

# Application-Layer Coding with Intermittent Feedback Under Delay and Duty-Cycle Constraints

Siddhartha S. Borkotoky<sup>\*†</sup>, Udo Schilcher<sup>\*‡</sup>, and Christian Raffelsberger<sup>\*</sup>

<sup>\*</sup>Lakeside Labs GmbH, 9020 Klagenfurt, Austria

<sup>†</sup>Indian Institute of Technology Bhubaneswar, Khordha 752050, India

<sup>‡</sup>University of Klagenfurt, 9020 Klagenfurt, Austria  
(borkotoky@iitbbs.ac.in)

**Abstract**—We propose two application-layer coding schemes for delay-constrained point-to-point packet communications with restrictions on the transmitter’s maximum duty-cycle. The schemes operate over GF(2) and utilize intermittently available receiver feedback for erasure correction. Applications that will benefit from the proposed schemes include wireless sensor networks in which energy-constrained sensors must deliver readings to a gateway within a deadline. Simulation results for independent Bernoulli erasure channels, Gilbert-Elliott channels, and Long Range (LoRa) communications demonstrate orders-of-magnitude reductions in the delivery failure rate as compared to feedback-assisted repetition redundancy and a blind coding scheme that does not utilize feedback.

**Index Terms**—Application-layer coding, delay-sensitivity, duty cycling, LoRa, wireless sensor networks.

## I. INTRODUCTION

Certain wireless networks are subject to limits on the transmitters’ maximum duty cycle. Such limits could be due to regulatory restrictions in certain frequency bands or could be enforced to restrict the transmission times of energy-constrained devices. Duty-cycle constraints limit the use of receiver feedback in multipoint-to-point communications as the receiver cannot acknowledge all receptions without violating the constraint when the number of transmitters is large. They also restrict the maximum number of retransmissions a sender can make and the amount of redundancy it can transmit for forward error correction.

The focus of this paper is on duty-cycle-constrained point-to-point transfer of delay-sensitive information. In the scenarios of interest, an information symbol (a collection of information bits) remains of interest to the receiver only for a limited amount of time after it is generated. Our work is motivated by use cases from a wireless sensor network in which sensors periodically transmit readings to a gateway using the LoRa technology [1] over the 868 MHz unlicensed frequency band, for which the maximum allowed duty cycle is 1% in the European Union [2]. Due to this restriction, the gateway acknowledges only some of the transmissions. The sensors, likewise, limit

their packet sizes to conserve energy and meet duty-cycle regulations. Each measurement must reach the gateway within a specified deadline. To improve communications reliability under such circumstances, we propose two application-layer coding schemes for erasure correction.

Although motivated by LoRa use cases, our proposed schemes can be implemented with any physical-layer technology. In our schemes, a transmitted packet contains the current information symbol, and possibly some past information symbols as well as some coded symbols (random linear combinations of information symbols). The schemes employ cumulative feedback that conveys the identity of the oldest unexpired-but-undelivered information symbol. Instead of blindly inserting coded symbols into every packet, current and past feedback is used to choose between transmitting coded symbols and retransmitting past information symbols. When coding occurs, the feedback is used to determine the number of information symbols to combine to produce a coded symbol as well as the set of information symbols from which the combined symbols are randomly chosen. The computational complexity is kept low by coding only at certain instances and restricting all coding operations to GF(2). In our design, we account for the possibility that the sender may not obtain receiver feedback after every transmission.

## II. RELATED WORK

Well-known application-layer codes such as LT codes [3] and Raptor codes [4] operate over large blocks of information symbols. They have strong erasure-correction capabilities, but large decoding delays render them unsuitable for delay-constrained applications.

Coding over small blocks of information is investigated in [5] under the assumption of perfect feedback. In [6], a coding scheme over a large finite field (GF(256)) for delay-sensitive multimedia applications is proposed. In contrast, we seek computationally simpler strategies that operate in GF(2). Unlike in [5] and [6], where each coded symbol is sent in a separate packet, we consider the transmission of coded symbols and information symbols in the same packet. By avoiding the extra packet overhead in this manner, lower energy consumption and smaller duty cycles are achieved. For example, transmitting a 3-byte sensor measurement followed by a 3-byte coded measurement in two separate packets using LoRa results in 67% higher duty-cycle and energy expenditure, compared to sending the measurement and the coded measurement in the same

This work has been supported by the K-project DeSSnet (Dependable, secure and time-aware sensor networks), which is funded within the context of COMET – Competence Centers for Excellent Technologies by the Austrian Ministry for Transport, Innovation and Technology (BMVIT), the Federal Ministry for Digital and Economic Affairs (BMDW), and the federal states of Styria and Carinthia; the COMET program is conducted by the Austrian Research Promotion Agency (FFG).

ACK / NACK	Seq. no. of oldest unexpired missing symbol ( $s_u$ )	No. of unexpired missing symbols ( $\beta$ )
---------------	--	---

Fig. 1: Feedback structure.

Current information symbol $s_i$	Information symbol $s_u$	Coded symbol or past information symbol
-------------------------------------	--------------------------	--

(a) If feedback was received for  $\mathbf{p}_{i-1}$ .

Current information symbol $s_j$	Coded symbol or past information symbol	Coded symbol or past information symbol
-------------------------------------	--	--

(b) If feedback was not received for  $\mathbf{p}_{i-1}$ .

Fig. 2: Payload structure for packet  $\mathbf{p}_i$  ( $b = 3$ ).

packet [7]. Additionally, in some situations, coded symbols can be transmitted without increasing the packet duration. For example, the duration of a LoRa packet remains constant for payload sizes of 1 byte through 4 bytes [7]. Therefore, a 1-byte measurement can be accompanied by up to three coded measurements of 1 byte each at no extra cost in terms of duty-cycling and energy expenditure.

The approach closest to ours is that of [8], where each packet contains the current information symbol and some coded symbols that are bitwise XORs of randomly selected past information symbols. However, the scheme assumes complete absence of feedback, and hence cannot benefit from any feedback availability. Furthermore, in contrast to the insertion of coded symbols into each packet in [8], our methods perform coding only when deemed necessary.

### III. SYSTEM MODEL

An *information symbol* is a collection of information bits (e.g., the binary representation of a sensor measurement). Every information symbol triggers a packet transmission. Let  $\delta_{\max}$  denote the application's *delay tolerance*, the time after which an information symbol expires (is no longer of interest to the receiver).

A *coded symbol* is a linear combination (bitwise XOR) of information symbols selected at random from a set of information symbols referred to as the *coding set*. The *degree* of the coded symbol is the number of information symbols combined and can vary from symbol to symbol. A coded symbol of degree  $d$  results in the recovery of an information symbol at the receiver if  $d - 1$  out of the  $d$  XORed information symbols have already been delivered. An information symbol is *delivered* if the receiver either receives a packet containing the uncoded symbol or is able to recover it from a coded symbol.

An information symbol and a coded symbol are both  $l$ -bits long. Let  $l_p$  be the maximum number of bits in a packet's payload, as determined by duty-cycle and energy constraints. Thus,  $b = l_p/l$  is the maximum number of symbols that can be included in the packet.

Following a transmission, the sender receives a feedback packet with a feedback reception probability (FRP)  $p_{\text{feedback}}$ . As shown in Fig. 1, the feedback packet contains a one-bit ACK or NACK that indicates whether the transmission was received correctly, the sequence number  $u$  of the oldest undelivered and unexpired information symbol,

TABLE I: Summary of key notation

Notation	Description
$s_i$	Information symbol no. $i$
$\mathbf{p}_i$	Packet no. $i$
$b$	Maximum number of symbols a packet can carry
$\mathbf{q}_i[k]$	$k$ -th symbol in the payload of $\mathbf{p}_i$ ( $1 \leq k \leq b$ )
$u$	Sequence number of oldest unexpired and undelivered information symbol ( $s_u$ )
$\beta$	Number of unexpired information symbols not yet delivered
$\mathbf{c}_d(\mathbf{v})$	A coded symbol produced by XORing $d$ randomly chosen elements from vector $\mathbf{v}$ , where $d$ is determined using (1)
$\mathbf{c}_D(\mathbf{v})$	A coded symbol produced by XORing $D$ randomly chosen entries of vector $\mathbf{v}$ , where $D$ is a uniform random variable
$\mathbf{v}[k]$	$k$ -th entry of vector $\mathbf{v}$
$\mathbf{v} \setminus \{x\}$	Set containing all entries of $\mathbf{v}$ except $x$

and the total number  $\beta$  of undelivered and unexpired information symbols. If all unexpired symbols up to and including  $s_j$  in the just-transmitted packet  $\mathbf{p}_j$  have been delivered, then  $u = j + 1$  in the feedback for  $\mathbf{p}_j$ .

The payload structure of the  $i$ -th packet  $\mathbf{p}_i$  is illustrated in Fig. 2 for  $b = 3$ . Packet  $\mathbf{p}_i$  includes the information symbol  $s_i$ . If an acknowledgement was received for  $\mathbf{p}_{i-1}$  and  $u \neq i$ , then  $\mathbf{p}_i$  includes  $s_u$  as well. The space for the remaining  $b - 2$  symbols carry either past information symbols or coded symbols, as determined by the coding algorithms.

A summary of the key notation used in the paper is given in Table I.

### IV. CODING SCHEMES

The two proposed coding schemes differ in their respective coding sets and in the determination of the degrees of the coded packets. In *windowed coding*, the coding set is a continuous window of information symbols starting from the oldest unexpired and undelivered symbol until the most recently transmitted one. In *selective coding*, some symbols from the aforementioned window are excluded based on receiver feedback. Descriptions and pseudocodes for the algorithms are given below.

#### A. Windowed Coding

The payload of packet  $\mathbf{p}_i$  depends on whether a feedback was received for the previous packet  $\mathbf{p}_{i-1}$ :

1) *Feedback received for  $\mathbf{p}_{i-1}$* : From the feedback, the sender knows that the set of sent but undelivered (and unexpired) information symbols is a subset of the set of entries in the vector  $\mathbf{w}_i = \{s_u, s_{u+1}, \dots, s_{i-1}\}$ . Note that  $\mathbf{w}_i$  has  $i - u$  entries, and if  $\beta = i - u$ , no symbol in  $\mathbf{w}_i$  has been delivered. There are the following possibilities:

(a)  $u = i$ : All information symbols up to this point have been delivered. Hence,  $\mathbf{p}_i$  contains only  $s_i$  in its payload.

(b)  $u < i, i - u \leq b - 1, \beta \geq 1$ : All entries of  $\mathbf{w}_i$ , which form a superset of all undelivered information symbols, can be included in the packet. Hence,  $\mathbf{p}_i$  contains  $s_i$  followed by the elements of  $\mathbf{w}$ .

(c)  $u < i, i - u > b - 1, 1 < \beta = i - u$ : None of the information symbols in  $\mathbf{w}_i$  have been delivered, but there is not enough space in  $\mathbf{p}_i$  to include all of them. Hence,  $\mathbf{p}_i$  contains  $s_i$  and the first  $b - 1$  elements of  $\mathbf{w}_i$  (i.e.,  $\{s_j : u \leq j \leq u + b - 2\}$ ).

(d)  $u < i, i - u > b - 1, 1 < \beta < i - u$ : Here,  $\beta$  out of the  $i - u$  entries of  $\mathbf{w}_i$  are undelivered; but the sender does not know which ones (except for  $s_u$ ), nor can it include the entire vector  $\mathbf{w}_i$  in  $\mathbf{p}_i$ . We leverage coding in this situation. Packet  $\mathbf{p}_i$  contains  $s_i, s_u$ , and  $b - 2$  coded symbols produced by random linear combinations of elements from  $\mathbf{a}_i = \mathbf{w}_i \setminus \{s_u\}$ . The degree of each coded symbol is  $d(i - u - 1, \beta - 1)$ , where

$$d(x, y) = \arg\max_{d'} \frac{y \binom{x-y}{d'-1}}{\binom{x}{d'}} \text{ for } x > y > 0. \quad (1)$$

The degree is chosen to maximize the probability that exactly one of the XORed information symbols belongs to the set of  $\beta - 1$  undelivered symbols, thus maximizing the probability that the coded symbol results in the recovery of one information symbol. Note that there are  $\beta$  undelivered information symbols, but the symbol  $s_u$  included in  $\mathbf{p}_i$  reduces that number to  $\beta - 1$ .

The maximization according to (1) is only an approximation if there are multiple coded symbols in the packet. A joint maximization of the individual degrees is possible, but is not employed to keep the complexity low.

2) *Feedback not received for  $\mathbf{p}_{i-1}$* : Define  $\mathbf{b}_i = \{s_{i-1}, s_{i-2}, \dots, s_{i-z}\}$ , where  $z = \min\{i - u_l, i - u_{\max}\}$ ,  $u_l$  is the sequence number included in the most recent feedback, and  $u_{\max}$  is the oldest unexpired information symbol at time  $i$ . There are two possibilities:

(a)  $z \leq b - 1$ : The entire vector  $\mathbf{b}_i$ , whose entries form a superset of all undelivered information symbols, can be included in the packet. Hence,  $\mathbf{p}_i$  contains  $s_i$  followed by the entries of  $\mathbf{b}_i$ .

(b)  $z > b - 1$ : This is similar to situation (d) above, but since the current number of undelivered symbols is unknown, degree selection via (1) is infeasible. Packet  $\mathbf{p}_i$  contains  $s_i$  followed by  $b - 1$  coded symbols. A coded symbol is the XOR of  $D$  information symbols chosen at random from  $\mathbf{b}_i$ , where  $D$  is a uniform random variable in the range  $[1, z]$ .

## B. Selective Coding

The sender maintains a list of the unexpired information symbols for which an ACK has not been received and whose sequence numbers are equal to or greater than  $u_l$ , the sequence number included in the most recent feedback. Let  $\mathbf{m}_i$  be the vector of such symbols at time  $i$ , and let  $n$  denote the number of entries in  $\mathbf{m}_i$ . The payload of  $\mathbf{p}_i$  is constructed as follows:

1) *Feedback received for  $\mathbf{p}_{i-1}$* : There are the following possibilities:

---

### Algorithm 1: Windowed coding

---

```

1  $\mathbf{q}_i[1] = s_i$ 
2 if Feedback received for  $\mathbf{p}_{i-1}$  then
    if  $u < i$  then
         $\mathbf{q}_i[2] = \mathbf{w}_i[1]$ 
        if  $\beta > 1$  then
            if  $i - u \leq b - 1$  then
                 $\mathbf{q}_i[k] = \mathbf{w}_i[k - 1], 3 \leq k \leq i - u + 1$ 
            else if  $i - u > b - 1$  and  $\beta = i - u$  then
                 $\mathbf{q}_i[k] = \mathbf{w}_i[k - 1], 3 \leq k \leq b$ 
            else if  $i - u > b - 1, 1 < \beta < i - u$  then
                 $\mathbf{q}_i[k] = \mathbf{c}_d(\mathbf{a}_i), 3 \leq k \leq b$ 
    else
        if  $z \leq b - 1$  then
             $\mathbf{q}_i[k] = \mathbf{b}_i[k], 2 \leq k \leq z + 1$ 
        else
             $\mathbf{q}_i[k] = \mathbf{c}_D(\mathbf{b}_i), 2 \leq k \leq b$ 

```

---



---

### Algorithm 2: Selective coding

---

```

1  $\mathbf{q}_i[1] = s_i$ 
2 if Feedback received for  $\mathbf{p}_{i-1}$  then
    if  $\beta \geq 1$  then
         $\mathbf{q}_i[2] = s_u$ 
        if  $\beta > 1$  then
            if  $n = \beta$  then
                 $\mathbf{q}_i[k] = \mathbf{n}_i[k - 2], 3 \leq k \leq \max\{n + 1, b\}$ 
            else if  $b - 1 \geq n > \beta$  then
                 $\mathbf{q}_i[k] = \mathbf{n}_i[k - 2], 3 \leq k \leq n + 1$ 
            else if  $n > \beta$  and  $n > b - 1$  then
                 $\mathbf{q}_i[k] = \mathbf{c}_d(\mathbf{n}_i), 3 \leq k \leq b$ 
    else
        if  $n < b - 1$  then
             $\mathbf{q}_i[k] = \mathbf{m}_i[k], 2 \leq k \leq n + 1$ 
        else
             $\mathbf{q}_i[k] = \mathbf{c}_D(\mathbf{m}_i), 2 \leq k \leq b$ 

```

---

(a)  $\beta = 0$ : The receiver has received all information symbols up to this point. Hence,  $\mathbf{p}_i$  contains only  $s_i$ .

(b)  $\beta = 1$ : There is only one undelivered symbol at the receiver, which is  $s_u$ . Hence,  $\mathbf{p}_i$  contains  $s_i$  and  $s_u$ .

(c)  $\beta > 1, n = \beta$ : The sender knows which symbols are undelivered. Here,  $\mathbf{p}_i$  contains  $s_i, s_u$ , and  $\max\{n - 1, b - 2\}$  information symbols from the vector  $\mathbf{n}_i = \mathbf{m}_i \setminus \{s_u\}$ .

(d)  $\beta > 1, b - 1 \geq n > \beta$ : The sender does not know which symbols are undelivered, but all potentially unde-

livered symbols can be included in  $\mathbf{p}_i$ . Here,  $\mathbf{p}_i$  contains  $s_i$  followed by the elements of  $\mathbf{m}_i$ .

(e)  $\beta > 1$ ,  $n > \beta$ ,  $n > b - 1$ : The sender knows a set of  $n$  symbols, out of which  $\beta$  are undelivered (although the sender does not know which ones), and not all  $n$  symbols fit in the packet. In this case,  $\mathbf{p}_i$  contains  $s_i$ ,  $s_u$ , and  $b - 2$  coded symbols produced by XORing  $d$  randomly chosen information symbols out of the  $n - 1$  symbols in  $\mathbf{n}_i = \mathbf{m}_i \setminus \{s_u\}$ . The degree  $d$  is determined according to (1) with  $x = n - 1$  and  $y = \beta - 1$ .

2) *Feedback not received for  $\mathbf{p}_{i-1}$* : There are two possibilities:

(a)  $n \leq b - 1$ : All potentially undelivered information symbols can be included in a packet. Hence,  $\mathbf{p}_i$  contains  $s_i$  followed by the information symbols in  $\mathbf{m}_i$ .

(b)  $n > b - 1$ : Similar to situation (e), but the current number of undelivered symbols is unknown. Packet  $\mathbf{p}_i$  contains  $s_i$  followed by  $b - 1$  coded symbols, each produced from the coding set comprising the elements of  $\mathbf{m}_i$  and with a degree chosen uniformly at random from  $[1, n]$ .

For both coding schemes, the maximization in (1) can be performed using a look-up table containing the values of  $d(x, y)$  for all valid  $(x, y)$  pairs. Since the applications are delay constrained, the number of pairs is not large. If there can be a maximum of  $q$  unexpired packets at any given instant, the look-up table has  $q(q + 1)/2$  entries. For our numerical results,  $q \leq 16$ , resulting in at most 136 table entries.

## V. PERFORMANCE EVALUATION

### A. Simulation Setup

We are interested in the fraction of information symbols that are not delivered before their expiration. This performance measure is called the *delivery failure rate* (DFR). We simulate periodic packet transmissions. Unless stated otherwise, an information symbol expires  $\delta_{\max} = 16$  packet intervals after it is first transmitted. For each data point in the numerical results, enough transmissions are simulated to obtain at least 100 delivery failures. Results are provided for three types of channels:

a) *Independent Bernoulli erasure channel*.

b) *Gilbert-Elliott channel*: Using a two-state Markov chain, packets transmitted during the “good” state are delivered and packets in the “bad” state are lost. The state is fixed during a packet transmission but may change from packet to packet. The transition probability from “good” to “bad” (“bad” to “good”) is denoted by  $p_{gb}$  ( $p_{bg}$ ).

c) *LoRa packet-erasure model*: The tool LoRaSim [9] is used to simulate LoRa communications in an industrial environment. The spreading factor [7] is 10 and each packet has a duration of 13 LoRa symbols, which corresponds to a payload of 4 bytes or lower [7]. Packet losses occur due to independent fading (Nakagami- $m$  with  $m = 2.5$ ) and interference from other LoRa transmitters. The nodes are simulated as points on a two-dimensional plane. The receiver is at the origin; the sender is at  $x = y = 36$  m. There are up to 200 other transmitters in the vicinity; the  $x$  and  $y$ -coordinates of each such transmitter are uniform random variables in the range [30 m, 42 m]. The pathloss exponent is 4, which results in short ranges.

For each setup, the arrival of feedback is an independent Bernoulli random process. Following a transmission, the sender receives feedback with probability  $p_{\text{feedback}}$ . The non-arrival of feedback may be due to disturbances in the feedback channel, or due to the inability of the receiver to send a feedback caused by duty-cycle restrictions. We do not attempt to derive a stochastic characterization of the feedback reception process but employ the simple Bernoulli model instead.

### B. Benchmark Schemes

Two benchmark schemes are simulated:

1) A *repetition-redundancy scheme* that utilizes feedback but does not perform XORing of symbols. Packet  $\mathbf{p}_i$  contains the current information symbol  $s_i$  and, if a feedback was received for  $\mathbf{p}_{i-1}$ , the symbol  $s_u$  whose sequence number was included in the feedback. The remainder of the payload carries the most recent information symbols that have not been acknowledged yet.

2) A *blind coding scheme* (similar to that of [8]) that does not utilize feedback; each packet contains the current information symbol followed by  $b - 1$  coded symbols formed by XORing randomly chosen unexpired information symbols. The degree of a coded symbol is  $\delta_{\max}/2$ . As shown in [8], the degree requires careful adjustment for optimal performance, but such adjustments are infeasible unless the channel model is known in advance.

### C. Simulation Results

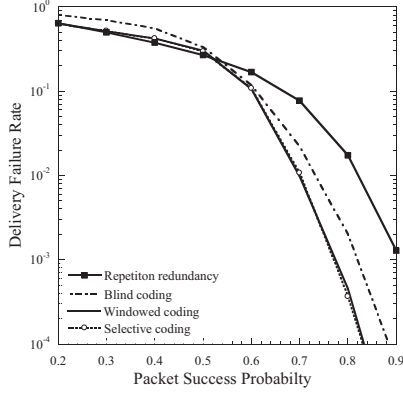
Fig. 3 shows the DFR over an independent Bernoulli channel for two sets of values of  $b$  and  $p_{\text{feedback}}$ . Fig. 3a shows that the proposed algorithms outperform repetition redundancy for packet success probabilities exceeding 0.5, providing up to two orders-of-magnitude improvement. Up to an order-of-magnitude improvement over the blind coding scheme is also seen. The two proposed schemes provide similar performance. In Fig. 3b, the DFR of each of the four schemes is lower than in Fig. 3a. This is due to the additional redundancy per packet and, except for the blind coding scheme, more frequent availability of feedback. In this scenario, selective coding outperforms windowed coding. Also, unlike Fig. 3a, the blind coding scheme performs much worse than the other three schemes due to its inability to exploit the greater availability of feedback.

Fig. 4 plots the DFR against FRP. The link from sender to receiver is an independent Bernoulli erasure channel with success probability 0.6. The proposed algorithms outperform repetition redundancy for any value of the FRP. For low FRPs, the two proposed algorithms give approximately the same performance. However, selective coding outperforms windowed coding when the FRP is high and, unlike windowed coding and repetition redundancy, its performance does not saturate at high FRP.

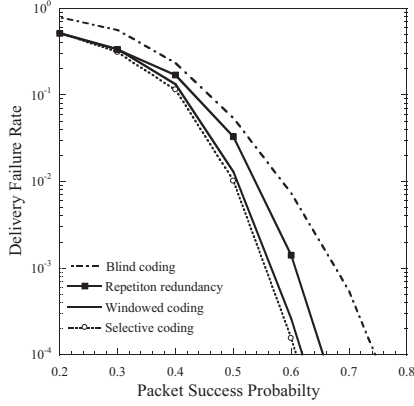
Results for the Gilbert-Elliott channel are shown in Fig. 5. The transition probability  $p_{gb}$  is fixed at 0.2 while  $p_{bg}$  is varied. The graphs exhibit similar patterns to those for the independent Bernoulli erasure channel, but the performance improvement due to the proposed algorithms is higher than for the independent Bernoulli channels.

Fig. 6 shows the delivery performance against the delay tolerance. Recall that an information symbol expires  $\delta_{\max}$



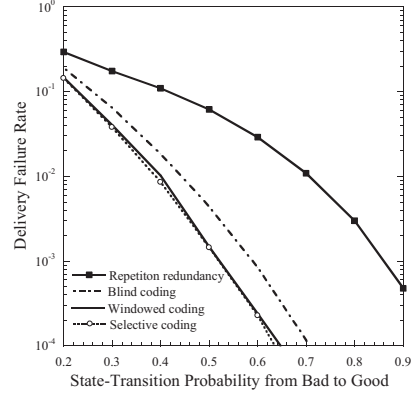


(a)  $b = 2$ ,  $p_{\text{feedback}} = 0.25$ .

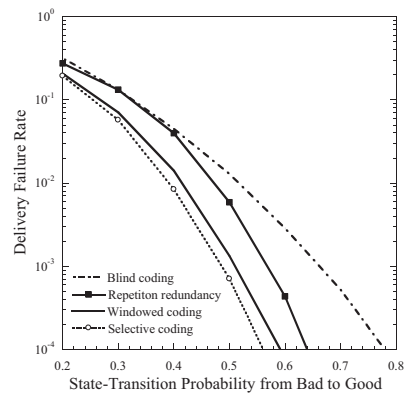


(b)  $b = 3$ ,  $p_{\text{feedback}} = 0.75$ .

Fig. 3: Delivery performance on an independent Bernoulli erasure channel.



(a)  $b = 2$ ,  $p_{\text{feedback}} = 0.3$ .



(b)  $b = 3$ ,  $p_{\text{feedback}} = 0.9$ .

Fig. 5: Delivery performance on a Gilbert-Elliott channel.

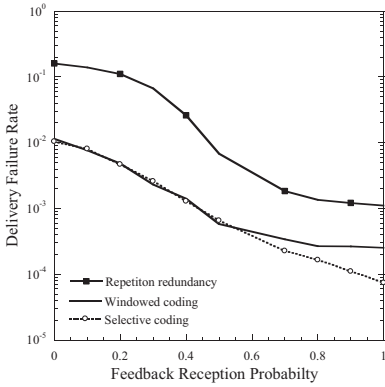


Fig. 4: Delivery performance as a function of FRP.

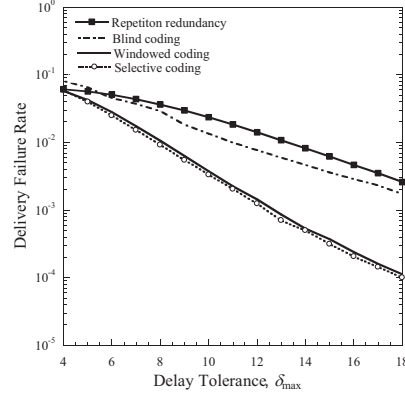


Fig. 6: Delivery performance versus delay tolerance.

packet intervals after its first transmission. The results are shown for a Gilbert-Elliott channel with  $p_{\text{gb}} = 0.3$ ,  $p_{\text{bg}} = 0.6$ , and  $p_{\text{feedback}} = 0.5$ . The number of symbols per packet is  $b = 3$ . We observe that higher delay tolerance results in better performance for each scheme, which is due to more opportunities to correct an erasure. Higher delay tolerance also leads to a greater performance advantage of the proposed algorithms over the benchmark schemes.

To assess the computational overhead due to coding, we plot in Fig. 7 the average number of symbols XORed per transmitted packet for the Gilbert-Elliott channel with

$p_{\text{gb}} = 0.2$  and for two different feedback reception probabilities. We observe that higher feedback availability reduces the number of symbol XORs for the proposed schemes by providing the sender with more information about the set of potentially undelivered packets. Similarly, the number decreases with better channel conditions owing to fewer undelivered information symbols. For each value of  $p_{\text{feedback}}$ , selective coding incurs lower complexity than windowed coding by further winnowing the coding set. Due to its use of a fixed degree, the number of XORs in the blind coding scheme is constant.

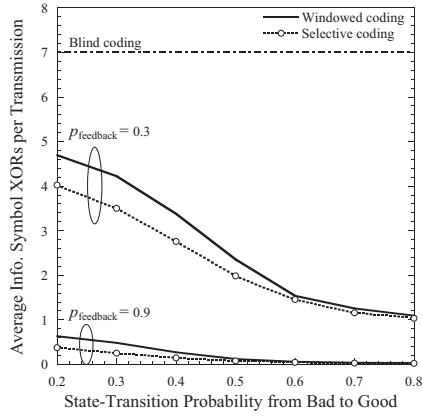


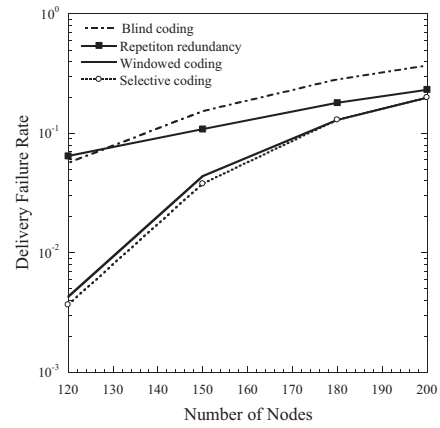
Fig. 7: Average number of information symbols XORed per transmission.

Fig. 8 shows the delivery performance for LoRa communications between a sensor and a gateway. Each sensor measurement is 1 byte long. For the LoRa parameters employed, the packet duration is constant for payload sizes of 1–4 bytes [7]. Therefore, up to 3 symbols in addition to the current information symbol (i.e., up to  $b = 4$ ) can be included without increasing the duty cycle and transmission energy. The FRP is 0.5. The delivery performance is plotted against the number of nearby nodes. The DFR increases with the number of nodes due to higher interference. As before, the proposed coding schemes outperform the benchmark schemes. For  $b = 4$ , we observe that the performance advantage of selective coding over windowed coding is higher than in the other scenarios considered so far.

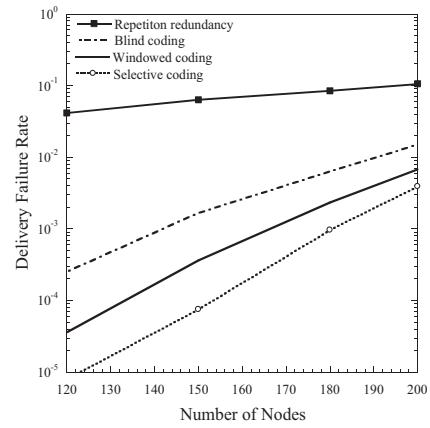
## VI. CONCLUSION

We proposed two application-layer coding schemes – windowed coding and selective coding – for delay-constrained communications with limited receiver feedback. The schemes are effective tools to improve communications reliability in various delay-sensitive and duty-cycle-constrained settings, such as those encountered in wireless sensor networks. As with any coding technique, there is some increase in computations relative to uncoded redundancy transmissions; however, the added complexity is much lower than for blind coding, and decreases with increasing packet success probability and feedback reception probability. Selective coding provides superior delivery performance and lower computational complexity than windowed coding at the cost of some increase in storage requirements (due to the need to maintain a list of unacknowledged symbols). The designer can choose one of the schemes depending on target delivery performance and the transmitting node’s computational capability and storage capacity.

Future work includes the following: (a) A computationally simple method for joint optimization of the degrees of the coded symbols when there are multiple coded symbols per packet. (b) Adaptation of the proposed schemes to the



(a)  $b = 2$ .



(b)  $b = 4$ .

Fig. 8: Delivery performance for LoRa.

point-to-multipoint scenario (e.g., a sensor sending data to multiple gateways).

## REFERENCES

- [1] U. Raza, P. Kulkarni, and M. Sooriyabandara, “Low power wide area networks: An overview,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 855–873, 2017.
- [2] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, “Understanding the limits of LoRaWAN,” *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 34–40, 2017.
- [3] M. Luby, “LT codes,” in *Proc. IEEE Symp. Found. Computer Science*, Vancouver, Canada, Nov. 2002, pp. 271–280.
- [4] A. Shokrollahi, “Raptor codes,” *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.
- [5] E. Drinea, L. Keller, and C. Fragouli, “Real-time delay with network coding and feedback,” *Physical Communication*, vol. 6, pp. 100–113, 2013.
- [6] S. Fong, S. Emara, B. Li, A. Khisti, W. Tan, X. Zhu, and J. Apostolopoulos, “Low-latency network-adaptive error control for interactive streaming,” in *27th ACM International Conference on Multimedia* (available: [arxiv.org/abs/1909.06709](https://arxiv.org/abs/1909.06709)), 2019.
- [7] SX1272/3/6/7/8: *LoRa Modem Designer’s Guide AN1200.13*. Semtech Corporation, 2013.
- [8] P. J. Marcelis, V. S. Rao, and R. V. Prasad, “DaRe: Data recovery through application layer coding for LoRaWAN,” in *Proc. IEEE/ACM Intern. Conf. on Internet-of-Things Design and Implementation (IoTDI)*, Pittsburgh, USA, Apr. 2017, pp. 97–108.
- [9] “LoRaSim,” [www.lancaster.ac.uk/scc/sites/loralorasil.html](http://www.lancaster.ac.uk/scc/sites/loralorasil.html), accessed February 2019.