TKN Telecommunication Networks Group

Technische Universität Berlin
Telecommunication Networks Group

What's new? Message reduction in sensor networks using Events

Andreas Köpke
koepke@tkn.tu-berlin.de

Berlin, May 19, 2008

TKN Technical Report TKN-08-007

TKN Technical Reports Series Editor:

Prof. Dr.-Ing. Adam Wolisz

Abstract

Observing the environment is the raison d' être of sensor networks, but the precise reconstruction of the measured process requires far too many messages for a low power, long lived sensor network. In this paper, we examine how the number of messages can be reduced using *events*. This paper examines four event definitions and compares the reduction in the number of transmitted messages. In the extreme case, they can reduced to two messages per node and hour starting from more than 3000 messages necessary for the reconstruction of the signal.

Contents

1 Introduction

The purpose of a Wireless Sensor Network (WSN) is to observe changes in the environment. To this end, the individual sensor nodes sample their sensors and send the values to some actuator or gateway. The sample rate is determined by the change rate of the observed process. If the sampled process is free of noise the maximum sampling interval is given by the Nyquist sampling theorem, however, if the sensor readings are noisy, more samples must be taken due to the necessary filtering. In this report, we show that each sensor node must transmit a light message every 600 ms and a temperature message every 30 s, if the noise is filtered at the node prior to transmission. But the network capacity is much smaller, and depends on the energy constraints and network size. As a rough number, the network can transport something like 10 messages per hour and node, assuming a two year life span on two AA batteries. As a result of the energy constraints, each sensor node faces a restriction how many messages it can transmit per hour. This paper deals with the question, how to relieve the tension between high sampling rate on the one hand and low network capacity on the other hand.

For many applications of sensor networks we are not interested in the raw samples, instead we are interested in *events* like "It is too dark in this room" or "It is too cold here". These events can be computed at the data sink using the raw samples, but often they can be computed at the data source. Here, the source transmits a message only if there is an event to report. An *event* can be defined in a variety of ways, and four such definitions are discussed in this paper.

We already introduced the base line definition using Nyquists sampling theorem, and return to it in Section 5.1. The second definition is based on the human perception and discussed in Section 2. The third definition is based on the statistical properties of the measured signal and discussed in Section 3. All the definitions presented so far focus on the "input" side – the sensor readings. In contrast, the fourth definition focuses on the "output" side of the problem: the effect that the reception of a message has on the control algorithm, it is presented in Section 4.

2 Human perception

This section gives a short introduction into human perception, it largely follows [4, ch. 4].

In a home automation setting, the control goal is to increase human comfort. Hence, the human perception can be used to derive a suitable definition of an event:

Definition 1 *An event occurs, when the reading of a sensor changes such that a human can just notice the difference.*

Before we can talk about how to bridge the gap between "sensor reading" and "human perception", it is first necessary to gain an understanding of human perception, because the same change of a variable is not always perceived in the same way. The sensors of a sensor node and the specialized nerves of a human behave quite similar: a change in an observed variable leads to a change in the sensor output. At this level, the human nerves already employ a filter: if other things are more important, the change is not reported. In the next step, the information of more than one nerve is combined and after this the information is interpreted and assigned an importance, involving prior experience, expectations and motivation. If the message is still important enough, the human notices it explicitly.

In a home automation setting it is important to keep all variables at a certain level; a human should not notice the changes. Hence, we will concentrate on the just noticeable difference to derive the definition of an event for the observed variables. The dependency of the level on individual preferences, age and time of the day have to be modelled in the target function of the control algorithm. The just noticeable difference depends on the observed variable, and we discuss it for temperature and illumination.

2.1 Illumination

The changes in illumination are perceived in a relative manner. The absolute change expressed e. g. Lux is important, but it depends on the illumination level. The Weber Law expresses this observation:

$$\frac{\Delta I}{I} = k$$

where k is some constant, I is the old value of the observed parameter, and ΔI the change between the old value and the current value. The constant k depends on the observed parameter, it is different for illumination than for, say, acoustic noise.

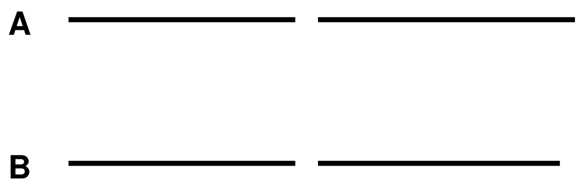

Figure 1: Illustration of the Weber Law

As an illustration of the Weber Law consider Figure 1, to make the observations it is useful to cover A or B, respectively. For A, most people would say that the right line is longer than the left line, while for B one can't be sure. For A, the right line is longer then the just noticeable difference of $k = 0.1$ (3 cm vs. 3.4 cm) while in B it is shorter than this threshold (3 cm vs. 3.2 cm). In your printout, the lines may have a different absolute length, but the conclusion will still hold.

For illumination changes a similar $k = 0.1$ is found. This allows us to define an event for Light:

Definition 2 *A light event occurs, when the illumination change is larger than the just noticeable difference as expressed on the Weber law, using* $k = 0.1$.

While this definition is certainly operational, it is not always valid. For one, the Weber law is not valid over the whole range of the physical process. Secondly, the human eye can adapt to changes in the illumination, if the change is slow enough it will not be perceived. Here, also the direction of the change is important: the eye adapts faster for changes from dark to light (usually within seconds), than the other way around (several minutes). In addition, very fast changes are not observed either. A refined definition of an event should take this into account.

2.2 Temperature

The human perception of temperature does not quite fit the Weber law. Humans do perceive temperature changes on an absolute scale: if it is 30°C and the temperature drops by 5 K to 25 °C, this change will be equally perceived as the drop from 0 °C to -5 °C. Also, there exists a time dependence: fast changes are perceived, while slow changes by the same absolute amount are not. In the range between 0 °C and 30 °C the just noticeable difference is about 0.5 K [3], it is larger outside this interesting region. Neglecting the time dependency, we can define an event for Temperature:

Definition 3 *A temperature event occurs, when the temperature change is larger than the just noticeable difference of 0.5 K.*

3 Statistical significance

The human perception gives an interesting insight into how large a change must be in order to be perceived by a human. Given the cheap sensors on a typical sensor node, the question arises whether such changes can be measured. This leads us to the second approach to define an event:

Definition 4 *An event occurs, when the reading of a sensor changes such that it is statistically significant at a level α. This implies that the change must exceed a certain Δ.*

This definition takes the noise in the sensor readings into account. The major noise source on a sensor node is the digital circuitry. This circuitry switches gates quite fast, causing sudden currents to flow that lead to fluctuations on the power supply lines. Another important factor is the power supply itself: a voltage regulator compares its output voltage to the desired reference voltage. This comparison is not perfect and introduces noise. The noise on the power supply has a direct influence on the quality of the Analog-to-Digital conversion. Besides these digital noise sources, the sensors themselves are also susceptible to noise. Depending on the sensor board design, they are more or less influenced by the digital noise. In addition, they produce an internal sensor noise, which is partly thermal noise.

This major disadvantage of this event definition is that some arbitrary α has to be chosen. The α does not only depend on the noise, but also on the sampling rate: an α of 1% implies 1% false positives: events are reported that are just noise. In a 100 node network where each node samples every second this implies one noise message per second: this can already exceed transport capacity of the sensor network. However, it also implies an unknown number of false negatives: some events are regarded as noise although they are real events. The processes that we observe in our home automation setting usually have a tendency: the temperature continues to fall, the sun continues to rise. So even if an event is discarded as noise, the trend in the process will still cause an event at a later point in time. We can therefore choose an α that suppresses a great deal of the noise messages.

The eyesIFXv2[2] platform that is used in our testbed features the LM61 temperature sensor and the photo resistor NSL19-M51 to measure the illumination. In order to measure the sensor noise, we set up a measurement using all 102 nodes in TKN WIreless Sensor network Testbed (TWIST) [1]. It can not be expected that the sensor noise is equal on all nodes, so having a large sample of sensors helps to find suitable borders.

For the measurements the sensors where sampled every Millisecond for 30 ms. This was repeated every 500 ms for the light sensor and every 2 s for the temperature sensor. From these measurements the mean and the empirical standard deviation where computed and transmitted to a PC for further evaluation.

3.1 Light sensor

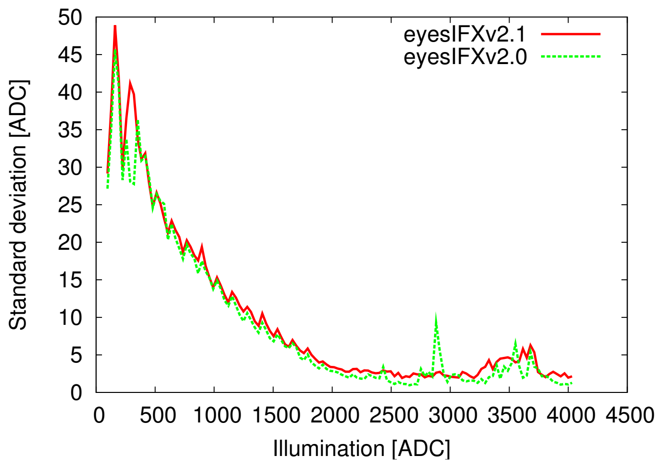

Figure 2: Light sensor noise

The result for the light sensor is shown in Figure 2. The sensor readings are given in their raw ADC values that could be converted to Lux. The photo resistance of the sensor scales logarithmically in the illumination, and a conversion into Lux does decrease the readability of the graph. From this Figure, we see that the sensor noise depends strongly on the level, the darker it is, the higher the resistance and the higher the noise. In addition, there is an increase of the sensor noise around 3500. This is due to the lamps in the offices: these old fluorescent lights flicker with a frequency of 50 Hz. At a significance level of $\alpha = 0.1\%$ the significance borders are defined as shown in Table 1, computed using the normal distribution with the measured standard deviations at each illumination level.

This leads to the following definition of an light event:

Definition 5 *A light event occurs, when the illumination change is larger than the change given in Table 1.*

Interval	Δ
0-499	165
500-999	99
1000-1499	66
1500-4096	33

Table 1: Light, significant change

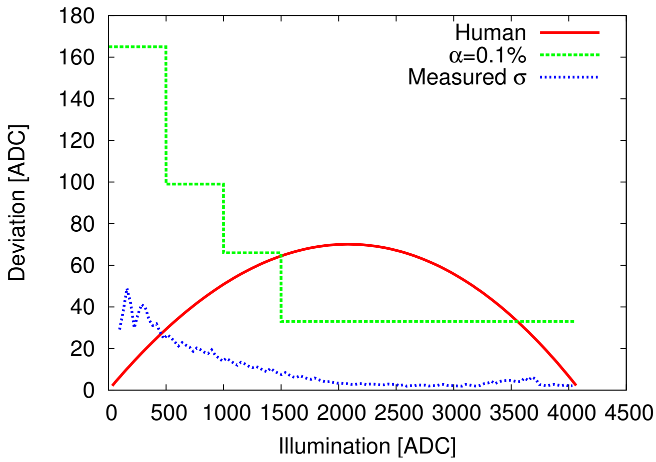

Figure 3: Comparison of Definitions

The Definition 2 and 5 are rather abstract. Figure 3 compares the implied Δs expressed in AD conversion units that can readily be used on the sensor node. This plot exposes a problem with the photo resistor used on the eyesIFX node: its logarithmic scaling makes it hard to find simple thresholds. When it is pretty dark, humans perceive more changes than the sensor, the same applies when it is very bright. Interestingly, this upper crossing coincides with the point where the lights in the offices are switched on: humans can perceive the flicker of the lamps if it changes slow enough. There is also a region where humans perceive fewer changes compared to the sensor.

3.2 Temperature sensor

The results for the temperature sensors is shown in Figure 4. For this sensor, there is a apparent dependency on the platform. A possible explanation is the different voltage regulator used on the nodes. The v2.0 nodes have a voltage regulator with a fixed output voltage, whereas on v2.1 the output voltage is customized using resistors. This increases the overall noise. The main noise source for this sensor is quantization noise. Although this can not be seen from this plot, the sensor noise of the temperature sensor is also temperature dependent: it increases with temperature, but only slightly. In our case, we can define a significant change at $\alpha = 0.1\%$ to be 0.33K. However, the temperature sensor on some nodes have a larger standard deviation, leading to a different border for the event definition:

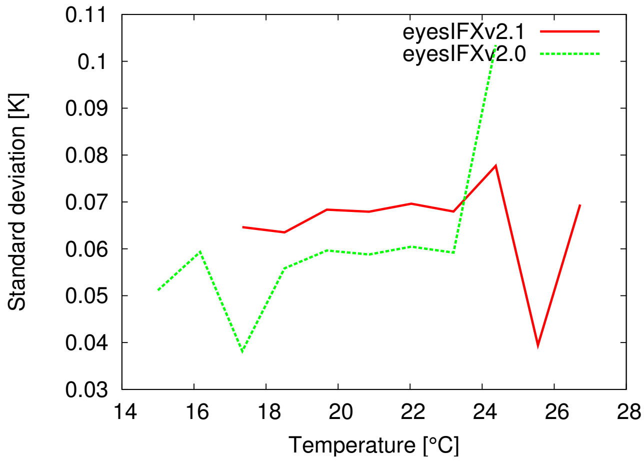

Figure 4: Temperature sensor noise

Definition 6 *A temperature event occurs, when the temperature change is larger than 0.5 K.*

4 Influence on controller

The last definition of an event that is examined in this paper includes the control loop. The lights in an office can e.g. be only on or off. This implies that the sensors have to report their readings only if the illumination level in the office falls below a certain threshold, or when it raises above a certain threshold. More general, the sensor nodes execute the control algorithm and transmit a message only if the result of the control algorithm changes.

The controllers for temperature, widely known as thermostats, can influence the heater in a granular fashion. In contrast to heaters, lamps can often only be switched on and off: A very coarse way to influence the illumination in a room. For this case, it suffices that the sensors send just two types of messages: "It is too dark" or "It is too bright". This will often lead to a further reduction of the number of messages send.

5 Results

5.1 Periodic sampling interval

The maximum sampling interval that allows the exact reconstruction of a noise free signal is given by Nyquists sampling theorem. It says that the sampling frequency should be twice as high as the highest frequency in the signal. To get an impression how fast changes occur in an home automation setting, we used TWIST. The measurement lasted for over a month, starting from December 14th, 2007 and ended on January 18th, 2008. The measurement included 102 eyesIFX nodes in the testbed.

5.1.1 Light sensors

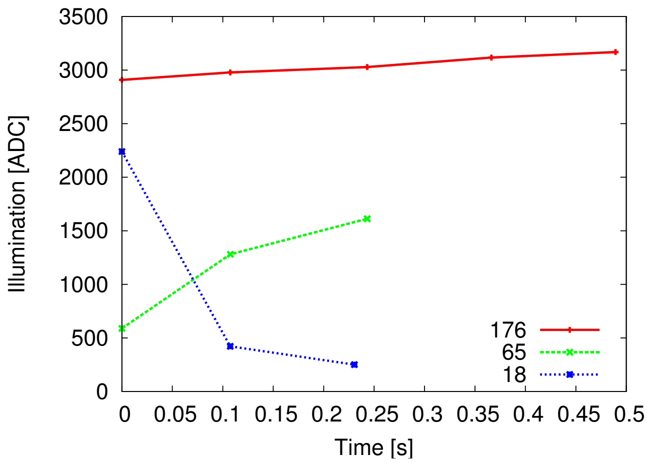

Figure 5: Fast illumination changes

The light sensors where sampled approximately every 100 ms. The samples are not completely equidistant, because the reference voltage generator needs some time before he provides a stable voltage. The reference is switched off when it is idle for some time. This introduces a jitter into the sampling: sometimes the reference is still on from an earlier measurement, sometimes it is off. For each sample, the sensor was read every 2 ms and 20 samples where taken. The average of these values is used as the sensor sample. This low pass filtering was necessary to filter out the flicker of the old fluorescent lamps in the offices. Besides this low pass filtering, no other filters where employed. Using the samples from the testbed, we sought three nodes that measured the fastest changes during the entire measurement campaign. The changes sampled by these nodes is shown in Figure 5. The minimum time difference between the samples is not surprising: it is determined by the ad hoc chosen sampling interval. From these plots it seems reasonable to increase the sampling interval. With the current sampling interval of 100 ms each node generates 10 messages per second. If we periodically extend the graphs in Figure 5 to full periods, the duration of a period would be 1200 ms. According to Nyquists sampling theorem, this means that we can increase the sampling interval to 600 ms, lowering the network load by a factor of six. This load can be further reduced by using the spatial redundancy of the sensor nodes. In TWIST, each room has two sensor nodes. With proper synchronization we could increase the sampling interval of each node to 1.2 s, lowering the overall network load by a factor of twelve compared with the ad hoc chosen sample interval.

5.1.2 Temperature sensors

Temperature is in general a slow process. The initial, ad hoc chosen sampling interval is therefore 1 s. To filter out the sensor and digitization noise, the readings where low pass filtered like the light sensor readings: The sensor was sampled every 2 ms and the mean of 20 samples was used as the real value. Figure 6 shows the

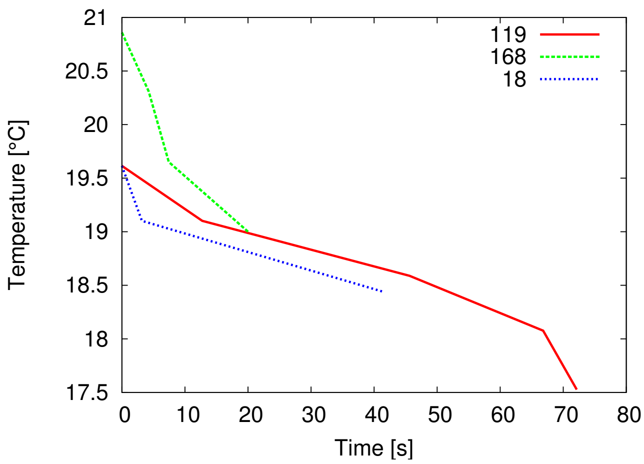

Figure 6: Fast temperature changes

fastest temperature changes that occurred during the measurement. Usually, the temperature drops on the scale of minutes, but sometimes the temperature drops significantly within 5 s. It seems reasonable to extend the sampling interval to 30 s or even a minute – the loss in precision is acceptable.

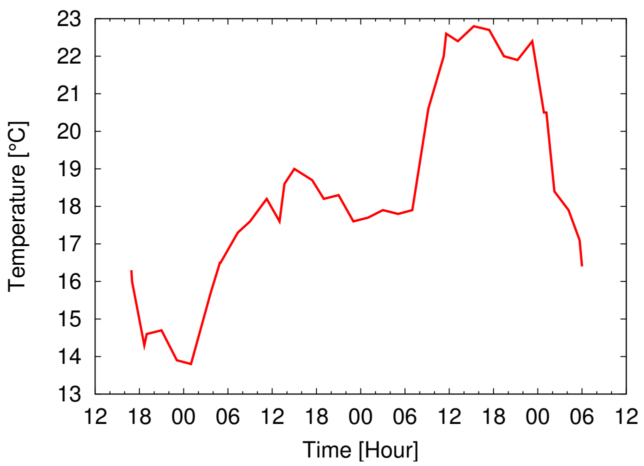

Figure 7: Roof server temperature

The temperature changes in Figure 6 are quite fast: opening a window in an office when the outside temperature is below 10°C results in a fast drop. The temperature inside a computer on the roof of our building does not change as fast, as can be seen from Figure 7. There, the temperature changes by 0.5 K in about 30 min.

5.2 Events

In the last section, we examined the potential reduction in network traffic if the sampling interval is increased. While this approach is interesting and does provide some data reduction, it has its limits. In this section, we want to examine the data reduction that can be achieved using the event definitions presented in Section 3. Because of the sensor noise we could not use the definitions based on human perception. However, both definitions are quite close in our case. For temperature, both definitions match. For light, the relative change is important for humans, and the logarithmic scale of the photo resistor introduces a similar feature into the measurements.

To assert how many messages would still be generated by the nodes, a long measurement was conducted. It started on December 14th 2007 at 20:16 o'clock CET and lasted till January 18th, 2008 at 16:37 o'clock. The measurement included all 102 eyesIFX nodes in TWIST. Because some of the supernodes responsible for the transmission of the measured values to the PC crashed, values from 17 nodes arrived only for a part of the measurement duration. These partial traces were excluded from the analysis, together with the two nodes from which the sensors where removed due to privacy concerns. Although the data of some nodes was lost, all other messages were transferred correctly: the results were send via the testbed, and not via the radio modem of the nodes.

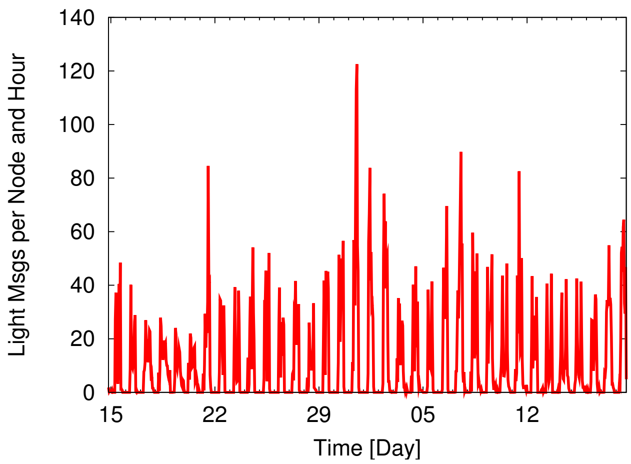

Figure 8: Number of illumination change messages

Each time a node senses an illumination change, it generates a message. The number of messages during one hour is counted, and this sum is divided by the number of nodes. The result is the average number of messages per node and hour as shown in Figure 8. The maximum is slightly above 120 messages per node and hour: a substantial reduction compared with the minimum of 3000 messages generated using the periodic sampling approach.

As similar approach is taken for the temperature changes and the result is shown in Figure 9. The number of messages drops again compared with the periodic sampling approach. However, the result is somewhat limited: Each room is equipped with a controlled heater that keeps the temperature at a certain level. Hence, the result shown in Figure 9 may underestimate the number of messages. On average, less than two message is send by a node per hour, which is quite a reduction compared with the 60 messages of the periodic sampling approach.

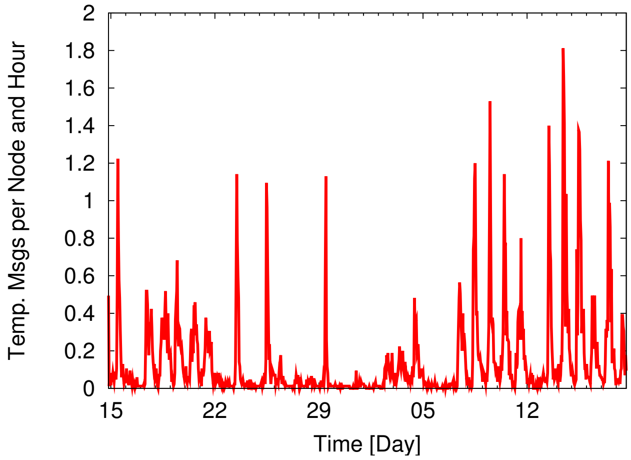

Figure 9: Number of Temperature change messages

Comparing this with Figure 7 shows that this peak load estimate is consistent in a rather different environment.

5.3 Effect on controller

The evaluation of the effect on the controller is a bit difficult, due to calibration issues. In principle, it is easy to calibrate the temperature sensor by comparing them to a reference temperature sensor, but this is tedious work given the large number of nodes. The situation for the light sensors is even worse. The nodes are mounted on the ceiling with the light sensor facing towards the floor of the room. Depending on the specific office layout, the may either look upon a white desk or on the grubby gray floor. These surfaces reflect a different amount of light and thus the light sensors need to be calibrated every time something changes in the office. In addition, the light resistor has a logarithmic scaled output and this makes it impossible to express the calibration as a linear equation.

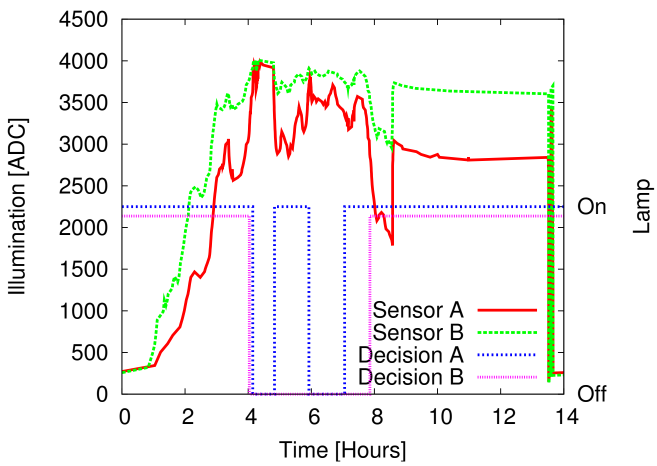

Figure 10: Measured Illumination and corresponding decisions

To get an impression on the potential of this approach, we nonetheless evaluate it based on the measurements used in Section 5.2. In a first step, we “calibrated” the light sensors, using the effect of the lamps on the illumination. Since the measurement took place during the winter, the day light influenced the measurements only between 8 o’clock and 16 o’clock. If we go into each room before 8 o’clock, switch on the lights, wait a while and switch them off again, we get a good impression on how the sensor readings change just because of the lamps. The cleaning staff helped us here: they enter the room before 8 o’clock, switch on the lights, do the cleaning and leave the room. This provides a set of samples on how much the lamps influence the light sensors. These samples where collected for each sensor node, converted to Lux and averaged. This way, a single calibration point L_c for each sensor according to its current environment is obtained. Based on this point, we defined a hysteresis for each sensor. When the illumination level as seen by this sensor falls below the calibration point L_c, it sends a “too dark” message. When the illumination rises above $2.4L_c$ it sends a “too bright” message. The actuators in the room would have to communicate to reach a consistent result, because the individual sensors may contradict each other. Figure 10 shows the measured illumination and the resulting decisions for two sensors. The sensor readings are plotted on the left y-Axis and the decisions are plotted on the right y-Axis. This plot shows that the decisions are “sane”: they do not oscillate too often. It also shows that in a real setting one would like to fine tune the values further. As an indication, take the sharp increase shortly before the ninth hour in this trace, when I switch on the lamps. This happens later then the sensors suggest as a result of my personal preferences. The plot also shows that this approach is incomplete: During the night the lamps remain on, even if no person is in the office. This approach should be combined with a presence detection.

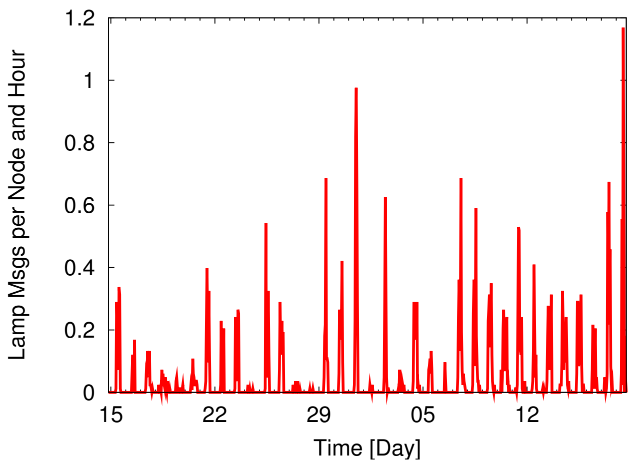

Figure 11: Number of “too dark” or “too bright” messages

The simple hysteresis allows a first glance on the necessary number of messages. One should keep in mind that this number depends on the control algorithm used and on the control granularity. Figure 11 shows that this approach can further reduce the number of light messages: each node sends less than two messages per hour, the maximum number of illumination change induced messages is even lower than the number of temperature change messages. Of course, a finer control of the lamps will in general increase the number of messages.

6 Conclusion

The number of messages that are send in a WSN is crucial. It is important to send as few messages as possible, while maintaining the goal of the application. For a home automation setting, we examined four different approaches to reduce the number of messages in the network. The first approach tries to maximize the time between two samples that still allows to capture all changes. This base line approach reduces the number of messages compared with the ad hoc chosen sampling intervals used in the measurement campaign. The second approach takes the human perception into account, but the sensors on the platform do not perfectly support such an approach: the sensor readings are noisy and not reliable enough. We therefore evaluated a statistical approach, where messages are only send when the difference to the previously send reading is statistically significant at $\alpha = 0.1\%$. This reduces the number of messages by about a factor of 30 compared with the periodic sampling approach. Still, the number of light messages can be quite high. To reduce them further, we introduced a simple control algorithm on each sensor node. This allows each node to send messages only, if they influence the result of the control algorithm. The number of light messages drops by a factor of 2500 compared to the periodic sampling approach. In passing, this paper shows that it is possible calibrate the sensors in situ.

References

[1] Vlado Handziski, Andreas Köpke, Andreas Willig, and Adam Wolisz. TWIST: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks. In *REALMAN '06: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, pages 63–70, New York, 2006. ACM Press.

[2] Infineon. *eyesIFX Wireless Sensor Network Development Kit*, November 2005. http://www.infineon.com.

[3] Robert F. Schmidt, Gerhard Thews, and Florian Lang. *Physiologie des Menschen*. Springer Verlag, Berlin, 29 edition, 2005.

[4] Philip G. Zimbardo. *Psychologie*. Springer, Berlin, 6 edition, 1995.