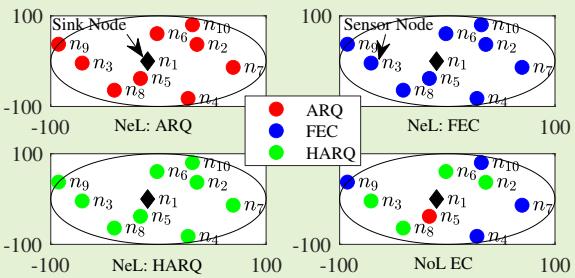
Node-Level Error Control Strategies for Prolonging the Lifetime of Wireless Sensor Networks

Nazli Tekin, Huseyin Ugur Yildiz, *Senior Member, IEEE*, and Vehbi Cagri Gungor

***Abstract*— In Wireless Sensor Networks (WSNs), energy-efficiency and reliability are two critical requirements for attaining a long-term stable communication performance. Using error control (EC) methods is a promising technique to improve the reliability of WSNs. EC methods are typically utilized at the network-level, where all sensor nodes use the same EC method. However, improper selection of EC methods on some nodes in the network-level strategy can reduce the energy-efficiency, thus the lifetime of WSNs. In this study, a node-level EC strategy is proposed via mixed-integer programming (MIP) formulations. The MIP model determines the optimum EC method (*i.e.*, automatic repeat request (ARQ), forward error correction (FEC), or hybrid ARQ (HARQ)) for each sensor node to maximize the network lifetime while guaranteeing a pre-determined reliability requirement. Five meta-heuristic approaches are developed to overcome the computational complexity of the MIP model. The performances of the MIP model and meta-heuristic approaches are evaluated for a wide range of parameters such as the number of nodes, network area, packet size, minimum desired reliability criterion, transmission power, and data rate. The results show that the node-level EC strategy provides at least 4.4% prolonged lifetimes and 4.0% better energy-efficiency than the network-level EC strategies. Furthermore, one of the developed meta-heuristic approaches (*i.e.*, extended golden section search) provides lifetimes within a 3.9% neighborhood of the optimal solutions, reducing the solution time of the MIP model by 89.6%.**



***Index Terms*— network lifetime, error control, mixed-integer programming, meta-heuristics, wireless sensor networks.**

I. Introduction

WIRELESS Sensor Networks (WSNs) are considered a superior technology for industrial applications (*e.g.*, condition monitoring, process automation, and environment sensing [1]). The quality of the wireless channel is harsh and non-stationary in the industrial environment due to different types of interference sources available in these areas (*e.g.*, metal objects, welding equipment, microwave ovens, *etc*. [2]). These harsh environmental conditions cause WSN nodes to dissipate their limited battery energy supplies rapidly, resulting in a degraded network lifetime.

Although the energy-efficiency is critical for most WSNs applications, reliability is also an essential issue in WSNs such that the collected data should be conveyed at the sink node reliably [3]. *Error control* (EC) is a crucial technique for improving the reliability of the network. The two most popular EC methods are the *automatic repeat request* (ARQ) and the *forward error correction* (FEC) [4]. In ARQ, retransmissions are used to guarantee reliability. In FEC, redundant bits are included before transmitting packets, eliminating the need for retransmission. The advantages of both ARQ and FEC can be exploited for further improvements in reliability (*i.e.*, Hybrid ARQ – HARQ).

The energy-efficiency of ARQ (or HARQ) is better than the energy-efficiency of FEC while communicating over short links. Short links have low bit error rates (BER) where the network reliability can be guaranteed using ARQ with low energy overhead since no extra retransmissions would be required when the channel conditions are good. However, as the communication distance gets large, the BER increases. Therefore, retransmissions of ARQ become costly, which diminishes the advantages of ARQ. In this case, FEC codes provide better energy-efficiency than ARQ, although FEC codes incur an extra energy overhead due to the transmission of redundant bits and decoding packets [5].

EC methods are typically applied at the network-level, where the same EC method is used throughout the nodes. In our earlier work, [6], we investigate the performance of ARQ, FEC, and HARQ methods applied at the network-level on WSNs lifetime. Our *motivation* is to further prolong the network lifetime by developing a *node-level EC strategy* such that each node in the network utilizes an optimum EC method (*i.e.*, ARQ, FEC, or HARQ) for maximizing the network

Nazli Tekin is with the Department of Software Engineering, Erciyes University, 38280 Kayseri, Turkey (e-mail: nazlitekin@erciyes.edu.tr).

Huseyin Ugur Yildiz is with the Department of Electrical and Electronics Engineering, TED University, 06420 Ankara, Turkey (e-mail: hugur.yildiz@tedu.edu.tr)

Vehbi Cagri Gungor is with the Department of Computer Engineering, Abdullah Gül University, 38080 Kayseri, Turkey (e-mail: cagri.gungor@agu.edu.tr).

lifetime. Although the node-level EC strategy can be modeled using a mixed-integer programming (MIP) framework, the computational complexity arises from the nature of MIP encourages us to develop meta-heuristic solutions for obtaining reasonable suboptimal solutions in polynomial-time [7].

Our novel contributions are enumerated as follows:

1) We propose a novel network lifetime maximization framework by performing an optimal assignment plan of EC methods at the node-level, called the node-level EC strategies, against the network-level EC strategies, which is an important research topic that has not been comprehensively investigated in the literature before.
2) We develop the node-level EC strategy using MIP formulations for network lifetime maximization. The major advantage of using MIP formulations is to abstract the sub-optimal behaviors of routing protocols that are not relevant to the research subject. In this way, the MIP model enables us to focus on the main benefits of using node-level EC strategy instead of network-level EC strategies for WSNs. The proposed MIP model incorporates the energy and the delay cost models of the three favorite EC methods (*i.e.*, ARQ, FEC, and HARQ) at the link layer. Furthermore, the MIP model utilizes the energy dissipation characteristics of the Mica2 node platform, which is a widely used node platform in WSN research. At the same time, the MIP model considers the harsh channel conditions in industrial environments. Nevertheless, the MIP model is a flexible framework, which can be easily adapted for other application areas by changing the node platform types and wireless channel parameters.
3) Using the MIP model, we quantitatively compare the lifetime performance and energy efficiency of the node-level EC strategy against the network-level EC strategies in WSNs for a wide range of parameter settings, such as the number of nodes, network area, minimum desired reliability criteria, transmission power, packet size, and data rate. Indeed, we aim to determine the improvement in the network lifetime and the energy efficiency of the network-level EC strategies with the node-level EC strategy for the parameter configurations mentioned above.
4) Solving the MIP model for large-scale WSNs, consisting of hundreds of nodes, is time-consuming since MIP models fall into the class of NP-hard problems. To solve the MIP model efficiently for large-scale WSNs without significant deviations from the optimal solutions, we develop five meta-heuristic approaches: Golden Section Search (GSS), Extended GSS (E-GSS), Simulated Annealing (SA), Extended SA (E-SA), and Genetic Algorithm (GA). For each meta-heuristic algorithm, we provide a time complexity analysis and investigate the near-optimal solution performance.

The rest of the paper is organized as follows. Section II provides a summary of previous works. Section III details the channel model, energy & delay models of various EC methods, network & MIP models, and proposed meta-heuristic approaches. Section IV discusses performance evaluation. Finally, Section V concludes the paper.

II. Related Work

In the last decade, EC methods in WSNs have attracted much attention in the literature. A recent survey in [8] provides a comprehensive overview of various EC methods used in WSNs literature. Previous studies on the utilization of EC methods in WSNs generally focus on evaluating the performance of these methods in terms of the energy-efficiency. In [5] and [9], cross-layer analyses of EC schemes are investigated for WSNs and underwater WSNs, respectively. In [10], the energy-efficiency of ARQ is compared to the energy-efficiency of FEC. Similarly, in [11], the energy-efficiency of ARQ, FEC, and HARQ methods are studied for wireless multimedia sensor networks (WMSNs). In [12], several EC methods (*i.e.*, Cyclic Redundancy Check–4 (CRC–4), Bose–Chaudhuri–Hocquenghem (BCH), Reed-Solomon (RS), and convolutional codes (CC)) are compared to HARQ in terms of BER and the energy-efficiency for code-division multiple access (CDMA) WSNs.

In [13], a magnetic induction-based channel coding strategy via BCH is proposed for the optimal energy planning in linear oil sensor networks. In [14], the effects of error-correcting capabilities of BCH and RS codes on the characteristic distance (*i.e.*, the single-hop distance where the energy consumption is minimized) of WSNs are investigated. In [15], a novel EC method, based on RS codes, CCs, and their concatenated codes, is introduced for attaining the energy-efficiency in cyber-physical systems-based smart buildings. In [16], an adaptive FEC scheme based on RS codes is proposed for solar-powered WSNs. In [17], a cognitive radio sensor networks-assisted HARQ method is developed for solving the inefficient utilization of the primary user channel. In [18], a design methodology of a reliable low-power WSNs is presented, where FEC, blind retransmissions, and ARQ are utilized to target a predetermined reliability criterion.

Sarvi *et al.* [19] develop an adaptive cross-layer EC protocol, which uses erasure coding as the FEC code at the application layer and the hybrid RS/ARQ method at the link layer for reliable energy-efficient multimedia streaming over WSNs. Another adaptive cross-layer EC method is introduced in [20] for distributed video coding over WMSNs. The erasure coding with RS and hybrid RS/ARQ methods are utilized at the application and the link layers. In [21], an EC framework that exploits the features of BCH and low-density parity-check (LDPC) codes is proposed for software-defined WSNs. This framework allows different types of FEC methods to be deployed in various links or clusters of WSNs to reduce the energy consumption of the sensor nodes. In [22], the advantages of turbo codes, a class of high-performance FEC codes, over ARQ are discussed for WSNs having a multi-hop extended star topology. In [23], a comparative study is carried out for determining the most suitable EC method among convolutional, turbo, and LDPC codes for industrial WSNs. In [24], a dynamic EC method is proposed that estimates the channel conditions and controls the errors. For low BERs, a simple EC method is used without transmissions, while for

high BERs, retransmissions are considered. In [25], a flexible EC protocol that adaptively changes the utilized EC code depending on the wireless channel conditions is developed for WSN-based smart grid applications. The performance of this adaptive EC protocol is compared to static RS and without-FEC mechanisms. In [26], the capabilities of HARQ are extended via estimating the network conditions for optimizing the energy-efficiency of CDMA-based WSNs. In [27], a two-dimensional single error correction and double error detection (2D SEC-DED) FEC method is introduced, which provides good performance when BER is high, and most of the errors are single-/double-bit.

Ez-Zazi *et al.* [28] propose two energy-efficient coding frameworks for clustered WSNs. In the first approach, the cluster head decides to correct the packets using an LDPC-based FEC code. In the second method, the cluster head decides to decode the received packets or verify the integrity of the packets using ARQ. The first and the second approaches are suitable when BER is low and high, respectively. In [29], an energy-efficient coding framework is developed for multi-hop WSNs. The developed method decides when to use FEC decoding or retransmissions by considering the propagation characteristics of the wireless channel.

The problem of maximizing the WSN lifetime can be formulated as an optimization problem [30]. One of the most popular optimization models to maximize the WSN lifetime is *network flow programming*. In network flow programming, the aim is to find an optimum data-routing plan for sensor nodes to deliver their collected data to the sink node in a manner that maximizes the network lifetime [31]. This problem can be formulated as a linear program (LP) where the objective function is defined as the maximization of WSN lifetime while constraints ensure both the flow balance and the energy balance [32]. Although these LP models usually assume fractional flows (splitting of packets into fractional portions) as decision variables, in practice, these models require flow decision variables to have integer values since flow variables are typically defined as the number of packets to be transmitted over links [33]. Network flow programming models are treated as either integer programs (IP) or MIP (depending on the usage of another binary variable or continuous variable) and no longer have a polynomial-time solution.

There has been little quantitative analysis on WSNs lifetime maximization via MIP formulations by utilizing EC methods in the literature. In [38], an MIP formulation is proposed to optimize the network lifetime and the video encoding rate for wireless video sensor networks. A hybrid EC scheme that integrates network coding and ARQ is employed to combat packet losses over wireless channels. In [39], a hybrid ARQ protocol is developed via an MIP model that maximizes WSN lifetime while the MIP model has a constraint to satisfy a pre-determined reliability criterion. In [40], a green WSN lifetime is maximized using MIP formulations incorporating a novel redundant residue number system-based EC method.

The computational complexity of MIP models can be reduced by heuristic methods that can find near-optimal solutions in a reasonable amount of time. A particular class of heuristics is called *meta-heuristics*, which are nature-inspired (based on some principles adopted from physics, biology, *etc.*) and problem-independent algorithms to approximately solve a wide range of challenging optimization problems [41]. Golden section search (GSS), genetic algorithm (GA), and simulated annealing (SA) are three widespread instances of meta-heuristics algorithms. Recent surveys in [42], [43] study the techniques for improving the network lifetime while attaining the energy-efficiency using meta-heuristics in WSNs. The most similar work to our study is performed by Xenakis *et al.* [36], where the authors jointly optimize the transmit power, packet transmission through EC methods, and topology control in WSNs for the energy-efficiency. The optimization problem is solved using SA meta-heuristic. However, EC methods are applied at the network-level rather than at the node-level. Moreover, the MIP model minimizes the energy consumption of the network instead of directly maximizing the network lifetime, as we aim in this study.

To the best of our knowledge, a study that solves WSNs lifetime maximization problem for the node-level EC strategy (which is modeled as MIP) via meta-heuristic methods has not been encountered in the literature yet. Moreover, the existing body of research on the utilization of EC methods for the WSN lifetime maximization (*i.e.*, [38]–[40]) generally adopt network-level strategies and provide comparative analyses on the energy-efficiency for various EC methods. Therefore, this study makes two major contributions to research on the utilization of EC methods in WSNs. First, a novel node-level EC strategy is formulated as an MIP model such that each node in the network can adopt an optimum EC method for maximizing the network lifetime. Second, several meta-heuristic methods are developed to solve this MIP model (*i.e.*, node-level EC strategy) to get near-optimal solutions in polynomial-time.

III. System Model

In this section, we first introduce the channel model. Then, energy and delay cost models of various EC methods are detailed where these models are adopted from [6] that are derived from the analyses provided in [5], [36]. Afterward, we present our network model and the node-level EC strategy. Finally, we describe our meta-heuristic approaches. Table I provides the terminology used in this paper.

A. Channel Model

We utilize the log-normal shadowing channel model detailed in [34]. The path loss over the link-(i,j) is defined as

$$\overline{\mathrm{PL}_{ij}} = \overline{\mathrm{PL}_0} + 10\eta \log_{10}(d_{ij}/d_0) + \overline{X_\sigma}, \tag{1}$$

where d_{ij} and d_0 refer to the distance of link-(i,j) and the reference distance, respectively. $\overline{\mathrm{PL}_0}$ is the path loss at d_0. η is the path loss exponent and $\overline{X_\sigma}$ models the shadowing phenomena, which is defined as a zero-mean Normal random variable with the standard deviation σ.

The signal-to-noise ratio (SNR) at the receiver node-j (where the transmitter node is defined as node-i) is [34]

$$\overline{\gamma_{ij}} = \overline{P_t} - \overline{\mathrm{PL}_{ij}} - \overline{P_n}, \tag{2}$$

TABLE I: Terminology used in this paper.

Symbol	Description	Unit	Value	Symbol	Description	Unit	Value
BER_{ij}	Bit error rate observed over link-(i,j)	–	–	BLER_{ij}	Block error rate observed over link-(i,j)	–	–
B_N	Noise bandwidth [34]	kHz	30	c	Speed of light in vacuum	m/s	3×10^8
d_0	Reference distance [35]	m	15	d_{ij}	Distance between node-i and node-j	m	–
$\bar{E}$	Avg. energy consumption per node per round	J	–	E_{acq}	Data acquisition energy [6]	μJ	600
E_{dec}	Energy cost of decoding a FEC code	J	–	E_{elec}	Energy consumed by the tx. electronics [6]	nJ/bit	50
E_{enc}	Energy cost of encoding [36]	J	≈ 0	E_i	Total energy consumed by node-i	J	–
E_{ini}	Initial battery energy of sensor nodes [6]	KJ	25	$E^k_{rx,ji}$	Reception energy cost for $k=\{$ARQ, FEC, HARQ$\}$	J	–
$E^k_{tx,ij}$	Transmission energy cost for $k=\{$ARQ, FEC, HARQ$\}$	J	–	$E_{rx}(q)$	Energy required to receive q bit-packet	J	–
$E_{tx}(q)$	Energy required to transmit q bit-packet	J	–	E^{to}_{ij}	Energy consumed before timeout	J	–
H	Network lifetime	rounds	–	$\mathcal{K}$	Set of EC methods	–	–
ℓ	Number of bits transmitted after a NACK packet has been received in HARQ	bits	–	$l_a(l_{na})$	ACK (NACK) packet size [6]	bytes	20
l^k_{pck}	Packet size for $k=\{$ARQ, FEC$\}$	bytes	–	l_h	Header size [6]	bytes	12
l_{pl}	Payload size	bytes	32–128	$n^k_{\text{ret},ij}$	Expected number of retransmissions for $k=\{$ARQ, FEC, HARQ$\}$	–	–
$\mathcal{P}$	Transmission power level	–	1–26	$\text{PER}^k_{ij}(q)$	Packet (of q bits) error rate for $k=\{$ARQ, FEC$\}$	–	–
$\overline{\text{PL}_{ij}}$	Path loss over the link-(i,j)	dB	–	$\overline{\text{PL}_0}$	Reference path loss [35]	dB	61.65
$\overline{P_n}$	Noise floor [34]	dBm	–115	P_{proc}	Processing power [5]	mW	24
P_{slp}	Sleep power [37]	μW	3	P_{std}	Standby power [37]	mW	35.4
P_{rx}	Reception power [37]	mW	35.4	$\overline{P_t}$	Output antenna power [37]	dBm	(–20)–(+5)
P_{tx}	Transmission power [37]	mW	25.8–76.2	R	Data rate [34]	kbps	19.2–38.4
R_{com}	Max. transmission range [37]	m	19.3–82.92	R_{net}	Network radius	m	10–40
s_i	Number of packets generated by node-i in each round	–	1	T_{acq}	Time required for data acquisition [6]	ms	20
T_{add}	Time required to perform addition in the CPU of the microcontroller	s	–	T^{blck}_{dec}	Time required to decode a FEC block-(n,k,t)	s	–
T^{bsy}_i	Total busy time of sensor node-i	s	–	t_{cyc}	Cycle duration [5]	ns	250
T_{dec}	Decoding time of a FEC code	s	–	T_{grd}	Guard time	s	–
T_{mult}	Time required to perform multiplication in the CPU of the microcontroller	s	–	T^p_{ij}	Round-trip propagation delay over link-(i,j)	s	–
T_{rnd}	Round duration	s	100	$T^k_{rx,ji}$	Reception time for $k=\{$ARQ, FEC, HARQ$\}$	s	–
$T^k_{tx,ij}$	Transmission time for $k=\{$ARQ, FEC, HARQ$\}$	s	–	V	Set of all nodes (including the sink node)	–	–
$\lvert V\rvert$	Number of all nodes (incl. the sink node)	–	101–221	W	Set of all sensor nodes (excl. the sink node)	–	–
$\lvert W\rvert$	Number of sensor nodes	–	100–220	x^k_{ij}	Number of packets flowing over the link-(i,j) using type-k EC method	–	–
$\overline{X_\sigma}$	Zero-mean Gaussian random variable with std. deviation, σ, to represent shadowing [35]	dB	7.35	$\overline{\gamma_{ij}}$	SNR at the receiver node-j	dB	–
η	Path loss exponent [35]	–	2.49	ϕ^k_{ji}	Loss rate of the received packets for $k=\{$ARQ, FEC, HARQ$\}$	–	–
ψ	Minimum desired packet delivery ratio	–	0.5–0.999	(n,k,t)	Block length, payload length, and error correction capability for FEC codes	bits	–

where $\overline{P_t}$ and $\overline{P_n}$ are the output antenna power and the noise floor (both in dB), respectively. For the non-coherent frequency-shift keying (FSK) modulation scheme (which is the modulation used in Mica2 platforms), the BER is [34]

$$\text{BER}_{ij} = 0.5 \exp\left(-0.5 \times 10^{0.1 \times \overline{\gamma_{ij}}} \times (B_N/R)\right), \quad (3)$$

where B_N is the noise bandwidth and R denotes the data rate. The packet error rate (PER) for ARQ is defined as [5]

$$\text{PER}^{\text{ARQ}}_{ij}(q) = 1 - (1 - \text{BER}_{ij})^q, \quad (4)$$

where q is the size of the packet (in bits). For a FEC block-(n,k,t), the block error rate is calculated as [5]

$$\text{BLER}_{ij} = \sum_{l=t+1}^{n} \binom{n}{l} (\text{BER}_{ij})^l (1 - \text{BER}_{ij})^{n-l}, \quad (5)$$

where n, k, and t denote the block length, the payload length, and the capability of error correction (all in bits), respectively. For a packet of size q bits (assuming that $q > n$), the PER of FEC can be estimated as [5]

$$\text{PER}^{\text{FEC}}_{ij}(q) = 1 - (1 - \text{BLER}_{ij})^{\lceil \frac{q}{k} \rceil}, \quad (6)$$

where $\lceil \frac{q}{k} \rceil$ is the number of blocks required to send q bits and $\lceil \cdot \rceil$ is the ceil function.

B. Energy Cost Models

1) ARQ: In ARQ, we consider a two-way handshake mechanism where the receiver node-j sends an acknowledgment (ACK) packet as soon as this node correctly receives the data packet transmitted by node-i [9]. The energy consumed for transmitting a data packet of size $l^{\text{ARQ}}_{pck} = l_{pl} + l_h$ (where l_{pl} and l_h are the payload and the header sizes, respectively) and receiving an ACK packet of size l_a is [6]

$$E^{\text{ARQ}}_{tx,ij} = E_{enc} + n^{\text{ARQ}}_{\text{ret},ij} \times \left[E_{tx}\left(l^{\text{ARQ}}_{pck}\right) + E_{rx}(l_a) + E^{to}_{ij}\right], \quad (7)$$

where E_{enc}, $E_{tx}\left(l^{\text{ARQ}}_{pck}\right)$, $E_{rx}(l_a)$, E^{to}_{ij}, and $n^{\text{ARQ}}_{\text{ret},ij}$ are the encoding energy cost, the data transmission energy cost, the ACK reception energy cost, the energy required before timeout, and the expected number of retransmissions, respectively. Since E_{enc} is negligible small, we neglect this term [36]. For a packet of size q bits, the transmission and the reception energy costs are expressed as [9]

$$\begin{aligned} E_{tx}(q) &= qE_{elec} + P_{tx}\,(q/R)\,, \\ E_{rx}(q) &= qE_{elec} + P_{rx}\,(q/R)\,, \end{aligned} \quad (8)$$

respectively, where E_{elec}, P_{tx}, and P_{rx} denote the energy consumed by the transmitter electronics, the transmission power, and the reception power, respectively. The Mica2 node platform has 26 discrete power levels (*i.e.*, $\mathcal{P}$), where $\overline{P_t}$, P_{tx}, and the corresponding maximum transmission range values (*i.e.*, R_{com}) for each power level are reported in Table II [37]. The energy required before timeout, E^{to}_{ij}, is formulated as $E^{to}_{ij} = P_{std} \times (T^p_{ij} + T_{grd})$ [6]. In this equation, P_{std}, $T^p_{ij} = 2 \times d_{ij}/c$, and $T_{grd} = 2 \times \max_{i,j} T^p_{ij}$ are the standby power, the round-trip propagation delay, and the guard time, respectively. Note that, the guard time is as twice as the maximum value of round-trip propagation delay [44]. c is the

TABLE II: Output antenna power ($\overline{P_t}$ in dBm), transmission power (P_{tx} in mW), and max. transmission range (R_{com} in m) for each power level ($\mathcal{P}$) for the Mica2 node platforms [37].

$\mathcal{P}$	$\overline{P_t}$	P_{tx}	R_{com}	$\mathcal{P}$	$\overline{P_t}$	P_{tx}	R_{com}
1	–20	25.8	19.30	14	–7	32.4	41.19
2	–19	26.4	20.46	15	–6	33.3	43.67
3	–18	27.0	21.69	16	–5	41.4	46.29
4	–17	27.0	22.99	17	–4	43.5	49.07
5	–16	27.3	24.38	18	–3	43.5	52.01
6	–15	27.9	25.84	19	–2	45.3	55.13
7	–14	27.9	27.39	20	–1	47.4	58.44
8	–13	28.5	29.03	21	0	50.4	61.95
9	–12	29.1	30.78	22	1	51.6	65.67
10	–11	29.7	32.62	23	2	55.5	69.61
11	–10	30.3	34.58	24	3	57.6	73.79
12	–9	31.2	36.66	25	4	63.9	78.22
13	–8	31.8	38.86	26	5	76.2	82.92

speed of light in vacuum. $n_{\text{ret},ij}^{\text{ARQ}}$ is calculated as [6]

$$n_{\text{ret},ij}^{\text{ARQ}} = \left[\left(1 - \text{PER}_{ij}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right)\right) \times \left(1 - \text{PER}_{ji}^{\text{ARQ}}\left(l_a\right)\right)\right]^{-1}. \tag{9}$$

The energy cost for the reception in ARQ is defined as [6]

$$E_{rx,ji}^{\text{ARQ}} = n_{\text{ret},ji}^{\text{ARQ}} \times \left[E_{rx}\left(l_{pck}^{\text{ARQ}}\right) + E_{tx}(l_a)\right]. \tag{10}$$

2) FEC: Redundancy bits are added before the transmission in FEC methods. Since FEC methods do not require retransmissions, ACK packets are not required. The size of a FEC packet is calculated as $l_{pck}^{\text{FEC}} = \lceil \frac{l_{pl}}{k} \rceil n + l_h$ [9]. Energy costs for transmission and reception for FEC are [6]

$$\begin{aligned} E_{tx}^{\text{FEC}} &= E_{enc} + E_{tx}\left(l_{pck}^{\text{FEC}}\right), \\ E_{rx}^{\text{FEC}} &= E_{rx}\left(l_{pck}^{\text{FEC}}\right) + E_{dec}, \end{aligned} \tag{11}$$

respectively, where $E_{dec} = P_{proc} \times T_{dec}$ is the energy consumption for decoding a FEC packet [36]. In this equation, P_{proc} is the processing power and T_{dec} is the decoding time for a FEC packet, which is calculated in Section III-C.2.

3) HARQ: A data packet of size l_{pck}^{ARQ} is sent with ARQ, and if this packet is replied with a negative ACK packet (NACK) of size $l_{na} = l_a$, then the transmitter re-sends the data encoded by FEC (*i.e.*, HARQ-I) or the redundant bits (*i.e.*, HARQ-II). From a general perspective, we define ℓ to be the number of bits transmitted after a NACK packet has been received [5]. Therefore, $\ell = l_{pck}^{\text{FEC}}$ for HARQ-I and $\ell = \lceil \frac{l_{pl}}{k} \rceil (n-k) + l_h$ for HARQ-II. As in [5], we choose BCH as the FEC code utilized for both HARQ-I and HARQ-II. The transmission and the reception energy costs for HARQ are calculated as [6]

$$\begin{aligned} E_{tx,ij}^{\text{HARQ}} &= E_{enc} + E_{tx}\left(l_{pck}^{\text{ARQ}}\right) + \text{PER}_{ij}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right) \\ &\quad \times \left[E_{rx}(l_{na}) + E_{tx}(\ell)\right], \\ E_{rx,ji}^{\text{HARQ}} &= E_{rx}\left(l_{pck}^{\text{ARQ}}\right) + \text{PER}_{ji}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right) \\ &\quad \times \left[E_{tx}(l_{na}) + E_{rx}(\ell) + E_{dec}\right]. \end{aligned} \tag{12}$$

C. Delay Cost Models

1) ARQ: Since ARQ employs a two-way handshake mechanism, the transmission time includes the time spent for sending a data packet of size l_{pck}^{ARQ} (*i.e.*, l_{pck}^{ARQ}/R) plus the round-trip propagation delay (T_{ij}^{p}) and the guard time (T_{grd}) that is the time spent for waiting for an ACK packet. On the other hand, the reception time includes the time spent for receiving the data packet of size l_{pck}^{ARQ}. Hence, the transmission and the reception times for ARQ are calculated as [6]

$$\begin{aligned} T_{tx,ij}^{\text{ARQ}} &= \left(l_{pck}^{\text{ARQ}}/R\right) + T_{ij}^{p} + T_{grd}, \\ T_{rx}^{\text{ARQ}} &= \left(l_{pck}^{\text{ARQ}}/R\right). \end{aligned} \tag{13}$$

2) FEC: Due to the lack of ACK packets in FEC, the transmission and the reception times are expressed as [6]

$$\begin{aligned} T_{tx,ij}^{\text{FEC}} &= \left(l_{pck}^{\text{FEC}}/R\right) + T_{ij}^{p}, \\ T_{rx}^{\text{FEC}} &= \left(l_{pck}^{\text{FEC}}/R\right) + T_{dec}, \end{aligned} \tag{14}$$

respectively. The decoding time for a FEC block-(n,k,t) is calculated as $T_{dec}^{blck} = (2nt + 2t^2) \times (T_{add} + T_{mult})$ [5], where T_{add} and T_{mult} stand for the times required to perform addition and multiplication in the central processing unit (CPU) of the microcontroller, respectively. Note that, Mica2 sensor node platform with 8-bit microcontroller operates addition and multiplication of 8 bits in 3 cycles where $m = \lfloor \log_2 n + 1 \rfloor$ [5]. Therefore, $T_{add} + T_{mult} = 3 \times \lceil m/8 \rceil \times t_{cyc}$, where t_{cyc} is the one cycle period [5]. Since $\lceil l_{pl}/k \rceil$ number of FEC blocks are received, the total decoding time for a FEC packet is calculated as $T_{dec} = T_{dec}^{blck} \times \lceil l_{pl}/k \rceil$ [6].

3) HARQ: In HARQ, the first transmission is performed using ARQ, while the second transmission is achieved using a BCH FEC code. Thus, the transmission and reception times are defined as [6]

$$\begin{aligned} T_{tx,ij}^{\text{HARQ}} &= T_{tx,ij}^{\text{ARQ}} + \text{PER}_{ij}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right) \times T_{tx,ij}^{\text{FEC}}, \\ T_{rx,ji}^{\text{HARQ}} &= T_{rx}^{\text{ARQ}} + \text{PER}_{ji}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right) \times T_{rx}^{\text{FEC}}. \end{aligned} \tag{15}$$

D. Network Model & The Node-Level EC Strategy

1) Network Model: We consider a disk-shaped network of radius R_{net} m. The network consists of a single sink node (*i.e.*, node-1), which is located in the middle of the network. The sets V and W are defined to represent sets of all nodes (including the sink node) and all sensor nodes, respectively. $|W|$ sensor nodes are uniformly distributed within the network. We define the network lifetime as the time until the first node exhausts its battery [30]. The network lifetime is organized as *rounds*, where T_{rnd} and H are defined as the round duration and the network lifetime (in rounds), respectively. s_i is defined as the number of packets generated by node-i at each round. Generated data are collected at the sink node at the end of each round.

2) MIP Formulation for the Node-Level EC Strategy: The node-level EC strategy is developed as a *network flow program* using MIP formulations in (16). The objective function of the MIP model is defined as the maximization of the network lifetime (*i.e.*, H) in (16a). The set $\mathcal{K}$ consists of the EC methods used throughout this work, where $\mathcal{K} = \{$ARQ, BCH(31,11,5), BCH(31,21,5), RS(15,11,2), HARQ-I, HARQ-II$\}$. As in [6], we choose BCH(31,11,5), BCH(31,21,5), and

RS(15,11,2) as FEC codes. x_{ij}^k, T_i^{bsy}, and E_i are the decision variables of the MIP model, where x_{ij}^k is an integer variable that denotes the number of packets flowing over link-(i,j) using type-k EC method through the network lifetime. T_i^{bsy} is a continuous variable that represents the total busy time of sensor node-i during the network lifetime. E_i is the total energy consumed by node-i throughout the network lifetime. The constraints of the MIP model are defined in (16b)–(16k).

The first constraint, (16b), ensures the flow balance at each sensor node-i, which states that the number of outgoing packets (*i.e.*, $\sum_{k\in\mathcal{K}}\sum_{j\in V} x_{ij}^k$) is the summation of incoming packets (*i.e.*, $\sum_{k\in\mathcal{K}}\sum_{j\in W} x_{ji}^k\phi_{ji}^k$) and the generated packets during the network lifetime (*i.e.*, $H\times s_i$).

$$\text{Maximize } H \tag{16a}$$

subject to:

$$\sum_{k\in\mathcal{K}}\sum_{j\in V} x_{ij}^k - \sum_{k\in\mathcal{K}}\sum_{j\in W} x_{ji}^k\phi_{ji}^k = H\times s_i, \forall i\in W \tag{16b}$$

$$\sum_{k\in\mathcal{K}}\sum_{j\in W} x_{j1}^k\phi_{j1}^k \geq \left(\sum_{i\in W} s_i\right)\times H\times\psi, \tag{16c}$$

$$T_i^{bsy} = \sum_{k\in\mathcal{K}}\left(\sum_{j\in V} n_{\text{ret},ij}^k T_{tx,ij}^k x_{ij}^k + \sum_{j\in W} n_{\text{ret},ji}^k T_{rx,ji}^k x_{ji}^k\right) + H\times T_{acq}, \forall i\in W \tag{16d}$$

$$T_i^{bsy} \leq H\times T_{rnd}, \forall i\in W \tag{16e}$$

$$\sum_{k\in\mathcal{K}}\left(\sum_{j\in V} E_{tx,ij}^k x_{ij}^k + \sum_{j\in W} E_{rx,ji}^k x_{ji}^k\right) + H\times E_{acq} + P_{slp}\times\left(H\times T_{rnd} - T_i^{bsy}\right) = E_i, \forall i\in W \tag{16f}$$

$$E_i \leq E_{ini}, \forall i\in W \tag{16g}$$

$$x_{ij}^k = 0 \text{ if } d_{ij} > R_{com}, \forall i\in W, \forall j\in V, \forall k\in\mathcal{K} \tag{16h}$$

$$x_{ij}^k \geq 0, \forall i\in W, \forall j\in V, \forall k\in\mathcal{K} \tag{16i}$$

$$T_i^{bsy} \geq 0, \forall i\in W \tag{16j}$$

$$E_i \geq 0, \forall i\in W \tag{16k}$$

The term, ϕ_{ji}^k, refers to the *loss rate* of the received packets, and calculated as [6]

$$\begin{aligned}
\phi_{ji}^{\text{ARQ}} &= 1 - \left[\text{PER}_{ji}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right)\times\text{PER}_{ji}^{\text{ARQ}}(l_a)\right]^{\left(n_{\text{ret},ji}^{\text{ARQ}}+1\right)},\\
\phi_{ji}^{\text{FEC}} &= 1 - \text{PER}_{ji}^{\text{FEC}}\left(l_{pck}^{\text{FEC}}\right),\\
\phi_{ji}^{\text{HARQ}} &= 1 - \left[\text{PER}_{ji}^{\text{ARQ}}\left(l_{pck}^{\text{ARQ}}\right)\times\text{PER}_{ji}^{\text{FEC}}\left(l_{pck}^{\text{FEC}}\right)\right],
\end{aligned} \tag{17}$$

for ARQ, FEC, and HARQ, respectively. (16c) is the reliability requirement constraint, which states at least ψ ratio of the generated packets from all sensor nodes during the network lifetime (*i.e.*, $(\sum_{i\in W} s_i)\times H\times\psi$) should be received by the sink node (node-1). Indeed, this constraint ensures that the desired packet delivery ratio (PDR) of the network should be at least ψ. (16d) calculates the total time required for the transmission, the reception, and the data acquisition (*i.e.*, $H\times T_{acq}$) of node-i during the network lifetime (busy time), where T_{acq} is the time required for data acquisition at each round. For FEC and HARQ, we set $n_{\text{ret},ij}^{\text{FEC}} = n_{\text{ret},ij}^{\text{HARQ}} = 1$ [36]. On the other hand, for ARQ, $n_{\text{ret},ij}^{\text{ARQ}}$ has already been derived in Eq. (9). The parameters, $T_{tx,ij}^k$ and $T_{rx,ji}^k$, are calculated for ARQ, FEC, and HARQ in Section III-C. (16e) states that the busy time of each sensor node cannot exceed the network lifetime in terms of seconds. (16f) calculates the total energy dissipated of node-i for the communication, the data acquisition (*i.e.*, $H\times E_{acq}$), and staying in the sleep mode (*i.e.*, $P_{slp}\times\left(H\times T_{rnd} - T_i^{bsy}\right)$. In this constraint, E_{acq} and P_{slp} are the energy consumption of data acquisition at each round and the sleep power, respectively. The parameters, $E_{tx,ij}^k$ and $E_{rx,ji}^k$, are defined for ARQ, FEC, and HARQ in Section III-B. (16g) limits the total energy consumed by node-i to the initial battery capacity of sensor nodes (E_{ini}). (16h) enforces the maximum transmission range constraint. Finally, (16i)–(16k) define the boundaries of the decision variables.

E. Meta-Heuristic Approaches

We solved the MIP model in (16) to optimality for small-sized randomly generated WSNs. We observe that sensor nodes mostly utilize HARQ-II, HARQ-I, and BCH(31,21,5) to maximize lifetime. Since HARQ-II has better energy-efficiency than HARQ-I, and BCH(31,21,5) is chosen as the FEC method for HARQ, our problem has turned into a *binary selection problem* [45] (*i.e.*, which nodes should use HARQ-II or BCH(31,21,5) to maximize the network lifetime). We define a *binary solution vector* $\mathbb{S} = [\mathcal{S}_2, \mathcal{S}_3, \ldots, \mathcal{S}_{|V|}]$ that is used as a baseline throughout this section. Note that, $\mathcal{S}_i = 1$, if node-i uses HARQ-II, and $\mathcal{S}_i = 0$, if node-i uses BCH(31,21,5). In the remaining part of this subsection, we detail our five meta-heuristic approaches.

1) Golden Section Search (GSS): The pseudo-code for the GSS algorithm is presented in Alg. 1. The *inputs* are $|V|$, `lb`, `ub`, and Φ, which denote the number of all nodes in the network, the lower & upper bounds of the search interval, and the golden ratio, respectively. Initially, we set `lb` $= 0$ and `ub` $= |V| - 1$. We define two interior points, λ_1 and λ_2 (*i.e.*, `lb` $< (\lambda_1, \lambda_2) <$ `ub`), as boundaries for the nodes, which are using HARQ-II and BCH(31,21,5) (line 1). In other words, from node-2 up to node-(λ_1+1), HARQ-II is utilized (*i.e.*, $\mathcal{S}_i = 1,\ \forall i \in \{2,3,\ldots,\lambda_1+1\}$), while the rest of the sensor nodes use BCH(31,21,5) (*i.e.*, $\mathcal{S}_i = 0,\ \forall i \in \{\lambda_1+2,\ldots,|V|\}$). The same procedure is also applied for λ_2. Then, the network lifetime is computed according to λ_1 as follows. In (16), we manually set $x_{ij}^k = 0,\ \forall i,\ \forall j,\ \forall k \in \{\text{ARQ},\ \text{BCH(31,11,5)},\ \text{RS(15,11,2)},\ \text{HARQ-I}\}$ since ARQ, BCH(31,11,5), RS(15,11,2), and HARQ-I are not preferred by the sensor nodes. Moreover, we set $x_{ij}^{\text{HARQ-II}} = 0,\ \forall i \in \{\lambda_1+2,\ldots,|V|\},\ \forall j$ and $x_{ij}^{\text{BCH(31,21,5)}} = 0, \forall i \in \{2,\ldots,\lambda_1+1\},\ \forall j$. Since the search space of x_{ij}^k is reduced by this manual operation, we can solve (16) in a reasonable time, and the lifetime is obtained as H_1 (line 3). A similar procedure is also applied for λ_2, where the lifetime obtained for this case is denoted as H_2 (line 4). While searching for the maximum lifetime, the GSS algorithm updates the interval according to

Algorithm 1 Pseudo-code for the GSS algorithm.

Input: Number of all nodes ($|V|$), Lower bound ($\texttt{lb} \leftarrow 0$), Upper bound ($\texttt{ub} \leftarrow |V|-1$), Golden ratio ($\Phi \leftarrow (-1+\sqrt{5})/2$).

Output: Lifetime, nodes using HARQ-II or BCH(31,21,5).

1: Compute $\lambda_1 \leftarrow \lceil(\texttt{ub}-\Phi\times(\texttt{ub}-\texttt{lb})\rceil$ and $\lambda_2 \leftarrow \lfloor(\texttt{lb}+\Phi\times(\texttt{ub}-\texttt{lb})\rfloor$;
2: **while** $\mid \texttt{ub}-\texttt{lb} \mid \geq 1$ **do**
3: Apply first λ_1+1 nodes to HARQ-II and remaining nodes to BCH(31,21,5). Compute the lifetime as H_1;
4: Apply first λ_2+1 nodes to HARQ-II and remaining nodes to BCH(31,21,5). Compute the lifetime as H_2;
5: **if** $H_1 < H_2$ **then**
6: Narrow the interval from the right (*i.e.*, $[\texttt{lb},\texttt{ub}]$ to $[\texttt{lb},\texttt{ub}=\lambda_2]$) and set $\lambda_2=\lambda_1$: $\texttt{ub}\leftarrow\lambda_2$; $\lambda_2\leftarrow\lambda_1$;
7: Update the intermediate point, λ_1, according to the new upper bound as: $\lambda_1 \leftarrow \lceil\texttt{ub}-\Phi\times(\texttt{ub}-\texttt{lb})\rceil$;
8: **else**
9: Narrow the interval from the left (*i.e.*, $[\texttt{lb},\texttt{ub}]$ to $[\texttt{lb}=\lambda_1,\texttt{ub}]$) and set $\lambda_1=\lambda_2$: $\texttt{lb}\leftarrow\lambda_1$; $\lambda_1\leftarrow\lambda_2$;
10: Update the intermediate point, λ_2, according to the new lower bound as: $\lambda_2 \leftarrow \lfloor\texttt{lb}+\Phi\times(\texttt{ub}-\texttt{lb})\rfloor$;
11: **end if**
12: **end while**
13: **Result:**
H_1 (or H_2); ▷ Network lifetime
$\mathcal{S}_i=1,\ i\in[2,\texttt{lb}]$; ▷ Nodes using HARQ-II
$\mathcal{S}_i=0,\ i\in[\texttt{lb}+1,|V|]$; ▷ Nodes using BCH(31,21,5)

H_1 and H_2. If $H_1 < H_2$, the upper bound of the interval is updated to $\texttt{ub}=\lambda_2$ (lines 5–7). On the other hand, if $H_1 \geq H_2$, the lower bound is updated to $\texttt{lb}=\lambda_1$ (lines 8–11). This narrowing process is repeated until $|\texttt{ub}-\texttt{lb}|<1$ (*i.e.*, $\texttt{ub}=\texttt{lb}$). H_1 (or H_2) is returned as the best lifetime. Furthermore, the GSS algorithm returns lb that indicates the threshold value for the number of sensor nodes that should use HARQ-II. In other words, from node-2 to node-lb, HARQ-II is used; while from node-$(\texttt{lb}+1)$ to node-$|V|$, BCH(31,21,5) is used (line 13).

We extend our GSS algorithm (*i.e.*, extended GSS – E-GSS) such that HARQ-II is set to the sensor nodes close to the sink node and BCH(31,21,5) is set to the sensor nodes far away from the sink node.

2) Simulated Annealing (SA): The SA algorithm is presented in Alg. 2. The *inputs* are $|V|$, the temperature (tmp), the cooling parameter (α), and the maximum cycle number (maxCycle). Initially, we set $\mathcal{S}_i=1,\ \forall i\in\left\{2,\ldots,\left\lceil\frac{|V|}{2}\right\rceil\right\}$ and $\mathcal{S}_i=0,\ \forall i\in\left\{\left\lceil\frac{|V|}{2}\right\rceil+1,\ldots,|V|\right\}$, in the current solution vector $\mathbb{S}$ (*i.e.*, the first half of the sensor nodes use HARQ-II, and the remaining ones use BCH(31,21,5)). The current lifetime (curLT) is obtained from this current

Algorithm 2 Pseudo-code for the SA algorithm.

Input: Number of all nodes ($|V|$), Initial temperature ($\texttt{tmp}\leftarrow 0$), Cooling parameter ($\alpha\leftarrow 0.95$), and Maximum cycle number ($\texttt{maxCycle}\leftarrow 50$).

Output: Lifetime, nodes using HARQ-II or BCH(31,21,5).

1: Initialize the parameters: Count ($\texttt{count}\leftarrow 0$), and Best Lifetime ($\texttt{bestLT}\leftarrow 0$);
2: Find an initial solution vector and set it as the initial temperature (tmp), the current solution vector (curSol), and the best solution vector (bestSol);
3: Compute the lifetime for the current solution vector and set as the current lifetime (curLT) and the best lifetime (bestLT);
4: **while** $\texttt{count}\leq\texttt{maxCycle}$ **do**
5: Assign the temperature ($\texttt{tmp}\leftarrow\texttt{tmp}\times\alpha$). Find a candidate solution vector (candSol) by taking the complement of one element in the current solution vector. Compute its lifetime as the candidate lifetime (candLT);
6: **if** $\texttt{candLT}>\texttt{curLT}$ **then**
7: Update the current lifetime (curLT) as the candidate lifetime (candLT): $\texttt{curLT}\leftarrow\texttt{candLT}$;
8: Update the current solution vector (curSol) as the candidate solution vector (candSol): $\texttt{curSol}\leftarrow\texttt{candSol}$;
9: **if** $\texttt{candLT}>\texttt{bestLT}$ **then**
10: Update the best lifetime (bestLT) as the candidate lifetime (candLT): $\texttt{bestLT}\leftarrow\texttt{candLT}$;
11: Update the best solution vector (bestSol) as the candidate solution vector (candSol): $\texttt{bestSol}\leftarrow\texttt{candSol}$;
12: $\texttt{count}\leftarrow 0$;
13: **end if**
14: **else**
15: $\texttt{count}\leftarrow\texttt{count}+1$;
16: $\Delta\texttt{Obj}\leftarrow\texttt{curLT}-\texttt{bestLT}$. Assign candSol as curSol with probability: $e^{-(\Delta\texttt{Obj}/\texttt{tmp})}$.
17: **end if**
18: **end while**
19: **Result:** bestLT and bestSol

solution vector (curSol) by manually setting $x_{ij}^k=0$ for the corresponding i and k values. Moreover, we store the current lifetime value as tmp (lines 1–3). Then, we find a candidate solution vector (candSol) by complementing the EC methods (*i.e.*, HARQ-II, or BCH(31,21,5)) of a randomly selected neighbor node. For instance, if a randomly selected neighbor node uses HARQ-II, it switches to use BCH(31,21,5) or vice versa. The candidate lifetime (candLT) is calculated using this candidate solution (line 5). If the candidate lifetime is greater than the current lifetime, we update the current solution vector and the current lifetime with the candidate solution vector and the candidate lifetime (lines 6–8). Moreover, if the candidate lifetime is greater than the best lifetime obtained so far (bestLT), the candidate lifetime and the candidate

Algorithm 3 Pseudo-code for the GA algorithm.

Input: Population size ($\texttt{popSize} \leftarrow 10$), Maximum generation ($\texttt{maxGen} \leftarrow 50$), Mutation probability ($\texttt{mutProb} \leftarrow 0.15$).

Output: Lifetime, nodes using HARQ-II or BCH(31,21,5).

1: Initialize the parameters: Generation ($\texttt{gen} \leftarrow 0$), Generate population with random chromosomes ($P \leftarrow \{\mathcal{C}_1, \ldots, \mathcal{C}_{\texttt{popSize}}\}$),
2: Calculate the best lifetime of the population ($\texttt{bestLT} \leftarrow \texttt{bestLT}(P)$).
3: **while** $\texttt{gen} \leq \texttt{maxGen}$ **do**
4: Select two chromosomes having the highest two lifetimes via the Roulette Wheel Selection algorithm: $\{\mathcal{C}_{S1}, \mathcal{C}_{S2}\} \leftarrow \texttt{RoulettteWheel}(P)$;
5: Crossover the selected chromosomes: $\mathcal{C}_{new} \leftarrow \texttt{CrossOver}(\mathcal{C}_{S1}, \mathcal{C}_{S2})$;
6: Mutate a random gene of the chromosome with probability, $\texttt{mutProb}$: $\mathcal{C}_{mut} \leftarrow \texttt{Mutation}(\mathcal{C}_{new})$;
7: Calculate the lifetime of the new chromosome ($\mathcal{C}_{new}$): $\texttt{newLT} \leftarrow \texttt{CalculateLifetime}(\mathcal{C}_{new})$;
8: **if** $\texttt{newLT} > \texttt{bestLT}$ **then**
9: Update the $\texttt{bestLT}$ with the $\texttt{newLT}$: $\texttt{bestLT} \leftarrow \texttt{newLT}$;
10: Select the worst chromosome yielding the lowest lifetime from the population: $\mathcal{C}_{worst} \leftarrow \texttt{SelectWorseChromosome}(P)$;
11: Remove it from the population: $P.\texttt{remove}(\mathcal{C}_{worst})$;
12: Add new chromosome to the population: $P.\texttt{add}(\mathcal{C}_{new})$;
13: **end if**
14: $\texttt{gen} \leftarrow \texttt{gen} + 1$;
15: **end while**
16: **Result:** $\texttt{bestLT}$ and P

solution vector are set as the best lifetime and the best solution vector ($\texttt{bestSol}$) (lines 9–13). Otherwise, we update the current solution vector and the current lifetime with the probability $e^{-(\Delta\texttt{Obj}/\texttt{tmp})}$, where $\Delta\texttt{Obj} = \texttt{curLT} - \texttt{bestLT}$ (lines 14–17). The algorithm continues until the maximum number of cycle limit has been reached. The SA algorithm returns the best lifetime and the corresponding best solution vector, containing the information on which nodes are using HARQ-II or BCH(31,21,5) (line 19).

We extend the SA algorithm by checking the distance of a random neighbor node to the sink node (*i.e.*, E-SA). This time, the initial solution vector is defined as half of the sensor nodes closest to the sink node are using HARQ-II while the other nodes are using BCH(31,21,5).

3) Genetic Algorithm (GA): The outline for the GA algorithm is provided in Alg. 3. In this algorithm, the *input* parameters are the population size ($\texttt{popSize}$), the maximum generation ($\texttt{maxGen}$), and the mutation probability ($\texttt{mutProb}$). Initially, the population (P) contains random solution vectors, which are called the *chromosomes* (*i.e.*, $P \leftarrow \{\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_{\texttt{popSize}}\}$). Each chromosome contains sensor nodes, called *genes* (*i.e.*, $\mathcal{C}_j = [\mathcal{G}_2, \mathcal{G}_3, \ldots, \mathcal{G}_{|V|}]^\intercal, \forall j \in \{1, 2, \ldots, \texttt{popSize}\}$). When $\mathcal{G}_i = 1$, HARQ-II is used in node-i. On the other hand, when $\mathcal{G}_i = 0$, BCH(31,21,5) is used (line 1). The lifetime of each chromosome, $\mathcal{C}_j$, is evaluated, and the best lifetime ($\texttt{bestLT}$) is declared (line 2). Using the *roulette wheel selection* (RWS), the chromosomes, having the highest two lifetimes, from the population are selected (*i.e.*, $\mathcal{C}_{S1}$ and $\mathcal{C}_{S2}$) (line 4). In RWS, the selection process is done randomly according to the lifetime values where the chromosomes with better lifetimes have higher chances to be selected for the crossover. The parent chromosomes, $\mathcal{C}_{S1}$ and $\mathcal{C}_{S2}$, go through the crossover and the mutation stages. In the crossover stage, a new child chromosome is produced by swapping the genes of two parents after a random cut point (*e.g.*, $\mathcal{C}_{S1} = [\mathbf{0\ 0\ 1} \mid 0\ 1 \ldots 0]$ and $\mathcal{C}_{S2} = [1\ 0\ 1 \mid \mathbf{0\ 1 \ldots 1}] \rightarrow \mathcal{C}_{new} = [\mathbf{0\ 0\ 1} \mid \mathbf{0\ 1 \ldots 1}]$) (line 5). The mutation stage occurs at the randomly selected gene with a pre-defined probability ($\texttt{mutProb}$). If the value of the selected gene is 1, it changes to 0, or vice versa (*e.g.*, $\mathcal{C}_{new} = [0\ 0\ 1\ 0\ 1\ 1 \ldots 1] \rightarrow \mathcal{C}_{mut} = [0\ 0\ 1\ \mathbf{1}\ 1\ 1 \ldots 1]$) (line 6). The new lifetime ($\texttt{newLT}$) is calculated from the new child chromosome, $\mathcal{C}_{new}$ (line 7). If the new lifetime is greater than the best lifetime ($\texttt{bestLT}$), the new child chromosome is added to the population (*i.e.*, $P \cup \{\mathcal{C}_{new}\}$) and the worst chromosome, which has the lowest lifetime, is removed from the population (*i.e.*, $P \setminus \{\mathcal{C}_{worst}\}$) (lines 8–13). These stages are continued until the termination criterion is met, and the best lifetime obtained so far ($\texttt{bestLT}$) with the population (P) are returned (lines 16).

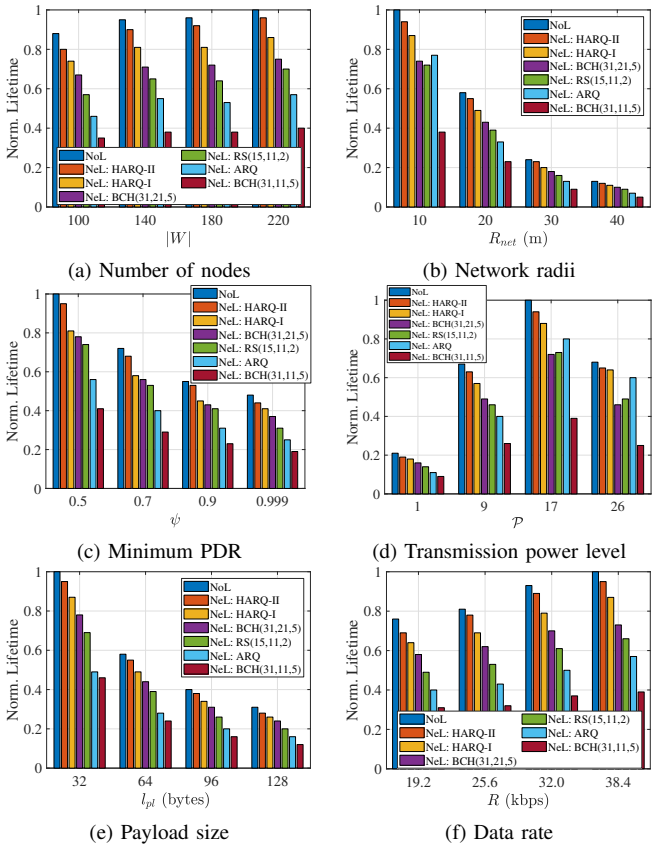
IV. Performance Results

In this section, we discuss the results of our numerical analysis. The MIP model and the meta-heuristic approaches are solved using the General Algebraic Modeling System (GAMS)[1]. The network-level (NeL) EC strategies are modeled by predefining $x_{ij}^k = 0$ for the undesired EC methods in (16). For example, the "NeL: ARQ" strategy is developed by forcing $x_{ij}^k = 0$ for $k = \{$BCH(31,11,5), BCH(31,21,5), RS(15,11,2), HARQ-I, HARQ-II$\}$. A similar interpretation can be applied to other NeL EC strategies. We randomly generate 20 network topologies and provide the average solutions throughout our analysis. The simulations are performed on HP Z640 Workstation with 16GB RAM and Intel(R) Xeon(R) E5-2620 v3 processor.

In Figs. 1–4, we choose $|W| = 100$, $R_{net} = 40$ m, $\psi = 0.999$, $\mathcal{P} = 1$, $l_{pl} = 128$ bytes, and $R = 19.2$ kbps. We investigate the effects of varying the number of nodes ($|W|$), the network radius (R_{net}), the minimum desired reliability criterion – minimum PDR (ψ), the transmission power level ($\mathcal{P}$), the packet size (l_{pl}), and the data rate (R) throughout the analysis. For this purpose, we vary one of the mentioned parameters (*i.e.*, $|W|$, R_{net}, ψ, $\mathcal{P}$, l_{pl}, or R), while the other parameters are kept constant.

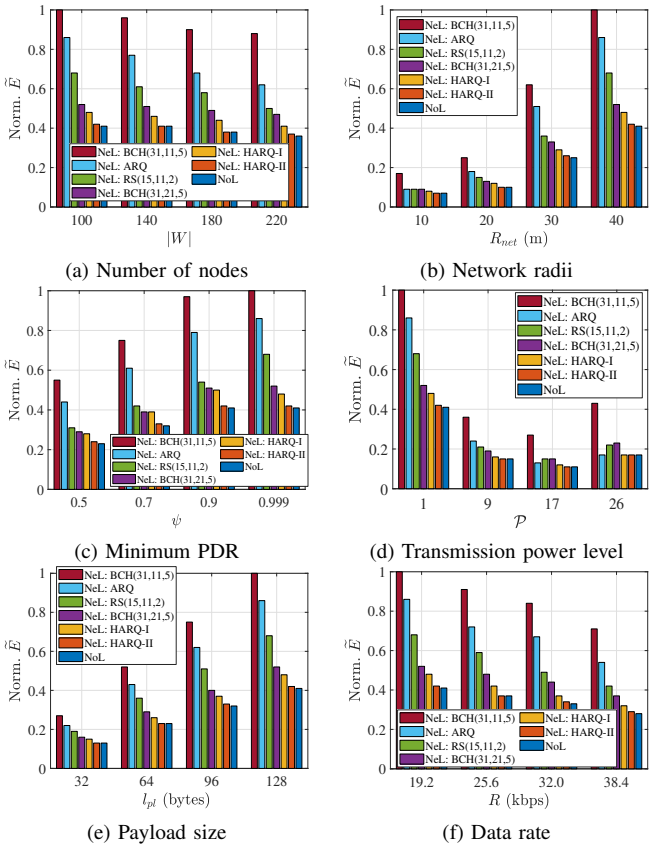
Figs. 1a–1f provides normalized lifetimes of both the NoL EC and the NeL EC strategies as $|W|$, R_{net}, ψ, $\mathcal{P}$, l_{pl}, or R

[1]GAMS Development Corporation, Washington, DC, USA. *General Algebraic Modeling System (GAMS) Release 25.0.3.* (2018). [Online]. Available: http://www.gams.com/.



(a) Number of nodes (b) Network radii (c) Minimum PDR (d) Transmission power level (e) Payload size (f) Data rate

Fig. 1: Comparison of normalized lifetimes of the NoL EC strategy against the NeL EC strategies.



(a) Number of nodes (b) Network radii (c) Minimum PDR (d) Transmission power level (e) Payload size (f) Data rate

Fig. 2: Normalized average energy consumption per node per round ($\widetilde{E}$) of the NoL EC and the NeL EC strategies.

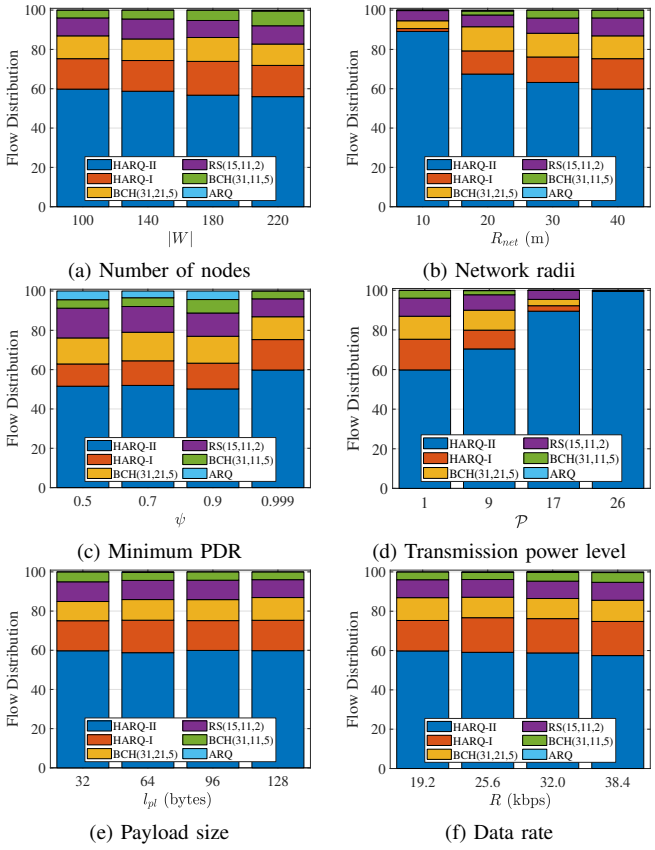
changes in each sub-figure. Normalized lifetimes are obtained by dividing each data point to the maximum data point in each sub-figure. Normalized lifetimes increase as $|W|$ increases for all EC strategies (in Fig. 1a) due to the decrement of the hop distance as the network becomes dense, resulting in nodes dissipating less energy for communication, thus improving lifetimes. On the other hand, normalized lifetimes drop as the network radius (R_{net}) increases (in Fig. 1b) since the energy required for communication increases in sparse networks. Moreover, as the minimum desired PDR (ψ) grows (in Fig. 1c), normalized lifetimes decrease since nodes need to dissipate more energy to satisfy the minimum required reliability criterion. We observe an interesting trend as the transmission power level ($\mathcal{P}$) changes. Our outcomes show that normalized lifetimes increase up to a specific power level (*i.e.*, $\mathcal{P} = 17$) and fall after $\mathcal{P} = 17$, regardless of the EC strategies (in Fig. 1d). The reason behind this trend is that we have high BERs on links for $\mathcal{P} \leq 17$. Although the BERs decrease as $\mathcal{P}$ is further increased, nodes start to waste energy, yielding short lifetimes. Normalized lifetimes reduce as the payload size (l_{pl}) increases (in Fig. 1e) since high l_{pl} values result in low BERs, where nodes should consume excessive energy to ensure reliable communication performance. Finally, normalized lifetimes increase as the data rate (R) is raised (in Fig. 1f) since the communication-related energies reduce as R increases.

The NoL EC strategy has the highest normalized lifetimes than all the NeL EC strategies such that NoL EC lifetimes are 4.4%–175.4% greater than the NeL EC strategies. In general, NeL: HARQ-II and NeL: HARQ-I strategies are the best two NeL EC strategies (in terms of lifetimes) after the NoL EC strategy such that these EC strategies have 4.2%–9.6% and 6.1%–19.7% shorter lifetimes than the NoL EC strategy, respectively. NeL: BCH(31,11,5) is the worst NeL EC strategy for all the parameter configurations. NeL: ARQ gives mediocre lifetimes after NeL: HARQ-II and NeL: HARQ-I strategies if the network radius is small (*i.e.*, $R_{net} = 10$ m), or the transmission power level is high (*i.e.*, $\mathcal{P} \geq 17$). However, for $R_{net} \neq 10$ m and $\mathcal{P} < 17$, NeL: BCH(31,21,5) is the most suitable option instead of NeL: ARQ after NeL: HARQ-II and NeL: HARQ-I strategies. For higher power levels (*i.e.*, $\mathcal{P} \geq 17$), NeL: RS(15,11,2) slightly provides prolonged lifetimes than NeL: BCH(31,21,5).

Figs. 2a–2f compare the average energy consumption per node per round (AEC) of the NoL EC and the NeL EC strategies. The AEC values (*i.e.*, $\widetilde{E}$) are calculated as

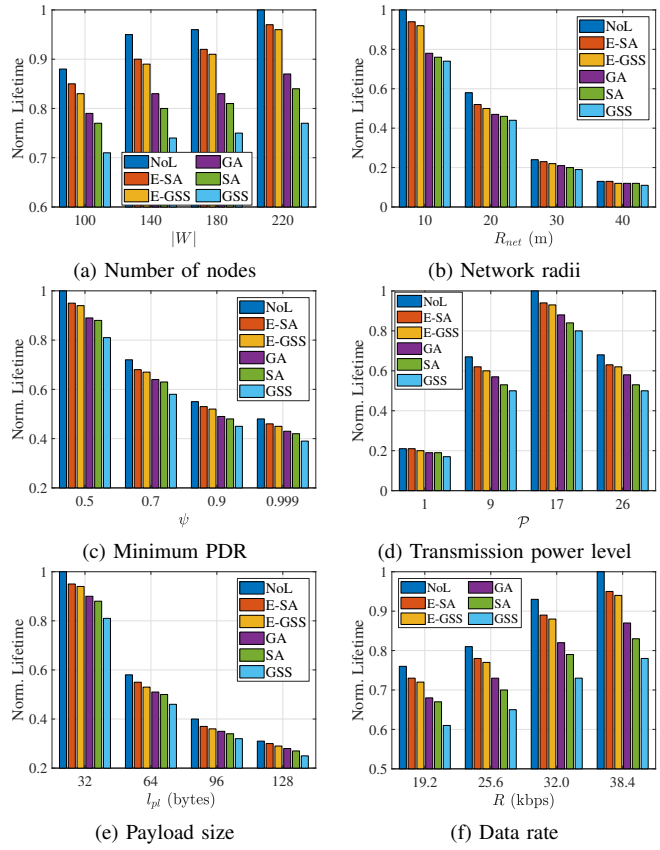
$$\widetilde{E} = \frac{\sum_{i \in W} E_i}{|W| \times H}. \tag{18}$$

In each sub-figure, we provide normalized AEC values such that $\widetilde{E}$ values obtained for each EC strategy are divided by the maximum $\widetilde{E}$ value attained. A high normalized $\widetilde{E}$ value indicates poor energy-efficiency and vice versa. Regardless of the parameter configuration, NeL: BCH(31,11,5)



(a) Number of nodes (b) Network radii (c) Minimum PDR (d) Transmission power level (e) Payload size (f) Data rate

Fig. 3: Distribution of flows according to the EC method usage (%) in the NoL EC strategy.



(a) Number of nodes (b) Network radii (c) Minimum PDR (d) Transmission power level (e) Payload size (f) Data rate

Fig. 4: Comparison of normalized lifetimes of the NoL EC strategy against the meta-heuristic approaches.

has the worst energy-efficiency among the NeL EC strategies, such that normalized $\widetilde{E}$ values are between 0.17 and 1.00. Our results show that NeL: BCH(31,11,5) has a mean energy-efficiency of 0.71 when the normalized $\widetilde{E}$ values obtained in all the parameter configurations are averaged. NeL: BCH(31,11,5) strategy is typically followed by NeL: ARQ strategy, where the normalized $\widetilde{E}$ values are in the 0.09–0.86 interval (NeL: BCH(31,11,5) has the mean energy-efficiency of 0.57). However, in Fig. 2d as $\mathcal{P} \geq 17$, NeL: RS(15,11,2) and NeL: BCH(31,21,5) have worse energy-efficiency performance than NeL: ARQ. For $\mathcal{P} \geq 17$, NeL: RS(15,11,2), NeL: BCH(31,21,5), and NeL: ARQ have mean energy-efficiency values of 0.19, 0.18, 0.15, respectively. On the other hand, NeL: HARQ-II yields the best energy-efficiency among the NeL EC strategies, having normalized $\widetilde{E}$ values in the 0.07–0.42 interval (the mean energy-efficiency of 0.30). The NoL EC strategy can further provide up to 4.0% lower normalized $\widetilde{E}$ values than NeL: HARQ-II (the NoL EC strategy has a mean energy-efficiency of 0.29). Hence, the NoL EC strategy has a better energy-efficiency than the NeL EC strategies.

In Figs. 3a–3f, we present the distribution of flows (packets) traversing over the network according to the EC method used for the NoL EC strategy. The flow distribution (*i.e.*, $\widetilde{x}_k$) is calculated using the optimal solutions of the x_{ij}^k variables defined in (16) while considering all the generated network topologies as

$$\widetilde{x}_k = \frac{\sum_{i \in W} \sum_{j \in V} x_{ij}^k}{\sum_{l \in \mathcal{K}} \sum_{i \in W} \sum_{j \in V} x_{ij}^l}, \ \forall k \in \mathcal{K}. \tag{19}$$

In each subplot, we provide *stacked* bar plots for each parameter in the x-axes such that the flow distribution sums up to 100%. For example, in Fig. 3a, when $|W| = 100$, our results show that 59.8%, 15.5%, 9.1%, 11.6%, 4.0%, 0.0% of the total flows are utilizing HARQ-II, HARQ-I, RS(15,11,2), BCH(31,21,5), BCH(31,11,5), and ARQ, respectively. Mathematically, $\widetilde{x}_{\text{HARQ-II}} = 59.8\%$, $\widetilde{x}_{\text{HARQ-I}} = 15.5\%$, $\ldots$, $\widetilde{x}_{\text{ARQ}} = 0.0\%$. Hence, $\sum_{k \in \mathcal{K}} \widetilde{x}_k = 100\%$. Regardless of the parameter configuration, the results show that at least half of the nodes are utilizing HARQ-II (*i.e.*, $\widetilde{x}_{\text{HARQ-II}} \geq 50.2\%$). On the average, majority of the nodes (86.8%) are utilizing HARQ-II (*i.e.*, $\widetilde{x}_{\text{HARQ-II}} = 63.2\%$), HARQ-I (*i.e.*, $\widetilde{x}_{\text{HARQ-I}} = 13.2\%$), and BCH(31,21,5) (*i.e.*, $\widetilde{x}_{\text{HARQ-I}} = 10.4\%$), while RS(15,11,2), BCH(31,11,5), ARQ are the least preferred EC methods in the NoL EC strategy. In Figs. 3a, 3e, and 3f, changing the parameters, $|W|$, l_{pl}, or R, has an insignificant effect on the flow distribution. On the other hand, as the network radii increase in Fig. 3b, the utilization of HARQ-II is reduced from 89.1% to 60.0% since the retransmission of redundant bits in HARQ-II is costly (in terms of energy) when the network becomes sparse. However, as ψ increases in Fig. 3c, the utilization percentage of HARQ-II increases from 51.6% to 60.0% as the retransmission of redundant

TABLE III: Average solution times (ASTs – s) for the NoL EC strategy and the meta-heuristic approaches wrt. $|W|$.

$\|W\|$	**NoL**	**GSS**	**E-GSS**	**SA**	**E-SA**	**GA**
100	15	12	15	71	66	74
140	125	30	37	180	164	190
180	538	56	87	350	297	359
220	1114	121	116	588	496	619

bits can be considered as a feasible choice to satisfy the tight reliability criterion. Similarly, as $\mathcal{P}$ increases in Fig. 3d, HARQ-II becomes the dominant method since BERs reduce as $\mathcal{P}$ grows; thus, the retransmission of redundant bits is minimized, where HARQ-II yields a high energy-efficiency.

Figs. 4a–4f show normalized lifetimes of both the NoL EC strategy and the meta-heuristic approaches (*i.e.*, GSS, E-GSS, SA, E-SA, and GA). As a general trend, E-SA has the highest normalized lifetimes among the meta-heuristic approaches, yielding 3.4%–10.0% lesser lifetimes than the NoL EC strategy. E-SA is followed by E-GSS, GA, SA, and GSS algorithms, respectively. GSS has the lowest normalized lifetimes, providing 18.2%–26.9% shorter lifetimes than the NoL EC strategy. Extended versions of GSS and SA algorithms (*i.e.*, E-GSS & E-SA) offer longer lifetimes than their unextended versions. E-GSS and E-SA have 13.0%–25.0% and 8.3%–23.8% higher lifetimes than GSS and SA, respectively. Finally, GA has mediocre lifetime performance.

Table III provides the average solution times (ASTs – in seconds) for the NoL EC strategy and the meta-heuristic approaches with respect to $|W|$. As $|W|$ increases, ASTs of all EC strategies significantly increase due to the increment in the search space of the optimization problems. When $|W| \leq 140$, only the ASTs of GSS (*i.e.*, 12–121 seconds) and E-GSS (*i.e.*, 15–116 seconds) are lower than the ASTs of the NoL EC strategy while the remaining algorithms (*i.e.*, SA, E-SA, and GA) have higher ASTs than the NoL EC strategy. On the other hand, as $|W| > 140$, ASTs of all the meta-heuristic approaches are shorter than the ASTs of the NoL EC strategy. Although E-SA produces the greatest lifetimes among the meta-heuristic approaches (*i.e.*, 3.4%–10.0% reduced lifetimes than the NoL EC strategy), it has a poor AST performance (*i.e.*, ASTs are between 66 and 496 seconds). On the other hand, E-GSS is a decent alternative for the E-SA, which can deliver 3.9%–13.6% lower lifetimes than the NoL EC strategy while reducing the ASTs of E-SA by 75.5% on the average. Besides, E-GSS decreases the AST of the NoL EC strategy by 89.6% while giving lifetimes within a 3.9% neighborhood of the NoL EC lifetimes.

V. Conclusion

In this work, we develop a node-level error control (NoL EC) strategy, which allows each node to utilize an optimum EC method (ARQ, FEC, or Hybrid ARQ) to maximize WSNs lifetime satisfying a pre-determined reliability requirement. The NoL EC strategy is constructed using MIP formulations that consider the energy dissipation characteristics of both the Mica2 WSN node platform and the three mentioned EC methods. We compare the lifetimes and the energy-efficiency of the NoL EC strategy against the network-level EC (NeL EC) strategies. We propose five meta-heuristic approaches to obtain near-optimal solutions for the MIP model in polynomial-time.

Our main conclusions are enumerated as follows:

1) The NoL EC strategy outperforms all the NeL EC strategies in terms of the network lifetimes. Our results show that the lifetimes obtained with the NeL EC strategies can be improved by at least 4.4% using the NoL EC strategy with the cost of extra computation time. For the NeL EC strategies, HARQ-II yields the highest lifetimes, while BCH(31,11,5) has the lowest lifetimes.
2) Among the NeL EC strategies, HARQ-II has the best efficiency, while BCH(31,11,5) has the worst energy-efficiency. Nevertheless, the NoL EC strategy can improve the energy-efficiency of NeL: HARQ-II strategy by at most 4.0%.
3) For the NoL EC strategy, HARQ-II is the preferred EC method for the nodes close to the sink node. On the other hand, BCH(31,21,5) is the favorite EC method for the nodes away from the sink node. Furthermore, in the NoL EC strategy, at least half of the nodes prefer to use the HARQ-II method.
4) The E-GSS algorithm can reduce the solution times of the NoL EC strategy by 89.6% while providing lifetimes within a 3.9% neighborhood of the NoL EC lifetimes.

References

[1] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, 2009.

[2] D. V. Queiroz, M. S. Alencar, R. D. Gomes, I. E. Fonseca, and C. Benavente-Peces, "Survey and systematic mapping of industrial wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 97, pp. 96–125, 2017.

[3] L. Lei, Y. Kuang, X. S. Shen, K. Yang, J. Qiao, and Z. Zhong, "Optimal reliability in energy harvesting industrial wireless sensor networks," *IEEE Trans. Wirel. Commun.*, vol. 15, no. 8, pp. 5399–5413, 2016.

[4] K. Yu, M. Gidlund, J. Åkerberg, and M. Bjorkman, "Reliable and low latency transmission in industrial wireless sensor networks," *Procedia Comput. Sci.*, vol. 5, pp. 866–873, 2011.

[5] M. C. Vuran and I. F. Akyildiz, "Error control in wireless sensor networks: A cross layer analysis," *IEEE/ACM Trans. Netw.*, vol. 17, no. 4, pp. 1186–1199, 2009.

[6] N. Tekin and V. C. Gungor, "The impact of error control schemes on lifetime of energy harvesting wireless sensor networks in industrial environments," *Comp. Stand. Inter.*, vol. 70, p. 103417, 2020.

[7] J. Park and S. Sahni, "An online heuristic for maximum lifetime routing in wireless sensor networks," *IEEE Trans. Comput.*, vol. 55, no. 8, pp. 1048–1056, 2006.

[8] R. Kadel, K. Paudel, D. B. Guruge, and S. J. Halder, "Opportunities and challenges for error control schemes for wireless sensor networks: A review," *Electronics*, vol. 9, no. 3, 2020.

[9] M. C. Domingo and M. C. Vuran, "Cross-layer analysis of error control in underwater wireless sensor networks," *Comput. Commun.*, vol. 35, no. 17, pp. 2162–2172, 2012.

[10] Z. Tian, D. Yuan, and Q. Liang, "Energy efficiency analysis of error control schemes in wireless sensor networks," in *Proc. Int. Wirel. Commun. Mobile Comput. Conf.*, 2008, pp. 401–405.

[11] M. Y. Naderi, H. R. Rabiee, M. Khansari, and M. Salehi, "Error control for multimedia communications in wireless sensor networks: A comparative performance analysis," *Ad Hoc Netw.*, vol. 10, no. 6, pp. 1028–1042, 2012.

[12] S. M. Razali, K. Mamat, and N. S. K. Bashah, "Implementation of hybrid ARQ (HARQ) error control algorithm for lifetime maximization and low overhead CDMA wireless sensor network (WSN)," in *Proc. IEEE Conf. Wirel. Sens. (ICWiSE)*, 2016, pp. 71–76.

[13] A. A. Alshehri, S. Lin, and I. F. Akyildiz, "Optimal energy planning for wireless self-contained sensor networks in oil reservoirs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–7.
[14] S. M. Chowdhury, A. Hossain, and S. Debnath, "Impact of error control code on characteristic distance in wireless sensor network," *Wirel. Pers. Commun.*, vol. 92, no. 4, pp. 1459–1471, 2017.
[15] M. Zhan, J. Wu, H. Wen, and P. Zhang, "A novel error correction mechanism for energy-efficient cyber-physical systems in smart building," *IEEE Access*, vol. 6, pp. 39 037–39 045, 2018.
[16] M. Kang, D. K. Noh, and I. Yoon, "Energy-aware control of error correction rate for solar-powered wireless sensor networks," *Sensors*, vol. 18, no. 8, p. 2599, 2018.
[17] F. Khan, A. ur Rehman, M. Usman, Z. Tan, and D. Puthal, "Performance of cognitive radio sensor networks using hybrid automatic repeat request: Stop-and-wait," *Mobile Netw. Appl.*, vol. 23, no. 3, pp. 479–488, 2018.
[18] O. Brini, D. Deslandes, and F. Nabki, "A system-level methodology for the design of reliable low-power wireless sensor networks," *Sensors*, vol. 19, no. 8, p. 1800, 2019.
[19] B. Sarvi, H. R. Rabiee, and K. Mizanian, "An adaptive cross-layer error control protocol for wireless multimedia sensor networks," *Ad Hoc Netw.*, vol. 56, pp. 173–185, 2017.
[20] M. Nikzad, A. Bohlooli, and K. Jamshidi, "An adaptive, cross layer error control scheme for distributed video coding over wireless multimedia sensor networks," *Multimed. Tools Appl.*, vol. 79, no. 43, pp. 32 999–33 021, 2020.
[21] R. Kadel, K. Ahmed, and A. Nepal, "Adaptive error control code implementation framework for software defined wireless sensor network (SDWSN)," in *Proc. Int. Telecommun. Netw. Appl. Conf. (ITNAC)*, 2017, pp. 1–6.
[22] A. Brokalakis, I. Chondroulis, and I. Papaefstathiou, "Extending the forward error correction paradigm for multi-hop wireless sensor networks," in *Proc. IFIP Int. Conf. New Technol. Mobil. Secur. (NTMS)*, 2018, pp. 1–5.
[23] B. Husain and A. Czylwik, "Channel coding and low latency HARQ for industrial wireless sensor networks," in *Proc. Wirel. Days (WD)*, 2019, pp. 1–5.
[24] M. Patil and R. C. Biradar, "Dynamic error control scheme based on channel characteristics in wireless sensor networks," in *Proc. IEEE Int. Conf. Recent Trends Electron. Inform. Commun. Technol. (RTEICT)*, 2017, pp. 736–741.
[25] M. Yigit, P. S. Boluk, and V. C. Gungor, "A new efficient error control algorithm for wireless sensor networks in smart grid," *Comp. Stand. Inter.*, vol. 63, pp. 27–42, 2019.
[26] S. Razali, K. Mamat, and N. S. K. Bashah, "Optimization of remaining energy and error rates for wireless sensor network," in *Proc. Int. Conf. Comput. Sci. Technol.*, 2017, pp. 130–140.
[27] G. S. Nikolic, M. K. Stojcev, T. R. Nikolic, B. D. Petrovic, G. S. Jovanovic, and B. R. Dimitrijevic, "Implementation and evaluation of 2D SEC-DED forward error correction scheme in wireless sensor networks," *Microelectron. Reliab.*, vol. 78, pp. 161–180, 2017.
[28] I. Ez-Zazi, M. Arioua, A. E. Oualkadi, and P. Lorenz, "On the performance of adaptive coding schemes for energy efficient and reliable clustered wireless sensor networks," *Ad Hoc Netw.*, vol. 64, pp. 99–111, 2017.
[29] I. Ez-Zazi, M. Arioua, and A. E. Oualkadi, "On the design of coding framework for energy efficient and reliable multi-hop sensor networks," *Proc. Comput. Sci.*, vol. 109, pp. 537–544, 2017.
[30] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. H. Hanzo, "A survey of network lifetime maximization techniques in wireless sensor networks," *IEEE Commun. Surv. Tut.*, vol. 19, no. 2, pp. 828–854, 2017.
[31] R. M. Curry and J. C. Smith, "A survey of optimization algorithms for wireless sensor network lifetime maximization," *Comput. Ind. Eng.*, vol. 101, pp. 145–166, 2016.
[32] S. Kurt, H. U. Yildiz, M. Yigit, B. Tavli, and V. C. Gungor, "Packet size optimization in wireless sensor networks for smart grid applications," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2392–2401, 2016.
[33] H. L. Bodlaender, R. B. Tan, T. C. van Dijk, and J. van Leeuwen, "Integer maximum flow in wireless sensor networks with energy constraint," in *Algorithm Theory – SWAT*. Springer Berlin Heidelberg, 2008, pp. 102–113.
[34] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Proc. Ann. IEEE Commun. Society Conf. Sens. Ad Hoc Commun. Netw. (SECON)*, 2004, pp. 517–526.
[35] E. Tanghe, W. Joseph, L. Verloock, L. Martens, H. Capoen, K. V. Herwegen, and W. Vantomme, "The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 mhz," *IEEE Trans. Wirel. Commun.*, vol. 7, no. 7, pp. 2740–2751, 2008.
[36] A. Xenakis, F. Foukalas, and G. Stamoulis, "Cross-layer energy-aware topology control through simulated annealing for WSNs," *Comput. Elec. Eng.*, vol. 56, pp. 576–590, 2016.
[37] J. Vales-Alonso, E. Egea-López, A. Martínez-Sala, P. Pavón-Mariño, M. Victoria Bueno-Delgado, and J. García-Haro, "Performance evaluation of MAC transmission power control in wireless sensor networks," *Comput. Netw.*, vol. 51, no. 6, pp. 1483–1498, 2007.
[38] C. Tan, J. Zou, M. Wang, and R. Zhang, "Network lifetime optimization for wireless video sensor networks with network coding/ARQ hybrid adaptive error-control scheme," *Comput. Netw.*, vol. 55, no. 9, pp. 2126–2137, 2011.
[39] J. Zhang and J. Long, "An energy-aware hybrid ARQ scheme with multi-ACKs for data sensing wireless sensor networks," *Sensors*, vol. 17, no. 6, 2017.
[40] C. Mahapatra, Z. Sheng, P. Kamalinejad, V. C. M. Leung, and S. Mirabbasi, "Optimal power control in green wireless sensor networks with wireless energy harvesting, wake-up radio and transmission control," *IEEE Access*, vol. 5, pp. 501–518, 2017.
[41] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inform. Sciences*, vol. 237, pp. 82–117, 2013.
[42] C. Tsai, T. Hong, and G. Shiu, "Metaheuristics for the lifetime of WSN: A review," *IEEE Sens. J.*, vol. 16, no. 9, pp. 2812–2831, 2016.
[43] N. Sharma and V. Gupta, "Meta-heuristic based optimization of WSNs energy and lifetime-a survey," in *Proc. Int. Conf. Cloud Comput. Data Sci. Eng. (Confluence)*, 2020, pp. 369–374.
[44] H. U. Yildiz, V. C. Gungor, and B. Tavli, "Packet size optimization for lifetime maximization in underwater acoustic sensor networks," *IEEE Trans. Ind. Inform.*, vol. 15, no. 2, pp. 719–729, 2019.
[45] H. U. Yildiz, K. Bicakci, B. Tavli, H. Gultekin, and D. Incebacak, "Maximizing wireless sensor network lifetime by communication/computation energy optimization of non-repudiation security service: Node level versus network level strategies," *Ad Hoc Netw.*, vol. 37, pp. 301–323, 2016.

Nazli Tekin received the B.S. degree in computer engineering from Koc University, Istanbul, Turkey, in 2011, and the Ph.D. degree in electrical and computer engineering from Abdullah Gül University, Kayseri, Turkey, in 2020. She is an Assistant Professor with the Department of Software Engineering, Erciyes University, Kayseri. Her current research interests include wireless sensor networks, the Industrial Internet of Things, and network security.

Huseyin Ugur Yildiz (Senior Member, IEEE) received the B.S. degree from Bilkent University, Ankara, Turkey, in 2009; the M.S. and Ph.D. degrees from the TOBB University of Economics and Technology, Ankara, in 2013 and 2016, respectively, all in electrical and electronics engineering. He is currently an Associate Professor in the Department of Electrical and Electronics Engineering at TED University, Ankara. His research focuses on the applications of optimization techniques for modeling and analyzing research problems on wireless communications, wireless networks, underwater acoustic networks, and smart grids.

Vehbi Cagri Gungor received the B.S. and M.S. degrees in electrical and electronics engineering from METU, Ankara, Turkey, in 2001 and 2003, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2007. He is currently a Full Professor and the Chair of Computer Engineering Department, Abdullah Gül University, Kayseri, Turkey. He has authored more than 100 articles in refereed journals and international conference proceedings. His current research interests are in next-generation wireless networks, wireless ad hoc and sensor networks, smart grid communications, and artificial intelligence. He has been serving as an Associate Editor for prestigious journals, such as IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS and Ad Hoc Networks (Elsevier).