Transmission Power Control for Interference Minimization in WSNs

Georg von Zengen, Felix Büsching, Wolf-Bastian Pöttner and Lars Wolf Institute of Operating Systems and Computer Networks TU Braunschweig

Email: [vonzengen|buesching|poettner|wolf]@ibr.cs.tu-bs.de

Abstract—Typical home automation systems consist of dense populated sensors and actuators in wireless networks. In apartment houses, networks are often deployed closely to each other by different neighbors. In such scenarios wireless networks suffer from packet loss caused by interference with other networks. Packet loss causes unnecessary delay and energy consumption. To overcome this, transmission power control can be used to lower the interference. However, reducing the interference is a rarely used capability of transmission power control. This paper presents transmission power control based on a P-controller. The feedback loop for the signal-to-noise ratio is realized by modifying the link layer acknowledgments of IEEE 802.15.4. The evaluation of the system is performed in a real-world testbed. It shows the benefit of reducing the transmission power in dense networks. The application layer loss can be lowered by 5%.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) become an emerging field for home automation. In home automation the network nodes usually represent actuators and sensors. Typical examples for sensors are light switches and thermostats, actuators are lamps or HVAC systems. This leads to a dense deployment of wireless nodes in these networks. Placing many nodes in the transmission range of each other causes a higher risk of collision and so increases the interference seen by a single node. Other sources of interference are different home automation networks deployed by nearby neighbors especially in apartment buildings.

This interference causes packet loss which can be minimized by using the lowest possible transmission power that still allows successful packet reception. Packet loss should be minimized since additional delay is caused by the resulting retransmission. By decreasing the transmission power the transmission range is also decreased. This lowers the number of nodes in the transmission range which lowers the possibility of collisions and so reduces the interference.

Transmission Power Control (TPC) is a field where lots of work has been done. Most of these aims at reducing the power consumed by transmissions to extend the lifetime of nodes. The aim of the TPC system presented in this paper is to reduce the interference caused by a transmission. To do so, a P-controller is introduced which controls the transmission power based on the Signal-to-Noise Ratio (SNR). The feedback of the SNR is given to the sender by utilizing the link layer acknowledgments of IEEE 802.15.4.

This paper is structured as follows: The next section discusses the related work which mostly tries to lower the transmission power to be able to save energy. Afterwards the design of the system is presented. The third part describes the implementation of our TPC system. Following the TPC approach is evaluated in a real-world testbed. The last part concludes this paper.

II. RELATED WORK

Most of the research regarding TPC is focused on battery powered networks. Due to this fact, most approaches try to minimize the energy consumption to extend the lifetime of battery powered devices.

In general there are three main categories. (A) Approaches which control the transmission power depending on the number of lost packets. (B) Methods controlling the power according to the Received Signal Strength Indicator (RSSI) or the Link Quality Indicator (LQI). (C) Adjusting the transmission power in relation to the number of neighbors is the third approach. In this section different variants of these groups will be discussed in detail.

A. TPC based on packet loss

Correia et al. [1] published a two phase algorithm to calculate and adjust the ideal transmission power. In the first phase the sender decreases the transmission power for every packet sent until acknowledgments from the receiver are no longer received. The ideal transmission power is the one at which the last acknowledgment (ACK) was received. After this value is determined, the second phase is entered.

In this phase the algorithm uses the ACKs to adjust the transmission power to environmental changes. This adjustment is triggered if a certain number of acknowledgments are received or lost in a row.

An advanced version is discussed in [2]. Here the authors added a mechanism that measures the noise power when no transmissions are active in the network. Together with the measurement of the received power this enables the receiver to calculate the SNR. If the SNR is below or above a threshold, a bit is set in the ACK to advise the sender to adjust its transmission power. Being an iterative method, it has the disadvantage that it cannot react to sudden changes of the noise power. Another presented method is called AEWMA [3] and stores the transmission power in every packet. The receiver calculates the ideal transmission power considering the SNR. Afterwards it sends the ACK-packet with the calculated power and also piggybacks the power to this packet. If the original sender wants to transmit the next packet it will use this power level. By using the calculated power to transmit the ACK this method assumes symmetric channel characteristics. However research [4] has shown that wireless channels are not necessarily symmetric.

B. TPC derived from LQI or RSSI

ATPC (Adaptive Transmission Power Control) by Lin et al. [5] describes a control loop mechanism for transmission power control. A sender transmits so called beacons at different power levels and stores these levels in a matrix. A receiver measures the power of the received packet and replies a feedback packet to the sender which contains the measured power. With this measurements the sender can calculate the ideal transmission power. In a second phase beacons will only be transmitted when no other packets have been sent for a certain time. A node will change into this phase after it has calculated the ideal power. In the second phase the ideal transmission power is calculated in a more lightweight way. This can be done because the static part of the link characteristics will be used from the former calculation.

Another approach was published by Pöttner et al. [6]. Here the sender stores the transmission power for every packet and the receiver measures the RSSI. But in this approach no extra packets are used to transfer the RSSI back to the sender. Instead it is piggybacked into the link layer acknowledgments. With this RSSI the sender calculates the attenuation of the link and adjusts the transmission power to compensate it. To smooth outliers an Exponential Weighted Moving Average (EWMA) is used. This method is able to react fast to changes of channel characteristics and does not need an initialization phase.

Both approaches assume that the link quality depends on the RSSI. This disregards the impact of the noise to the link quality.

C. TPC based on the number of neighbors

Jeong et al. [7] developed the theory that there is a coherence between the transmission power and the number of nodes that receive a packet. Based on this an algorithm was realized that in the beginning sends broadcast packets. If the packet is received by another node, it returns an ACK. The original sender will receive these ACKs from all nodes that received the broadcast packet. If the number of ACKs is higher than a configured value the transmission power is decreased by a certain step. If the target value was undershot the transmission power is increased by half a step. This procedure will be repeated until the number of neighbors is equal to the configured value.

By using the same transmission power for all transmissions, this approach causes unnecessary interference when communicating with nearby neighbors.

III. SYSTEM DESIGN

In contrast to the approaches presented in Section II, our approach aims at minimizing the packet loss caused by interference using the SNR to adjust the transmission power. The SNR is the ratio between the received signal power and the noise power. If both the signal power and the noise power are presented in Decibel the SNR can be calculated as described in the following formula:

$$SNR = RSS - noise \tag{1}$$

Where Receiver Signal Strength (RSS) is the signal strength received by the receiver given in dBm. RSS can be derived from the RSSI. The way to do this depends on the used radio

chip. Noise in this case means everything but the signal used to transmit the packet from the sender to the receiver. That means the interference from other links in the same network and other networks is also included. This decision was made because it is difficult to differentiate between background noise and interference. Moreover having the two values separated will not give an advantage.

As shown by Zamalloa et al. [4] the bit error rate theoretically correlates with SNR and leads to a Packet Reception Rate (PRR).

If the difference between noise and signal is greater, the bit error rate is supposed to be lower which results in a higher PRR. This coherence can also be discovered in Figure 1. Where SNR rises almost linear form 80% to 100% PRR, which indicates a tight coherence. The data presented in this figure was gathered in an experiment which ran in the testbed described in Section V. These results motivate the use of SNR as a control variable for the TPC.

We now describe the design of our TPC approach. First the measurement of the SNR is discussed. The obtained values are transmitted using the link layer acknowledgments. The third part describes the control circuit to adjust the transmission power.



Figure 1. Correlation between SNR to PRR

A. Measurement

To get the most precise value for the noise it is measured between the reception of the packet and the transmission of its acknowledgments. At this time no node in the transmission range of the sender should transmit a packet, thus the noise can be measured without interference from its own network.

B. Acknowledgments

After the SNR is measured it is piggybacked into the acknowledgment and transmitted to the sender of the packet. To be accepted by the sender, according to [8], the ACK has to be transmitted within 864 μ s. Using the ACKs to transfer the SNR to the sender has the advantage that no extra packet has to be transmitted. It was inspired be the work of Pöttner et al. [9] where the RSS is piggybacked to the link layer ACKs. To piggyback the SNR into the ACKs, these messages have to be generated in software. The whole procedure of measuring, piggybacking, generating and sending the acknowledgment is shown in Figure 2.



Figure 2. Sequence diagram of software acknowledgments

C. Adjustment of the Transmission Power

A control circuit is used to adjust the transmission power . The receiver acts as the sensor which measures the SNR and transfers it back to the sender. The sender then calculates the new transmission power. There is a target SNR (SNR_p) which is set to a predefined value at which the mean PRR is high enough for the application of the network. From the receiver the sender gets SNR_c, which represents the SNR for the last packet. With SNR_p and SNR_c the difference *e* can be calculated as follows:

$$e = \mathrm{SNR}_p - \mathrm{SNR}_c \tag{2}$$

The new transmission power level P_t is calculated with the following formula:

$$P_t = P_{t-1} + K_p * e (3)$$

Where P_{t-1} is the transmission power level for the last packet and K_p is a predefined factor to scale the error *e*.

As the last formula shows the controller designed here is a so-called P-controller with a proportional factor K_p . The P-controller was chosen because more advanced controllers like a PID-controller would need a constant time between the measurements. However, such a constant inter-measurement time cannot be guaranteed in our system because application packets are usually not sent in a precisely constant interval and the transmission of extra packets should be avoided. So the Pcontroller was the only choice that would not add a significant amount of communication overhead.

Due to the fact that the P_t needed to reach SNR_p depends on the distance between the sender and receiver and other environmental variables, every link needs its own control circuit with its own P_t , P_{t-1} , and e. In this system broadcast packets are not affected by the TPC because it is assumed that these packets should be received by as many nodes as possible. K_p and SNR_p can be used identically for all links. Figure 3 shows the control circuit with its parts and variables. The red *Link* box represents the link where the disturbance variables, *noise* and *attenuation*, effect the transmission power P. The RSS measured by the receiver can be calculated as following:

$$RSS = P - noise - attenuation \tag{4}$$

The receiver radio is represented by the *Sensor* box. If the *attenuation* is assumed as constant over time and the *noise* is assumed as negligibly. The necessary transmission power level

 P_t could easily be calculated from the transmission power P and the *attenuation*.



Figure 3. Control circuit for transmission power

IV. IMPLEMENTATION

The following section outlines the implementation of the system described in Section III. It is implemented for the Contiki¹ operation system. The hardware platform is the Tmote Sky [10]. The first part discusses the measurement of the necessary values. These values are piggybacked into link layer acknowledgments as described in the second part. The last part describes how these values are used to adjust the transmission power.

A. Measurement of SNR

The necessary values, to calculate the SNR, are the RSS and the noise with formula 1 they result in SNR. The RSS represents the signal strength of the last 8 symbols of an IEEE 802.15.4 packet and can be read out of the *RSSI_VAL* register of the CC2420.

The noise measurement necessary for the SNR calculation is done between the reception of the packet and the transmission of the ACK. This is possible because the automatic acknowledgment functionality of the CC2420 is deactivated and the ACKs are generated and send by the software. The noise value is smoothed by a EWMA with the following formula:

$$noise_t = noise_{new} * 0.2 + noise_{t-1} * 0.8 \tag{5}$$

Where $noise_{new}$ is the noise value that was measured for the last packet and $noise_{t-1}$ is the result from the last execution of the formula. This is done to smooth outliers resulting from reflections and interference.

0 7	11	17	23
First Byte of Frame Control Field (FCF)	DSN	Noise	SNR
Table I. DATA ORDER IN A TPC-ACK			

¹http://www.contiki-os.org/

B. ACK Generation

To be able to generate ACKs in software and meet the timing restrictions of the IEEE 802.15.4 [8] standard the FIFO of the CC2420 [11] is read out in 6 byte blocks. Reading the first block is triggered by the *FIFOP*-pin of the CC2420 that is connected to an interrupt pin of the msp430. The *FIFOP*-pin switches to "low" if the preconfigured threshold of bytes in the FIFO is exceeded. Here this threshold is defined as 6 bytes.

To generate the normal ACK only the Data Sequence Number (DSN) is needed. But for the TPC-ACK the noise and SNR are also needed. Because the ACK have just a size of three bytes, SNR and noise has to be compressed. This is done by inverting and decreasing the noise by 60, thereby it is possible to encode the noise in 6 bits. The operation of decreasing by 60 does not compromise the value of the noise because it is in most cases higher than 60. This assumption is supported by Figure 1. The SNR is encoded in 6 bits, this is possible because this value of the SNR is in most cases below 64. By shortening the DSN to four bits the length of DSN, SNR, and noise is 16 bits. So the last two byte of the ACK are used to transport the SNR and noise. The first byte is used for the *frame type* and other information from the Frame Control Field (FCF). Shortening the FCF is possible because all necessary information are in the first byte. Putting data into the last two bytes of the ACK is possible because in IEEE 802.15.4 it is specified that all information except the DSN and some information from the first byte of the FCF should be ignored when receiving an ACK. Table I shows the order of the data in the ACK.

After generating the ACK it is send back to the sender of the original packet.

C. Transmission Power Adjustment

If a node receives an ACK it will decode the noise and SNR from the packet. This decoding is implemented in the driver for the radio chip. By taking the SNR and using it in the following formula the transmission power for the next packet is calculated.

$$P_t = P_{t-1} + K_p * (SNR_p - SNR_c)$$
(6)

As seen from the formula the controller implemented here is a proportional controller. P_t is the transmission power level for the next packet and P_{t-1} the one for the last packet. K_p is the factor with which the difference between the proposed SNR (SNR_p) and the SNR of the last packet (SNR_c) is scaled. With this factor the slope of the transmission power curve, after a change of the noise, can be adjusted. The former transmission power level (P_{t-1}) is needed because otherwise the formula would return zero if the proposed SNR is reached.

After P_t is calculated it will be stored in the list that holds the P_t and P_{t-1} for all known neighbors. If the node intends to send a unicast packet, P_t for the destination node is loaded from the list and the packet is transmitted using this P_t .

For the first packet that a node transmits to another node, the highest power level is selected.

For packets without an ACK the formula 2 is calculated with SNR_c equals zero. That results in the highest possible increase of P_t . Figure 3 shows that the calculation of P_t is done at the sender and that noise (RSS_{noise}) and SNR of the last packet (SNR_c) are calculated at the receiver. The figure also shows

the list of the nodes that are stored with the values P_t and P_{t-1} .

V. EVALUATION

All experiments were performed in the *Wisebed* [12] testbed shown in Figure 4. Experiments that utilize one link use node 19 as the sender and node 7 as the receiver. If an experiment utilizes two links node 8 is used as an additional sender and node 9 as a receiver. Every sender sends 40 unicast packets per second to its receiver using *rime mesh* [13]. To be able to send 40 packets per second the channel check rate was increased to 64 checks per second.



Figure 4. Testbed in the first floor of the Computer Science Building

A. Selection of Evaluation Parameters

One of the most important parameters for our TPC approach is the SNR that every packet is supposed to reach (SNR_p) . As shown in Figure 5 it has an influence on the packet loss at the link. To generate Figure 5 over 120000 packets were analyzed for each configuration.

The figure shows a strong relation between the retransmission probability and SNR_p . In the following SNR_p 15 dBm will be used. A retransmission probability of 0.07 means a link layer packet loss of 7%. This loss can be statistically compensated by the three link layer retransmissions.



Figure 5. Probability of packet loss to proposed SNR

B. Controller behavior over time

This section analyzes the behavior of the controller over time. Analyzing this is important to be able to give a statement whether the controller is adjusting the transmission power as it is supposed to be.

To be able to recognize the different lines over time in Figure 6 each printed value represents 10 with a moving average smoothed values. Due to this, the power level (green line) which is normally an integer value has steps of 0.1 in this figure. The values plotted in Figure 6 are an exemplary subset of the data collected in Section V-A. Therefore, the time does not start at zero.

In the figure the red dashed line represents SNR_p which is at 15 dBm for the whole time. The blue line represents the measured SNR for the last packet (SNR_c) . By comparing these lines it can be observed, that the controller is not able to adjust SNR_c exactly to SNR_p . Here the controller suffers from the inexact adjustability of the power levels. The fact that the power level is between 11 and 10.5 leads to the observation that the power level was frequently switching the power levels around 10.5. By comparing the power level and SNR_c a strong relation can be observed. This means the controller is able to adjust SNR_c by adjusting the power level.



Figure 6. SNR and transmission power level over time

C. Adaption to sudden changes of the noise

This experiment analyzes the reaction of the controller to a sudden rise of the noise power on the current channel. This is important to be able to estimate the time the controller needs to adjust the transmission power level according to the new noise power.

To evaluate this, the transmission power level of the link between node 9 and 8 is plotted over time in Figure 7. Where the dashed green line represents the time the disturber (node 15) was activated. The disturber was realized by transmitting packets as fast as possible without performing a clear channel assessment.

As the figure shows, the controller adjusts the power in less than 0.25 s and needs just seven measurements to reach a higher level. So, this experiment proved that the controller is able to adjust the transmission power very fast without overshooting.

D. Influence of TPC on the retransmission probability

The goal of this experiment is to evaluate a network using TPC in comparison to one without TPC. In Figure 8 the link



Figure 7. Reaction to suddenly rising noise

layer retransmission probability for a network with a single link is plotted. The left bar represents the network using TPC, the left one represents the network without TPC. The thinner bars represent the standard deviation. To measure the probability, the sender counts how many packets it has to retransmit in one second. By knowing that 40 packets were sent in this time the probability for every packet to be retransmitted can be calculated. The average value and the standard deviation are plotted.

As the figure shows, the retransmission probability in the network using TPC is increased by less than 0.01. This extra retransmission probability is below the standard deviation of the network without TPC. So the extra probability is almost negligible.

Another interesting measurement is the retransmission probability in two neighboring networks. Therefore, a measurement of the retransmission probability using two links was made. The first link was between node 19 and 7, the second between nodes 8 and 9. These links were chosen because they are close enough to interfere with each other but not overlapping. This is the case when networks are deployed in neighboring apartments. In Figure 9 the right bar represents the retransmission probability between nodes 7 and 19 while using TPC on both links. The left bar represents the probability between the same nodes while using TPC on one of the links.

Here TPC decreases the retransmission probability by 0.02. This is because of the decreased interference from the other



Figure 8. Comparison of retransmission probability on a single link for TPC enabled and disabled

link.

A greater influence can be observed by analyzing the application layer packet loss of the same link. Here, TPC decreases the packet loss by almost five percent, as shown in Figure 10. The higher influence on the application layer packet loss can be explained due to the fact that a collided packet will be retransmitted by both senders. In this case the overall risk of collisions rises due to the increased number of packets. By causing more collisions the maximum number of retransmissions is reached more often. This results in a higher application layer packet loss.



Figure 9. Comparison of retransmission probability on two links for TPC enabled and disabled



Figure 10. Application layer packet loss for two links with enabled and disabled $\ensuremath{\text{TPC}}$

VI. CONCLUSION

This paper introduces a system which aims at reducing the packet loss caused by interference between different wireless links. To reach this goal a TPC approach based on SNR is presented. The transmission power is adjusted according to the SNR using a P-controller. The feedback for the P-Controller is realized by piggybacking SNR information onto the link layer acknowledgments of IEEE 802.15.4. By using these acknowledgments no overhead is created to adjust the transmission power. After discussing the design and implementation, a real world evaluation was performed. This evaluation showed that the system is able to reduce the application layer packet loss by 5%. Other experiments showed that it reacts fast to suddenly rising noise levels which is an advantage of the P-controller compared to iterative TPC approaches.

This system supports the dense deployment of many sensor nodes in a small area. Additionally, it can help to extend the lifetime of densely deployed WSNs by reducing the interference at nearly no costs.

For the future we plan to gather more data from our testbed to be able to find better parameters for the TPC. Better parameters will lead to further reductions of the packet loss.

REFERENCES

- [1] L. H. A. Correia, D. F. Macedo, D. A. C. Silva, A. L. dos Santos, A. A. F. Loureiro, and J. M. S. Nogueira, "Transmission power control in mac protocols for wireless sensor networks," in *Proceedings of the 8th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '05. New York, NY, USA: ACM, 2005, pp. 282–289.
- [2] L. H. A. Correia, D. F. Macedo, A. L. dos Santos, A. A. F. Loureiro, and J. M. S. Nogueira, "Transmission power control techniques for wireless sensor networks," *Comput. Netw.*, vol. 51, no. 17, pp. 4765–4779, Dec. 2007.
- [3] L. Correia and J. Nogueira, "Transmission power control techniques for mac protocols in wireless sensor networks," in *Network Operations and Management Symposium*, 2008. NOMS 2008. IEEE, 2008, pp. 1049– 1054.
- [4] M. Z. n. Zamalloa and B. Krishnamachari, "An analysis of unreliability and asymmetry in low-power wireless links," ACM Transactions on Sensor Networks, no. 2, Jun. 2007.
- [5] S. Lin, J. Zhang, G. Zhou, L. Gu, T. He, and J. A. Stankovic, "ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, ser. SenSys '06, 2006.
- [6] W.-B. Pöttner and L. Wolf, "Probe-based Transmission Power Control for Dependable Wireless Sensor Networks," in *Distributed Computing* in Sensor Systems (DCOSS), Cambridge, MA, USA, 2013.
- [7] J. Jeong, D. Culler, and J.-H. Oh, "Empirical analysis of transmission power control algorithms for wireless sensor networks," in *Networked Sensing Systems*, 2007. INSS '07. Fourth International Conference on, 2007.
- [8] IEEE, "IEEE Standard for Local and metropolitan area networks— Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)," 2011.
- [9] W.-B. Pöttner, S. Schildt, D. Meyer, and L. Wolf, "Piggy-Backing Link Quality Measurements to IEEE 802.15.4 Acknowledgements," in 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems. Ieee, Oct. 2011.
- [10] M. Corporation, "Tmote Sky Datasheet," 2006. [Online]. Available: http://www.eecs.harvard.edu/~konrad/projects/ shimmer/references/tmote-sky-datasheet.pdf
- [11] Chipcon, "SmartRF CC2420 Datasheet," 2004. [Online]. Available: http://www.ti.com/lit/gpn/cc2420
- [12] H. Hellbrück, M. Pagel, and A. Kroller, "Using and Operating Wireless Sensor Network Testbeds with WISEBED," in Ad Hoc Networking Workshop (Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean, 2011.
- [13] A. Dunkels, F. Österlind, and Z. He, "An adaptive communication architecture for wireless sensor networks," in *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems*, ser. SenSys '07. New York, NY, USA: ACM, 2007.