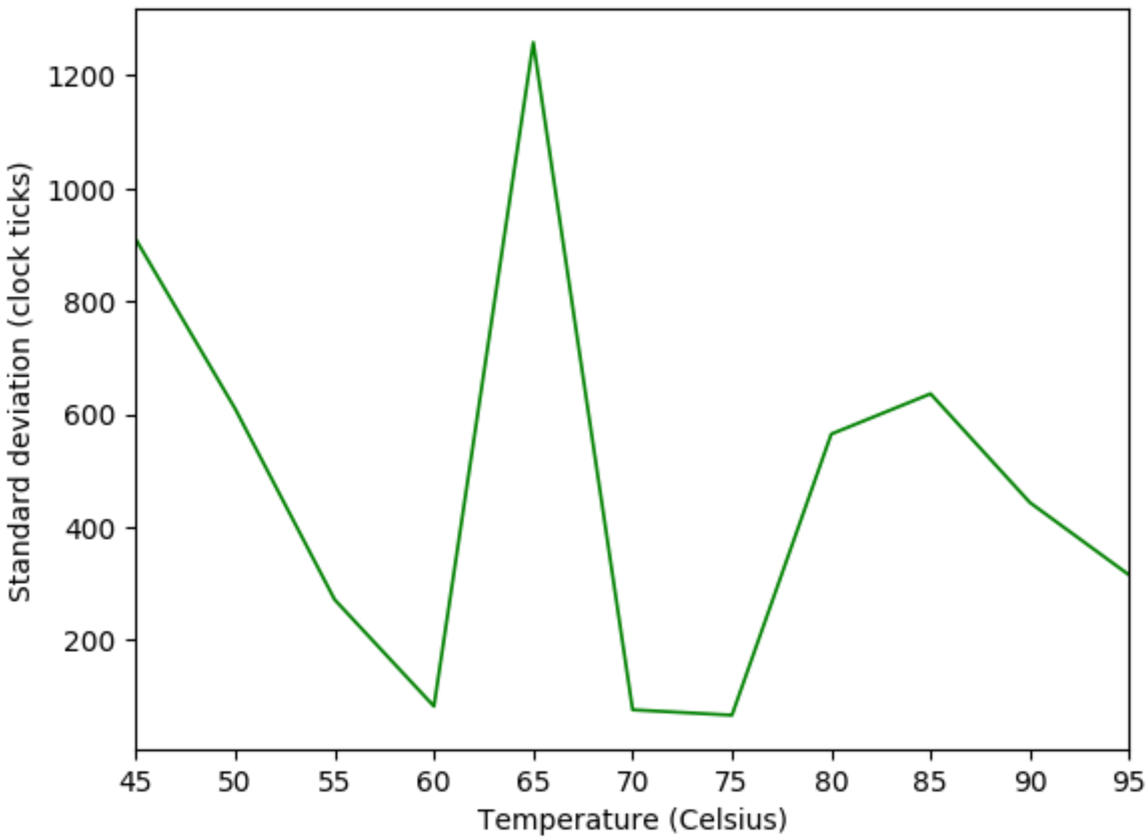
This section presents the results obtained from the experiments in which the gateway was placed inside a polystyrene box. With the lamp on, we allowed the air inside the box to reach the desired temperature, as informed by the digital thermometer, then waited at least 1 hour for the temperature to stabilize, dimming the lamp when necessary. We discarded the experiments in which the deviation between the desired temperature and the one shown on the thermometer surpassed $\pm 1°C$ at any moment during its execution.

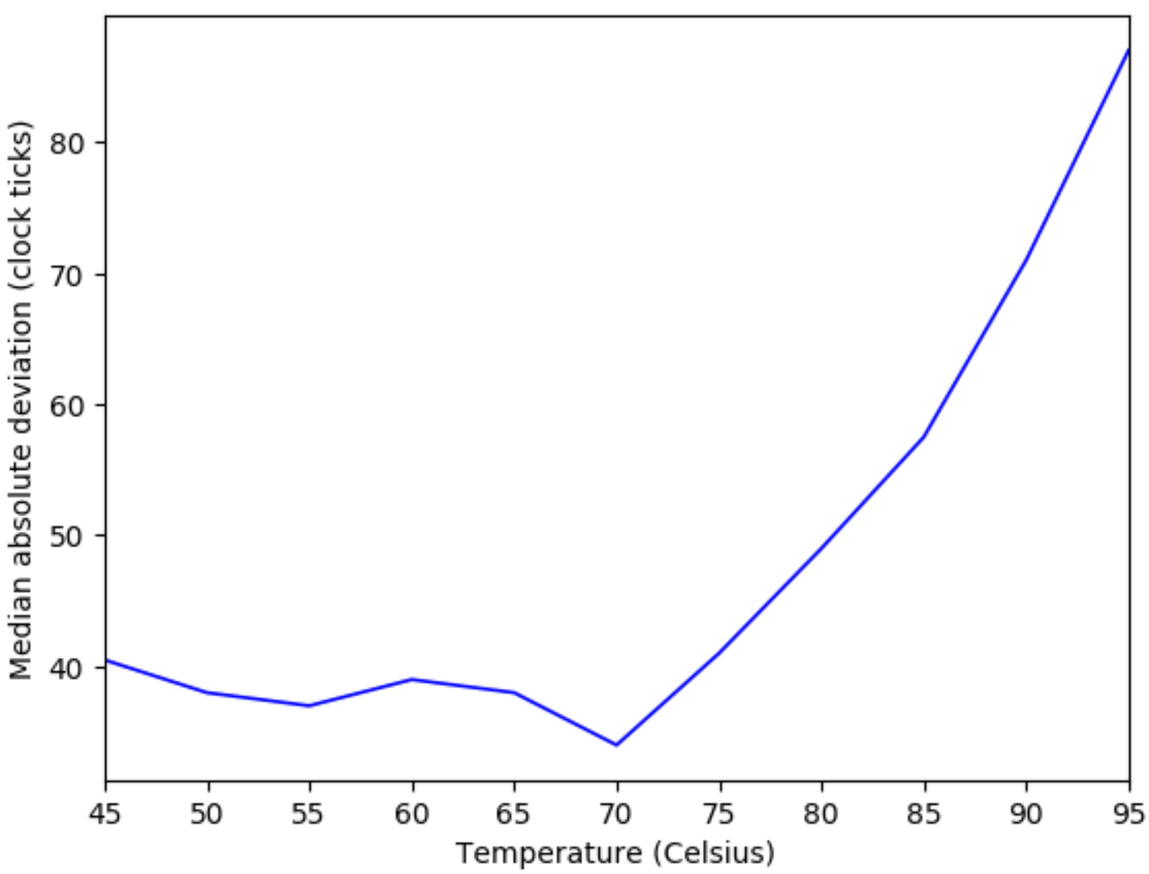
```
In group : 45C
    Average: 553.965
    Std dev: 908.989
    MAD: 40.5
    Loss: 0.350%
    Over 40ppm: 0.175%

In group : 50C
    Average: 587.801
    Std dev: 608.963
    MAD: 38.0
    Loss: 0.393%
    Over 40ppm: 0.000%

In group : 55C
    Average: 578.098
    Std dev: 271.469
    MAD: 37.0
    Loss: 0.350%
    Over 40ppm: 0.000%

In group : 60C
    Average: 549.115
    Std dev: 82.640
    MAD: 39.0
    Loss: 0.394%
    Over 40ppm: 0.000%

In group : 65C
    Average: 521.931
    Std dev: 1258.136
    MAD: 38.0
    Loss: 0.262%
    Over 40ppm: 0.962%
```

```
In group : 70C
    Average: 156.602
    Std dev: 76.421
    MAD: 34.0
    Loss: 0.262%
    Over 40ppm: 0.000%

In group : 75C
    Average: 228.379
    Std dev: 66.873
    MAD: 41.0
    Loss: 0.262%
    Over 40ppm: 0.000%

In group : 80C
    Average: 687.415
    Std dev: 564.640
    MAD: 49.0
    Loss: 0.262%
    Over 40ppm: 0.000%

In group : 85C
    Average: 1356.602
    Std dev: 635.996
    MAD: 57.5
    Loss: 0.350%
    Over 40ppm: 0.350%

In group : 90C
    Average: 1975.607
    Std dev: 443.361
    MAD: 71.0
    Loss: 0.131%
    Over 40ppm: 0.262%

In group : 95C
    Average: 2979.806
    Std dev: 314.979
    MAD: 87.0
    Loss: 0.790%
    Over 40ppm: 0.707%
```

Aside from the pathological case at 65°C, where nearly 1% of the adjustments were above the goal, we can see that the protocol remained within the specification up until 95°C, where the average was already high, the packet loss was tremendous, the MAD was the highest one yet. There is a valley between 65°C and 75°C where the synchronization was excellent. The curve of the MAD seems to follow that of the average, while the standard deviation proved itself to be quite erratic, being heavily influenced by the outliers.



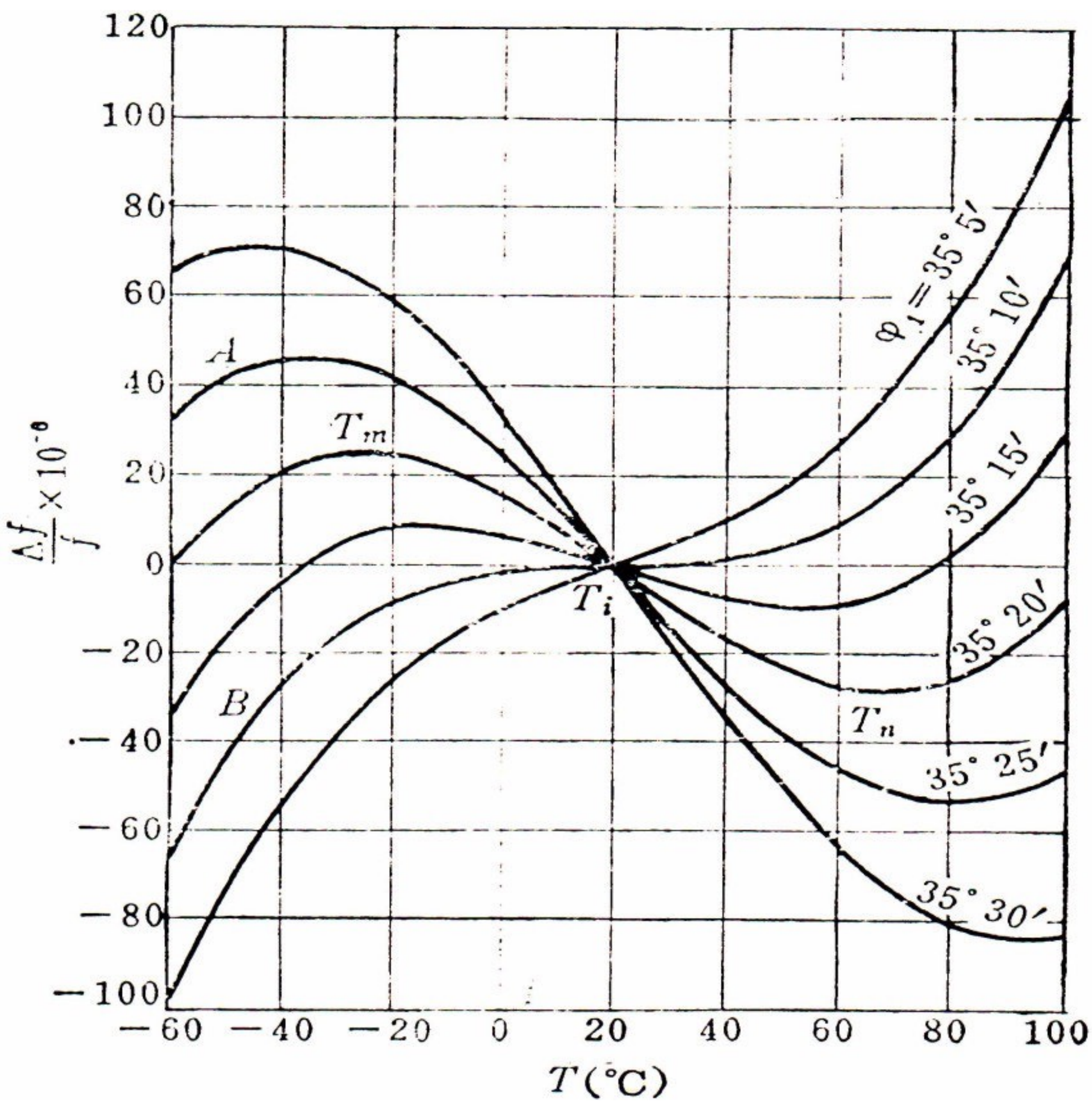

Standard deviation (clock ticks)
1200
1000
800
600
400
200
45
50
55
60
65
70
75
80
85
90
95
Temperature (Celsius)


Median absolute deviation (clock ticks)
80
70
60
50
40
45
50
55
60
65
70
75
80
85
90
95
Temperature (Celsius)

6. Conclusion

The protocol behaved within the expectations. Given that the eMote3 are being used in solar energy plants, it is reassuring to know that it functions within the specifications even in temperatures as high as 95°C. At an update period of 3 seconds, the goal of 40ppm is achieved when the offset is smaller than 3840, and the average was around 2979. Higher temperatures could not only damage the hardware, but cause the TSTP to no longer provide the expected quality.

Shown below is the temperature vs. frequency plot of various AT-cut crystals. Each plot is labeled according to the angle of the cut. Figure was taken from [CNC].



This suggests that the difference in frequency between the crystal in each node will increase with the temperature, and so will the number of offsets above the goal of 3840.

What the temperature vs. standard deviation plot suggests is that this is not a good measure of deviation for this problem, mostly because it is greatly affected by the seemingly random outliers, discussed in section 3. Further investigation on what causes these pairs of outliers is required. Since they appear in pairs, and usually result in an offset that is larger than 3848, a good start is that their rate is approximately equal to half of the "Over 40ppm" rate for temperatures between 40°C and 60°C, and seem to spike at 65°C. Therefore, the user of the network should be made aware, and still decide if it is wise to choose to tolerate these deviations that happen less than 1% of the time. It is possible that one of the units is simply faulty, in which case performing these experiments again to test them before deployment is recommended.

When experiments were performed in different parts of the room, it became clear that with this particular hardware, nodes should be kept reasonably close. A server running inside a metallic case between the two nodes did not seem to impact the quality of the synchronization, but distances of 11m and 12m did. When operating 11m away from its sink, the node was also receiving direct sunlight, and while the average of the offsets was good, perhaps due to the difference in temperature somewhat compensating for differences in the angles of the cuts on the crystal, the other statistics show that the deviation was high, and that the antennae were having difficulties communicating correctly. In a real scenario, this could lead to more power being drained, due to the fact that several attempts to make the packets reach the other side will need to be made, and the signal will need to be strengthened.

7. Future Work

There are two main issues with how the experiments were performed that can be dealt with in future experiments and in attempts to replicate these results.

The first one of them is them is the fact that many elements were not precisely verified. The amount of devices running at once when the spatial experiments were made was not constant, nor was the radio bands which they used. For future experiments, it would be advantageous to have similar nodes communication in the same channel and at an adjustable rate to make sure that the interference is known. Also, the experiments required supervision by a human, who had to watch the thermometer and dim the light accordingly. Automating the process, and perhaps adding multiple thermometers for a more precise measure, are improvements that can be taken into account.

Another issue is that we were trusting the timestamps themselves to measure the offset, but the real offset was never measured. With a function that allows an interruption to take place when the clock reaches a specified, with microsecond precision, we can send a pulse to an I/O pin, and use an oscilloscope to verify to real offset. This provides an even better baseline to compare results.

For future improvements that can use this experiment both as base knowledge, and as a mean to verify the quality of the alterations on the protocol, it is important to note that TSTP is not temperature-compensated, but the hardware that it runs on has support for external thermometers, and the protocol headers allow the nodes to report their current temperature. This information can be used to introduce temperature-compensated time synchronization. Schmid's work provides insight on how to design such protocol, and a good starting point is available in [SCH].

References

[CNC] H. Zhou, C. Nicholls, T. Kunz, H. Schwartz, *Frequency Accuracy & Stability Dependencies of Crystal Oscillators*, Carleton University, Systems and Computer Engineering, Technical Report SCE-08-12 (2008)

[SCH] T. Schmid, Z. Charbiwala, R. Shea, M. B. Srivastava, *Temperature Compensated Time Synchronization,* IEEE Embedded Systems Letters, vol. 1, no. 2, August 2009

[TP0] D. Resner, A. A. Fröhlich, L. F. Wanner, *Speculative Precision Time Protocol: submicrosecond clock synchronization for the IoT* (2016)

[TP1] D. Resner, A. A. Fröhlich, *TSTP MAC: A Foundation for the Trustful Space-Time Protocol* (2016)