A novel multi-objective optimizer framework for TDMA-based medium access control in IoT

Prathap Siddavaatam[1] · Reza Sedaghat[1]

Abstract Recently, cross-layer optimization has been emphasized by researchers for Internet of Things (IoT) where billions of devices communicate intelligently making every day communication highly informative. Medium Access Control (MAC) sub-layer is part of data link layer in IoT protocol stack. The energy consumption of things is greatly affected by MAC protocol which controls the node radio functionalities. In this article, we propose the design of an efficient MAC for IoT by minimizing its energy consumption at various levels. The minimization model is subject to reduction in distortion levels while simultaneously extending network lifetime. This article synthesizes and optimizes MAC layer of IoT based protocols using a novel delta diagram synthesizer and accurately captures both the high heterogeneity of the IoT and the impact of the Internet as part of the MAC layer architecture. Furthermore a novel hybrid Whale-Artificial Bee Colony optimizer framework is proposed to obtain optimal nodes based on energy consumption and the communication parameters among things, by exploiting the interrelations among different layer functionalities in the IoT. This novel system saves considerable computation resources of the resource constrained IoT devices and adapts to various data sources with different distortion levels rapidly.

✉ Reza Sedaghat
rsedagha@ee.ryerson.ca;
https://www.ee.ryerson.ca/opr/

Prathap Siddavaatam
psiddava@ee.ryerson.ca

[1] OPRA-Labs, Ryerson University, 350 Victoria St, Toronto, ON M5B 2K3, Canada

1 Introduction

Over the past decade wireless sensor networks (WSNs) have been the subject of intensive research in cross-layer optimization. WSNs comprise a vast number of sensor nodes deployed for collecting surrounding context and environment information [32]. A myriad of medium access and routing protocols [18, 28] along with many physical layers have been proposed for WSNs. These kind of protocols have been made energy-aware [31, 34]; fusion and aggregation strategies were employed [22]; location enriched basic infrastructures have been deployed [11], timing [40], and security protocols [12] enhanced; operating systems were specifically designed to support such higher-level abstractions [6]. Today, we stand at the cusp of Internet of Things (IoT), which is expected to massively span across these interconnecting WSNs.

The vast heterogeneousness in capacities of Internet of Things (IoT) device hardware and the diverse communication methods used among things warrant a cross-layer based design [2, 15] for IoT. The IoT aims to network various things, e.g., smart watches, phones, tablets, sensors, actuators, cars and other mobile devices, utilizing a variety of existing physical network infrastructures, including WiFi, Bluetooth, 802.15.4, Z-wave, and LTE-advanced [3, 5]. Basically, IoT is realized by amalgamating communication capabilities with functionalities of sensing, identification and actuation into everyday things and then initiating communication in expansive Internet technologies.

2 Synthesizing TDMA based MAC protocols

Schemes based on scheduling in the initialization phase are responsible for designating collision free links between neighboring nodes. These links are assigned as code division multiple access (CDMA), frequency division multiplexing (FDM) bands, or time division multiplexing (TDM) slot based spectrum codes. TDMA schemes are favored by IoT Networks due to the complexities faced by the CDMA and the FDMA schemes [30].

System time in the TDMA schemes is split into slots, that are then allocated to the neighboring nodes. Authorization of the participant on resources with regular time is managed by the schedule. While generally monitored by the central authority, the schedule may also be computed, fixed or a hybrid on demand. Since a node can only access its allocated time slot, there is no conflict with the neighbors. Prime advantages of using scheduling based schemes are implicitly avoidance of idle listening, minimum collisions and less overhearing. Also an anticipated and bounded end to end delay can be administered by scheduling based schemes. Considering that a node must wait for its allocated time slot before accessing the channel,the average queuing delay is generally high. Allocating conflict free TDMA schedules, however is a complex task. Major concerns with these schemes are lacking of adaptability, extra and overhead traffic, low throughput and reduced scalability. Since the nodes are only allowed to communicate with the central authority, it is also impractical for scheduling based schemes to have peer-to-peer based communication.

While many designs of wireless MAC protocols established on TDMA were suggested [7], despite needing global topology information, most of them could not be adaptable for very large size networks [1, 42]. To reduce the problem of acquiring global topology information in nigger networks, many distributed slot assignment schemes were proposed like, TRAMA [41],PACT [4] and DRAND [17]. Local interference and topology information at every node is obtained by the multicolor-GCF (M-GCF) [9], Green Conflict Free (GCF), or the depth first search (DFS) [42] algorithms. In all of these approaches, schedules are composed by exchanging messages between local nodes with the interference range. The cost increases corresponding to the schedule length [7] when global topology information is demanded in distributed scheduling in order to make it more flexible. A TDMA-MAC protocol derived from TRAMA is the Flow-Aware Medium Access (FLAMA) [23], it is optimized for monitoring applications periodically. The gist of FLAMA is evading the overhead linked to the traffic information exchange. FLAMA set up first flows before using a pull-based mechanism due to the message flow in periodic reporting applications being relatively stable. As a result, the data is transferred only when requested specifically.

Classical slot reservation algorithms are often not flexible and quite complex. Hence, researchers have explored simpler schemes that also aim in achieving better energy efficiency. A protocol to demonstrate the joint interconnect in the fair rate allocation and maximum throughput in IoT was proposed in [19] for slot reuse based TDMA in consideration.

3 Related work

We discuss some of the recent attempts at standardization of point-to-point communications with respect to data link layer for IoT that are only relevant to our model in regards to OSI protocol [39] stack.

The most widely used data link layer standards are Bluetooth Low Energy (BLE) [16, 29] and ZigBee [10]. IEEE 802.11ah [27], on the other hand, is the most convenient standard employed due to existing and widely separated infrastructure of IEEE 802.11 [14]. Also, IEEE 802.15.4 is the most commonly used IoT standard for Medium Access Control (MAC) sublayer of data link layer. However, some providers would seek for more reliable and secured technology and have recently employed Home-Plug [21] for LAN connectivity. In addition, the newly arising LoRaWAN [37] has made huge strides in usage of low-power Wide Area Networks (WANs) having low cost, mobility, security, and bi-directional data link communication for IoT applications.

3.1 Time-division multiple access (TDMA)

Used in digital 2G cellular systems, a channel access method for shared medium networks as shown by Fig. 1 is known as Time-Division Multiple Access (TDMA). In this protocol, a signal gets divided into different slots of time which enables multiple users to share the same frequency channel [13]. Despite being similar to time division multiplexing, TDMA has multiple transmitters rather than just one. Each user uses their own allocated time slot. Data is transmitted in swift progression hereby permitting multiple stations to share the same frequency channel despite using only a part of the channel capacity. Figure 1 shows the TDMA structure where the data stream is divided into frames and the frames are further divided into time slots.

Fig. 1 TDMA frame structure
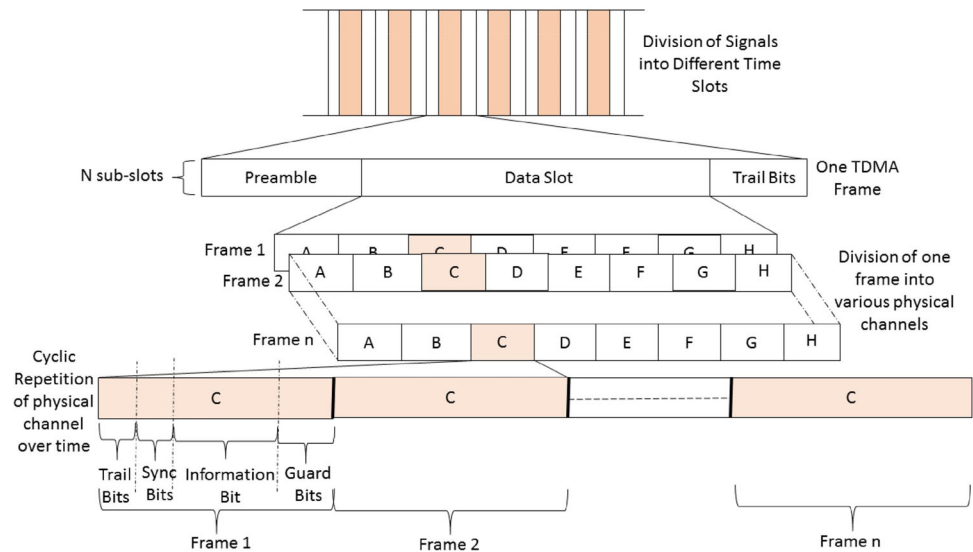
3.2 IoT system overview

3.2.1 Protocol set-up

The various cross-layered protocols that were developed by researchers for Ad-Hoc networks [24], Wireless Mesh Networks [46] and Wireless Sensor Networks (WSNs) [25] could not be used in experimenting and incorporating the suggested IoT framework. The main issue why synthesizing these networks into the IoT structure was due to the heterogeneity of the devices in these networks. The hardware specifications, the Quality-of-Service requirements and data precision requirements of all of these devices were different than those being used in the listed protocols of [8, 24, 25] where the hardware specifications in the WSNs are identical and thus the communication requirements can be shared since the objectives are shared too. Unlike the flat networks proposed without the Internet in [24, 25, 46], the IoT network comprises of the Internet. As a result, the reference architecture resembles practical scenarios which are generally integrated and organized hierarchically. A cross-layered protocol named the intel-LEACH [34], was developed to centralize the routing, MAC and the physical layer functionalities into a common communication scheme. intel-LEACH also picks
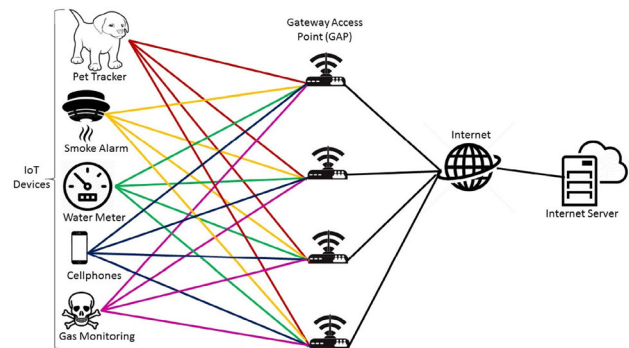

Fig. 2 An example of network architecture for IoT

a node with high yields based on a strategy of dynamic optimization which is autonomous of the layout of the network architecture. The intel-LEACH protocol is used on the proposed on the IoT network model because it is regarded as an appropriate fit in performing comparison evaluations and for testing the performance metrics with the state-of-the-art protocols.

3.2.2 Network architecture

The network architecture of the IoT used as a reference in the course of this article is depicted in Fig. 2 The following elements form an essential part in the network:

Gateway Access Point (GAP)	Devices that act as an administrator of communication between the Internet and things are know as Gateway Access Points (GAP). They have a higher computation capacity. GAP domain is referred to as a set of things that are in complete control of a single GAP. For routing though a GAP, there are two types of possibilities of communications during the routing process. These are: inter-GAP and intra-GAP. Inter-GAP communication is responsible for dealing with things in different GAP domains. The GAP behaves as both, a coordinator and a gateway. The optimization algorithms are locally initiated by exploiting the network knowledge by the GAP. Intra-GAP communication deals with things in the same GAP domain. It is desirable to use the GAP for coordination in the IoT network because of the different hardware capabilities in things.
Things	Objects that have different abilities owing to their different hardware specifications with respect to memory and data storage capacity, computation, transmission power or communication. Fig 2 shows different appliances and other equipments that are all examples of things. All of these things do not have the ability to communicate with the GAP directly, thus multi-hop links are a critical requirement. GAP is able to communicate with all of the things within its domain in one hop. Thus, GAP may have a higher transmission power compared to the things due to the asymmetry of the links.
Internet	A primitive component of the IoT network is the Internet. When the network is synthesized using Delta-Diagrams, it is characterized by a stochastic packet loss model [43] and a stochastic queuing delay model. Most of the things are connected to the Internet via the common gateway access point (GAP) as shown in Fig. 2.

4 MAC layer design synthesis

In this section, we propose a novel optimization model for the IoT using a parameter synthesizing road map provided by a delta diagram illustrated in Fig. 4 for the MAC layer. Furthermore the delta diagram synthesizer can be generalized and extended to any layer chosen by the optimization model adopted by a communication system [35]. For the IoT network architecture specified in Fig. 2, the delta diagram is applied to synthesize MAC layer of the protocol stack described in Fig. 3.

Our model assumes K heterogeneous sources that transmit data wired or wirelessly to a central Base Station (BS). The nodes access the up-link channel using TDMA based MAC protocol, and time is partitioned into frames, where frame m corresponds to the time interval $[\mathcal{T}_m, \mathcal{T}_{m+1})$. Every duty cycle of the node periodically produces data and evaluates the compression ratio resulting in final transmission to the common receiver.

We adopt a resource allocation approach which involves centralized management [32] to estimate resource availability and environmental dynamics, coordinate the allocation of resources across applications and nodes, and therefore adapt the protocol energy parameters at each level based on the synthesizer road map provided by delta diagram. This approach assists in integrating scattered communication functionalities into a united coherent optimization model and provide an flexible solution for MAC layer design and control. We use this approach to jointly control and synthesize a select case of Quality of Service (QoS) requirements. Based on our selected QoS case, we synthesize the MAC layer of network architecture represented in Fig. 2 using the delta diagram in Fig. 5 that results in mapping respective energy parameters. The synthesizer produces these parameters based on differentiated services for applications having contrasting QoS requirements, ranging from error-limited applications or minimum energy consumption applications to highly-delay-sensitive applications or any combination of them. We can model this case as a multi-objective optimization problem that must simultaneously optimize multiple conflicting objectives of QoS requirements subject to distortion constraints of delta diagram synthesizer, given by a minimization fitness function that produces a triple as follows,

$$\mathcal{F}^{\mathtt{fit}} = \underset{x}{\text{minimize}}\, J(\mathbf{x})$$

$$\Longleftrightarrow \underset{x}{\text{minimize}}\, J(\mathbf{x}) = \left(\, x_0,\ x_1,\ x_2,\ x_3\, \right)$$

$$= \left(\, \xi_{min}^{\mathtt{pe}},\ \xi_{min}^{\mathtt{te}},\ \xi_{min}^{\mathtt{sd}},\ \xi_{min}^{\mathtt{ec}}\, \right) \quad (1)$$

$$\text{subject to } \mathcal{E}_{\mathtt{D}}^m \leq \mathcal{E}_{\mathtt{DTH}}^{\mathtt{m}},\ \forall\, m$$

$$\xi_{\mathtt{Total}}^m \leq \xi_{\mathtt{max}}^m,\ \forall\, m$$

Fig. 3 Open systems interconnection reference model magnifying network, data link and physical layers
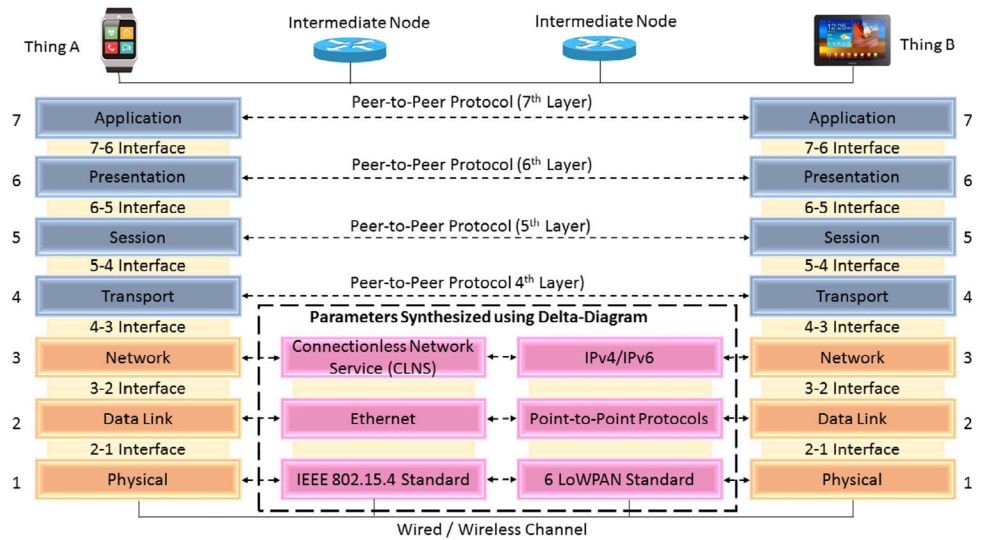
The fitness function $\mathcal{F}^{\texttt{fit}}$ is a minimization function. It takes a vector $\mathbf{x}$ of dimensions 4×1 having elements x_0, x_1, x_2 and x_3 and produces a minimized tuple ($\xi_{min}^{\texttt{pe}}$, $\xi_{min}^{\texttt{te}}$, $\xi_{min}^{\texttt{sd}}$, $\xi_{min}^{\texttt{ec}}$) for a given instance when subjected to constraint conditions in Eq. (1). For the IoT network architecture in Fig. 2, we consider the end-to-end communication as point-to-point communication due the vast diversity of end nodes and its heterogeneousness with respect to its computing capacity, storage, etc. The terms in Eq. (1) are as follows: $\xi_{min}^{\texttt{pe}}$ is processing energy for compressing data, $\xi_{min}^{\texttt{te}}$ is transmission energy of the thing, $\xi_{min}^{\texttt{sd}}$ is combination of equipment circuitry sink energy and sensing detection energy, and finally $\xi_{min}^{\texttt{ec}}$ is the encryption/decryption cost for data to be transmitted or received. Note that all the above discussed energies must be optimally minimal. The total energy consumption $\xi_{\texttt{Total}}^m$ of a thing in frame m is obtained by summing all the energies (ξ's) mentioned above.

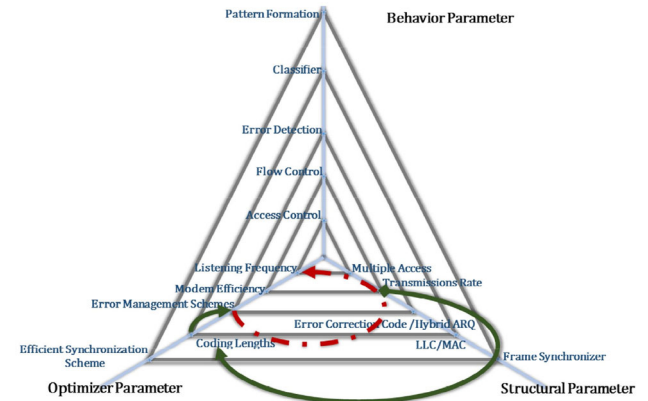**Fig. 4** Delta diagram synthesizer for Medium Access Control (MAC) layer

Fig. 5 Energy parameter synthesizing final action for data link layer synthesis

The average physical rate of user $\ell \in K$ in frame m is denoted using Shannon's bound [38] as,

$$\mathfrak{r}_\ell^m = B \, \log_{10}(1 + \mathsf{SNR}_\ell^m)$$
$$\Longleftrightarrow \mathfrak{r}_\ell^m = B \, \log_{10}(1 + g \cdot \mathsf{P}_\ell^m) \tag{2}$$

where SNR is the signal-to-noise ratio, P_ℓ^m is transmission power, g is noise normalized channel gain and B is the bandwidth of the channel. The channels gains for every thing ℓ is affected by fast fading evolving independently over time and every user.

The processing energy $\xi^{\texttt{pe}}$ for compressing data is obtained by exploiting the results from [45] that is characteristically expressed as,

$$\xi^{\texttt{pe}} = \xi_{\texttt{Proc}}^m \cdot \mathfrak{d}_\ell^m \cdot \mathfrak{C}_{\texttt{f}}(\, \{\mathfrak{C}_{\texttt{r}}\}_\ell^m \,) \tag{3}$$

where $\xi_{\texttt{Proc}}^m$ is the energy consumed per duty cycle of the thing's processor, $\mathfrak{d}_\ell^m$ is magnitude of data generated in frame m by node ℓ, $\mathfrak{C}_{\texttt{f}}$ is a function returning the number of duty cycles of the processor per bit required to compress the
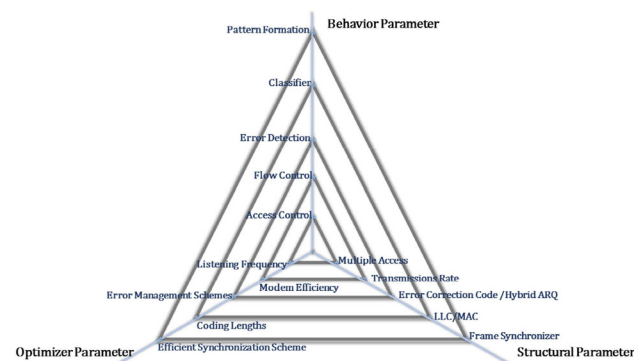
original signal and takes data compression ratio $\{\mathfrak{C}_r\}_\ell^m$ as an argument. The $\{\mathfrak{C}_r\}_\ell^m$ requires less energy for higher compression and depends on the algorithm employed. For example, temporal compression (TC) and low pass filter (LPF) algorithms can be approximated as linear viz., $\mathfrak{C}_f(\{\mathfrak{C}_r\}_\ell^m) = a_\ell^m \cdot \{\mathfrak{C}_r\}_\ell^m - b_\ell^m$ where a_ℓ^m and b_ℓ^m are coefficients [45]. Consequently the energy consumption becomes,

$$\xi^{\text{pe}} = \xi_{\text{Proc}}^m \cdot a_\ell^m \cdot \{\mathfrak{C}_r\}_\ell^m \cdot \mathfrak{d}_\ell^m + \xi_{\text{Proc}}^m \cdot \{\mathfrak{C}_r\}_\ell^m \cdot b_\ell^m$$

$$\text{where } \{\mathfrak{C}_r\}_\ell^m = \frac{\text{Uncompressed Size}}{\text{Compressed Size}}$$

$$\iff \{\mathfrak{C}_r\}_\ell^m = \frac{\mathfrak{d}_\ell^m}{\mathfrak{d}_\ell^{\hat{m}}}$$

$$(4)$$

In our model, we assume that $\mathfrak{C}_f$ to be increasing in $\{\mathfrak{C}_r\}_\ell^m$. The transmission energy ξ^{te} of a thing ℓ for a time interval $\mathcal{T}_\ell^m$ seconds having fixed power P_ℓ^m is calculated as,

$$\xi^{\text{te}} = \frac{\mathsf{P}_\ell^m \cdot \mathcal{T}_\ell^m}{\psi_{Ant}^m} \qquad (5)$$

where ψ_{Ant}^m is a constant term that is proportional to the efficiency of the power amplifier for antenna. The combined equipment circuitry sink and sensing energy ξ^{sd} consists of total energy expended to sense the environment and the eddy losses due to circuitry, such as the energy spent for node switches between sleep and active modes, the synchronization costs, as well as energy spent for transmission and given by,

$$\xi^{\text{sd}} = \mathfrak{R}_\ell^{\mathfrak{m}} \cdot \mathcal{T}_\ell^{\mathfrak{m}} + \mathfrak{c}_\ell^{\mathfrak{m}}$$
$$\text{where } c_\ell^m = s_\ell^m + e_\ell^m \qquad (6)$$

In Eq. 6, $\mathfrak{R}_\ell^{\mathfrak{m}}$ represents the rate of equipment circuitry power consumption for transmitting data and c_ℓ^m is a constant denoting the sensing s_ℓ^m and equipment e_ℓ^m sink costs respectively. It is important to note that there is no wastage of energy due to collisions and overlapping since TDMA-based MAC has been adopted in this model and therefore devices get exclusive access to communication channel for their respective slot (Fig. 1). The encryption/decryption cost [44] for data to be transmitted or received ξ^{ec} of a thing ℓ for a time interval $\mathcal{T}_\ell^m$ seconds having fixed power P_ℓ^m is calculated as,

$$\xi^{\text{te}} = \frac{\mathsf{P}_\ell^m \cdot \mathcal{T}_\ell^m}{\phi_{\text{size}}^m}$$
$$\text{where } \phi_{\text{size}}^m = \frac{\mu}{\mathsf{P}_{\text{load}}} \qquad (7)$$

where ϕ_{size}^m is a function that returns a term that is proportional to the pay load P_{load} and block size μ to be encrypted. For a block cipher, μ is the block size, while for a stream cipher, it represents the keystream block size which is the amount of keystream produced at one time to be used in the bit-wise XOR of μ bits in parallel. We apply a novel hybrid multi-objective optimizer algorithm WOABC to derive top η number of optimal minimized tuples specified in Eq. 1. The working of WOABC algorithm is described in Sect. 5.3.

5 WAOABC cross-layer optimizer framework

5.1 Whale optimization algorithm

We adopt a WAOABC framework proposed in [26] and apply it to extract a minimized tuple using fitness function defined in Eq. (1) by synthesizing the parameters discussed in delta diagram Fig. 5. In [26], the core function lies in closely mimicking the behavior of the whales in prey hunting. The WAO consists of two steps: The Encircling Step and the Prey Attacking Step. This algorithm closely mimics the social behavior of these big exquisite creatures that mostly move around in groups. These prey hunt for small schools of fish or krills close to the surface of the water body. Their method of scavenging for food is also known as the Bubble-Net feeding Method. In this bubble-net feeding method, it was observed by the researchers that the whales encircle their prey in the form of a spiral of bubbles that entraps the smaller fish. Two methods in which they do this is by the "Upward Spiral Method" and the "Double Loops" Method. The mathematical modeling of the whales hunting activities is described in the following steps.

5.1.1 Prey encircling

Once the prey is spotted by the humpback whales, they encircle it as shown in Fig. 6a. Theoretically, we are unaware of the position of the optimal design in the search space in the beginning. Thus it is assumed that the best solution in the current iteration is the prey or is at least close to it. Once this best search agent is established, the other ones try to update their positions towards the best. The following equations represent this behavior:
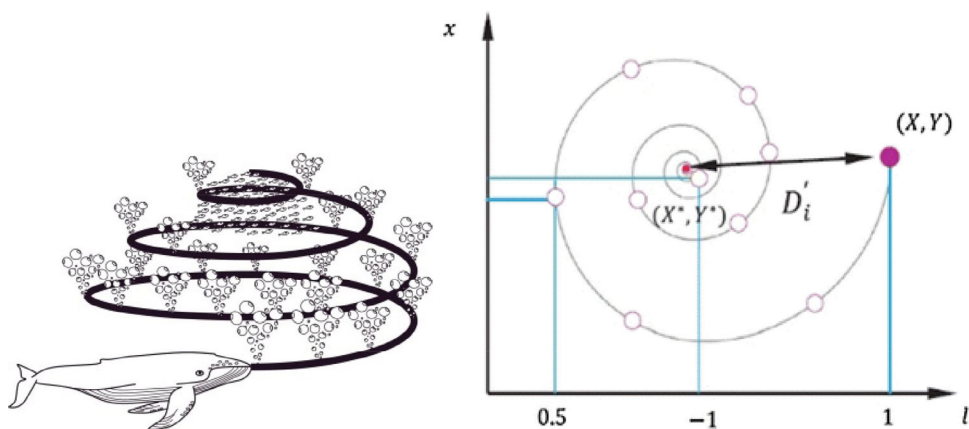
$$\overrightarrow{D} = |\ \overrightarrow{C} \times \overrightarrow{X}^*(t) - X(t)\ | \qquad (8)$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X^*}(t) - \overrightarrow{A} \times D \qquad (9)$$

where t is the current iteration, A and C are coefficient vectors and X^* is the position vector.

To calculate A and C, we use the following two equations:

Fig. 6 Whale optimization algorithm

(a) Bubble-net feeding behavior of humpback whales.

(b) Model of Bubble-net search spiral position updating model [28]

$$\overrightarrow{A} = 2\overrightarrow{a}\,\overrightarrow{r} - \overrightarrow{a} \tag{10}$$

$$\overrightarrow{C} = 2\overrightarrow{r} \tag{11}$$

$\overrightarrow{a}$ is reduced linearly from 2 to 0 and $\overrightarrow{r}$ is a random vector in the range [0,1]

5.1.2 Spiral update position

Once the encircling and the spiral formation is done, the next step of the humpback whales is to attack their prey as depicted in Fig. 6b. In the mathematical modeling of the WOA algorithm, the helix shaped movement of the whales is mimicked to create a spiral equation between the position of the prey and the whale. The equation is as follows:

$$\overrightarrow{X}(t+1) = \overrightarrow{D'} \cdot e^{b\ell} \cdot cos(2\pi\ell) + \overrightarrow{X^*}(t) \tag{12}$$

Here D' specifies the best solution so far, which is the distance of the ith whale to the prey. It is given by the following equation:

$$\overrightarrow{D} = |\overrightarrow{X^*}(t) - \overrightarrow{X}(t) \tag{13}$$

The logarithmic spiral is designated by b and l is a random number in [− 1,1]. The humpback whales swim around the prey in a spiral path in a shrinking circle, both at once. Thus a probability of 0.5 is assumed for choosing between the spiral model or the shrinking encircling method to update the position of the whales, which can be seen from the following equation,

$$\overrightarrow{X}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D} & \text{if } p < 0.5 \\ \overrightarrow{X}(t+1) = \overrightarrow{D'} \cdot e^{bl} \cdot cos(2\pi\ell) + \overrightarrow{X^*}(t) & \text{if } p > 0.5 \end{cases} \tag{14}$$

where p is a random number between [0,1]. A random search is conducted by the humpback whales in addition to

the bubble-net method where their positions keep getting updated according to those of the other search agents. This helps in finding the global optimal solution in the search space. $\overrightarrow{a} > 1$ and $\overrightarrow{a} < 1$ are used to move the whale away from the reference whale and is mathematically computed as:

$$\overrightarrow{D} = |\overrightarrow{C}\overrightarrow{X_{rand}} - X| \tag{15}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X_{rand}} - \overrightarrow{A} \cdot \overrightarrow{D} \tag{16}$$

5.2 Artificial Bee Colony algorithm

The Artificial Bee Colony (ABC) algorithm [20] consists of colonies of bees in three different groups namely the employed bees, the onlooker bees and the scout bees. There is just one employee bee per food source. In the colony, the first half are the employee bees and the other half are the onlooker bees. Scout bees are the bees who were employed before but their food sources ran out or were abandoned.

The food source with the most amount of nectar in it, is the best possible solution of the optimization problem. Here the amount of nectar in the food source correlates to the fitness of the solution. The number of solutions in the population are equal to the number of the employed or the onlooker bees. When the mathematical modeling of this algorithm is done, first the initial population is generated with various food source positions, which are also the solutions of the population. Each solution is a D dimensional vector where D represents the number of parameters of optimization. Once the initialization is complete, the employed,onlooker and the scout bees are subject to the search processes. The position of the food source and the nectar amount is embedded into the brain of the employed bee, who goes out in search for it. If the employed bee succeeds in finding a new food position with a greater

nectar amount in it, she forgets the old one otherwise she still remembers the old one. The nectar information given by the employed bee is evaluated by the onlooker bees and a food source is chosen. If the onlooker bee succeeds in finding a new food position with a greater nectar amount in it, she forgets the old one otherwise she still remembers the old one. The food source chosen by the onlooker bee depends on the probability value of that food source, which is calculated by the following equation:

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \qquad (17)$$

where SN is the total number of food sources which is the same as the number of employed bees. and fit_i is the value of the fitness function or the nectar amount in the food source. The candidate food position is updated using the following equation:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \qquad (18)$$

where k and j are randomly chosen indexes and k $\neq$ i, ϕ_{ij} is a random number between [− 1,1]. ϕ_{ij} represents a comparison of two food sources as seen visually by a bee and is also responsible for controlling the production of the food sources from the neighboring sites. As the difference between $(x_{ij} - x_{kj})$ decreases, it means that the search is approaching the most optimum solution. The Limit of Abandonment is a crucial control parameter of the ABC algorithm which is set up at the start of the algorithm. Now, we choose the best properties of the WAO and ABC algorithms to construct a hybrid WAOABC algorithm to shortlist η number of near optimal solutions for Eq. 1

5.3 Hybrid WOAABC algorithm

Researchers have been working on developing many hybrid approaches in Evolutionary Algorithms in order to enhance the exploration performance of the algorithms in the search space. Our attempt is to hybridize the Whale Optimization Algorithm with the Artificial Bee Colony algorithm to generate a hybrid which consists of functionalities of the two individual approaches. The capability of exploitation in the Artificial Bee Colony algorithm coupled with the capability of exploration in Whale Optimizer Algorithm is improved in hybridizing of both algorithms. Since in the WOA algorithm, the whales use the bubble-net foraging method to trap the prey, it is used in the exploration phase. The position of the whale that is supposed to find the most optimal solution is replaced with the position of the artificial bee, which is equivalent to the updated position of the whale but is more highly efficient to move the solution to the global best solution. As a result, with the mix of the best characteristics from both, the

WOA and the ABC algorithm, is it easier and guaranteed that the global best solution will be reached, thereby avoiding the local best or the local optima problems.

The mathematical modeling of the WOABC Algorithm is as follows: In WOA algorithm, the position of the humpback whale with probability greater than 0.5 is given by:

$$\overrightarrow{X}(t+1) = \overrightarrow{D\prime} \cdot e^{bl} \cdot cos(2\pi\ell) + \overrightarrow{X^*}(t) \qquad (19)$$

where l is a random number between [− 1,1], which along with 'b' defines the shape of the logarithmic spiral. In the ABC algorithm, the updated new position of the bee is given by the following equation:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \qquad (20)$$

where ϕ is responsible for comparing two food sources visually seen by the bee and also for controlling the creation of food sources from the neighboring sites. The ϕ from the ABC algorithm replaces the ℓ in the WOA algorithm. Since ϕ is also in the range of [− 1,1], it does not affect the ability of the whales to form the spirals in the bubble-net method, however, is enables them to improve the exploration method as the ϕ also looks in the neighboring sites for similar prey.

In the equation, $\overrightarrow{A} = 2\overrightarrow{a}\overrightarrow{r} - \overrightarrow{a}$, a represents the Shrinking Encircling Mechanism, which means that once the prey is circled around by the whales, they attack on it and thus the spiral shrinks in on them. This is replaced by the following equation from the ABC algorithm:

$$a = (2/\text{Max\_Iterations}) \qquad (21)$$

In the ABC algorithm, "a" signifies the acceleration coefficient upper bound. It decreases from 2 to 0. In order to update the whale position when the probability is less than 0.5, for the equation,

$$\overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D} \qquad (22)$$

The parameter A is replaced the ϕ from the ABC algorithm again, for further exploration purposes. The probability value in the WOA algorithm is taken as a random number,however in the ABC algorithm it is calculated as per the following equation:

$$\frac{fit_i}{\sum_{i=1}^{SN} fit_i} \qquad (23)$$

In the WOABC algorithm, the search for a better position for the whale is based on the probability found by the ABC algorithm rather than just a random number. This makes the search more detailed and thus better than the individual searches of both, the WOA or the ABC algorithm.

Now we apply the hybrid WOABC algorithm to find the near optimal solutions in the form of tuples that satisfy the fitness function in Eq. 1 using the pseudo code in Algorithm 1.

Algorithm 1 WOABC Pseudo Code

1: **procedure** WOABC
2:     [(x_0, x_1, x_2, x_3)] ¡- initialize()
3:     find the fitness of each of each search agent
4:     Initialize best solution to infinity
5:     **while** t **do** ¡ maximum number of iterations
6:         Discard(SearchSpace > BeyondTheBoundary (Search Agents))
7:         Calculate objective function of the search space using Eq. 1
8:         Update the leader
9:         Update the position A of the Search Agents
10:         Calculate probability p for search agents using ABC
11:         **if** $p < 0.5$ **then**
12:             **if** $ABS(A) < 1$ **then**
13:                 Update new position of bee as position of the whale
14:             **else if** $ABS(A) > 1$ **then**
15:                 Select a random search member
16:                 Update the position of the whale with ϕ instead of A
17:             **end if**
18:         **else if** $p > 0.5$ **then**
19:             Update position at current search agent using Cosine Eqn.
20:         **end if**
21:         Find fitness of all search members using Eq. 1
22:         Update new bee position, best search agent & best energy score
23:         $t < -t + 1$
24:     **end while**
25: **return** [$(\xi_{min}^{pe}, \xi_{min}^{te}, \xi_{min}^{sd}, \xi_{min}^{ec})$] with magnitude η
26: **end procedure**

---

6 Results and analysis

The intel-LEACH protocol [34] was incorporated to test our framework from synthesis to optimization steps in practical point-to-point IoT scenarios. The operation is briefly listed in following phases.

1. Transmission phase

   - Check route validity from point to point nodes and following initialization routines of MAC operation described in Sect. 4.
   - Failure mitigation by generating Route Request (RR) packet containing the destination thing ID directed towards nearest GAP.

2. Service phase

   - GAP transmits its ID in broadcast mode periodically
   - GAP has sufficiently large power to directly communicate with every thing in its domain (Sect. 3.2.2).
   - The things register themselves to GAP with Network Association (NAS) packet.

3. Messaging phase

   - Upon receiving packet data, the thing sends a Route Acknowledge (RA) packet to the previous hop in the route to show its alive.
   - The above process is repeated in multi-hops scenario until source is reached.
   - Data is transmitted by following the optimal route with the chosen communication parameters defined by delta diagram in Fig. 5 and according to description in Sect. 4.
   - Computation complexity is shifted from things to GAP which reduces multi-flow problems.

4. Routing phase

   - GAP receives the RR packet via several paths, the intermediate nodes are earmarked as priority candidates for data transmission
   - GAP initiates the WOABC framework for potential paths and QoS requirements to find the near optimal path and the associated communication parameters, as explained in Sect. 5.3.

The performance of the hybrid WOABC MAC optimizer framework and state of the art solutions was compared and analyzed by setting up the network with different parameters and methodologies described as follows.

7 Energy analysis

A critical cost analysis for energy benchmarking WOABC-Crosslayer Framework with traditional crosslayer models having different modulation levels is depicted in Fig. 8. It can clearly interpreted from Fig. 8 that our model based on delta diagram outlasts all the other models in terms of network longevity and energy conservation.
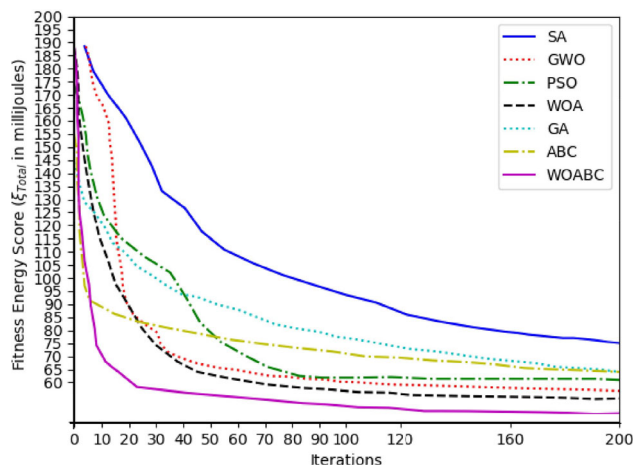

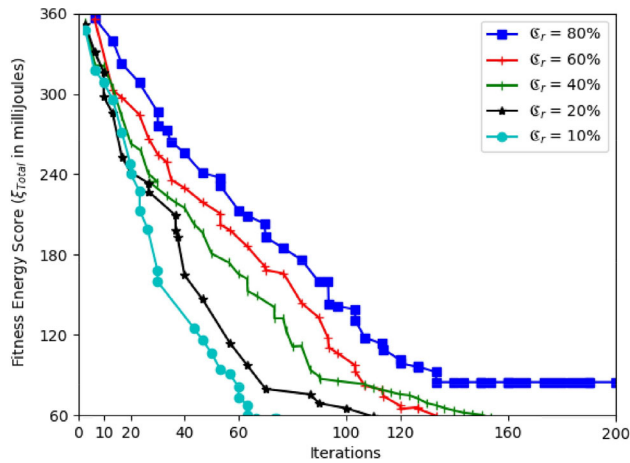**Fig. 7** Comparison of energy dissipation among various optimization algorithms

Fig. 8 Energy dissipation rate over iterations for various compression ratios of Eq. 1

The comparison of energy dissipation among various optimization algorithms is shown in Fig. 7. We begin the study of evolutionary algorithms by applying the multi-objective optimization (MOO) for energy factor described in fitness function of Eq. 1. Initially we choose MOO algorithms such as Simulated Annealing (SA) algorithm and the Genetic Algorithms (GA) for our model. However, SA algorithm suffers from extreme slowness thereby searching for an optimal solution is not very efficient and feasible. Thus as depicted in Fig 9, the curve for SA predicates worst performance among all other comparative algorithms for energy model in Eq. 1. Although relatively the Genetic Algorithm (GA) performs better than SA, it encounters issues about not only termination time, but also with the convergence rate. This is due to the fact that GA has a major drawback of getting stuck in local minima, making it unsuitable for multi-objective based optimization problems. Particle Swarm Optimization (PSO) overcomes
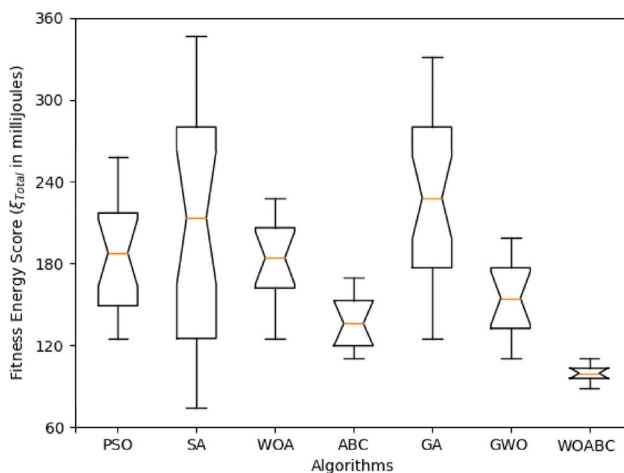
the above challenges faced by SA and GA, however it still falls short with problems related to high dimensional space. Furthermore, Hybrid Grey Wolf Optimizer Sine Cosine Algorithm (HGWOSCA) [36], Grey Wolf Optimizer (GWO) [33] and Artificial Bee Colony (ABC) [20] methods overcome most of the problems faced by other algorithms described earlier. But the WOABC based framework described in Alg. 1 performs optimally for our the specific context and able to find the best possible fitness energy scores with least number of iterations. The Fig. 9 show that the box-plot of WOABC is significantly lower and narrower than other algorithms. Figure 9 shows that the box-plot of SA is super narrow implying worst suitability for our model; while WOABC is under the minima of other algorithms. This means that WOABC tends to find the global minimum and significantly outperforms other algorithms.

8 Conclusion

In this article, we presented the novel energy optimized design for MAC-Layer of the IoT protocol stack. The approach uses a Delta-diagram based synthesizer to identify the energy parameters of MAC layer to be considered for optimization. The parameters chosen are completely dependent on the type of network and topology of IoT setup and hence it is highly customizable. Furthermore, synthesis process enables parsing differing levels of parameters and moving across different domains of Behavior, Structural and Optimizer requirements of the chosen protocol for IoT.

We constructed a model for deriving the fitness function which simultaneously minimizes energy requirements across different paradigms for the MAC layer of network protocol subjected to the QoS constraints.

We introduced a hybrid WOABC algorithm to optimize and search the near optimal minimized tuple resulting from model search space. The results and analysis of the methodology shows that our approach using a test protocol outperforms other cross-layered approaches significantly and is highly flexible in nature so that it can adapted to any type QoS requirements of different type of MAC layers across various network protocols.

Fig. 9 Comparative tests of algorithms for delta diagram synthesizer using box-plot

References

1. Ahmad A, Hanzálek Z (2018) An energy efficient schedule for IEEE 802.15.4/ZigBee cluster tree wsn with multiple collision domains and period crossing constraint. IEEE Trans Ind Inf 14(1):12–23

2. Akyildiz IF, Vuran MC (2010) XLP: a cross-layer protocol for efficient communication in wireless sensor networks. IEEE Trans Mobile Comput 9:1578–1591

3. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M (2016) Internet of things: a survey on enabling technologies, protocols, and applications. IEEE Commun Surv Tutor 17(4):2347–2376

4. Alfayez F, Hammoudeh M, Abuarqoub A (2015) A survey on MAC protocols for duty-cycled wireless sensor networks. Procedia Comput Sci 73:482–489

5. Chai F, Zhu T, Kang KD (2016) A link-correlation-aware cross-layer protocol for IOT devices. In: 2016 IEEE international conference on communications (ICC), pp 1–6

6. Dixon C, Mahajan R, Agarwal S, Brush AJ, Lee B, Saroiu S, Bahl P (2012) An operating system for the home. In: Proceedings of the 9th USENIX conference on networked systems design and implementation. NSDI'12, USENIX Association, Berkeley, CA, USA, pp 25–25

7. Dong M, Ota K, Liu A, Guo M (2016) Joint optimization of lifetime and transport delay under reliability constraint wireless sensor networks. IEEE Trans Parallel Distrib Syst 27(1):225–236

8. Dziri A, Ammar AB, Terre M (June 2017) Performance analysis of mimo cooperative relays for wireless sensor networks. In: 2017 13th international wireless communications and mobile computing conference (IWCMC), pp 2029–2033

9. El Amine Seddik M, Toldov V, Clavier L, Mitton N (2018) From outage probability to ALOHA MAC layer performance analysis in distributed WSNs. In: WCNC 2018—IEEE wireless communications and networking conference. Barcelona, Spain. https://hal.inria.fr/hal-01677687

10. Essa AA, Zhang X, Wu P, Abuzneid A (2017) Zigbee network using low power techniques and modified leach protocol. In: 2017 IEEE long island systems, applications and technology conference (LISAT), pp 1–5

11. Fröhlich AA, Okazaki AM, Steiner RV, Oliveira P, Martina JE (2013) A cross-layer approach to trustfulness in the internet of things. In: 16th IEEE international symposium on object/component/service-oriented real-time distributed computing (ISORC 2013), pp 1–8

12. Granjal J, Monteiro E, Silva JS (2015) Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun Surv Tutor 17(3):1294–1312

13. Guowang M (2016) Fundamentals of mobile data networks, 1st edn. Cambridge University Press, Cambridge

14. Gutierrez JA, Naeve M, Callaway E, Bourgeois M, Mitter V, Heile B (2001) IEEE 802.15.4: A developing standard for low-power low-cost wireless personal area networks. IEEE Netw 15(5):12–19

15. Han C, Jornet JM, Fadel E, Akyildiz IF (2013) A cross-layer communication module for the internet of things. Comput Netw 57(3):622–633

16. Hansen CJ (2015) Internetworking with bluetooth low energy. GetMobile Mobile Comput Commun 19(2):34–38

17. Hermeto RT, Gallais A, Theoleyre F (2017) Scheduling for IEEE802.15.4-TSCH and slow channel hopping MAC in low power industrial wireless networks: a survey. Comput Commun 114:84–105

18. Huang P, Xiao L, Soltani S, Mutka MW, Xi N (2013) The evolution of MAC protocols in wireless sensor networks: a survey. IEEE Commun Surv Tutor 15(1):101–120

19. Kafi MA, Ben-Othman J, Ouadjaout A, Bagaa M, Badache N (2017) Refiacc: reliable, efficient, fair and interference-aware congestion control protocol for wireless sensor networks. Comput Commun 101:1–11

20. Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J Glob Optim 39(3):459–471

21. Latchman HA, Katar S, Yonge L, Gavette S (2013) Homeplug AV and IEEE 1901: a handbook for PLC designers and users, 1st edn. Wiley-IEEE Press, Hoboken

22. Levis P, Patel N, Culler D, Shenker S (2004) Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks. In: Proceedings of the first USENIX/ACM symposium on networked systems design and implementation (NSDI), pp 15–28

23. Majumder AB, Gupta S (2018) An energy-efficient congestion avoidance priority-based routing algorithm for body area network. In: Bhattacharyya S, Sen S, Dutta M, Biswas P, Chattopadhyay H (eds) Industry interactive innovations in science, engineering and technology. Springer, Singapore, pp 545–552

24. Malik A, Qadir J, Ahmad B, Yau KLA, Ullah U (2015) QOS in IEEE 802.11-based wireless networks: a contemporary review. J Netw Comput Appl 55:24–46

25. Mendes LD, Rodrigues JJ (2011) A survey on cross-layer solutions for wireless sensor networks. J Netw Comput Appl 34(2):523–534

26. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67

27. Park M (2015) IEEE 802.11ah: sub-1-GHz license-exempt operation for the internet of things. IEEE Commun Mag 53(9):145–151

28. Patil M, Biradar RC (2012) A survey on routing protocols in wireless sensor networks. In: 2012 18th IEEE international conference on networks (ICON), pp 86–91

29. Ray PP, Agarwal S (2016) Bluetooth 5 and internet of things: potential and architecture. In: 2016 international conference on signal processing, communication, power and embedded system (SCOPES), pp 1461–1465

30. Ren J, Zhang Y, Zhang K, Liu A, Chen J, Shen XS (2016) Lifetime and energy hole evolution analysis in data-gathering wireless sensor networks. IEEE Trans Ind Inf 12(2):788–800

31. Resner D, d. Araujo GM, Fröhlich AA (2016) On the impact of dynamic routing metrics on a geographic protocol for wsns. In: 2016 VI Brazilian symposium on computing systems engineering (SBESC), pp 109–115

32. Sah DK, Amgoth T (2018) Parametric survey on cross-layer designs for wireless sensor networks. Comput Sci Rev 27:112–134

33. Siddavaatam P, Sedaghat R, Sharma AT (2017a) Grey wolf optimizer driven design space exploration: a novel framework for multi-objective trade-off in architectural synthesis. Microelectron Reliab 55(6):990–1004

34. Siddavaatam P, Sedaghat R, Sharma AT (2017b) Intel-leach: an optimal framework for node selection using dynamic clustering for wireless sensor networks. In: 2017 12th IEEE international conference for internet technology and secured transactions (ICITST), pp 136–146

35. Siddavaatam P, Sedaghat R (2018) A delta-diagram based synthesis for cross layer optimization modeling of IoT. Springer, Berlin, pp 1–24

36. Singh N, Singh S (2017) A novel hybrid gwo-sca approach for optimization problems. Int J Eng Sci Technol 20(6):1586–1601

37. Sinha RS, Wei Y, Hwang SH (2017) A survey on LPWA technology: LoRa and NB-IoT. ICT Express 3(1):14–21

38. Sloane NJA, Wyner AD (1993) Coding theorems for a discrete source with a fidelity criterion. Institute of radio engineers, international convention record, vol 7, 1959. Wiley-IEEE Press, p 968

39. Tanenbaum A, Wetherall D (2011) Computer networks. Pearson Prentice Hall, Upper Saddle River

40. Vilajosana X, Wang Q, Chraim F, Watteyne T, Chang T, Pister KSJ (2014) A realistic energy consumption model for TSCH networks. IEEE Sens J 14(2):482–489

41. Wu M, Tan L, Xiong N (2016) Data prediction, compression, and recovery in clustered wireless sensor networks for environmental monitoring applications. Inf Sci 329:800–818

42. Yajnanarayana V, Magnusson KEG, Brandt R, Dwivedi S, Händel P (2017) Optimal scheduling for interference mitigation by range information. IEEE Trans Mobile Comput 16(11):3167–3181

43. Zhang H, Quan W, Song J, Jiang Z, Yu S (2016) Link state prediction-based reliable transmission for high-speed railway networks. IEEE Trans Veh Technol 65(12):9617–9629

44. Zhang X, Heys HM, Li C (2010) Energy efficiency of symmetric key cryptographic algorithms in wireless sensor networks. In: 2010 25th Biennial symposium on communications, pp 168–172

45. Zordan D, Martinez B, Vilajosana I, Rossi M (2014) On the performance of lossy compression schemes for energy constrained sensor networking. ACM Trans Sen Netw 11(1):15:1–15:34

46. Zuo J, Dong C, Ng SX, Yang LL, Hanzo L (2015) Cross-layer aided energy-efficient routing design for ad hoc networks. IEEE Commun Surv Tutor 17(3):1214–1238