Aalborg Universitet

AALBORG UNIVERSITY

Data Recovery using Side Information from the Wireless M-Bus Protocol

Melchior Jacobsen, Rasmus; Popovski, Petar

Published in:
IEEE Global Conference on Signal and Information Processing (GlobalSIP 2013)

DOI (link to publication from Publisher):
10.1109/GlobalSIP.2013.6736927

Publication date:
2013

Document Version
Early version, also known as pre-print

Link to publication from Aalborg University

Citation for published version (APA):
Melchior Jacobsen, R., & Popovski, P. (2013). Data Recovery using Side Information from the Wireless M-Bus Protocol. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP 2013)* (pp. 511-514). IEEE Press. https://doi.org/10.1109/GlobalSIP.2013.6736927

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: December 05, 2025

Data Recovery using Side Information from the Wireless M-Bus Protocol

Rasmus Melchior Jacobsen*†
*Kamstrup A/S
Denmark
rmj@kamstrup.dk

Petar Popovski†
†Aalborg University, Dept. of Electronic Systems
Denmark
petarp@es.aau.dk

***Abstract*—Dedicated infrastructure for wireless battery-powered smart meters involves a costly setup. To reduce the cost, one often made compromise is the link reliability, where a majority of packets are allowed to be in error, as long as a few packets are received over a long time frame. On the other hand, each meter transmits frequently to accommodate for drive-by/walk-by remote reading applications, leaving redundancy in the transmissions seen on a dedicated receiver. We utilize the redundancy and the inherent timing structure of the protocol in order to recover metering data from multiple erroneous packet receptions. From an actual deployment a receiver sensitivity gain is found to be around 1dB, equivalent to increasing the number of read meters by 14% in an experiment over 66 hours.**

***Index Terms*—WM-Bus, Packet Recovery, Protocol Coding**

I. Introduction

Operation lifetime is the key parameter for battery-powered wireless smart meters for heat, cooling and water. This impacts many elements in the design of a meter, in particular the transmission protocol, which for this type of device is mainly dominated by Wireless M-Bus [1] in Europe. As a receiving module is power-hungry, most meters in this class are transmitter-only devices. The rate of meter transmissions varies with the manufacturer, the product, and the actual meter use case. One typical operational mode has frequent (e.g. four packets per minute), short transmissions which enable drive-by/walk-by remote reading. An alternative system operation features a large-scale infrastructure deployment, where deployed receivers aim to read the surrounding meters and achieve certain reliability. The system can tolerate a significant amount of packet errors, and reliability can be satisfied if at least some data is received over a time frame of minutes, or even a much longer period, e.g. day or month. Wireless M-Bus meters are allowed to retransmit the same encrypted data in up to 300 seconds. This mitigates the energy-consuming encryption step, and as a consequence, it introduces a significant amount of repetitions of the same encrypted data.

In this paper we show the gain that can be obtained by recovering data from multiple erroneous packets on a stationary receiver, demonstrated experimentally in an actual deployment. The main instruments for the recovery process are: 1) the data redundancy in the transmissions, together with 2) the inherent strict transmission scheduling on the meters, which acts as side information when we associate multiple erroneous arrivals with a specific meter. A common way to combine erroneous packets is in the baseband domain, using noisy samples from multiple receivers or multiple antennas. However, in our setting each erroneous packet contains only hard bit decisions, and we combine packets across multiple arrivals over time. The idea is illustrated in Fig. 1, where to correctly identify packets from the same meter, we utilize the strict packet transmission interval function. This function allows to relate a packet arrival to past and future arrivals, and from this derive information about the underlying meter who transmitted the data. In addition to know if a packet is transmitted from a particular meter, we identify which adjacent packets belongs to the same data set. When identified, a data recovery procedure can use these multiple observations and ultimately recover the data.

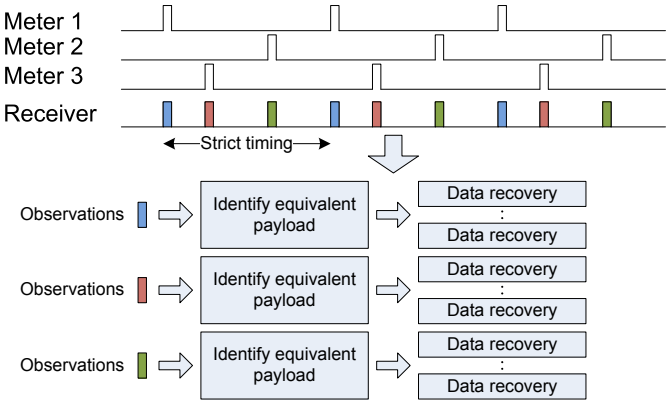


Fig. 1: Packets with predictable transmission intervals and their mapping to independent observations for packet recovery.

Field-test experiments indicate a possible recovery gain, as seen in Fig. 2, where the received signal strength is related to the number of erroneously detected packets. The figure is from one particular deployment, but others show a similar trend of a large number of erroneously received packets that are acquired at the edge of the receiver sensitivity.

II. Background and System Model

Consider packets from a Wireless M-Bus conforming meter. The specification allows for a variety of modes of operation, with a large freedom for customization from the individual vendor. The model presented here is typical, and is the operation of Wireless M-Bus meters from e.g. Kamstrup [2].

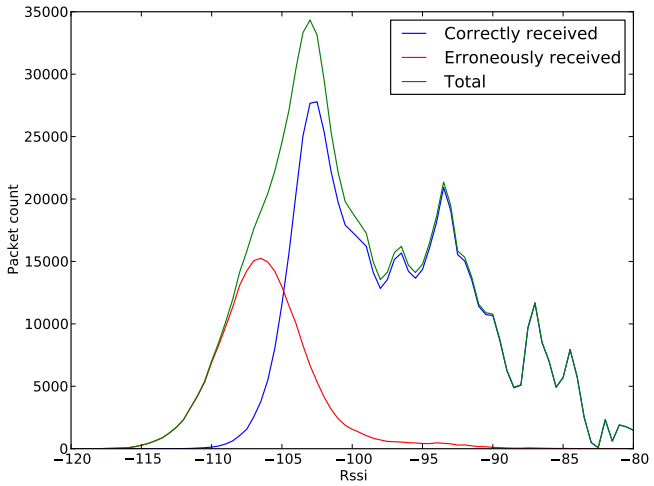


Fig. 2: Histogram of packet arrivals for various signal strengths with a configured sync word qualifier of 30/32 bits. The data is from the dataset considered throughout the paper.

Let the packet i belong to a session of eight packets. The eight packets within a session all contain the same sampled data set. There are two types of packet layout within a session: *Full Packet* is the first packet in a session, and contains all information to interpret the packet payload so that for each data record, static record header information such as the type of data, unit of data, etc. is included. *Compact Packet* is the packet layout for the following seven transmissions, which include only the data records, while the header information must be derived from a previously sent full packet to interpret the data records. The seven compact packet transmissions are almost repetitions of the same packet, except the update of Access Number (ACC) and the link-layer CRC for each transmission. The ACC is a one byte transmission counter incremented for each transmission, and we denote the ACC for the ith packet from a meter x_i, and its received version y_i which may be in error if the packet is detected to be in error. Let the indicator function $r(i,j)$ identify if the two packets i and $(i+j)$ belongs to the same repetition, so that $r(i,j)$ evaluates to 1 if packet i and $(i+j)$ are both compact frames from the same session, and 0 otherwise.

A packet i transmitted at time t_i, uniquely identifies the transmission timeslot for any past and future transmission $(i+j)$ from the same meter. The slot for the jth transmission relative to the ith packet is defined as:

$$\big[a_{x_i}(j),\ b_{x_i}(j)\big) = \big[t^{nom}_{x_i}(j) - \theta_{x_i}(j),\ t^{nom}_{x_i}(j) - \theta_{x_i}(j) + \tau_{x_i}(j)\big),$$

where $t^{nom}_{x_i}(j)$ is the nominal interval, $\theta_{x_i}(j)$ is the time the transmission is allowed to arrive earlier, and $\tau_{x_i}(j)$ is duration of the timeslot where the $(i+j)$th packet can arrive. The limits on $\theta_{x_i}(j)$ and $\tau_{x_i}(j)$ are defined by the specification and depends on the operating conditions for a given meter.

It is important to note that the slot for packet $(i+j)$ depends on the ACC x_i in the ith packet. With this relation, the timing side information element can now be exemplified as follows: If the receiver observes two erroneous packets, then by their presence at specific arrival times, this time difference inherently contains information about the transmitted ACCs in the two packets. But in contrast to the error prone packet payload, the arrival time is a very reliable measure [3]. It is important to note that the, on the receiver, expected arrival time of packet $(i+j)$ *depends on the received, and possibly erroneous ACC* y_i. This introduces an uncertainty in the establishment of the timing relation between erroneously received packets, an uncertainty which is not present in other setups where a recovery operation achieves multiple observations by means of multiple receivers or multiple antennas.

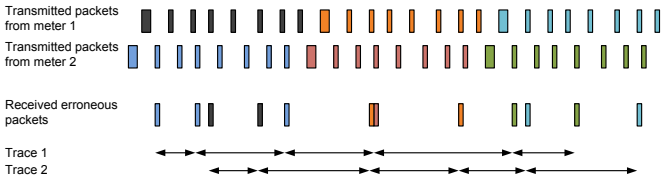


Fig. 3: Synchronous packet arrival times, pairing of packets, and constructed traces.

III. Packet Grouping

Before any packet recovery can be applied, one needs to identify which packets belongs to which session from which meter. The packets to be recovered are inherently erroneous, so any information within the packets cannot be fully trusted, ultimately affecting the way the receiver can distinguish packets. We will group packets by sessions heuristically:

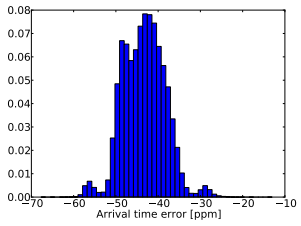
1) Find all pairs of two packets from the same meter.
2) From the pairs build a packet trace which is a complete identified sequence of packets from the same meter.
3) From the traces, identify packets belonging to the same session, which is to be recovered.

The three steps in packet grouping are exemplified in Fig. 3. The transmitted packets from a meter follows a specific timing interval. The interval is predicted by the receiver, and the erroneous packets are paired (indicated by arrows) through their arrival time such that the timing has an error correcting feature. Next, a trace is constructed by linking pairs together into a sequence of pairs. Finally, from the constructed trace, the individual sessions (indicated by color) are identified.

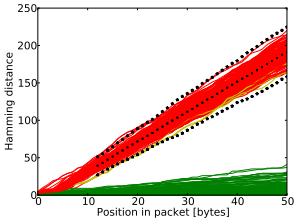
A. Pairing of Packets

It has been shown in [3] how to build pairs of two packets originating from the same meter based on the packet arrival time. The approach is similar to reconstructing location pointers to (forward and backward) messages in slot selection algorithms, as seen in e.g. RFID [4].

To create pointers based on the unreliable y_i, a series of *virtual slots* are created in which the next arrival from the same meter may arrive, one for each likely (for the receiver unknown) actual transmitted x_i. Each virtual slot ξ have a penalty associated, given by the Hamming distance $H(y_i, \xi - j)$, where ξ is the value of the virtual slot in which the $(i+j)$th packet can arrive. For example, if $j = 1$ (two adjacent packets from same meter) and $\xi = y_i + 1$ which is the expected



(a) Accumulative meter transmission jitter for the meters in the experiment.



(b) Cumulative Hamming distance between adjacent packets. The dots are mean values of the red lines, and the stars are the 3σ distance.

Fig. 4: Experiment values for calibration and grouping.

increment of the ACC, then no bit error occurred in y_i if paired with an arrival in slot ξ as $H(y_i, \xi - 1) = 0$. If $y_i = 0$ is the observed ACC, then for the virtual slot $\xi' = 2$, a single bit error must have occurred in the received ACC y_i, as $H(y_i, \xi' - 1) = 1$, etc.

Upon the arrival of a packet in a virtual slot, a decision is made on whether the packet i who initiated the setup of the slot, and the just arrived packet α should be paired (is α really the $(i+j)$th packet from the same meter?). A pairing decision is made, if the total Hamming distance satisfies:

$$H(\xi - j, y_i) + H(\xi, y_\alpha) + H(ID_i, ID_\alpha) \leq M,$$

where M is a threshold value, and ID_x is a static field in the header of packet x including e.g. meter identification. $H(\xi - j, y_i)$ is the penalty of using the virtual slot ξ setup by packet i, and $H(\xi, y_\alpha)$ is the penalty for packet α of picking the virtual slot ξ.

A premise for proper packet pairing is that the arrival time is sampled with sufficient precision. This is illustrated in Fig. 3 where to distinguish e.g. the orange from the red arrival, the receiver must be able to precisely predict the expected arrival time. The exact required precision depends on the number of meters within range of the receiver, but also on how well the transmission timing on the meters behave. We propose a simple adaptive timer calibration in order to align any timing offset incurred by the meters. The method does not depend on any factory pre-calibration or online temperature compensation. Instead, it uses the arrival time of the packets as reference. This puts a relaxing set of requirements on the receiver, while it also precisely captures any fluctuations in the setup due to varying operation conditions. The time for the virtual slot ξ is defined from a local imprecise clock as:

$$\big[a_{\xi-j}(j),\ b_{\xi-j}(j)\big) = \big[a^l_{\xi-j}(j)\nu_a,\ b^l_{\xi-j}(j)\nu_b\big),$$

where $a^l_{\xi-j}(j)$ and $b^l_{\xi-j}(j)$ are the slot boundaries according to the local receiver clock, which are compensated with $\nu_a = \mu - \sigma$ and $\nu_b = \mu + \sigma$, where μ and σ are calibration values.

From the considered experiment, the arrival time at the receiver relative to the expected arrival time according to the local (uncalibrated) timer clock is in Fig. 4a, suggesting that a compensation on the receiver should be made for around $\mu = -42$ ppm. The standard deviation of $\sigma \approx 5$ ppm provides a tight bound on the expected arrival time of a future packet, and is well within the Wireless M-Bus specification.

B. Session Grouping

The last step before actual data recovery is to identify sessions with the same packet data within each trace. A session number s is included in each packet to denote the session the packet belongs to, but this field is unreliable due to possible packet errors. As an alternative we will use the encrypted payload itself as an indicator to relate two packets to the same session. In general, it can be assumed that any two cleartext payloads reveal encrypted equivalents with a large variation when the encryption initialization vector is changed[1]. That is:

$$H(i, j) \ll H(i, j'),$$

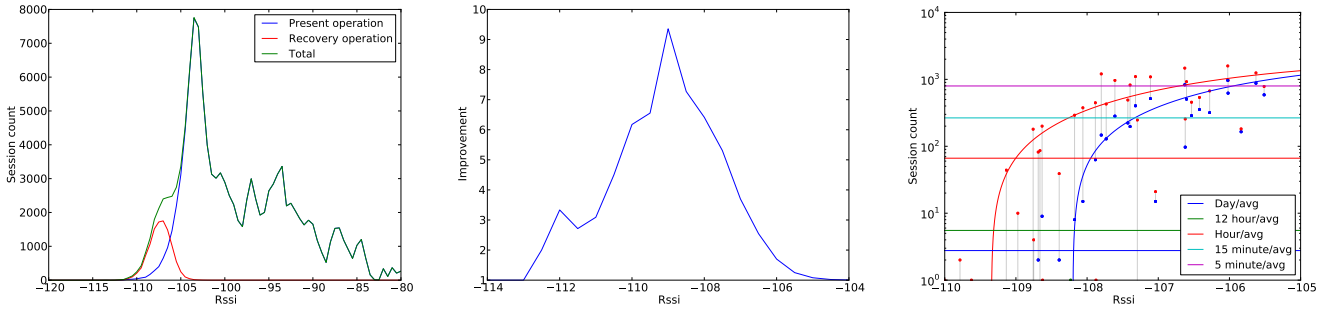
where $H(i, j)$ is the Hamming distance between packet i and j, when $r(i, j) = 1$ and $r(i, j') = 0$. This is illustrated in Fig. 4b showing the distance between adjacent packets at a given position within the packet. The positions does not include the full packet, but only the four bytes of the session number and the encrypted payload. In the beginning where the session number is located, the distance is expectedly low, as only a few bits are updated when the session number changes. At the positions where the encrypted payload is located, the distance diverges rapidly with an average distance of 4 bits/byte, and it is easy to identify packets from the same session. From the example in Fig. 3, if two same-color packets are compared, this will belong to the green set of lines, while if packets from two different sessions are compared, the result belongs to the red set of lines.

IV. Recovery Procedure

The recovery step is the data processing part working with the hard converted bits from the erroneously received packets grouped by session. The procedure is a mix of bitwise majority voting, and use of the error correcting capabilities of a CRC. This type of correcting measure is well studied. The majority voting mechanism has been used in [5] which also includes a simple example, and the error correcting capabilities of CRC is detailed in [6] with several hardware optimizations, e.g. [7].

Data integrity wise, we are not restricted to only test a limited set of bit combinations during recovery. If only CRC is used to verify the recovered packet, the search space is mainly limited by the used CRC polynomial. The minimum bit error generating polynomial for the generator polynomial used in Wireless M-Bus is $z^i(z^{151}+1)$, hence the minimum Hamming distance is 2 for messages over 151 bits. This suggest that a different verification strategy should be used together with plain link layer CRC. It can be shown that by transforming the reconstructed compact packet into its full packet equivalent (by inserting the static data record headers) which is already done at the receiver to interpret the data in the compact packet, this operation guarantees a final minimum Hamming distance of 4[2]. This distance is considered to be sufficient, and is

[1]The session number is part of the 128 bit AES-CTR initialization vector.
[2]Except for a limited set of compact packet lengths.



(a) Unique sessions. (b) Improvement. (c) Sessions per meter.

Fig. 5: Field test results.

equivalent to what the receiver sees without recovery. Further data integrity should be validated on higher layers with other message authentication such as CMAC.

V. Field Test Results

We will through an experiment on a dedicated receiver, validate the data-bit-recovery performance working on multiple erroneous packets. The receiver is located in a city with a scattered meter deployment, and does only cover a fraction of the full setup. The experiment is run over 66 hours and packets are observed from 84 different meters (without recovery). When comparing present operation with recovery operation, the quality is measured as follows: If *any* compact packet successfully arrives without recovery at the receiver, we count one full session as known. If *no* successful compact packet arrives for a session, the recovery operation is conducted, and if successfully validated, it counts as one recovered session.

Fig. 5a shows the number of received/recovered sessions for a given average RSSI, considered cumulatively across all the meters, possibly with more than one counted session per meter. The figure shows a clear improvement just below the original receiver sensitivity, in the region that features the majority of erroneously received packets from Fig. 2. Fig. 5b shows the improvement due to recovery: at -109 dBm the receiver finds nine times more sessions when applying recovery.

Because of the relaxed reliability requirements, we hope for a few sessions from many meters. To see if this applies, consider Fig. 5c, where the number of unique sessions per meter is shown, where a meter is identified from the average RSSI over the experiment. A blue point is the number of received sessions from a single meter during the experiment without recovery. When a blue point is in a pair with a red point, the difference between the two points indicates the improvement due to recovery. If a red point is isolated, then it corresponds to a meter being (fully) recovered with the given number of sessions in a way where no single successful packet was received through the experiment. The horizontal lines are average lines: if. e.g. a point is located on the hour/avg line, the number of sessions received is on average one per hour, averaged over the experiment time. The experimental results from Fig. 5c suggest a gain of around 1 dB, but for the given deployment where a large number devices are located at the edge of receiver sensitivity, this 1 dB means a significant increase of 12 meters which are now also seen by the receiver.

To further increase the gain it is relevant to consider the limiting element of the operation. One element is the requirement that at least two packets must be detected within the sequence of seven repeated compact frames before the recovery operation starts. To make this event more likely, we believe that the main limitation is the sync word qualifier in the receiver. In the experiment it is $30/32$ bits, hence a sync word is detected if 30 out of 32 bits are as expected. Lowering this requirement will clearly increase the number of false positive sync words, but will also relax the criteria for receiving an erroneous packet. Because of hardware limitations, it has not been possible to loosen the sync word qualifier.

VI. Conclusion

We show how the packet arrival time can be used as a side information for error correction at the receiver. Using this as basis, packets with equivalent payload can be grouped, and used as input for a simple recovery procedure. The results from an actual deployment show a significant increase of received data in the region around the receiver sensitivity, and when the data are grouped by the transmitting meter, an increase in the number of discovered meters suggests a receiver sensitivity gain of around 1dB compared to operation without recovery.

References

[1] European Standard, "Communication systems for meters and remote reading of meters - part 4: Wireless meter readout (radio meter reading for operation in the 868 mhz to 870 mhz srd band)," *EN13757-4:2005*.

[2] Kamstrup, "Multical 21," http://kamstrup.com/media/22027/file.pdf.

[3] R. M. Jacobsen and P. Popovski, "A framework for reliable reception of wireless metering data using protocol side information," in *Global Telecommunications Conference, 2013. GLOBECOM '13. IEEE*, 2013, to appear.

[4] F. Ricciato and P. Castiglione, "Pseudo-random aloha for enhanced collision-recovery in rfid," *Communications Letters, IEEE*, vol. 17, no. 3, pp. 608–611, 2013.

[5] M. Schmidt and K. Schilling, "Ground station majority voting for communication improvement in ground station networks," in *SpaceOps 2010 conference*, 2010.

[6] S. Lin and D. J. Costello, *Error Control Coding, Second Edition*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2004.

[7] Y. Pan, N. Ge, and Z. Dong, "Crc look-up table optimization for single-bit error correction," *Tsinghua Science and Technology*, vol. 12, no. 5, pp. 620–623, 2007.