

RESEARCH

Open Access

# Turbo codes for multi-hop wireless sensor networks with decode-and-forward mechanism

Nashat Abughalieh<sup>1,2\*</sup>, Kris Steenhaut<sup>2</sup>, Ann Nowé<sup>2</sup> and Alagan Anpalagan<sup>1</sup>

## Abstract

Wireless channels are prone to many impairments, such as noise and fading. Weak channels between the nodes in the wireless sensor network (WSN) can cause reception of erroneous packets. Retransmission mechanisms are mainly used to tackle the problem of erroneous reception in WSN communication protocols. Weak channels can cause high number of retransmissions in order to deliver a packet correctly, which will consume high energy of both the transmitting and the receiving nodes. Error correcting codes (ECCs) can be used to reduce number of retransmissions, but most ECCs have complex decoding algorithms, which leads to high processing energy consumption at the receiving nodes in the WSN. In this paper, we present a low power consumption decode-and-forward approach for the multi-hop WSNs; a serial concatenation convolutional codes (SCCC) encoder is implemented at the source node while the complex iterative decoding algorithm is shifted to the sink (base station). The intermediate nodes run a Viterbi decoding algorithm to decode only the inner code of the SCCC encoder. We investigate the effect of changing constraint length of both the inner and the outer codes and the effect of changing encoding block size. We show that most packets can be decoded at the base station at low signal-to-noise ratio (SNR) channels with the penalty of small energy loss in decoding the packet at the nodes in the network.

## 1 Introduction

Wireless sensor networks (WSNs) consist of small wireless nodes equipped with sensors, small scale processor, and radio frequency unit, all powered by batteries or scavenging the energy from the environment. The nodes in the WSN communicate their data in multi-hop and aggregate it at the base station as shown in Figure 1. WSNs have a large set of possible applications like environmental monitoring, home automation, military, industrial, and medical applications. They distinguish themselves in their unique capability of gathering information in remote, isolated, and often harsh environments, such as battlefields and deep water.

Different WSN applications require different network design criteria, i.e., for field monitoring applications, the

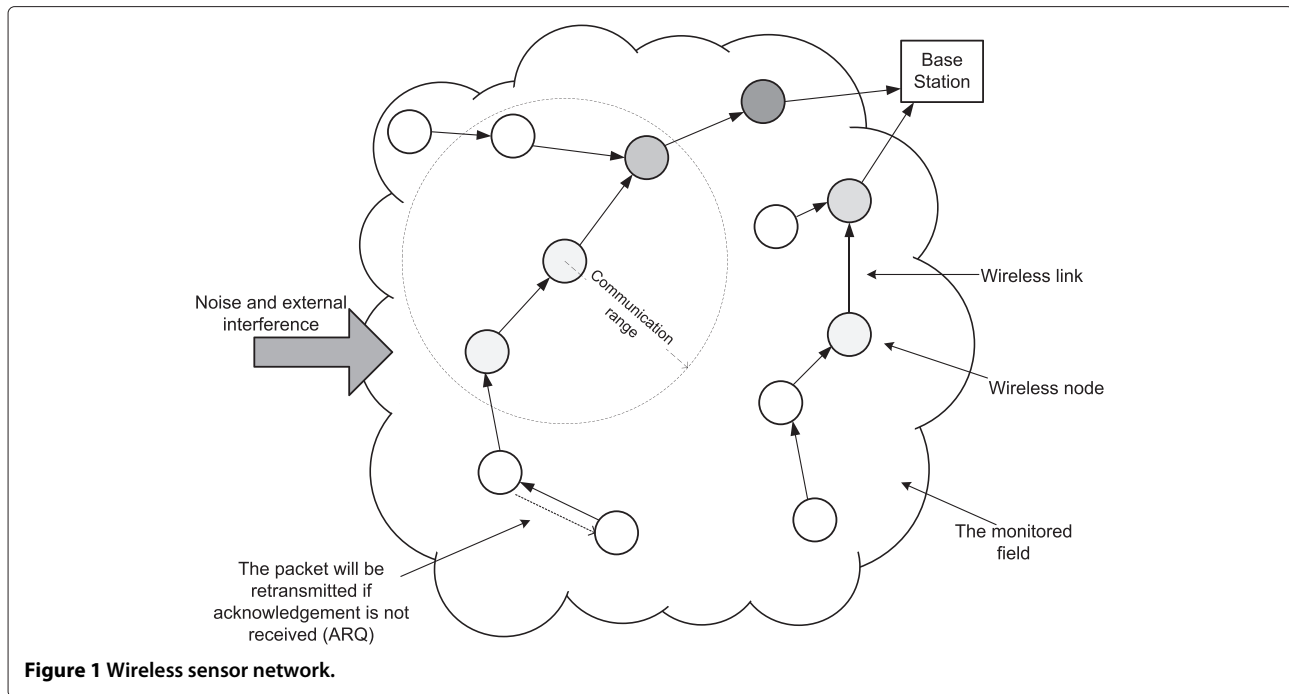
network size, expandability, lifetime, and energy scavenging are among the most important design criteria. While in body area networks (BANs), the nodes have strict transmission power limits, and channels between the nodes are prone to high noise and shadowing, and network reliability and delay are the most important design criteria in such networks. In other real-time applications, such as in camera-surveillance wireless sensor networks, the nodes have high transmission data rate and network throughput is the bottleneck in the design of such networks.

This paper targets WSN applications with noisy channels or high data rates. In such applications, retransmitting failed packets without applying forward error correction techniques can lead to high energy losses and delay. This paper studies different error correction techniques for WSNs, taking into account both communication and processing energy consumption at the nodes and error correction capability of each technique. It also presents the new design of forward error correction technique that builds on the serial concatenation convolutional codes

\*Correspondence: nabughal@net.ryerson.ca

<sup>1</sup>Department of Electrical and Computer Engineering, Ryerson University, 350 Victoria St., Toronto ON M5B 2K3, Canada

<sup>2</sup>Department of Electronics and Informatics, ETRO Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussel, Belgium



with decode-and-forward mechanism at the intermediate nodes between the source node and the base station in the WSN.

Due to the limited battery or scavenged energy in the nodes of the WSN, minimizing power consumption at the nodes is essential for efficient and sustainable network operation. The energy of the nodes in the WSNs is utilized by three main components of the node: the sensors, the transceiver, and the processor. The transceiver is known to have the highest consumption. Many previous works suggested protocols for medium access control (MAC), network topology, and routing to minimize the consumption of the transceiver [1-4]. The MAC protocols control sleep and wake periods of the transceivers to minimize energy consumption and schedules transmissions and receptions in the network to reduce packet collision possibility. Routing protocols defines routing paths from the source nodes through the other nodes to the base station. Routing paths are optimized to maximize data compression and network throughput [5] or network lifetime [6] or minimizing power consumption in the network [7]. Different routing and MAC protocols [8,9] are presented in the literature to tackle different WSN problems, but most of these works neglect the noisy nature of the wireless channels. This paper studies the communication between the nodes over noisy channels and possible implementations of error correction techniques at the physical layer level. MAC and routing problems are considered to be tackled at MAC and network layers, and they are out of the scope of this paper.

Transmission over the wireless channel is prone to noise, fading and shadowing, which leads to the reception of erroneous packets. Automatic repeat request (ARQ) is currently used to tackle the problem of erroneous transmission in WSNs by retransmitting the erroneous packets. One retransmission of a packet doubles the power consumption at both the transmitter and the receiver. The undesirable scenario is to have successively failed retransmissions when the nodes are communicating over weak channels.

To reduce packet error rate caused by the noisy channels, error correcting codes (ECCs) are commonly used. The use of ECCs not only reduces the power consumption caused by the retransmissions of erroneous packets but also leads to reduced delivery time of the packets to the base station. The deployment of ECCs comes with an increase in the processor power consumption at the sensor nodes caused by encoding and decoding processes. However, the deployment of ECC techniques reduces the power consumption at the transceiver, caused by the retransmissions of the packets, but this reduction comes at the cost of transmitting redundant parity bits which together with the data in the packets are used in the decoding process to correct errors. The need for powerful ECC is crucial in some applications when retransmissions are expensive in latency or energy. Strong ECCs provide high error correction performance at low channel quality but have more complex decoders than simpler ECCs. If the power consumption caused by the transmission of the encoder's redundancy bits together

with the power consumption of the encoding and decoding processes outweigh the power saving due to the reduced retransmissions, then the ECC technique is not efficient.

This paper presents the following:

- Performance and energy consumption analysis of multi-hop wireless sensor network running state-of-the-art error control technique (ARQ), communicating over weak channels.
- Analysis of encoding and decoding processing power requirement and energy consumption at the nodes in the multi-hop wireless sensor network, when implementing conventional error control coding algorithms (Bose, Chaudhuri, and Hocquenghem (BCH), Reed-Solomon and convolutional codes).
- The performance improvement and the reduced energy consumption at the wireless nodes in the network when implementing the first error control code design in this paper. This design consists of implementing parallel or serial concatenation turbo code, with coding rate  $\frac{1}{3}$ , at the source node and turbo decoding algorithm at the base station.
- The second error control code design for multi-hop wireless network that updates the first design in this paper with partial decoding algorithm running at the sensor nodes and the compromise between performance-processing energy of the designed ECC.

This paper is organized as follows: the literature review is presented in Section 2. Section 3 discusses the wireless channel model used in the paper. The error correcting codes and their processing energy consumption are discussed in Section 4. The proposed design is analyzed in Section 5 and evaluated in Section 6. Section 7 concludes the paper.

## 2 Related work

Error correcting codes for wireless sensor networks are proposed in the literature but focus primarily on conventional ECCs such as BCH and Reed-Solomon and convolutional codes [10-12]. A hybrid scheme combining energy-efficient ECC techniques and ARQ is studied in [13]. In [14], convolutional codes are analyzed for power consumption in a slow Rayleigh fading channel. Results show that the energy required for encoding data is negligible. However, performing Viterbi decoding on an Intel StrongARM SA-1110 Microprocessor is energy-intensive since the average energy consumption grows exponentially with the constraint length of the code, independent of the code rate. In [10], a comparison was made between different error correction codes on field-programmable gate array (FPGA) and ASIC in terms of complexity,

power consumption, and correction capability. The study demonstrates special features of the convolutional codes compared to the BCH and Reed-Solomon codes, that is,

- Increasing the constraint length of the convolutional code reduces the probability of errors at the receiver, without increasing the coding rate.
- The power consumption at the convolutional encoder is almost lower than all encoders in BCH and Reed-Solomon at same coding gain and almost insensitive to changes in constraint length.

Pellenz et al. [15] used the above features of the convolutional code to select the appropriate constraint length of the convolutional encoder for each link in WSNs according to the link quality estimates. Short constraint lengths are assigned to high quality channels while long constraint lengths are assigned to weak channels. In [16], advanced error correction codes using turbo codes are tested for single-hop WSNs.

Turbo codes have a superior performance over the conventional error control codes, which inspired Zhong et al. [17] to suggest using a power efficient single-hop sensor network instead of multi-hop sensor network by deploying turbo codes to extend the communication range. The high power consumption of the turbo decoder prevents the authors in [16] and [17] from implementing turbo codes on multi-hop WSNs.

In [18], a relay is used to help the transmitting nodes that communicate with the base station. The relay performs a column-base linear block encoding on the received row-base linear block-encoded packets sent by the nodes. The relay sends parity bits to the base station which performs turbo decoding using the received packets from the nodes together with the parity bits from the relay. Youssef and Graell i Amat [19] also studied the communication between multiple sources and the base station, with the help of one relay; the relay decodes the convolutionally encoded data from all sources then mixes and encodes them, then sends the output to the base station. Both [18] and [19] consider, in their network model, that a base station is always in the communication range of the transmitting node, which is not always true in WSNs. The sensor nodes in the WSN have limited transmission range, due to the limitation in energy sources. Only near nodes to the base station are able to communicate with it in one hop.

In this paper, we investigate the use of the serial concatenated convolutional codes (SCCC) in multi-hop WSNs with turbo decoding at the base station and partial decoding at the routing sensor nodes.

### 3 Channel model

Wireless communication between the nodes in sensor networks suffers mainly from the free path loss attenuation and the additive white Gaussian noise (AWGN) that limit the transmission range of the nodes [20]. Multi-path fading is an additional degradation to the communication quality; fading distortion is caused by the reception of the same transmitted signals with different traveling paths. The problem is that these signals are shifted in time and phase with different amplitudes. The narrow-band fading process is classically modeled using the Rayleigh probability distribution [21]. The channel quality between two nodes, in terms of packet or bit error probability, depends on the average signal-to-noise ratio (SNR) at the receiver. The instantaneous SNR depends on the channel gain during a block of data transmission and follows the Rayleigh distribution. The path loss model can be defined by [21]:

$$\frac{P_t}{P_r} [\text{dB}] = \mathcal{L} [\text{dB}] - 10\gamma \log_{10} \left[ \frac{d}{d_0} \right] - \psi_{dB} \quad (1)$$

In this approximation,  $P_t$  and  $P_r$  are the power of the transmitted and the received signals, respectively.  $d_0$  is a reference distance for the antenna far field, and  $\gamma$  is the path-loss exponent. The values for  $d_0$  and  $\gamma$  can be obtained to approximate either an analytical or empirical model.  $\psi_{dB}$  is a Gaussian-distributed random variable that represents the shadowing effect,  $\psi_{dB}$  has zero mean and variance  $\sigma_{\psi_{dB}}^2$ .  $\mathcal{L}$  is a unit-less constant that depends on the antenna characteristics and the average channel attenuation. For simplicity, we consider the nodes to have omnidirectional antennas, so that

$$\mathcal{L} [\text{dB}] = 20 \log_{10} \left( \frac{\lambda}{4\pi d_0} \right), \quad (2)$$

where  $\lambda$  is the wavelength. The value of  $\gamma$  depends on the propagation environment; for propagation that approximately follows a free-space model,  $\gamma$  is set to 2. The value of  $\gamma$  for more complex environments can be obtained via a minimum mean-square error (MMSE) fit to empirical measurements. For an office building (same floor),  $\gamma$  can range between 1.6 and 3.5.  $\gamma < 2$  is caused by the wave guide constructed by the corridors. For a home environment,  $\gamma$  is found to be 3. In our analysis, we set  $\gamma$  to 3.

The combined effects of path loss and shadowing have an important influence on WSN's connectivity and performance. In wireless communication systems, there is typically a target minimum received power level  $P_{\min}$ , below which performance of the system becomes poor and the packet error rate becomes high. However, with shadowing, the received power at any given distance from the transmitter is log-normally distributed with some probability of falling below  $P_{\min}$ . The probability that the

received power falls below  $P_{\min}$  at distance  $d$  is known as the *outage probability* and given by the following:

$$p(P_r(d) \leq P_{\min}) = 1 - Q \left( \frac{P_{\min} - \left( P_t + 10 \log_{10} \mathcal{L} - 10\gamma \log_{10} \left( \frac{d}{d_0} \right) \right)}{\sigma_{\psi_{dB}}} \right), \quad (3)$$

where  $Q(\cdot)$  is the Q-function. The AWGN and the slow fading channel<sup>a</sup> significantly decrease WSN's performance in terms of energy consumption, because in general, the packets received with errors must be retransmitted. Without loss of generality, let us consider the error bits to be uniformly distributed in the packet, then the packet error rate (PER) can be estimated for a specific data packet size as

$$\text{PER} = 1 - (1 - \text{BER})^m, \quad (4)$$

where  $m$  is the length of the data bits in the packet.

### 4 Error correction codes and their processing power

In this section, we study the processing power consumption of different error correcting algorithms, This section presents empirical analysis of energy consumption caused by transmitting redundant bits in the coded data packets. The processing and the transmission energy consumption are used to calculate the extra energy consumption of the serial concatenation convolutional encoder at the source node and energy consumption of the decoding operation running at the intermediate nodes.

The standard error control strategy employed in WSNs is the ARQ protocol. In such a protocol, the sensor nodes have only error detection capability to identify the correct packets. After the reception of the packet, the receiving node sends an acknowledgment packet to the transmitter if the node failed to receive the packet. The transmitting node retransmits the acknowledged packets. However, when the channel condition stays poor for a long period, a large number of retransmissions may be required to deliver the packet correctly, which significantly increases the energy consumption of the network.

Error correcting codes can reduce the number of retransmissions, but they introduce redundancy in the data packet according to the coding rate. The redundancy in the data allows the receiver to receive the packet correctly, even with error bits in the packet. Error correcting codes can provide a better bit error rate (BER) performance of the communication for the same SNR compared to an ARQ, and can provide the same BER at a lower SNR. This difference in the SNR required to achieve a certain

BER for a particular coding and decoding algorithm compared to the uncoded case is called coding gain. Typically, there is a tradeoff between coding gain and decoder complexity. Very long codes provide a higher gain but require complex decoders with high power consumption.

As stated before, several types of error correcting codes are studied in the literature for WSNs, including Reed-Solomon codes [10,12], and BCH codes [10] as block codes. The terminology  $(n, k, t)$  of the block code indicates the codeword length  $n$ , dataword length  $k$ , and the error correction capability as  $t$  bits per codeword. Convolutional codes are also suggested for WSNs [10,15]; the terminology  $(n, k, K)$  of the convolutional codes indicates the rate of the code as  $k/n$ ; so for each  $k$  data bits the encoder outputs  $n$  bits.  $K$  is the constraint length of the convolutional encode. The number of memory elements used in the convolutional encoder is  $K - 1$ . A convolutional encoder typically consisting of a shift register of length  $K - 1$  and generates codes that mix input bit with memory values using an XOR operation to generate the output sequence. Convolutional codes are decoded on a trellis using either Viterbi decoding, MAP decoding, or sequential decoding.

Balakrishnan et al. [10] studied the power consumption of encoders and decoders for three ECCs (BCH, Reed-Solomon, and Convolutional codes decoded with Viterbi algorithm) for different code complexities. They used FPGA and ASIC implementations of the ECCs to measure the power consumption of each encoding and decoding algorithm and to calculate the error correction capability of each code. Figure 2 shows the result they obtained in their experiments for the implementations of the ECC on ASIC design [10].

From Figure 2, we can conclude that

- BCH codes are appropriate codes for WSNs in terms of power consumption of the encoding and the decoding algorithms, but BCH codes have a low error correction capability.

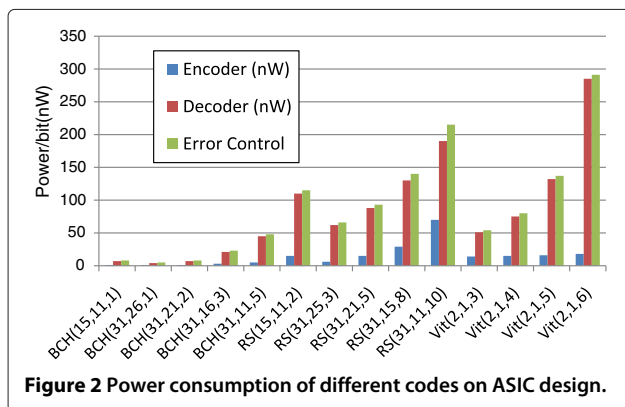


Figure 2 Power consumption of different codes on ASIC design.

- Reed-Solomon codes and convolutional codes have higher error correction capability but with higher power consumption, especially at the decoders. High complex decoding algorithms are not suitable for sensor nodes with limited energy source, memory size, and processing power.
- Both encoding and decoding power consumption and the error correction capability of the block codes increase with the code size. The encoding power consumption of convolutional codes is constant for different constraint lengths while the error correction capability and decoder complexity rise exponentially with increasing constraint length.

The redundant parity bits generated by error correcting systems increase the power consumption of the transceiver circuit at the sensor nodes. But the additional power consumption caused by transmitting a coded packet is less than the power consumption of the packet's retransmissions. Schmidt et al. [16] measured the power consumption of transmitting one uncoded packet with MICAz<sup>b</sup> node and compared it to the power consumption of transmitting a coded packet using repetition codes. They used a repetition code with rates  $\frac{1}{3}$  and  $\frac{1}{6}$  so that each data bit in the payload of the packet is repeated three or six times before modulation. Their results are shown in Table 1.

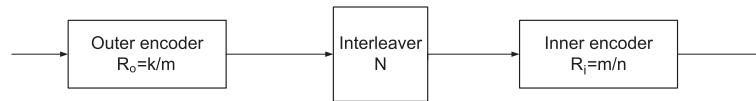
As a consequence of the high overhead for transmitting a packet, e.g., transceiver calibrations, preamble, and packet header, the energy consumption for sending one packet using the repetition code  $\frac{1}{3}$  does not even double the energy consumption for a packet transmission in the uncoded case. As we can see from Table 1, transmitting a packet with repetition code at coding rate  $\frac{1}{6}$  consumes less than 2.3 times the energy consumption of the uncoded transmission.

Error correction codes with iterative decoding algorithms, such as turbo decoding with component MAP decoders for each component code, and the sum-product algorithm (SPA) for low-density parity-check codes (LDPCs) have not been suggested before for WSNs with multi-hop communication due to the high complexity of the decoding algorithms.

Table 1 Comparison between repetition codes at different coding rates

Code	Rate	Energy [ $\mu$ ]
Uncoded	1	104.73
Repetition 1/3	$\frac{1}{3}$	128.97
Repetition 1/6	$\frac{1}{6}$	233.18

The coding runs at MICAz nodes. The table shows the transmission energy consumption with different codes. The energy consumption includes encoder processing power consumption.



**Figure 3** Block diagram of serial concatenation convolutional SCCC encoder.

In this paper, we analyze a scheme of deploying error correcting codes for WSNs by using serial concatenation convolutional codes (SCCC) for WSNs with intuitive turbo decoding, so that the encoding process takes place at the source node and the decoding process at the base station with partial decode-and-forward operations at the routing nodes.

The SCCC's encoder consists of two convolutional encoders separated by an interleaver that makes SCCC encoders have low complexity and power consumption. The first convolutional encoder is often called the outer encoder and has a coding rate  $k/m$ , while the second encoder has a coding rate  $m/n$  and is known as the inner encoder. The two encoders are separated by a random interleaver of size  $N$  as shown in Figure 3, and the total rate of the SCCC encoder is  $k/n$ .

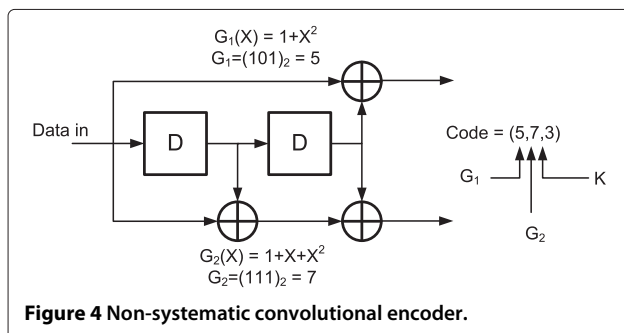
Convolutional encoders have two main types: the non-systematic convolutional encoder, as shown in the example in Figure 4, where the functions of the generating codes are implemented in feed-forward structure and generate output parity bits. The second type is the systematic convolutional encoder, as shown in the example in Figure 5. In this type of convolutional encoder, some of the code generating functions are implemented in the feedback structure. In this type, the output normally consists of the uncoded data together with the generated parity bits from the feed-forward functions. The functions of the generating codes can be represented with polynomial equation or binary sequence as shown in Figures 4 and 5.

The decoder of the SCCC code generally consists of two soft-input soft-output (SISO) decoders as shown in Figure 6. The decoders are also connected serially and separated by a de-interleaver. The inner decoder receives

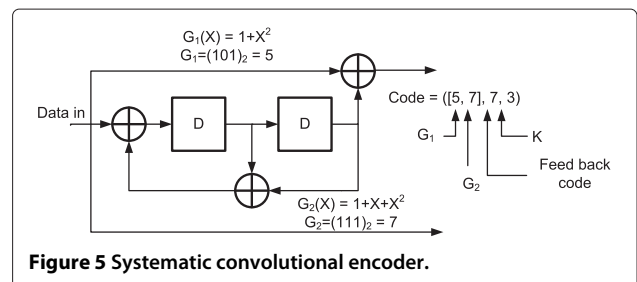
demodulated data and passes its soft output to the outer decoder through the de-interleaver. The outer decoder's soft output is fed back to the inner decoder through an interleaver, and the inner decoder uses this feedback for the next decoding iteration. The decoding process continues until the maximum number of iterations is reached or the output of the two decoders matches [22]. The interleaver in the SCCC encoder has a main role in the performance of the error correcting code: it introduces the randomness necessary for iterative decoding. In order to understand the performance of SCCC and the effect of design factors on the performance of the code, we need to formulate the error bounds of the SCCC's.

From [23], we can summarize the main rules for designing the SCCC as follows:

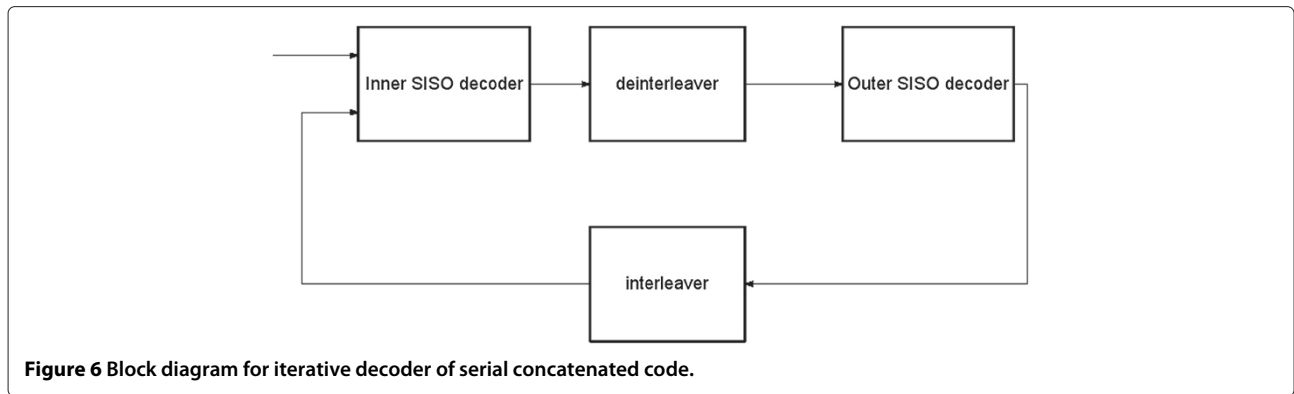
- For moderate to high SNR and large interleaver memory size  $N$ , increasing the interleaver length yields a gain in performance.
- To increase the interleaver gain, the free distance of the outer encoder should be high, which is often achieved by non-recursive convolutional coding.
- The use of a recursive convolutional inner encoder always yields an interleaver gain.
- Primitive feedback polynomial in the inner code maximizes the interleaver gain.
- To improve the performance of the SCCC at low SNR, the feedback polynomial of the inner encoder should have a factor of  $(1 + X)$ , which means that it cannot be primitive, and thus this contradicts with the previous design note. The choice between a primitive or non-primitive feedback polynomial is a trade-off between performance at low and high SNRs, respectively.



**Figure 4** Non-systematic convolutional encoder.



**Figure 5** Systematic convolutional encoder.



**Figure 6** Block diagram for iterative decoder of serial concatenated code.

### 5 Proposed design

Wireless sensor nodes have limited processing capability, so error correcting codes are not generally adopted into WSN standards. The empirical results of [10] depict that the decoder complexity of all error correcting codes have linear relation with correction capability. Shifting the decoder to the base station can solve the problem, but a high coding gain of the code is required to correct the accumulated errors in the communication over multi-hops.

SCCCs have superior performance over conventional ECCs. For testing the performance of the WSNs with SCCC, an SCCC encoder is placed at the source node with non-recursive convolutional outer encoder with rate  $\frac{1}{2}$  while the inner encoder is a recursive convolutional encoder with rate  $\frac{2}{3}$ . Figure 7 shows an example of an SCCC encoder circuit. The two encoders are separated by a random interleaver. The total code rate is  $\frac{1}{3}$ .

The decoder is placed at the base station, where enough processing power and energy are available, and it consists of two *a posteriori* probability (APP) decoders connected serially and separated by a de-interleaver. A feedback from the outer decoder is attached to the inner decoder through an interleaver.

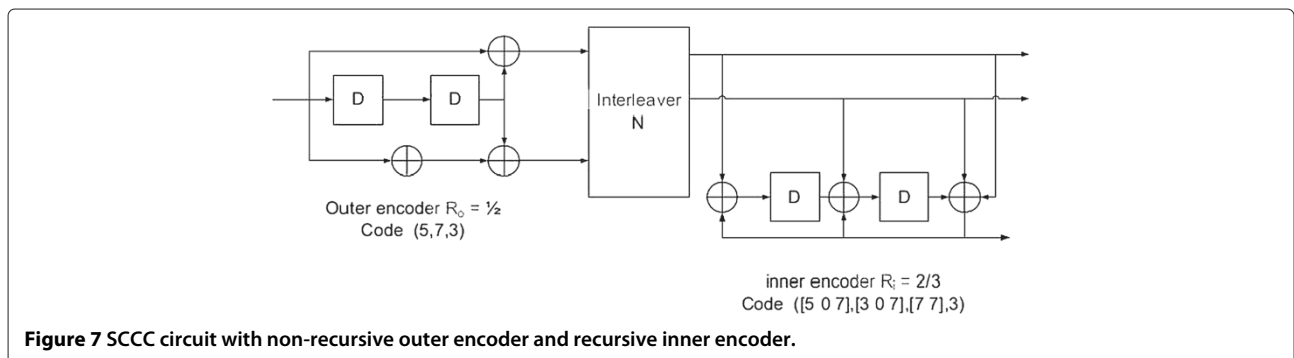
The maximum number of iterations at the decoder is 12. However, the decoder takes advantage of the CRC check

field at the end of each packet to check the correctness of the packet and stop the iterative decoding process.

Two scenarios are considered for the multi-hop WSN's design:

- The intermediate nodes are not decoding the packet.
- A Viterbi decoding algorithm is running at the intermediate nodes to decode only the inner code at each hop to reduce accumulated errors from multi-hop transmissions. After decoding the data with the Viterbi algorithm, the node encodes the output of the Viterbi decoder using the same inner code and transmits the coded data to the next hop.

The source node can configure the SCCC encoder to cope with the quality of the channel between the nodes in the WSN. By increasing the interleaver memory, the performance can be improved but at the cost of increased latency at the source node. If the WSN application is delay sensitive or the memory at the node is small, the source node can increase the free distance of the outer code by increasing the constraint length  $K$  and using longer generating codes. Increasing the free distance of the outer code will increase the interleaver gain for intermediate and high SNRs [23].



**Figure 7** SCCC circuit with non-recursive outer encoder and recursive inner encoder.

**Table 2 Viterbi decoder power consumption on ASIC design**

K (constraint length)	Power consumption on ASIC (nW)
3	51
4	75
5	132
6	285

## 6 Evaluation

### 6.1 Processing power consumption estimate of the design

We estimate the power consumption of the SCCC encoder used at the source node and the power consumption of the Viterbi decoder algorithm used at the routing nodes, using the measurements presented in [10], where Balakrishnan et al. used ASIC implementation of convolutional encoders and Viterbi decoders for estimating their power consumption. In this paper, we consider the designed encoders and decoders to be implemented in nodes' hardware. The results in [10], shown in Figure 2, are used in our calculations, because they were measured from ASIC designs, which reflects the real consumption of the hardware circuits. The figure shows that the convolutional encoders with different constraint lengths have approximately the same power consumption of 20 nW/bit, while the Viterbi decoders have exponentially increasing power consumption with the increment of the constraint length.

The SCCC encoder circuit consists of two convolutional encoders and the interleaver memory. If the power consumption of reading from and writing to the interleaver memory is neglected, then the total power consumption of the SCCC encoder is 40 nW/bit. The power consumption of the Viterbi decoder with increasing constraint length is given in Table 2.

### 6.2 Transmission energy consumption

The transmission and the reception operations are the main energy-consuming processes running at the wireless sensor node. By considering the nodes to be synchronized and only active during the reception or the transmission of a packet, the jointly consumed energy for both the transmitting and the receiving nodes can be defined as follows:

$$E_{\text{trans}}^{\text{bit}} = \frac{P_{\text{tx}} + P_{\text{rx}}}{R_b} \text{ [J/bit]}, \quad (5)$$

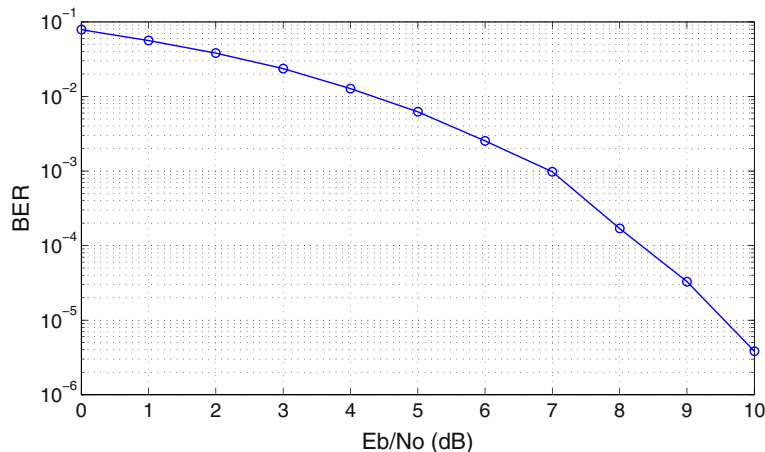
where  $P_{\text{tx}}$  and  $P_{\text{rx}}$  are the dissipated powers at the transmitter and receiver circuits, respectively, and  $R_b$  is the transmission rate in bits per second. The overall transmission energy for a coded data frame can be expressed as

$$E_{\text{trans}}^{\text{packet}} = \frac{E_{\text{trans}}^{\text{bit}} \cdot L}{R} + k \text{ [J/packet]}, \quad (6)$$

where  $L$  is the length of the packet,  $R$  is the coding rate,  $k$  is the overhead energy consumption of transmission and reception operations of one packet.

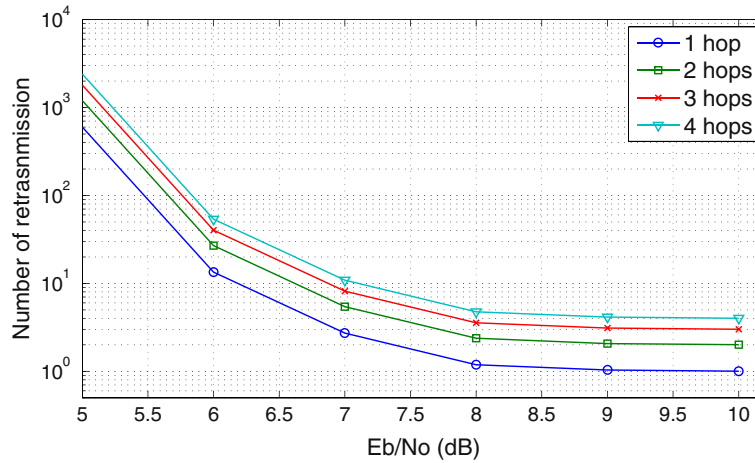
### 6.3 Network energy consumption with ARQ technique

In wireless sensor networks with an ARQ protocol, retransmissions are expected if the channel remains noisy. With the low data rate of WSNs, a slow narrowband fading channel is expected between the nodes. If the size of packets is small, then over the duration of a packet the channel remains approximately constant, hence, the AWGN model of the channel is appropriate. Let all channels associated with each hop be independent and identically Gaussian distributed. The packet error rate over each hop for the ARQ scheme can be approximated as  $\text{PER} = 1 - (1 - \text{BER})^m$  where  $m$  is the number of data



**Figure 8 Performance in terms of BER vs. SNR for OQPSK modulation between sensor nodes without ECC.** The curve was estimated using MatLab.





**Figure 9** Normalized power consumption vs. SNR for different number of hops.

bits in the packet. For an AWGN channel using quadrature phase-shift keying (QPSK) modulation, BER is given by

$$BER = Q(\sqrt{2SNR}). \quad (7)$$

The expected number of retransmissions of a data frame  $N_r$  depends on PER as

$$N_r = \frac{1}{1 - PER}. \quad (8)$$

Then, the total energy consumption for the transmission of one packet at one hop using ARQ can be expressed as

$$E = N_r \cdot E_{trans}^{packet} \text{ J/packet}. \quad (9)$$

Therefore, using (8) with the BER curve shown in Figure 8, it is possible to estimate the total energy consumption to deliver a packet as a function of the received SNR. With packet size of 128 bytes ( $m = 1,024$  bits), the normalized power consumption as a function of SNR for different numbers of hops is shown in Figure 9.

#### 6.4 BER and power consumption of WSN using SCCC with decode-and-forward at routing nodes

The simulated network is shown in Figure 10, where node A sends its packets to the base station through the other nodes in multi-hop communication. A SCCC is implemented at node A, and the Viterbi decoders are running

at the intermediate nodes. The base station runs the turbo decoder, and the performance of the network is measured by the bit error rate at the output of the turbo decoder. The performance of the WSN is evaluated at different interleaver memory sizes, different outer codes chosen from Table 3 and different inner codes chosen from Table 4. The inner codes tested are recursive convolutional codes with constraint length of 3 and a rate of 2/3. These inner codes are chosen with a short constraint length ( $K = 3$ ) because they will be decoded at each intermediate node. With  $K = 3$ , the Viterbi algorithm consumes about 50 nW/bit, which is as low as the energy consumption of the turbo encoder.

The WSN is evaluated, first with a Viterbi decoding algorithm running at intermediate nodes, second without decoding algorithms running at intermediate nodes. The BER performance curves of the WSN with decode-and-forward are shown in Figures 11 and 12. Figure 11 depicts the performance of the WSN with one communication hop between the nodes and the base station. Two symbol sizes are tested: 256 and 1,024 bits per symbol. Smaller symbol packets can converge faster, because large symbols are received with many errors at low SNR and require many iterations to converge or cannot converge. At higher SNR, decoding with larger symbols outperforms decoding with smaller symbols. The performance of the turbo decoders increases linearly with symbol size [21]. Figure 12a,b depicts the performance of WSNs for



**Figure 10** Multi-hop WSN. Node A transmits data to the base station through other nodes.

**Table 3 Code polynomials for the outer encoder**

	$K$	Code	Code function	$d_{free}$
$G_{01}$	8	345, 237	(8, [345, 237])	10
$G_{02}$	9	753, 561	(9, [345, 237])	12

different number of hops with the Viterbi algorithm running at the nodes at symbol sizes of 256 and 1,024 bits, respectively. Generally, the decoders with larger symbol size outperform the decoders with smaller symbol size for different network size.

Figure 12b demonstrates that even with a five-hop WSN, all packets are delivered at a channel SNR of more than 1 dB, while in the WSNs that do not deploy ECC, communication is not possible between the nodes at a channel SNR of less than 3 dB as depicted in Figure 9.

Figure 13a,b depicts the effect of using different outer codes with different free distances. The WSNs with the outer code that has a higher free distance performs better at higher SNR, but decoding complexity increases exponentially with the order of the code. The performance of turbo decoder with smaller outer code order is better at low SNR, but we can see from the results that both cases have high PER (almost 0.5) at low SNR, and that is a nature of all iterative decoders: at low SNR, the packets have many errors and the decoder will not converge, but with increasing the SNR, a falling edge happens and the PER drops to a very low value. Figure 13c,d depicts the effect of the feedback polynomial of the inner code: for low SNR, the inner encoder with non-primitive feedback with factor  $(1 + X)$  has better performance over the inner encoder with primitive feedback, while the primitive feedback inner encoder performs better with high SNR.

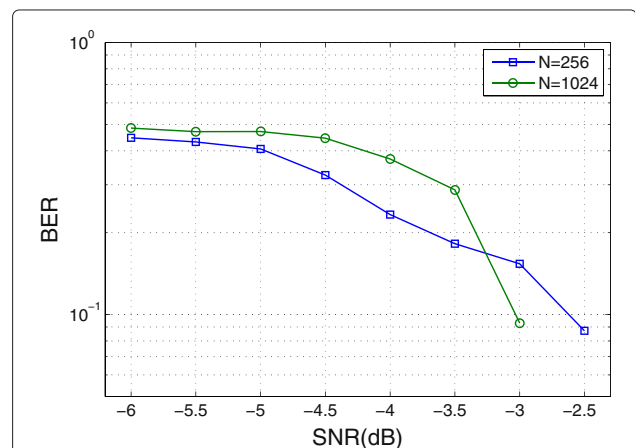
The additional consumption in the network caused by deploying an SCCC encoder and the Viterbi algorithm is divided into the additional processing consumption and the additional transceiver consumption.

- The error control processing power consumption in the network consists of the SCCC encoder consumption at the source node and the Viterbi decoder with the convolutional encoder consumption at the routing nodes. The additional processing consumption in the whole network is as follows:

$$P_{proc} = 2P_{conv} + n(P_{vit} + P_{conv}) \text{ [W/bit]}, \quad (10)$$

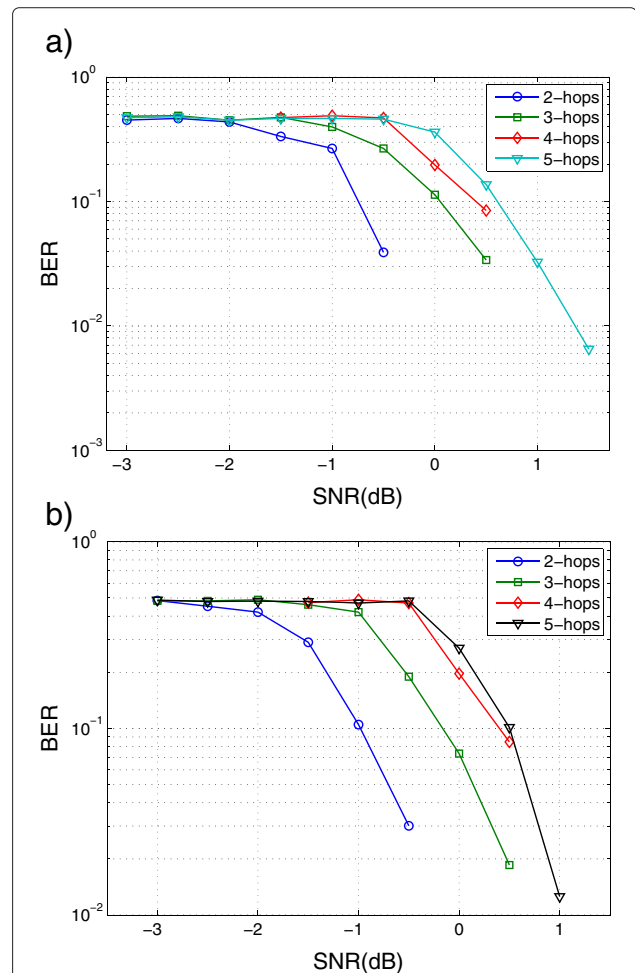
**Table 4 Code polynomials for the inner encoder**

	$K$	Code	Code function	Feedback polynomial state
$G_{11}$	3	5,6,7	(3, [705], [076], [77])	Primitive
$G_{12}$	3	4,7,5	(3, [504], [057], [55])	Non-primitive and $(1 + X)$ factor in feedback polynomial

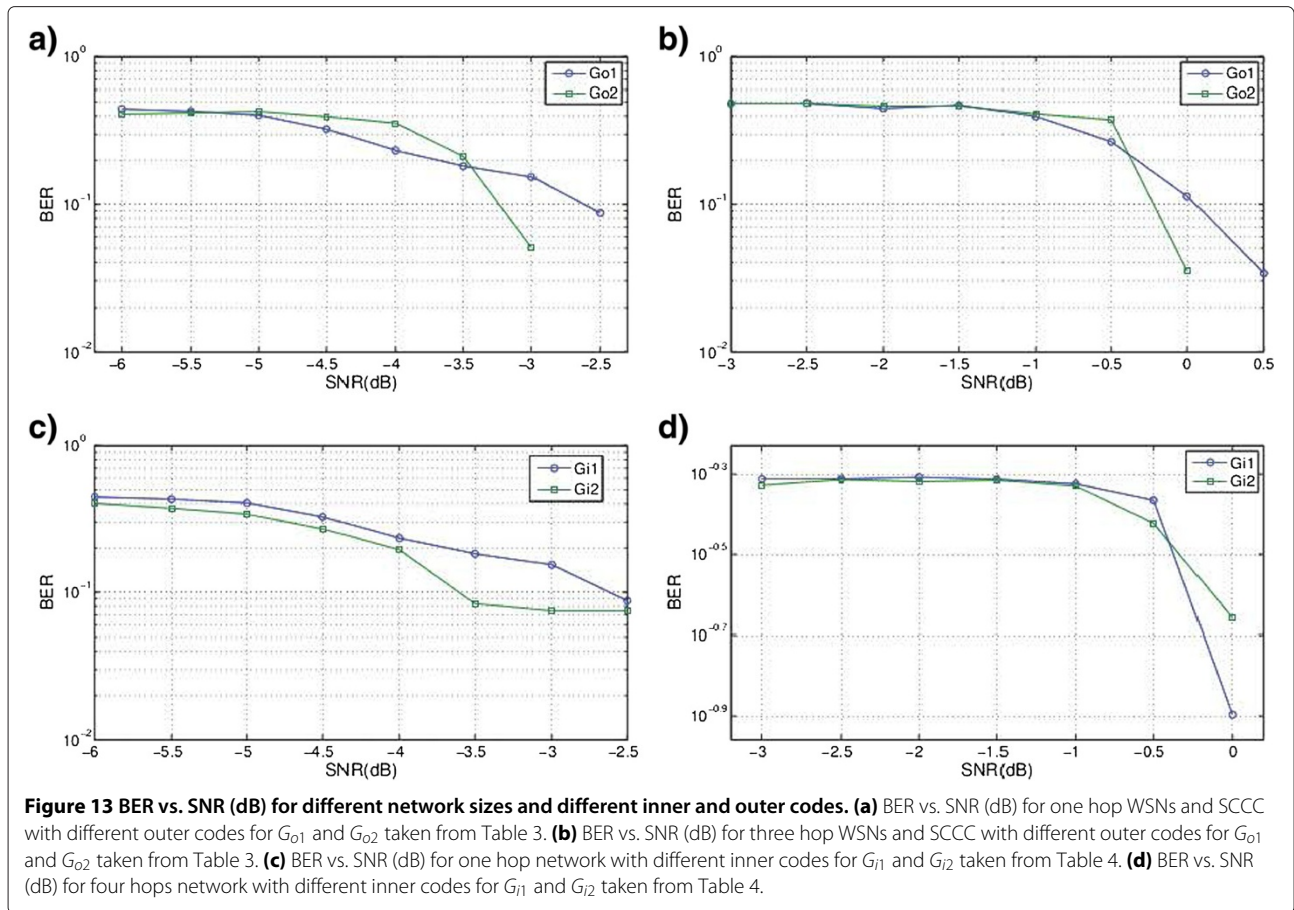


**Figure 11 BER vs. SNR (dB) for 1 hop WSNs with SCCC code.**

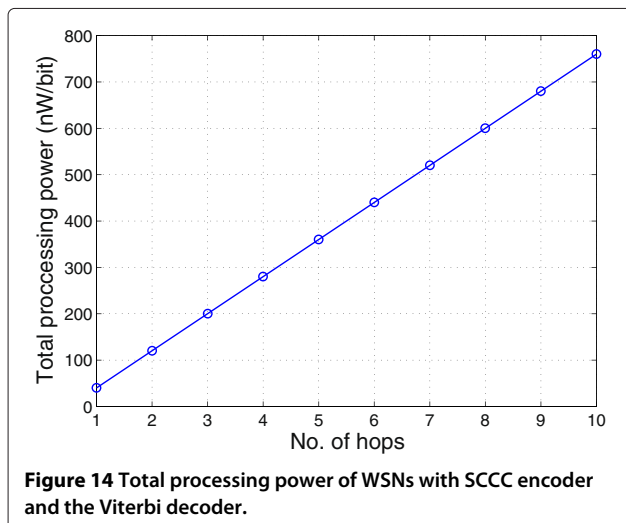
Intermediate node is running Viterbi decoder and network Performance is evaluated at different interleaver memory size.



**Figure 12 BER vs. SNR(dB) for 2, 3, 4 and 5 hop WSNs. (a) BER vs. SNR(dB) for 2, 3, 4 and 5 hop WSNs with SCCC code and the Viterbi decoder at intermediate node with symbol size of 256 bits. (b) BER vs. SNR(dB) for 2, 3, 4 and 5 hop WSNs with SCCC code and Viterbi decoder at intermediate node with symbol size of 1,024 bits.**



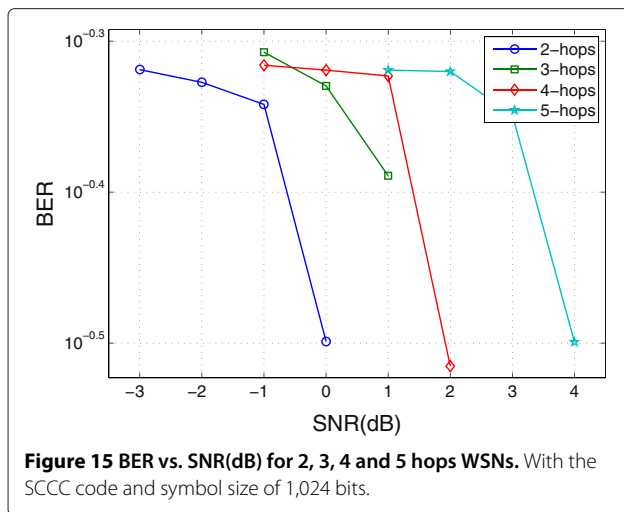
where  $P_{proc}$  is the total processing power consumption,  $P_{conv}$  is the power consumption of the convolutional encoder, and  $P_{vit}$  is the power consumption of the Viterbi decoder. From (6) and Figure 2, we can calculate linear relation between the processing energy consumption at the nodes and the number of hops as shown in Figure 14.



- The increment in energy consumption, caused by the redundant bits in the coded data over uncoded data, is empirically measured in [16], and the results were shown in Table 1. Empirical studies show that transmitting coded data at rate  $\frac{1}{3}$  will increase the transmission energy of the packet by 25%.

### 6.5 BER and power consumption of the WSN with SCCC without decoding at the intermediate nodes

We ran simulations of the WSNs with SCCC code at the source node  $A$  and without decoding the packets at the intermediate nodes. Figure 15 depicts the performance of the network for different number of hops in the communication path of the packet, with turbo decoder running at the base station with symbol size of 1,024 bits. Figure 15 shows that the performance of the SCCC is still superior to ARQ, but the SCCC system performs better when the intermediate nodes run the Viterbi decoding process. For this configuration, the additional energy consumption in the system is only the transceiver power consumption which adds 25% extra energy consumption per packet transmission. By going back to Figure 9 for SNR less than 10 dB, we can see that retransmissions, because of failed packets delivery, for SNR less than 10 dB, there is



a probability of having retransmissions, which means that the SCCC encoder performs better at low SNR.

## 7 Conclusions

In this paper, we considered the problem of delivering collected data from the sensor nodes to the base station on a multi-hop wireless network with noisy channels in an energy and time-efficient way. Error correcting codes are proposed to replace the retransmission mechanism widely deployed in wireless sensor networks in order to tackle the problem of high energy consumption caused by the retransmissions of erroneous packets. Due to the latency and energy consumption of the most error-correcting codes, serial concatenation convolutional codes are suggested for WSNs with their powerful turbo decoding algorithm running at the base station. The intermediate nodes can run the Viterbi algorithm to decode the inner coded data from the source node and encode it again. The Viterbi algorithm running at the intermediate nodes, together with the turbo decoder at the base station, can correct most packets at low SNR. In the WSN with five hops and SCCC encoding with symbol size of 1,024 bits, the turbo decoder decoded all the packets when the SNR is more than 2.5 dB at all the considered channels between the nodes. After adding the Viterbi decoder algorithm, which decodes the inner encoder with constraint length of 3, to the intermediate nodes, the turbo decoder decoded all the packets when the SNR is more than 1 dB for considered all the channels between the nodes. The Viterbi algorithm together with the convolutional inner encoder consumes 80 nW/bit in processing the packet when the constraint length of the inner encoder is 3. The SCCC encoder consumes 40 nW/bit in the encoding process. The redundancy in the coded data increases the transmission energy consumption of a packet by 25%. To improve the performance of the SCCC code, the constraint length of the

inner encoder and the Viterbi decoder can be increased. This improvement comes at the cost of extra processing energy consumption at the intermediate nodes.

## Endnotes

<sup>a</sup>The data rate in WSNs communication is relatively low with BPSK or QPSK modulation, and this leads to a slow fading channel model.

<sup>b</sup>The MICAz is a 2.4 GHz Mote module designed by Moog Crossbow (Milpitas, CA, USA) and it uses ZigBee communication standards [24].

## Competing interests

The authors declare that they have no competing interests.

## Acknowledgement

Part of the work presented in this article was supported by the FWO (Fonds Wetenschappelijk Onderzoek), project G.0219.09N.

Received: 24 May 2013 Accepted: 12 November 2014

Published: 29 November 2014

## References

1. G Halkes, K Langendoen, Crankshaft: an energy-efficient MAC-protocol for dense wireless sensor networks. *Wireless Sensor Networks: 4th European Conference, EWSN 2007, Delft, The Netherlands, January 2007*, 228–244
2. G Anastasi, M Conti, M Di Francesco, A Passarella, Energy conservation in wireless sensor networks: a survey. *Ad Hoc Netw.* **7**(3), 537–568 (2009). doi:10.1016/j.adhoc.2008.06.003
3. WR Heinzelman, A Chandrakasan, H Balakrishnan, in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, vol. 00*. Energy-efficient communication protocol for wireless microsensor networks. (IEEE Comput. Soc., 2000). doi:10.1109/HICSS.2000.926982. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=926982
4. C Ceken, An energy efficient and delay sensitive centralized MAC protocol for wireless sensor networks. *Comput. Stand. Interfaces.* **30**(1–2), 20–31 (2008). doi:10.1016/j.csi.2007.06.001
5. R Cristescu, B Beferull-Lozano, M Vetterli, R Wattenhofer, Network correlated data gathering with explicit communication: NP-completeness and algorithms. *IEEE/ACM Trans. Netw.* **14**(1), 41–54 (2006). doi:10.1109/TNET.2005.863711
6. N Abughalieh, Y-A Le Borgne, K Steenhaut, A Nowe, in *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*. Lifetime optimization for wireless sensor networks with correlated data gathering (Avignon-France, 2010), pp. 266–272. http://ieeexplore.ieee.org/xpl/freeabs\_all.jsp?arnumber=5518801
7. BPS Sahoo, S Rath, D Puthal, Article: Energy efficient protocols for wireless sensor networks: a survey and approach. *Int. J. Comput. App. Foundation of Computer Science, New York.* **44**(18), 43–48 (2012)
8. K Akkaya, M Younis, A survey on routing protocols for wireless sensor networks. *Ad Hoc Netw.* **3**, 325–349 (2005)
9. K Kredon II, P Mohapatra, Medium access control in wireless sensor networks. *Comput. Netw.* **51**(4), 961–994 (2007). doi:10.1016/j.comnet.2006.06.012
10. G Balakrishnan, M Yang, Y Jiang, Y Kim, in *Information Technology, 2007. Fourth International Conference On*. Performance analysis of error control codes for wireless sensor networks, (2007), pp. 876–879. doi:10.1109/ITNG.2007.149
11. J Jeong, CT Ee, in *International Workshop on Wireless Sensor Networks (WWSN), 2007*. Forward error correction in sensor networks, (2007)
12. H Soude, M Agueh, J Mehat, in *Proceedings of the 6th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks - PE-WASUN '09*. Towards an optimal Reed Solomon codes selection for sensor networks (ACM Press New York, USA, 2009).

- doi:10.1145/1641876.1641912. http://portal.acm.org/citation.cfm?doid=1641876.1641912
13. Y Jin, J Chang, D Le, in *2010 International Conference on Computational Intelligence and Software Engineering*. A high energy efficiency link layer adaptive error control mechanism for wireless sensor networks (IEEE, 2010), pp. 1–4. doi:10.1109/CISE.2010.5676934. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=567693423
  14. E Shih, S-h Cho, N Ickes, R Min, A Sinha, A Wang, A Chandrakasan, in *MobiCom '01*. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks, (2001), pp. 272–287. doi:10.1145/381677.381703
  15. ME Pellenz, RD Souza, MSP Fonseca, Error control coding in wireless sensor networks. *Telecommun. Syst.* **44**(1–2), 61–68 (2009). doi:10.1007/s11235-009-9222-5
  16. D Schmidt, M Berning, N Wehn, in *Design, Automation, and Test in Europe Conference and Exhibition*. Error correction in single-hop wireless sensor networks - a case study (Nice, 2009), pp. 1296–1301. doi:1530-1591
  17. LC Zhong, JM Rabaey, A Wolisz, Does proper coding make single hop wireless sensor networks reality: the power consumption perspective. *IEEE Wireless Commun. Netw. Conf.*, 664–669 (2005). doi:10.1109/WCNC.2005.1424587
  18. RM Pyndiah, F Guilloud, K Amis, in *Turbo Codes and Iterative Information Processing (ISTC), 2010 6th International Symposium On*. Multiple source cooperative coding using turbo product codes with a noisy relay, (2010), pp. 98–102. doi:10.1109/ISTC.2010.5613814
  19. R Youssef, I Graell, A Amat, Distributed serially concatenated codes for multi-source cooperative relay networks. *Wireless Commun. IEEE Trans.* **10**(1), 253–263 (2011). doi:10.1109/TWC.2010.102810.100422
  20. T Rappaport, *Wireless Communications: Principles and Practice*, 2nd edn. (Prentice Hall PTR, New Jersey, 2001), pp. 70–79
  21. A Goldsmith, *Wireless Communication*. (Cambridge University Press, New York, 2005), pp. 46–53
  22. S Benedetto, D Divsalar, G Montorsi, F Pollara, Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding. *IEEE Trans. Inform. Theory.* **44**, 909–926 (1998)
  23. S Benedetto, G Montorsi, Design of parallel concatenated convolutional codes. *IEEE Trans. Commun.* **44**(5), 591–600 (1996). doi:10.1109/26.494303
  24. Crossbow Inc, MICAz Wireless Measurement System (2004). http://www.xbow.com. Accessed in November 2011

doi:10.1186/1687-1499-2014-204

**Cite this article as:** Abughalieh et al.: Turbo codes for multi-hop wireless sensor networks with decode-and-forward mechanism. *EURASIP Journal on Wireless Communications and Networking* 2014 **2014**:204.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](http://springeropen.com)

---