Data link layer

Important task of link layer is the formation & maintenance of communication links betn neighboring sensor nodes along with reliable and efficient transfer of information among links. The reliability is achieved irrespective of time variable conditions of wireless link. Then mechanisms are developed by considering reliability, energy and error rates. The increased error rates or increased reliability targets increase the energy consumption. The error-control techniques are designed with a goal to achieve certain reliability level in packet transmission.

→ In general reliability improvement costs more energy. Therefore, these goals are set carefully. ~~However~~

Tasks and requirement

Network layer use the link-layer services for packet delivery. A reliable commn link betn neighbor for packet transmission is based on the following aspects:

(It includes user data & Protocol informatn for link & MAC layer)

(i) framing: Here user data is fragmented and formatted into (frames or packets) which include user data and Protocol related information for link layer (and underlying MAC layer). The format and size of packet has impact on througput and energy consumption metrics.

(ii) Error control: Here the effect of errors is compensated. The efficiency & energy consumption of different error control techniques depend upon the error patterns on the link.

(iii) Flow control: These techniques introduce some signaling technique to slowdown the transmission of the transmitter so that the receiver with lack of buffer space can manage the data reception. For WSN, most flow control techniques are integrated with error control protocols.

(iv) Link management: This mechanism involves discovery, setup, maintenance, of links to neighbors. The (main part of link management process is the estimation of link quality) which is (used by higher-layer Protocols for routing decisions and topology control purposes.)

Error control:

These techniques deal with transmission errors in wireless links. The data transport services of link layer are:

→ Error free, in-sequence (the data sequence in transmitter data link layer (DLL) must be followed in receiver's DLL. The sequence A, then B data is maintained.), duplicate free, and loss free (the receiver's DLL must get any piece of information.

Cause and characteristics of transmission errors

Physical phenomenon like reflection, diffraction, and scattering of waveforms leads to fast fading & intersymbol interference. path loss, attenuation cause slow fading.

Format of IEEE 802.11 / 802.11 b physical layer frame

Sync(128bit)	SFD(16bit)	Signal (8bit)	Service(8bit)	Length(16bit)	MPDU(variable)

← Preamble → ← PHY header →

(Provide equalizatn & receiver synchronizatn) (Describes modulatn type & data length)

The above is format of Physical-Layer Protocol Data Unit (PPDU) of the IEEE 802.11 WLAN standard with DSSS Physical layer. PPDU is subdivided into preamble, a Phy header, and the MAC-layer Protocol Data unit (MPDU) data part. This carries MAC packet.

→ Preamble is a constant bit pattern and useful for equalization purposes and allow receiver to acquire bit and frame synchronization. Physical header describes the modulation scheme & length used in data.

→ The end of PHY header and begining of actual MPDU is indicated by fixed SFD. packet losses occur if (i) receiver fail to acquire bit/frame synchronization, (ii) the SFD is wrong, iii) bit errors in the remaining PHY header lead to incorrect header checksum. If the received bits are not same as transmitted bits then bit error occurs. ARQ Protocols provide checksums to detect bit errors and drops the entire packet.

ARQ techniques (Feedback based tech.)

Here the retransmitting node's link layer accepts data Packet, (creates link layer Packet by Prepending a header and a checksum) and transmits this packet to receiver. Receiver checks integrity of the Packet with the help of checksum and Provides feedback to the transmitter regarding success of packet transmission. Upon receiving negative feedback, transmitter performs retransmission.

The key ingredients of ARQ Protocols are

(i) Packet formatting : The DLL at transmitter accepts user data (U) with Some size. The DLL Prepends a header H to packets, which contain control and address information. This address distinguishes intended receiver at the transmitter. The control information include sequence numbers, flags

(ii) Checksum : It is also called frame check sequence. The checksum is appended to a packet after packet formatting Process. Checksum is a function of U & H i.e f(U,H). A major class of checksums are CRC values (8, 16, or 32-bit wide). The CRC values computed through linear feedback shift register. The receiver repeats the checksum calculation for received header H' & data U'. If the newly computed checksum is same as one carried in the Packet, the receiver accepts the Packet as correct.

(iii) Feedback generation : This step provide transmitter about the outcome of Packet transmission. The mechanism for obtaining feedback are timers (at transmitter) and acknowledgement packets. If there is +ve acknowledgement the receiver confirms the Packet reception. A -ve acknowledgement is sent when receiver detects a reception failure. Since both data and acknowledgment packets can be lost, the transmitter uses timers. Transmitter sets timer when the last bit of data Packet has been sent to alert itself when the likely time for receiving the acknowledgement has expired.

Retransmission : Upon receiving -ve feedback for a packet, transmitter perform retransmissions. Hence, transmitter has to buffer the packets, and the transmitter need to decide when to retransmit and what to retransmit.

The ARQ protocol differ in their buffer requirements and retransmission strategies. The Standard ARQ Protocols are:

(i) Alternating bit:

Here transmitter buffers one packet, sends it and sets a timer. The receiver either received the packet and sends positive acknowledgement else it sends negative acknowledgement. If transmitter receives the acknowledgment the buffer is freed and next packet is transmitted. The packets are alternately numbered with 0 and 1. Retransmitted Packet have same sequence number. This sequence number allow receiver to detect duplicates. Alternating bit can Provide loss free, duplicate-free and in-sequence delivery of data

(ii) Goback N: Alternating bit is inefficient in case of 'long fat pipes', i.e links where multiple packets can be in transmit during a round-trip time. Goback N Protocol allows transmitter to have multiple unacknowledged frames. The transmitter keeps a buffer upto N packets, called as window. Each packet in the window has own timer, start with packet transmission. The receiver accepts frame in sequence and drops frame that correctly received. The receiver shows acknowledgement by acknowledging the last packet of the sequence in the single frame. If timer for oldest frame at transmitter expires due to non acknowledgement, then that frame and all other frame in window are retransmitted. (frame constitutes of many Packets; Window consists of many frames).

(only erroneous packet tested not entire window)

(iii) Selective Repeat / Selective Reject: This has similarities to Goback N. However, unlike Goback N Protocol, in Selective Repeat the receiver also has N buffers and uses them to buffer frames arriving out of sequence. To have in sequence delivery of data to user, the receiver keeps out of sequence packets in buffers until missing Packets have arrived. The receiver can use both positive and negative acknowledgement. Send and wait and Selective Repeat have important Property that only erroneous Packets are retransmitted while Goback N also retransmits correctly received packets.

How to use acknowledgement :

The energy can be saved by reducing the number of acknowledgement packets, i.e if a single acknowledgement packet acknowledges multiple data packets. In selective Repeat case a scheme named 'windowed feedback with Selective Repeat' where the receiver sends acknowledgement either after receiving agreed number W of packets after timeout or a duplicate packet reception. Another scheme named 'Instantaneous feedback with Selective Repeat Scheme' where the receiver in addition sends -ve ack. upon receiving out of order packet. Upon reaching end of the window without having received any feedback, the transmitter repeats the newest frame, triggering acknowledgement generation in the receiver.

* If delay is premium, the instantaneous feedback scheme is preferable but if throughput is more important the windowed feedback scheme is better. because increasing window size leads to decreased energy consumption but also increased delays. for windowed feedback scheme increased window size decrease energy consumption with moderate delays.

Who to retransmit :

In ARQ protocols transmitter has to decide at which point in time it has to retransmit. This depends on channel error characteristics.

→ In case of static BSC any time instant is good as any other, we can't improve successfull transmission by waiting.

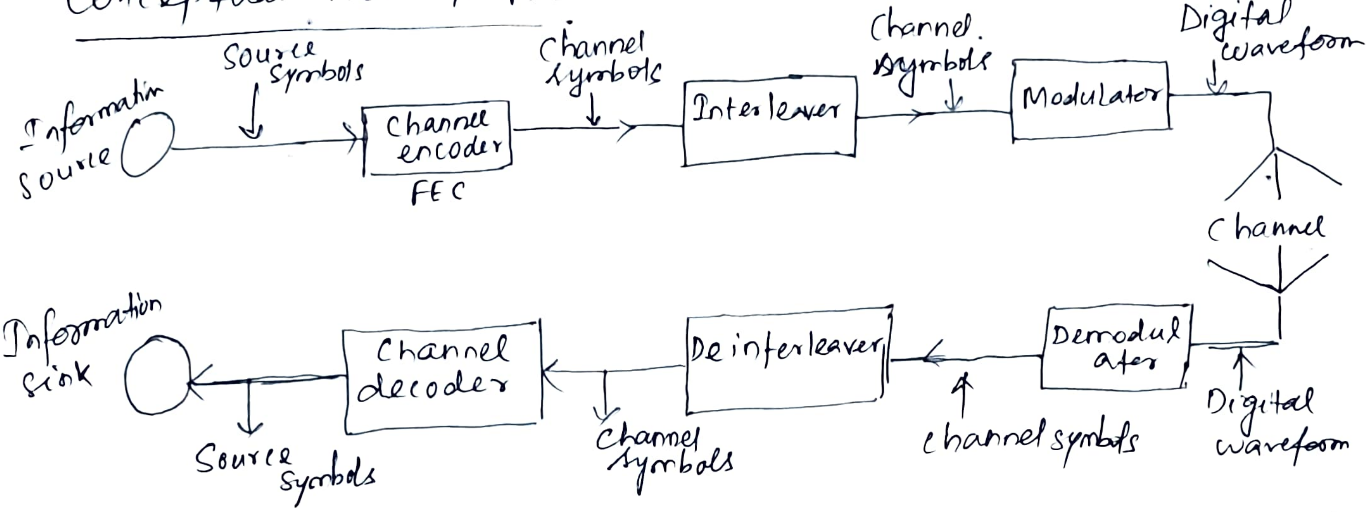
→ In fading channels, if packet hit by deep fade and if packet length is short than avg. fade duration an immediate retransmission occurs in deep fade which is waste of energy. Hence, in fading channel it is wise to postpone retransmissions.

Probing protocol :- This protocol distinguishes two different channel modes i.e normal mode and probing mode. During normal mode the transmitting node sends packet according to ARQ protocol. If transmitter received -ve feedback it switches into the probing mode. In this mode transmitter periodically sends small probe packets. These probes are ack. by receiver. After receiving such probe ack. transmitter assumes both forward & backward channel is ok and continue normal mode.

FEC techniques : (User symbols used, it is an openloop tech.)

In this technique transmitter accepts a stream or block of user data, adds suitable redundancy and transmit the results to receiver.

Conceptual view of FEC



Depending on the amount and structure of redundancy, the receiver is able to correct some bit errors. FEC is an 'openloop technique' i.e. there is no feedback from receiver. Hence, the transmitter uses some coding method all the time. Two widely used codes are block codes and convolutional codes. The turbo codes has a potential to achieve shanon capacity of channels, but requires complex implementation hence these turbocodes are not considered for WSNs.

Block-coded FEC. (Provides channel symbols from source symbols)
Code stands for mapping of 2^k source words to 2^n channel words

These codes takes a block or a word of a number k of P-ary source symbols and produces a block consisting of n of q-ary channel symbols. Generally $p=q=2$, $n \geq k$ and the symbols corresponds to bits.

→ Different source blocks are coded independent of each other. Basically there is a mapping from 2^k different source words into 2^n different channel bits. This mapping is called a code, the ratio $\frac{k}{n}$ is the code rate. (Small code rate equals high redundancy and small useful data rate.

→ The reliable correct bit number (t) in a channel of block length n bits depend on coding scheme. However the upper bound is imposed by (Hamming bound that states that a block with k user bits mapped to n channel bits can correct upto t bit errors if)

$$2^{n-k} \geq \sum_{i=0}^{t} \binom{n}{i} \quad \text{holds}$$

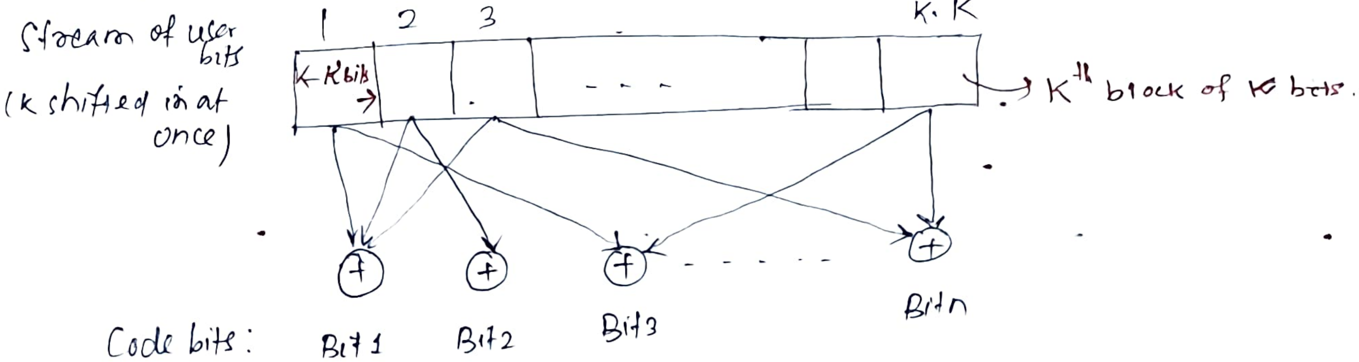
→ Block codes has important metric as Hamming distance. Hamming distance of two valid channel words w_1 & w_2 is defined as the number of bits in which they differ and Hamming distance d_{min} of whole code is defined as the minimum Hamming distance of all pairs of valid channel words. Any code with Hamming distance d_{min} can reliably detect upto and including $d_{min}-1$ bit errors and can reliably correct upto and including $\frac{d_{min}-1}{2}$ bit errors. (4)

→ As FEC uses open-loop technique, an overhead at transmitting node occurs due to transmitting extra bits for computations related to coding. The BCH codes uses a linear feedback shift register and uses negligible cost. For decoding the energy cost for algorithms (depend on block length n and code rate of correctable bits.)

→ If error in channel vary with time, the residual error rate of FEC varies too. If bit error rate is small all the time, a large FEC overhead not justified. Conversely for high rate of errors (deep fade in Rayleigh fading channel) very low code rate needed to ensure error free transmission. Decoding codes with low code rate is energy consuming. Therefore, it is appropriate to await the end of the fade and continue with moderate code rate in the following good channel period.

Convolutional codes :

Here k bits of user data is mapped to n channel symbols. However, coding of two successive k-bit blocks is not independent.



(Operation of convolutional coding)

→ Encoding Procedure runs in steps. During each step, k bits of user data are shifted into shift register. The length of shift register is K.k bits where K is constraint length of code. The k user bits are shifted on one side of register while on other side the 'oldest' block of 'k' bits removed.

→ The ~~o/px of~~ n-modulo-2 adders sums up a specific subset of K.k registers. O/px of n-adders transmitted once per step. The coding scheme has property that k bits are present in shift register for K steps and thus (coding of k user bits depends not only on these bits themselves but also on previous (K-1).k user bits.)

→ Similar to block codes encoding of convolutional codes is cheap in terms of energy. For decoding convolutional codes Viterbi algorithm used, whose memory requirements of receiver depend (exponentially on constraint length K)

Interleaving:

Many coding schemes, particularly convolutional schemes, don't acheive optimum performance when subjected to bursty error. A common technique to circumvent this is interleaving.

→ An interleaver in transmitting node accepts fixed length data packet generated by FEC encoder, which consists of multiple coded blocks.

→ Bits in the packet are permuted before transmission. The deinterleaver at receiving node inverts the permutation before it is sent to FEC decoder.

→ When error burst hits the permuted packet, the deinterleaving spreads this concentrated burst over this entire packet length. Hence, error burst are spread over multiple coding blocks instead of concentrating to one or a few blocks. This improves the chance of sucessful decoding of each block.

Summary (error control):

The conclusion is both increased reliability requirements and increased channel error rates demands more energy. FEC coding have real gain in energy efficiency for given reliability and medium bit error rates.

→ For both convolutional and block coding, the energy for encoding is negligible but decoding energy is significant. Block codes tend to be more energy efficient than convolutional codes but convolutional codes have better error correction capabilities.

→ ARQ schemes adapt their overhead to channel conditions. On excellent channels overhead is acknowledgement frames. However, for channel errors standard ARQ retransmit whole packets, even if few bits are wrong.

Framing: Deals with packet size issue

Generally link layer constructs a frame and then it is transmitted. There is an important issue with packet size. Large framing overhead favours large packet sizes to achieve reasonable energy efficiency per bit.

→ Again large packets are more susceptible to bit errors if FEC is not applied. (Requires fragmentatⁿ & reassembly work at link layer adding complexity)

Adaptive schemes: (used to handle diff error condⁿs (slow & fast fading) through channel estimatⁿ, which means finding prob. of bit error rates)

Wireless channel error conditions fluctuate both fast and slow fading ~~scale~~. For this scenarios the transmitter and receiver nodes continuously estimate channel conditions and adapt packet size accordingly.

→ The (instantaneous channel condition) is estimated through collecting channel quality at receiver and then transmitting back to transmitter with acknowledgement frames to adjust the frame size by transmitter. For this the metrics used by receiver are

(i) The success information obtained by pure ARQ scheme used to estimate the instantaneous bit error rate.

(ii) In FEC, the FEC decoder might provide information about number of corrected errors, which receiver feeds back to transmitter.

(iii) If transceiver delivers additional receive information like RSSI, this can also be taken into account.

Adaptive schemes use either framing table or MLE for getting optimal packet size

Intermediate checksum schemes. Long packets txted without fragmentation, suitable when only few bits are error

This scheme allows to use long packets without a fragmentation/reassembly scheme. This scheme is suitable for cases where only few bits in packet are erroneous. Retransmission is restricted to those parts of packet where errors occured actually. Intermediate checksum scheme can also be integrated with ARQ protocols.

Traditional frame's

O	S	h

Head bits (Source, dest. address, frame length, Control information.)

Trailer bits Contains frame Checksum

Traditional framing

Traditional

DLL/MAC header	User data	FCS

Intermediate checksum framing

DLL/MAC header		FCS		FCS		FCS

& checksum calculation is computationally intensive

→ In traditional framing with u bits of user data, header of o bits in front of user data and trailer of h bits behind user data. The header carries source & destination address, frame length information whereas trailer consists of frame's checksum. The total frame size is O+S+h. Therefore, all O+S+h bits need to be transmitted again during retransmission. Similarly, if receiver detects error whole frame is discarded & transmitter retransmit the frame again.

→ In intermediate checksum scheme u user bits are partitioned into L chunks of each (chunk size is c bits) to which checksum h' bits is appended. A frame header with all chunks of size $o' \geq o$. The overall frame size $o' + L \cdot (c + h')$ bits. The frame header in intermediate checksum includes a separate header checksum and information about chunk size/number. (For reception if header is correct), each chunk is checked separately and buffers correct chunks. If all chunks are correct receiver delivers the frame to upper layer. If chunks are incorrect, faulty chunks are indicated to transmitter with incomplete acknowledgement. The transmitter retransmits only faulty chunks. In this way retransmission frame is (smaller), consume (less energy), (less affected by errors), and reaches receiver at (smaller delay). The intermediate checksum may impose high overhead.

Intermediate checksum schemes

Combining Packet-Size optimization & FEC

Packet size optimization can also be combined with FEC investigate BCH codes & convolutional codes regarding energy efficiency.

→ for BSC channel with moderate bit error rate of 0.001, the optimum packet length with respect to energy can be significantly increased when BCH codes are used as compared to case without error correction.

→ for datagram traffic & for small packet sizes (50 bytes), a combined FEC & ARQ shows same values on ~~ergo~~ energy per useful bit and expected packet delay for both good & bad channel. The ARQ only scheme offers small delays and small energy consumption for both good channel and large delays as well as large energy consumption are encountered in case of bad channel. for packet size (1500 bytes) ARQ+FEC offers better delays at high energy costs for good channel while for bad channel both scheme have same average delay with FEC+ARQ requiring significantly more energy.

→ for speech source for low bit error rates during good channel period the ARQ only scheme require lesser energy than FEC+ARQ & both schemes drops no packet. For high bit error rates ARQ only scheme requires much more energy and at same time drops much more packets the ARQ+FEC scheme.

Framing summary:

The two aspects of framing (choice of packet size, intermediate checksum schemes) have common theme. They try to reduce the amount of information that is retransmitted in case of errors without imposing too much additional overhead.

→ Significant energy savings are possible but additional complexities like obtaining necessary feedback for adaptive schemes or fragmention and reassembly process impose energy as well as memory costs.

Link management:

The upper layers, specifically routing protocols need to know about available neighbors and also about link quality of their neighbors.

this quality informat? provide sensible routing decisions by avoiding bad links with a high chance of packet loss. some realization of linker

→ Link quality is not binary, i.e. there may be more link qualities than "good" and "bad". A link quality is characterized by the probability of loosing a packet over this link.

→ The quality of link is time variable because of mobility or when some obstacle has moved betn nodes.

→ The quality has to be estimated, either by sending probe packets and evaluating the responses or passively by overhearing the neighbor's transmission.

Link quality characteristics :

the link quality is expressed interms of packet loss rates. In a measurement of link quality in a 13x13 grid of 169 motes in open parking place, spaced 2 feet apart. Here one node at a time transmits packets and all other try to receive them, the findings are:

(i) For given transmitter power, there is no deterministic relationship betn distance & link quality. Nodes at same distance from transmitter can experience widely varying packet loss rates.

(ii) The region around a node having certain packet loss doesn't have shape of circle but is irregularly shaped.

(iii) There is significant degree of asymmetric links. In an assymetric link packets sent from node A to node B are received by B with few losses but conversely A receives B packet with much higher loss probability.

(iv) Packet loss rate is time vareable even when the neighbors are stationary.

The quality of packet reception rate vs distance relation shows three different regions. such as a) effective region, (b) Poor region (c) transitional region.

Effective region: Receiving nodes have a distance of at most 10 ft to the transmitter and consistently more than 90% of the packets are received by nodes in the region.

Poor region: Starts at distance of 40ft betn tx & Rx and the nodes consistently have loss rates beyond 90%.

transitional region: ~~In between the~~ In this region in betn, the variance of the experienced loss rates for nodes at the same distance is significant.

Link quality estimation:

If a node wants to estimate the quality of link toward a neighboring node, it has to do so by receiving packets from the neighbor and judging their quality or computing loss rates. Due to large variations in loss rates for same distance it is not sufficient to derive the packet loss information from the known distance to next node.

The several properties for an estimator

Precision: It should collect enough results and give statistical meaningful results.

Agility: It should detect significantly changing link conditions quickly which result due to node movements.

Stability: Estimation should ~~detect~~ be immune to short/transient fluctuations in link quality.

Efficiency: It should avoid too much listening for other nodes transmission since this can cost precious energy.

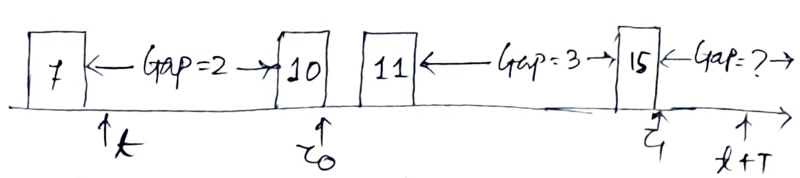
Estimators are passive and active estimators

In Active estimator the nodes send special measurement packet and collect responses from neighbors. By repeatedly doing so the loss statistics are obtained.

In passive estimator node overhears transmission of it's neighbors and estimates the loss rates from observing the neighbor's sequence numbers, packet losses are detected from gaps in received numbers.

Passive estimation technique

→ Input events to estimator are packet arrivals, packet losses have to be inferred indirectly from these using sequence numbers.



→ Suppose the estimator produce an estimate of packet loss rate in interval $(t, t+T)$ at time $t+T$, where T is the observation period. The estimator also determined the last sequence number it observed before time t is number 7. At time τ_0 when packet 10 arrived it finds that 2 packets are missing during interval (t, τ_0). Again from time t to τ_1 it finds 3 packets are received from 8. But which value shall be produced during time $t+T$. Therefore, to have reliable estimate, the estimator need to know the no. of packets lost in the last gap.

→ Some estimators such as pure moving average, time moving average window mean exponentially weighted moving average (WMEWMA) is used. The latter estimator produce predictions $\hat{P}_n$ only at time $t_n = n \cdot T + t$, denoted by $\hat{P}_1, \hat{P}_2, \hat{P}_3 \ldots$ the estimation sequence at these times. The estimator has 2 parameters $\alpha \in (0, 1)$ and observation period $T \in N$.

At an update instant t_n let r_n be the number of received packets in $(t_{n-1}, t_n]$ and f_n is the no. of packets identified as lost during $(t_{n-1}, t_n]$. Then

$$\mu_n = \frac{r_n}{r_n + f_n}, \quad \hat{P}_n = \alpha \cdot \hat{P}_{n-1} + (1-\alpha) \cdot \mu_n$$

Summary:

The design of link layer must focus on energy efficiency. Several techniques to save energy such as FEC or packet size optimization can be used.

→ If sensor n/wk is characterized by periodic data transfer, the different adaptation mechanisms are used as energy is drained continuously & we can reduce drain rate. On other hand if n/wk waits for occurance of rare event before its transmission, then it starts with robust settings (FEC, small packets) from the very begining.