A Novel Duty Cycle Based Cross Layer Model for Energy Efficient Routing in IWSN Based IoT Application

Ghanshyam Singh[1*], Pallavi Joshi[1], and Ajay Singh Raghuvanshi[1]
[1]Electronics and Communication Department, National Institute of Technology, Raipur (C.G.), India 492010
[e-mail: gsingh.phd2016.etc@nitrr..ac.in, pjoshi.phd2017.etc@nitrr..ac.in, asraghuvanshi.etc@nitrr..ac.in]
[*]Corresponding author: Ghanshyam Singh

Received June 21, 2021; revised October 31, 2021; revised March 3, 2022; revised May 21, 2022; accepted June 9, 2022; published June 30, 2022

Abstract

Wireless Sensor Network (WSN) is considered as an integral part of the Internet of Things (IoT) for collecting real-time data from the site having many applications in industry 4.0 and smart cities. The task of nodes is to sense the environment and send the relevant information over the internet. Though this task seems very straightforward but it is vulnerable to certain issues like energy consumption, delay, throughput, etc. To efficiently address these issues, this work develops a cross-layer model for the optimization between MAC and the Network layer of the OSI model for WSN. A high value of duty cycle for nodes is selected to control the delay and further enhances data transmission reliability. A node measurement prediction system based on the Kalman filter has been introduced, which uses the constraint based on covariance value to decide the scheduling scheme of the nodes. The concept of duty cycle for node scheduling is employed with a greedy data forwarding scheme. The proposed Duty Cycle-based Greedy Routing (DCGR) scheme aims to minimize the hop count, thereby mitigating the energy consumption rate. The proposed algorithm is tested using a real-world wastewater treatment dataset. The proposed method marks an 87.5% increase in the energy efficiency and reduction in the network latency by 61% when validated with other similar pre-existing schemes.

Keywords: Cross-layer optimization, Duty cycle, Greedy routing, Kalman filter, Node scheduling.

1. Introduction

The Internet of things (IoT) technology enables a way to connect remote objects or people, or other cyber-physical things with ease. Industrial wireless sensor network (IWSN) is also one of the vital applications of IoT, which can be visualized as an integral part of Industry 4.0, which includes cloud computing, cyber-physical systems and IoT [1]. In this context, the wireless sensor network as a part of IWSN is a ubiquitous yet powerful technology used to sense, collect, aggregate and fuse the data to a central controller. Many IWSN applications like e-healthcare, environment and habitat monitoring, smart transportation, etc., are defined by the structure, standards and models used in wireless technologies. One of the primary challenges in designing a WSN is the implementation of cross-layer (CL) optimization. **Fig. 1** gives the cross-layer framework used in this paper. It shows various layers in the protocol stack, the models and standards used and how they are interconnected to form cross-layer modules. Various layers have their own importance; like the physical layer focuses on data frames and their structures, the MAC layer gives information about accessing the medium [2]. The MAC layer also provides some standards for efficient MAC protocol working.

The network layer emphasizes the paths for forwarding and transmission of data. It uses various routing protocols for this task. The transport layer uses different protocols for controlling the traffic and ensures reliable delivery in the transmission of packets. It also manages the resources well. Finally, the application layer provides the user with the interface to access and operate the data. Every layer is connected to its consecutive layer and vice versa, making a transmission and reception system structure. The transmission system is such that the flow of the data is from the application layer, then the transport layer and so on, and finally to the physical layer. In contrast, the flow is just opposite in the receiving node, i.e., from physical towards application. The cross-layer model manages all these operations. The balance and optimization in the cross-layer can be enhanced by adopting various efficient methods at each layer. This paper deals with the optimization in the two layers: MAC and Network layer.

In [3], the authors present cooperation between the Network and MAC layers to precisely enable the wake-up scheduling in the nodes in multimedia WSN. The data forwarding adopts multi-path routing for extensive size networks. The method focuses on enhancing the system throughput and speed of data transmission. Cross-layer architectures for cognitive radio networks have been discussed in [4]. It gives layer-centric and centralized solutions for cross-layer design problems. It aims to hike the quality-of-service parameters in the network. A survey on various cross-layer architectures in WSN has been discussed in [5]. The paper gives an extensive parameter-oriented review from various research papers on cross-layer designs (CLD). It provides a classification of the CLDs based on the different quality of service metrics like security, energy optimization, etc. The duty-cycled MAC scheduling is of two types: Synchronous and Asynchronous. Synchronous scheduling schemes such as T-MAC, S-MAC require synchronized scheduling of the nodes with their neighbor for timely awake and sleep actions. This way, the communication between the nodes is synchronized. But such type of scheduling is not appropriate for dynamic, energy constraint networks as these may face some issues like complexity in computation and communication overhead. The popular MAC scheduling schemes are based on asynchronous protocols where no time synchronization among the nodes is required; instead, the receiver checks for the transmission by waking up periodically. The protocols such as X-MAC [6] and PB-MAC [7] fall in the asynchronous scheduling scheme. Shi et al. [8] presents an asynchronous duty cycle-based approach in a cross-layer WSN model to enhance QoS.

The asynchronous protocols are prevalent and they can handle heterogeneous and resource constraint networks. Since this paper deals with duty cycle-controlled scheduling, asynchronous scheduling serves as a motivation for our work. The varying duty cycles can be determined by asynchronous scheduling protocol as it helps maintain the sleep and awake times, thereby enhancing the system's reliability and throughput.

The original contributions of this paper are that we demonstrate the efficient clustering of a 500-node wireless network using k-means clustering and particle swarm optimization. This combination gives a more accurate cluster head selection and makes the clustering very efficient by reducing the unnecessary transmission delay caused due to the absence of clustering. The network's energy is limited and thus a Kalman filter-based data prediction scheme is adopted during the sensing of the data by the nodes. This helps in the efficient collection of data at the aggregators as well as at the base station node. We propose a cross-layer data transmission scheme using duty-cycled scheduling at the MAC layer and optimizes the network layer parameters. The variation in the duty cycle regulates the sleep and awake durations of the nodes. Thus, the scheduling of the nodes which contribute their information to the destination is done with ease.

The organization of the paper can be summarized as: In section 2, the related work is discussed. It deals with cross-layer related researches, which enhances the efficiency of the WSN and IoT networks. Section 3 gives the proposed model overview and its framework in the diagram form. Section 4 deals with the explanation of various analytical approaches used in the proposed model. The working of the proposed model is presented in section 5. Section 6 discusses the results and visualizes the results in the form of different analytical graphs. Section 7 concludes the whole work with future scope.

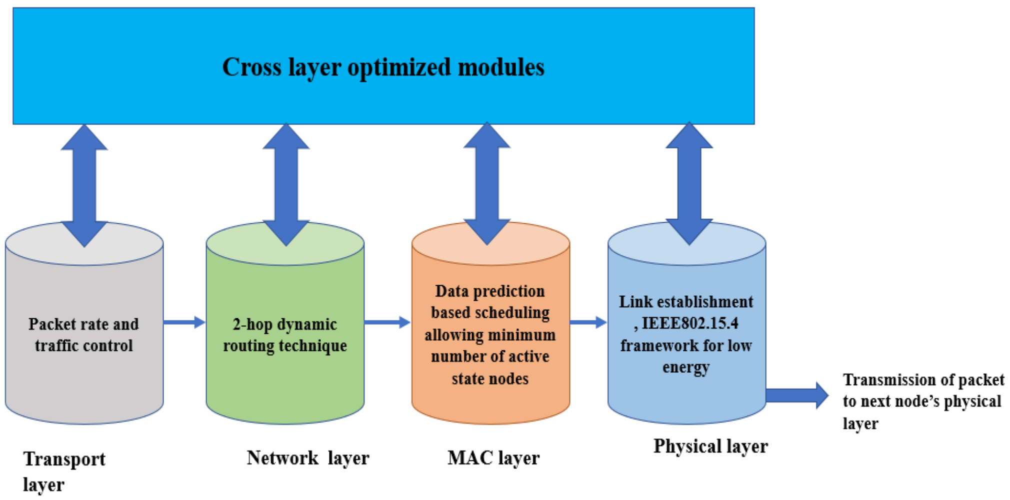

Fig. 1. Cross-layer framework used in proposed work

2. Related Works

This section covers the research relevant to the clustering technique, duty cycle scheduling approach and cross-layer optimization. The most vital thing is to achieve the optimized network performance indicators using scheduling and cross-layer design in a clustered WSN. Apart from this, one of the major challenges seen in the sensor networks in many industrial applications is the redundancy of the data. The final destination or the processing unit needs only useful data. Since the data collected at each interval may contain repeated or unusual values, the data estimation must be done to filter out the useless data, those causing energy and memory consumption and increased delay in the transmission. This enhances the reliability of the network and makes it more efficient. We produce literature of those works which focus on these paradigms.

The placement of the nodes in a sensor network has its importance. There are some techniques that can be used to enhance the sensing coverage and connectivity of the sensor nodes by placing them in the appropriate location so that the distance between the nodes and the base station can be optimized. One of the techniques is clustering. Some approaches like low energy adaptive (LEACH) clustering hierarchy [9], distributed energy-efficient (DEEC) clustering [10] and many more give an idea to route the data packets using clustering. Smaragdakis et al. [11]. Efficient routing can be achieved by using multi-path protocols. Jemili et al. [12] proposed a fast multi-path routing (FMRP) protocol that avoids correlated transmission paths by constructing the disjoint paths in the wireless sensor network to eliminate the collision during the transmission. In [3], the multi-path routing and the cross-layer management scheme that uses duty-cycle scheduling at MAC are investigated for the multimedia sensor network and compared with the FMRP. The proposed scheme ensures simultaneous data transmission by considering non-correlated and disjoint paths to avoid collisions; this method is named as cross-layer correlation-free (CL-CFMPR) multi-path routing technique.

The scheduling in nodes is another aspect of reducing the energy consumption and enhancing the network lifetime. The scheduling task divides the duration of the nodes into active and inactive periods, such that the node will be in a sleep mode when it is not required in data transmission or other tasks of the network. This saves an appreciable amount of energy and time in the network and enhances reliability and node efficiency. Many research works use sleep-awake scheduling, duty-cycle scheduling, which uses varying duty-cycles to obtain optimized QoS parameters. To improve the network coverage, Coverage guaranteed Distributed Sleep awake Scheduling (CDSWS) is applied in [13], which aims to conserve energy by regulating the sleep and awake durations of the nodes in each cluster. It works on three phases that are initialization, cluster formation and sleep-awake scheduling. Shu et al. [14] proposed another method of enhancing the connectivity of the k-neighborhood nodes (GCKN) using sleep scheduling based on geographic distance in IWSN. This proposed scheme solves the network isolation problem, which is responsible for the fast drain of energy of nodes and sleep scheduling problem in which nodes don't get the opportunity to sleep even if they are not active. This method enhances the network lifetime by approximately 41% when compared with other similar techniques.

In [15], the authors use the 2-phase geographic greedy forwarding algorithm for multi-path routing along with the cross-layer optimization between physical, MAC and network layers. The physical layer contributes to the energy harvesting in the nodes depending on their energy levels, the MAC layer implements the duty-cycled scheduling of the nodes to save the energy of the network and the greedy forwarding scheme for routing is applied on the duty-cycled scheduled nodes so that the nodes will be awake whenever it is required. This paper uses asynchronous duty cycle-based MAC scheduling, which adopts the estimation theory using Kalman filter similar to the one used in [16] to select the essential node with useful information to reduce unnecessary delay in the network.

The [3], cross-layer optimization proves itself to be one of the dominant features in solving various issues of WSN like degradation of QoS parameter, transmission energy requirement, network latency and many more. In [5], the authors have surveyed various cross-layer optimization frameworks from a QoS parameter perspective. The paper discusses multiple CL design models using different optimization approaches like non-linear constraint (like Karush-Kuhn Tucker condition) [17], linear constraint, etc., in cross-layer designs. Various challenges in WSN can be solved by adopting a cross-layer optimization framework.

In [18], a cross-layer model is designed to optimize the node's transmission range. It uses a threshold called the expected transmission range (ETRT) threshold. The paper deals with three layers, i.e., physical, data link and network layers in an IoT-enabled WSN protocol stack. In [19], the dynamic routing protocol along with the sleep scheduling protocol of MAC is combined and implemented via a cross-layer interaction platform in a web-based multi-hop WSN to enhance the system's performance metrics. To get the best cooperation between MAC and routing layers, [3] and [15] adopted a multi-path routing technique that also employs duty cycle mode. The proposed method promises 50% enhancement in the goodput when compared with other similar techniques.

Dao et al. [20] introduced a scheme called Deadline Aware Scheduling and Forwarding (DASF) scheme and Vu et al. [21] proposed a scheduling scheme called Delay constraint Duty-cycled Scheduling (DDS), both aim to reduce end-to-end delay and energy consumption in a duty-cycled WSN. These two methods, i.e., DASF and DDS, use the central limit theorem for determining the delay distributions of each group of nodes. The duty cycle is calculated using these distributions, which helps mitigate the network's energy utilization. Scheduling for low duty-cycled WSN called Robust Multi-pipeline Scheduling (RMS) is proposed in [22]. The multi-pipeline scheduling technique is implemented to overcome unreliable communication links and guarantee the timely delivery of the packets in low-duty-cycled sensor networks. A CL design called Reducing Delay and Maximizing Lifetime (RDML) scheme is proposed in [23], which uses a large duty cycle for residual energy nods to improve the reliability of transmission and to mitigate the probability of retransmission in an IWSN. The method gives a 43% enhancement in network lifetime.

Xu et al. [24] proposed a Cross-layer Optimized Opportunistic Routing (COOR) scheme to improve reliability of communication link and minimize the delay. In this scheme, two optimization techniques used to choose preferred node. One method chooses the node having higher communication reliability and second one prefer the node having longer communication range. Through these methods number of hops is reduced and reliability increased. A cross-layer mechanism is proposed in [25] for energy efficient cluster routing. In which the node selection is provided by fuzzy based CH selection method. To reduce congestion occurrence, network is combined through non-similar clusters. To achieve energy efficient transmission ABC optimization algorithm is utilized. A cross-layer based QoS aware routing protocol is proposed in [26] by utilizing multi objective ant colony optimization. The cross-layer approach is used for relay node selection by enabling interation between MAC and routing layers. Lahane et al. [27] proposed a cross-layer routing approach based on hybrid clustering. In this approach the optimal selection of cluster head is achieved by moth flame integrated dragonfly algorithm taking consideration of parameters as energy consumption, throughput, delay etc.

The above methods are up-and-coming in the cross-layer domain of WSN and IoT networks. Some method focuses on varying the duty cycle to control the end-to-end delay, while some consider efficient routing methods to enhance the performance of the sensor network. Our paper focuses on the duty-cycled scheduling and delay bound aspects like optimized clustering to cover maximum performance metrics to prove the model's efficacy in real-time applications. These techniques give novelty to the proposed model and make it distinct from other methods. Also, a cross-layer optimization framework is implemented to optimize the functioning between different layers of the network.

3. Proposed model

The experimentation is done on a water treatment dataset in which all the water pollutants and other water treating factors are monitored in a water treatment plant. It is designed for the industry-based application, i.e., water treatment plant. As shown in **Fig. 2**, the sensor nodes are deployed in a water treatment plant area. Each unit senses the parameters of that particular unit and aggregates the information, which is finally going to a fusion center or base station. A real-time data set for a water treatment plant is used in this paper. The sensor nodes are deployed in a real-time water treatment plant and the clusters are made. Each cluster has its own cluster head. All the cluster members send their data which is the estimated value of the measured data. In this paper, a Kalman filter-based estimation approach is adopted to predict the data to avoid the redundancy in data transmission and boost up the system's speed.

The nodes are differentiated as affected and normal nodes based on the threshold defined on their estimated values. The scheduling scheme based on the duty cycle is adopted for the nodes to differentiate them. The normal nodes for which correct prediction is computed are considered to have 75% duty-cycle based sleep-wake scheduling, i.e., alternate sleep and awake scheduling in an epoch. Initially, all nodes are programmed with a 75% duty cycle at MAC. After considering the constraints for predicted data values, the duty cycle of selected nodes is changed. The data is transmitted from the affected nodes to the cluster heads and from the cluster heads to the base station. The base station is connected with the cloud and many IoT devices access the data from the cloud for further processing. This type of scenario is used in the application where any event has to be detected in a water treatment plant. If the estimation for data of any node within a cluster is not correct, then that node is identified and the unit associated with that node or the region is acutely monitored to avoid any accidents.
Following assumptions are considered in this paper:

1. The transmission range and the communication range of each node are fixed.
2. The communication range of the cluster heads must be such that all the nodes in that cluster must be in their transmission radius.
3. The initial energy of all the nodes as well as the cluster heads is kept constant.
4. The duty cycle scheduling governs the time taken for a data packet to be transmitted and received.
5. The nodes are assumed to take a maximum of 2-hop routing paths to reach the destination node.

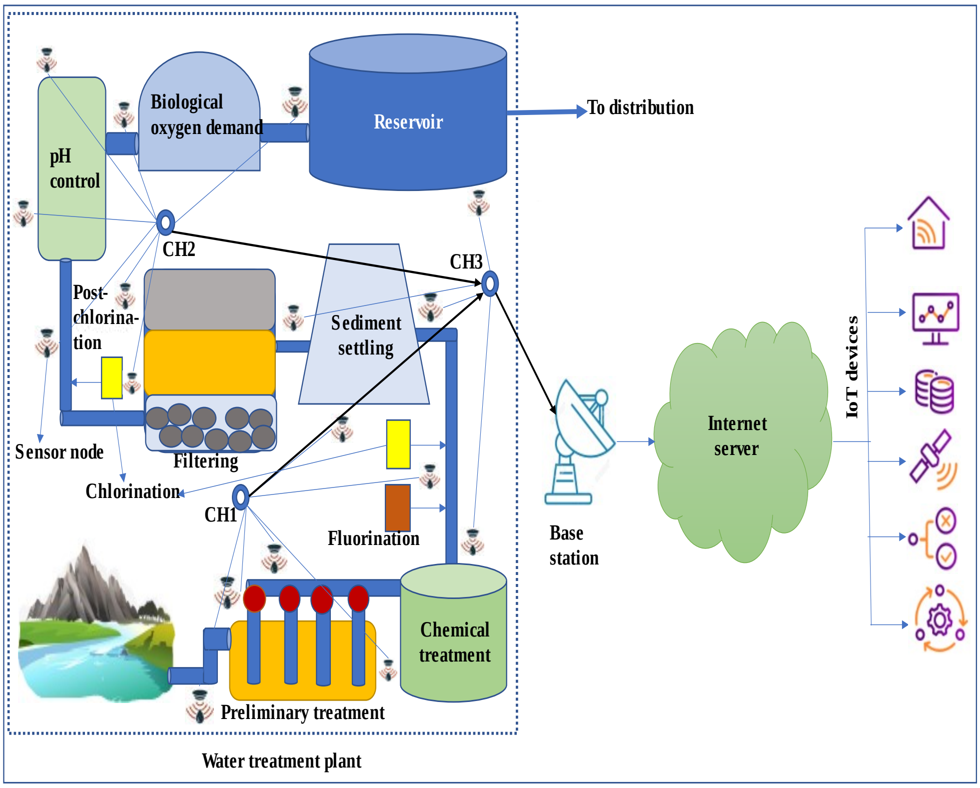

Fig. 2. Framework of the proposed model

4. Analytical approaches used in the proposed model

4.1 Node deployment using clustering

It employs a random node deployment strategy. The physical layer of the WSN architecture takes care of the energy handling issues. For this purpose, the IEEE 802.15.4 standard is used to ensure low communication cost as well as minimum energy requirement for nodes for low data rates.

K-means clustering with PSO optimization

The k-means algorithm is a clustering algorithm firstly proposed by Stuart Lloyd in 1957. This algorithm groups the data points following a similar pattern or similarity measure like distance and divides them into clusters based on these measures. In this paper, the Euclidean distance between each node belonging to a cluster is made minimum. It also aims to find the best centroid points for each cluster called cluster heads.

PSO is a swarm intelligence algorithm that works on the theory of bird flocking. The initial step is to create a population of birds over a search space. The population consists of particles (birds). Each particle has a fixed velocity and position and the position is updated every time. The velocity is updated using particle momentum, the best position reached and the number of particles. This updating leads the particles to reach the next position.

PSO-KM is the optimization approach adopted to make the clusters in a global optimal region. The K-means clustering is applied to the search spaces to form clusters and cluster heads. The slow convergence of the PSO algorithm has been compensated by the fast convergence rate of the K-means algorithm. The PSO initially starts the optimization of clustering by considering the locations of the cluster heads and then the position and velocity of these cluster centroids are optimized to have an easy transmission of data from clusters to the base station. The steps followed by the hybrid PSO-KM algorithm are as under:

Step 1: Implementing K-means clustering algorithm till one of the following conditions is achieved when-

a) Iteration number surpasses the predefined max value.
b) The cluster centroids cannot be changed.
c) Cluster membership does not change.

*Step 2:*The points are updated using equations:

$$V_j^{(i)} = \propto * V_j^{(i-1)} + c_1 * r_1\left(B_j - Y_j^{(i-1)}\right) + c_2 * r_2\left(G - Y_j^{(i-1)}\right) \quad (1)$$

$$Y_j^{(i)} = Y_j^{(i-1)} + V_j^{(i)} \quad (2)$$

Where $Y_j^{(i)} = \left(Y_{j1}^{(i)}, Y_{j2}^{(i)}, \dots, Y_{jm}^{(i)}\right)$ is the position vector of jth particle at ith iteration in m-dimensional search space. Similarly, $V_j^{(i)} = \left(V_{j1}^{(i)}, V_{j2}^{(i)}, \dots, V_{jm}^{(i)}\right)$ is the velocity vector of jth particle at ith iteration in m-dimensional search space. $B = \left(B_{j1}, B_{j2}, \dots, B_{jm}\right)$is the personal bet position of particle 'j' and $G = (g_1, g_2, \dots, g_m)$ is the global best for all the particles in the search space. '$\propto$' is the inertia weight which controls the impact of the previous velocity of a particle on the current velocity. c_1, c_2 are the acceleration coefficients and r_1, r_2 are the random variables uniformly distributed in the range [0,1].

Step 3: After updating the particle position and velocity, each particle is considered as a clustering problem solution. In terms of clustering, each datapoint represents the cluster head. So, the i^{th} data point can be initialized as:

$$Y_j^{(0)} = \left(p_{j1}^{(0)}, p_{j2}^{(0)}, \dots, p_{jk}^{(0)}\right) \quad (3)$$

Where $z_{jk}^{(0)}$ is the k^{th} cluster head in the solution given by the j^{th} datapoint.

Step 4: Evaluating the objective function for datapoint 'j' in the swarm given by (4). Based on the clustering criteria, the fitness for each point is evaluated.

$$F(j) = \frac{\sum_{i=1}^{k} \sum_{\forall x \in c_k} (x - z_{ji})^2}{D} \quad (4)$$

Where c_k is the subset of data belonging to cluster 'k', D is the number of data points input to clustering. The aim is to minimize the objective function so as to reduce the dispersion in clusters.

Step 5: If the iterations surpass a predefined level, then execute step 1 otherwise go to step 6.

Step 6: The updated position and velocities of particles using (1) and (2), are checked for the boundary condition, i.e., $[Y_{min}, Y_{max}]$ for the position and $[V_{min}, V_{max}]$ for the velocity. If the particle excels their boundaries, then the new position is fixed to Y_{min} or Y_{max} and the new velocity is fixed to V_{min}, V_{max} respectively.

Step 7: Reduce the value of $\propto$ once the swarm converges to the optimal solution.

Step 8: If the global best 'G' remains the same for the number of iterations, then go to step 1 otherwise, execute step 5.

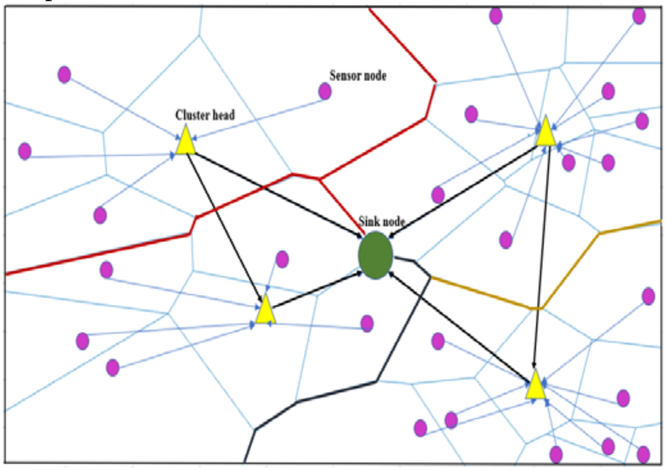

Fig. 3. Voronoi diagram for a clustered WSN scenario.

4.2 Estimation of sensor data

4.2.1 Kalman Filter

In 1958 Rudolf Emil Kalman presented the notion of Kalman filter (KF) used in estimation problems. It is a recursive prediction filter for estimating states of a system. It executes in two stages: prediction stage and correction stage. The prediction based on the Kalman calculations obliterates the randomness from the predicted data. In this phase, the estimation of 'x' (a data value) is generated on current time from the previous time. The outcomes of these calculations are covariance and prediction model. Hence, it is two states prediction and update. The recursion value of Kalman filter is due to the repetition of prediction and correction stages i.e., the previous time step becomes initial value for each time step. Let 'x' be a data value which has linear gaussian properties and its relation between two states can be represented in matrix form. Additionally, the noise around sensor is gaussian and it has mean. The 'x' belongs to a N-dimension space.

The Probability density function of a normal or gaussian distribution is-

$$\mathcal{N}(x, m, \Sigma^2) = \frac{1}{\Sigma\sqrt{2\pi}} e^{\frac{-(x-m)}{2\Sigma^2}} \quad (5)$$

Where 'm' and 'Σ^2' are the mean and variance of the distribution in N-dimensional space. Bigger the variance, higher is the uncertainty. If we multiply the mean and variance by a gain matrix F we get,

$m_{new} = F.m$ and $\Sigma^2_{new} = \mathrm{F}^2\Sigma^2$

Also,

$\mathcal{N}$ (x, m_0, Σ_0^2) $\mathcal{N}$(x, m_1, Σ_1^2) = $\mathcal{N}$(x, m_{new}, Σ^2_{new})

$m_{new} = m_0 + \frac{\Sigma_0^2(m_1 - m_0)}{\Sigma_0^2 + \Sigma_1^2}$

$m_{new} = m_0 + k(m_1 - m_0)$

Where k=$\frac{\Sigma_0^2}{\Sigma_0^2 + \Sigma_1^2}$ is the gain.

If Σ_0^2 is a huge number, 'k' goes to 1 and new mean becomes

$$m_{new} = m_1$$

But if the uncertainty is huge then the result of function of 2 data is closer to other one.

Let say if Σ_1^2 goes to ∞, then k goes to 0. Then

$$m_{new} = m_0$$

This is the essence of Kalman filter. This 'k' tells how much we get closer to the measurements or model. The limits for 'k' are 0 and 1.

4.2.2 Sensor data estimation based on Kalman filter

The framework of KF exhibits the update given by the measurements of sensors and their prediction using representation of base state vector in system matrix form. Consider a dynamic system with linear properties represented by following equations:
The state vector is given in (6)

$$x = \begin{bmatrix} d \\ v \end{bmatrix} \tag{6}$$

Where d is the distance and v is the velocity are the two parameters which defines the state of the vector 'x'.
The vector form representation for the 'x' vector and its covariance can be expressed as-

$$\hat{x} = \begin{bmatrix} position \\ velocity \end{bmatrix} \text{ and } cov = \begin{bmatrix} \Sigma_{pp} & \Sigma_{pv} \\ \Sigma_{vp} & \Sigma_{vv} \end{bmatrix}$$

The matrix P_k is symmetric that is, its transpose is equal to the original matrix. For a gaussian function given by-
As stated in (6), the state vector or measured value 'x' by a sensor is considered here as well. The state and error estimation of sensor values is done by employing Kalman filter. In prediction state, the prediction is done using dynamic model. x^- is the priori value and x^+ is the posteriori value.

$$\dot{x}^-(t) = D.x^-(t) \tag{7}$$

The solution $x^-(t)$ of the differential equation (7) is given by:

$$x^-(t) = \varphi_o^t.x^-(t_o) \tag{8}$$

φ_o^t is a state transition matrix which can transfer initial state $x(t_o)$ to its corresponding state x(t) at time (t).
From (7) and (8)

$$\dot{x}^-(t) = D.x^-(t) = D.\varphi_o^t.x^-(t_o) \tag{9}$$

$\varphi_o^t = I$ is the initial matrix because $x(t_o) = I\, x(t_o)$
The covariance matrix of the predicted state is expressed in (10). It shows how much uncertainty exists between the states.

$$cov^-(t_j) = \varphi_{t_{j-1}}^t.cov(t_{j-1})\left(\varphi_{t_{j-1}}^{t_j}\right)^T + Q \tag{10}$$

Generalized form of (10) is

$$cov^-(t_j) = \varphi_{t_{j-1}}^t.cov(t_{j-1})\left(\varphi_{t_{j-1}}^{t_j}\right)^T + \int_{t_{j-1}}^{t_j} Q(t)dt \tag{11}$$

Correction stage-
The posterior state is represented by

$$x^+(t_j) = x^-(t_j) + \Delta x(t_j) \tag{12}$$

Here the predicted vector $x^-(t_j)$ is improved because of observations at the epoch (t_j).
Covariance matrix for correction stage is given by (13)

$$cov^+(t_j) = cov^-(t_j) + \Delta cov(t_j) \tag{13}$$

Δcov is minimized for Kalman filter and can be calculated using (14)

$$\Delta cov(t_j) = E\left[\Delta x(t_j).\Delta x(t_j)^T\right] \tag{14}$$

Where 'E' is the mathematical operator which denotes expectation of the two values $\Delta x(t_j)$ *and* $\Delta x(t_j)^T$. The above (14) is compared with the standard equation of Kalman filter for finding the new state estimation error. [28]

$$\Delta x(t_j) = k(t_j).\left(r(t_j) - r^-(t_j)\right) \tag{15}$$

Where $\left(r(t_j) - r^-(t_j)\right)$ is the measurement residual which contributes to the dissimilarity between the predicted and actual measurements and k=gain matrix which is given by:

$$k(t) = cov^- H^T \left(Hcov^- H^T + R(t_j)\right)^{-1} \tag{16}$$

The estimation of predicted value ($\dot{x}^-(t)$) and covariance of error are combined with the measurements of sensor with its covariance to acquire updated error covariance and estimate matrix. **Fig. 4** accompanied by **Fig. 10** explain well the relationship between prediction and filtering the data by employing Kalman filter.

Q, H and R are the tuning parameters which gives the information about the system. Finally, the corrected state is given by

$$x^+(t_j) = x^-(t_j) + k(t_j).\left(r(t_j) - r^-(t_j)\right) \tag{17}$$

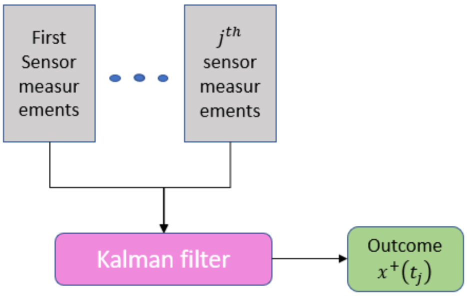

Fig. 4. Generation of corrected output from various sensor measurements and their estimated values using the Kalman filter.

4.3 A fast Network layer routing algorithm based on MAC scheduling

We consider here a clustered WSN, a network with its node clustered, and each group of nodes have their distinct boundaries, as illustrated in **Fig. 3**. The clustering makes a network such that it becomes vital for several standards and applications such as IEEE 802.15.4, automation control, tracking process, etc. The clustering allows local data collection and aggregation with high energy efficiency and low latency even in the presence of multi-hop routing. The clustered topology is designed such that broadly there exists two-level communication. The first level is when all the member nodes of a particular cluster send their data to their respective cluster head. The second level deals with inter-cluster communication, i.e., transmission happens using the greedy approach from one cluster head to another cluster head. The cluster head receives periodic data from other sensor nodes. In this paper, the nodes sense the data and the Kalman filter estimates the next time-series data to reduce redundancy, thereby successfully aggregating the data. The estimation works in such a way that if |det $(ME'(k)M^T + C)$| is greater than the predetermined threshold (λ) [16], then the nodes are scheduled, as shown in **Fig. 5**. E is the estimated error covariance, M is the measurement matrix, and C is the covariance matrix for known measurement noise. In **Fig. 5**, A1, A2, A3, A4 are each epoch representing the time for off state and on state. Synchronously, the IEEE 802.15.4 works at the physical layer in which the nodes send the preamble and wait for the acknowledgment (ACK) to be received. Thus sending a data frame is followed by a wait duration for ACK, i.e., mackAckWaitDuration [29] in the IEEE 802.15.4 standard. The active time of the affected nodes with the criteria mentioned above exceeds the active transmission time of the normal nodes, and thus the affected nodes are likely to send their data at a higher rate as compared to normal ones. The estimation goes correct with the normal nodes, and thus even if the transmission rate for normal nodes declines, it hardly affects the network's working. **Fig. 6** gives a cross-layer design of how the physical layer is interacting with the MAC, and MAC further impacts the routing layer.

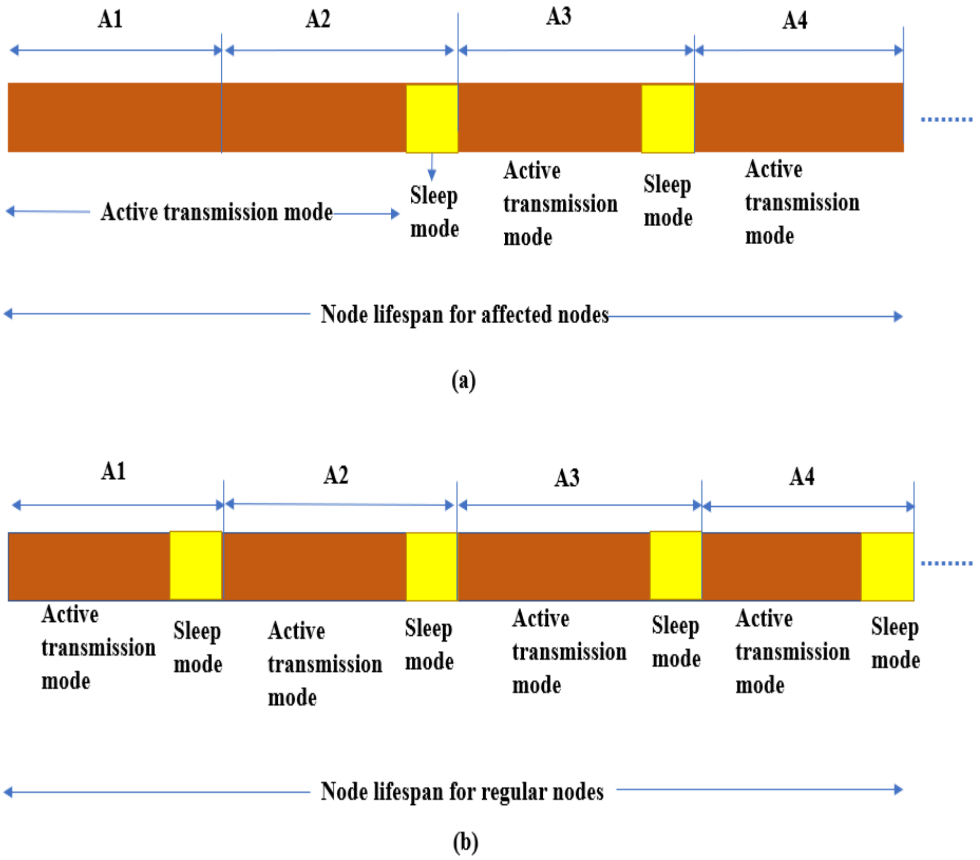

Fig. 5. Duty cycle dynamics for energy-efficient sleep scheduling scheme

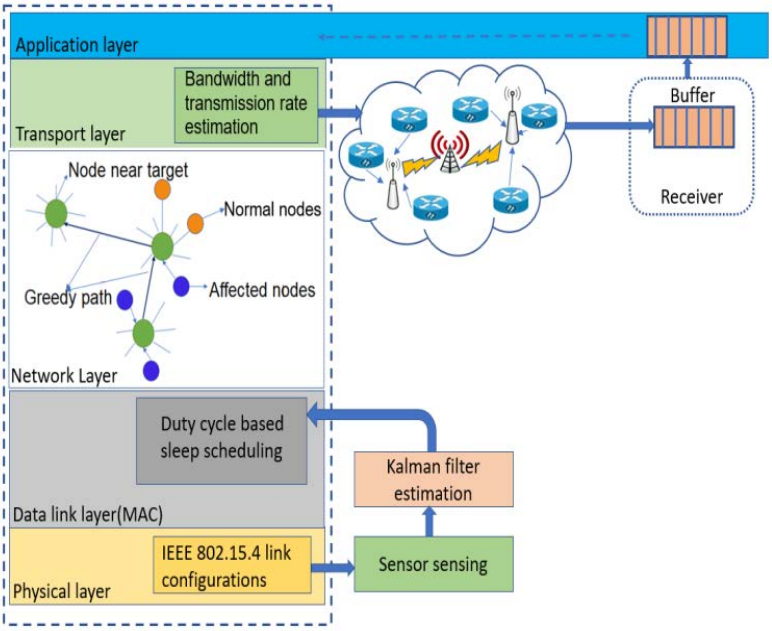

Fig. 6. Optimized Cross-layer scheme of proposed work

Duty cycle based greedy routing

A greedy routing approach [30] has been adopted here that takes those nodes into account, which are scheduled according to the Kalman filter-based estimation. In this routing, the clustered network is formed in a tree-based structure, and the source node forwards the data to the next neighboring node, provided the neighbor node is close to the sink node. **Algorithm.2** gives the pseudo-code for greedy distance-based routing, which focuses on choosing those neighbor nodes with minimum Euclidean distance from the destination. Thus, the greedy routing aims to reduce the number of hops in each transmission. The transmission is done from one cluster head to another cluster head nearest the sink or base station.

Definition 1:

A graph (V, E) in which given distance function is $D: V \times V \rightarrow G^+$, the greedy routing assumes a decision: For a given sink node 's', an affected node $'v'$ with the neighbors $N(v)$ transmits a packet to its neighbor $y \in N(v)$ such that $(y, s) = \min_{x \in N(v)} D(v, s)$ where x is a data point, and y is also a datapoint neighbor to x. The affected node vector will be therefore $= \{v_1, v_2, \dots, v_j\}$. The 2-hop distance subnetwork is formed, and the packets are assigned to each node for transmission. The nodes are iteratively checked using the Kalman filter estimation technique on the sensed data. The affected nodes are scheduled, and the packets are reduced by one (-1) as the nodes are being added to distinguish them from normal nodes. The greedy mode of routing is enabled and forwards the data to the next node (cluster head), which is nearest to the sink node. This forwarding of data continues till the sink node is encountered.

5. Implementation of proposed model

The methodology of the paper works in three sections. The first one is the formation of an optimized cluster in the network using the PSO-KM algorithm, which aims to reduce the dispersion between the clusters using the fitness function given in (3). The K-Means algorithm is a clustering algorithm used to cluster various data points based on their distance. It divides the data points into a predefined number of clusters based on the Euclidean distance, with each cluster having its own centroid point. The other data points are in the vicinity of that centroid. In this paper, the K-Means approach is used to cluster the nodes with their location. The centroids are obtained and treated as the cluster heads. Further, to optimize this clustering, a population-based metaheuristic approach called particle swarm optimization is implemented on the cluster heads so that the cluster heads can obtain the position in close proximity with the base station and other cluster heads, and thus, inter-cluster communication can be done with low loss of energy and time. **Fig. 7** depicts the flow schema of the proposed model. **Algorithm.1** shows a pseudo-code for PSO-KM clustering. After an optimized clustering topology is formed, the data sensed by the sensor nodes is estimated for future values. The Kalman filter is used to predict the data, and the sensor nodes that surpass the threshold value are selected as the affected nodes targeted for data transmission. The affected nodes possess the wrong predicted value, which does not match with the future values, and thus those nodes are scheduled. The duty cycle for affected nodes is shown in **Fig. 5**, and with the active and sleep time, the data transmission is regulated. The transmission is governed by a routing algorithm called the greedy routing algorithm. The algorithm for recursive covariance Kalman filter with duty cycle-based greedy routing is given in **Algorithm.2**.

Algorithm.1 PSO-KM clustering

1: **Input:**(1) set of sensor nodes $A = \{s_1, s_2, \dots s_n\}$
    (2) node coordinates as datafile
    (3) number ‘k’ for desired clusters
2: **Output:**k clusters with optimized cluster heads.
3: **Start K-Means algorithm**
4: Randomly choosing k-datapoints from A as initial centroid points.
5: Repeat iteratively
6: Assigning each $s_1, s_2, \dots$ to that cluster which has nearest centroid point.
7: Computing new mean taking each cluster.
8: Repeat the process till criteria for convergent is met.
9: After some iterations, the datapoints are clustered with ‘k’ number of clusters with a centroid for each.
10: The datapoint nearest to the centroid is taken as cluster head.
11: Run the K-Means clustering While maximum rounds or low error criteria is not achieved.
12: k=k+1
13: **Begin:**
    For each cluster head ‘c’
    Initialize position Y within the feasible range
    Initialize velocity V within the feasible range
    Iteration i=1
    Do
14: **For** each cluster head c

Computing fitness value using equation (3)
15: **If** (fitness value>p_best_c)
Set current fitness as p_best_c
16: **End if**
17: **End for**
Selecting cluster head with best fitness as g_best
18: **For** each cluster head c
Computing velocity using below equation
19: $V_j^{(i)} = \propto * V_j^{(i-1)} + c_1 * r_1 \left(B_j - Y_j^{(i-1)}\right) + c_2 * r_2 \left(G - Y_j^{(i-1)}\right)$
Updating position of particle using below equation
20: $Y_j^{(i)} = Y_j^{(i-1)} + V_j^{(i)}$
21: **End for**
22: **End for**

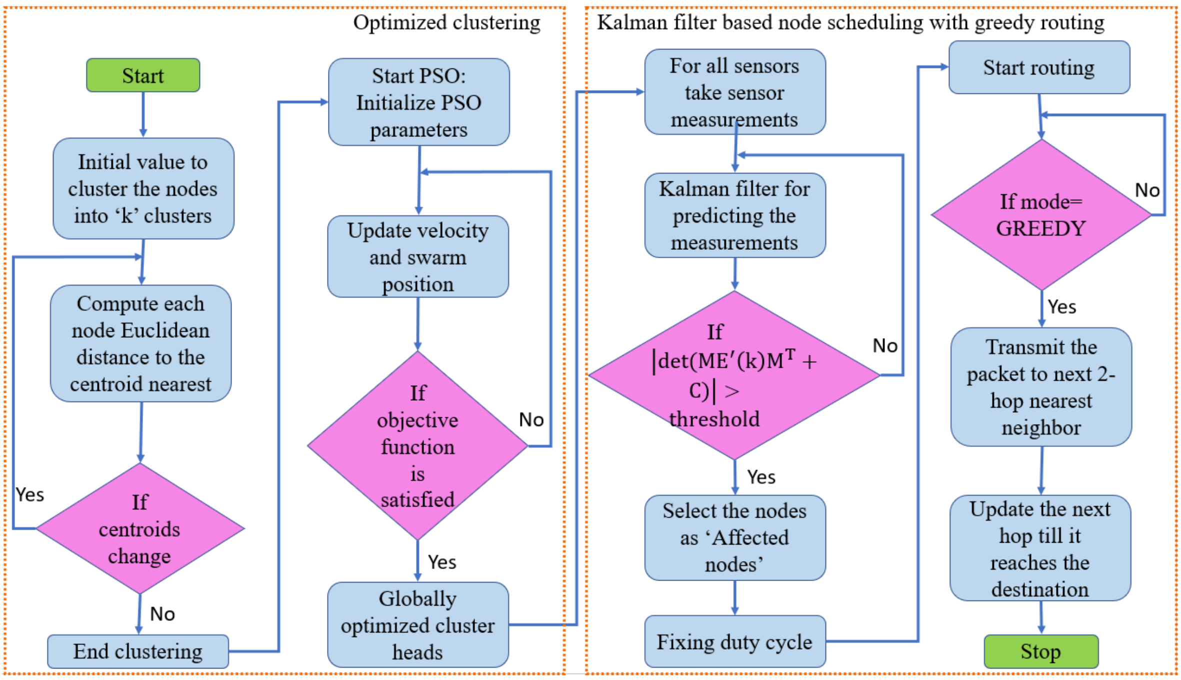

Fig. 7. Flow chart of the proposed method

Algorithm.2 RCKF based DCGR

1: **Input:**$\hat{y}_0$, $P_{x,0|0}$, G_k,H_k, X_k , measurement matrix M, Q_k&$C_k \forall k$
2: set 'k' value equal to 1
3: **Prediction stage:**
4: $\hat{y}_{k|k-1} = G_{k-1}\hat{y}_{k-1|k-1}$
5: **Prediction error covariance calculation:**
6: $P_{y,k|k-1} = G_{k-1}P_{y,k-1|k-1}G'_{k-1} + Q_k$
7: **Objective:** Set of nodes 'X' with predicted data beyond the threshold 'λ' are scheduled to 'm' data links.

8: **Assumption:** every y_i of X has some time of arrival a_i and deadline time d_i with minimal arrival and maximal deadline be T_0 and T_m respectively
9: Sorting of nodes in 'X' through thresholding
10: **For** each node n
  selected node vector $v = \{v_1, v_2, \dots, v_j\}$
  Initialize 2-hop distance subnetwork $H_i = (L_i, V_i)$, where L_i is the edge and V_i are the vertices of the tree.
  assigning S packets to each node in subnetwork
11: **If** pac$(v_j) \leftarrow \hat{y}_{k|k-1}, \forall v_j \in v_i$
  add nodes into set of unchecked nodes $v'' \leftarrow v$
  Adding the nodes in a queue based on their prediction
12: **End if**
13: **For** sensed value checking each node for sleep-awake scheduling
14: The node $v_j$is schedulable if |det $(ME'(k)M^T + C)| > \lambda$ for $\forall\ v_j \in \{v_i\} \cup (n(v_i) \cap A\ v)$
15: **If** v_j is schedulable, add it to the assigned active nodes
  n=n+1
  $A\ v \leftarrow A\ v \cup \{v_j\}$ and decrease the tickets of v_j and all its neighbors by 1.
  $pac(v_j) = pac(v_j - 1)$
  $for\ \forall\ v_j \in \{v_j\} \cup n(v_j)$
  Remove v_j from $v' \leftarrow v'' - \{v_j\}$
16: **End if**
17: **If** v'' is empty, then continue
  Data forwarded is pac(v_j)
  Introducing sleep time T_s once in two epochs
  In T_{on} time of v_j node
18: **End if**
19: **End for**
20: Route (v_j+1,v_j, t)
21: **If** (mode= GREEDY) then
  Next=GREEDY (v_j+1, D)
22: **If** next='NULL', then
  Return NEXT EDGE (v_j+1,v_j)
23: **Else**
  Return next
24: **End else**
25: **End if**
26: **End if**
27: **End for**

λ is a threshold value which has been taken from recursive covariance estimation idea in [16].

6. Performance evaluation

This section deals with the outcome of performance metrics for the application used in the proposed method; the results are presented using various analytical graphs, and a comparison with other existing approaches is also included. The experiment is carried out on a real-time dataset of the water treatment plant. The wastewater is treated daily and is measured. The state of the treatment plant is daily observed and classified to predict the upcoming faults in it using the probability density function and state variables. The data set consists of 38 features. The nodes are varied from 100 to 500. The description of the dataset is given in **Table 1**. The wireless sensor network for 500 nodes with an area of deployment 200×200 m^2 is taken into consideration. The nodes are randomly distributed and are clustered using the PSO-KM algorithm for more optimal clusters and cluster heads. The sensors for each unit in the treatment process sense the parameters like PH, biological oxygen demand, chemical demand of oxygen, water flow, suspended solids, and so on. These parameters are collected by the cluster heads for every round of iteration. For predicting faults at each treatment process of the plant, the state of the plant is classified. The algorithms for the deployment and clustering and the performance metrics are analyzed using MATLAB 2016a version. The nodes in a cluster are destined to forward their data to the cluster head, and every cluster head either forwards the data to its nearest cluster head or may send it directly to the base station (BS) depending on the distance between the cluster head and BS. The performance in terms of energy consumption, throughput, number of alive nodes, etc., is analyzed at the base station. **Fig. 8** shows the clustering of randomly distributed nodes in a given area of interest. **Fig. 9** and **Fig. 10** show the data of pH values and BOD collected for 500 days. The description of the dataset is as follows:

Table 1. Dataset description

Characteristics of the dataset	Multivariate nature
Nature of attributes used	Real-time and integer type
Instances	527
Attributes	38
Technique used	Clustering

Table 2. Parameter settings for proposed model

Parameters	Value
Deployment area	200X200 m^2
Number of nodes	500
Deployment	Random
Number of base station	1
MAC standard	IEEE802.15.4
Packet size	500 bits
Data transmission model	Routing through clustering
Data rate	200 kbps
Simulation time	1000 sec
Duration from sleep to idle	1500 μs
Duration from idle to sleep	300 μs

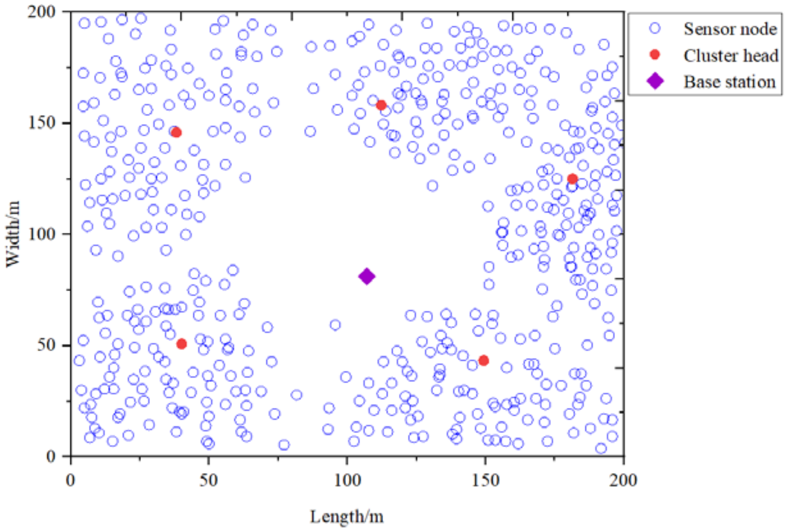

Fig. 8. Clustering of 500 nodes (including BS) randomly distributed in the area 200X200m^2

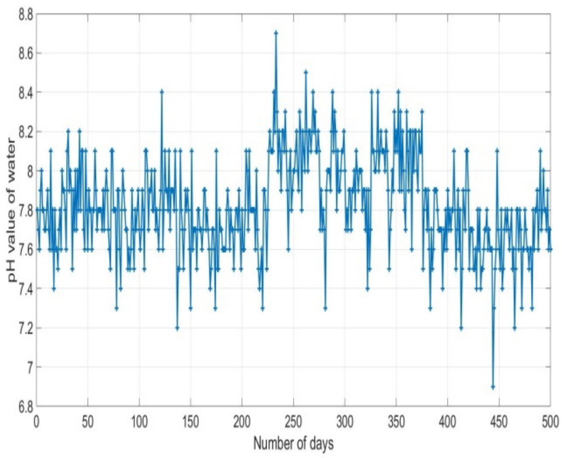

Fig. 9. The data values for the pH of the water collected for 500 days.

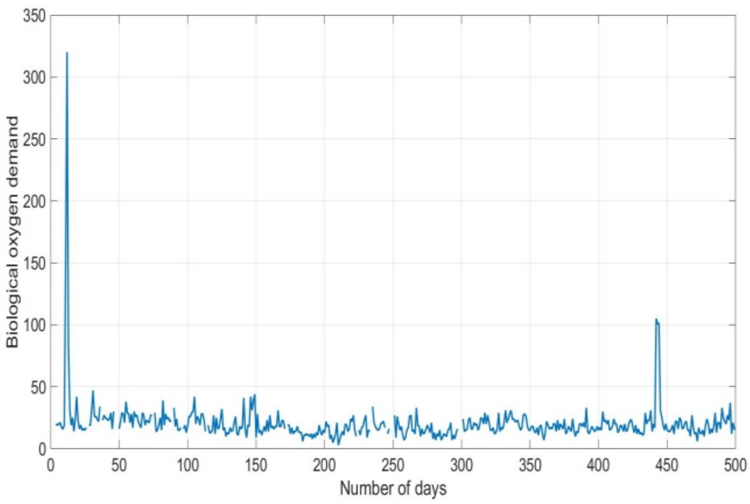

Fig. 10. The data values for the biological oxygen demand collected for 500 days.

Performance measures

The performance metrics as probability density function, state estimation error, energy consumption, goodput, duty cycle, sleep scheduling, number of alive nodes, throughput and delay are analyzed for the proposed method and are validated with other similar state-of-the-art techniques which inhibits the approaches like clustering, duty cycle scheduling, and cross-layer optimization i.e., LEACH[16], DEEC[10], SEP[11], Greedy routing[15], CL-CFMPR[3], FMRP[12], DASF[20], CDSWS[13], DDS[21], GCKN[14], X-MAC[6], PB-MAC[7], and RMS[22]. The probability density function for the RCKF method is shown in **Fig. 11**, which includes the PDF for actual measurement, predicted state, and optimal predicted state. The optimal state prediction estimate has the highest PDF value with relatively less variance. **Fig. 12** gives the state estimation errors between RCKF used in the proposed method and the Kalman filter for the difference between the measured and actual position and velocity variables of the nodes.

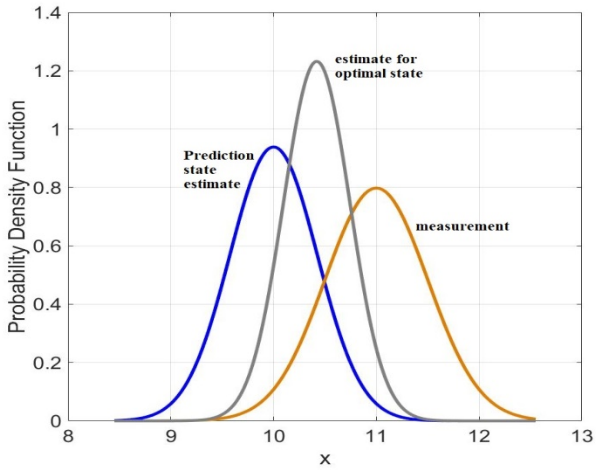

Fig. 11. Probability density function curves for measured, predicted, optimal predicted states of RCKF.

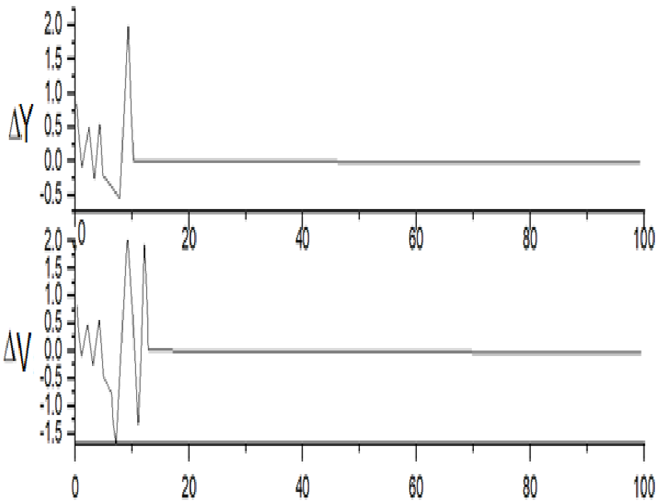

Fig. 12. State estimation errors between threshold-based KF and KF.

Fig. 13 depicts the total energy consumption of the network. The proposed method records a 28% decrease in energy consumption. From the figure, it is clearly observed that the energy consumption for the proposed model varies between 0.28 joule to 0.34 joule. The cross-layer optimization and duty cycle scheduling save more energy as compared to the methods adopting optimal and unequal duty cycle approaches. Fig. 14 shows the goodput comparison of the proposed method with FMRP and CL-CFMPR. The graph shows that there is an enhancement in the system goodput when greedy method for routing is used along with the duty cycle scheduling in MAC.

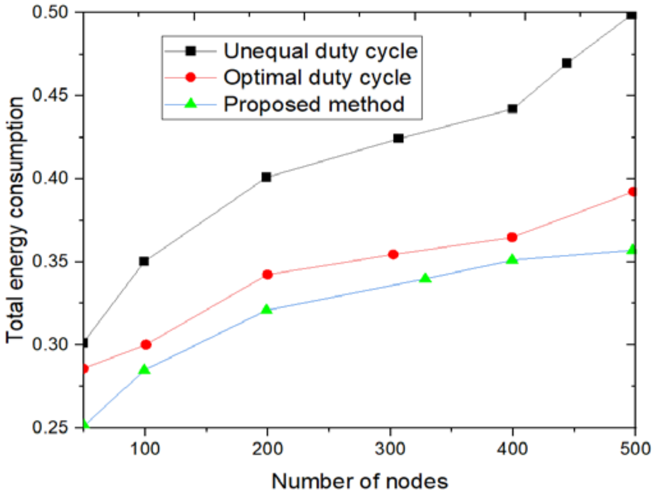

Fig. 13. Total energy consumption (in joule) for the duty cycle enabled network against the number of nodes.

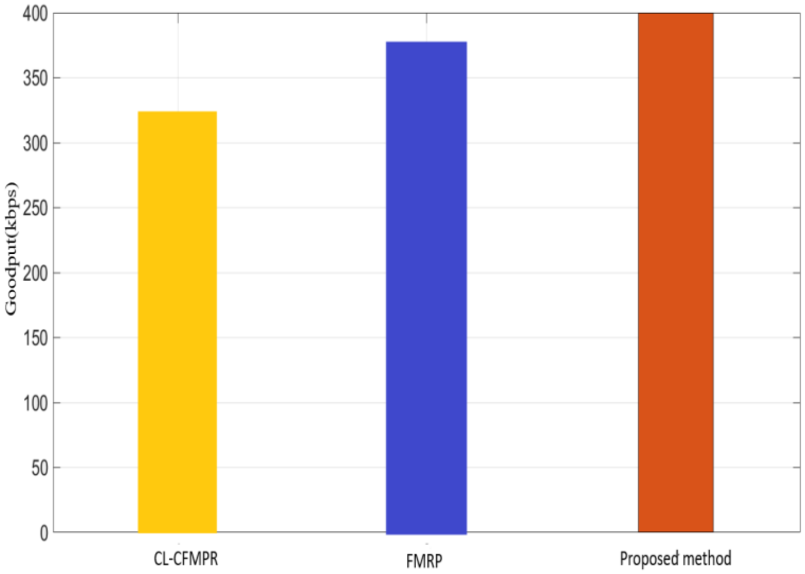

Fig. 14. System goodput

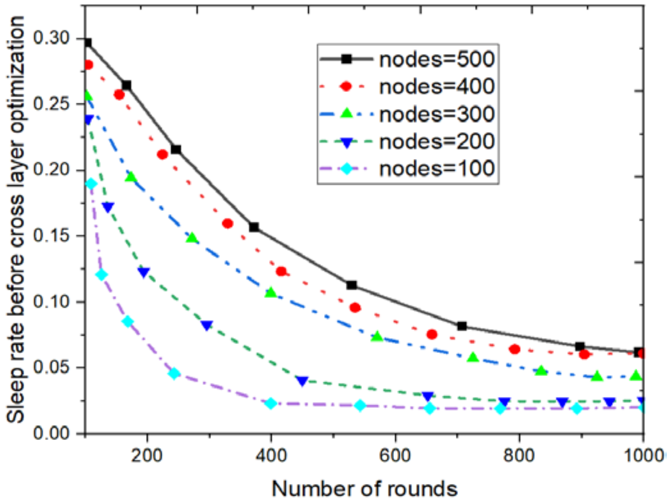

Fig. 15. Sleep rate before cross-layer optimization versus number of rounds.

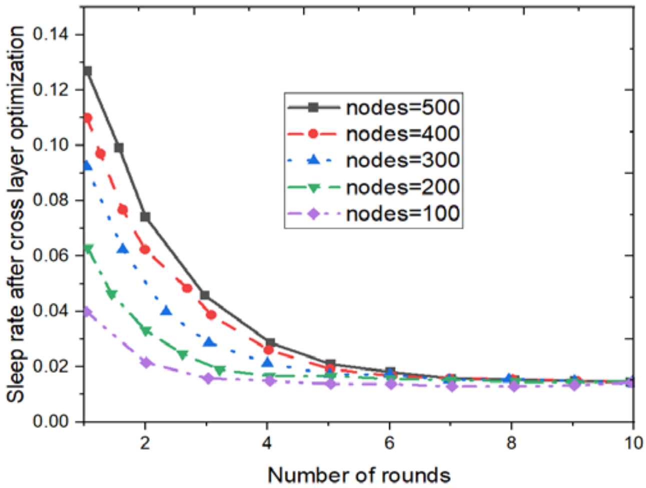

Fig. 16. Sleep rate after the cross-layer optimization against number of rounds.

Fig. 15 and **Fig. 16** illustrate the sleep rates before and after the cross-layer optimization. The figures clearly show that when the number of nodes increases, the sleep rate increases, leading to efficient energy management and controls delay. The sleep rate before cross-layer optimization is high for 500 nodes (0.30), and it gradually decreases while reaching 100 nodes (0.19). Similarly, the sleep rates after cross-layer optimization are high for 500 nodes (0.128) and then reduce with the number of nodes, and as the number of rounds increases, they meet at the same value, i.e., 0.02.

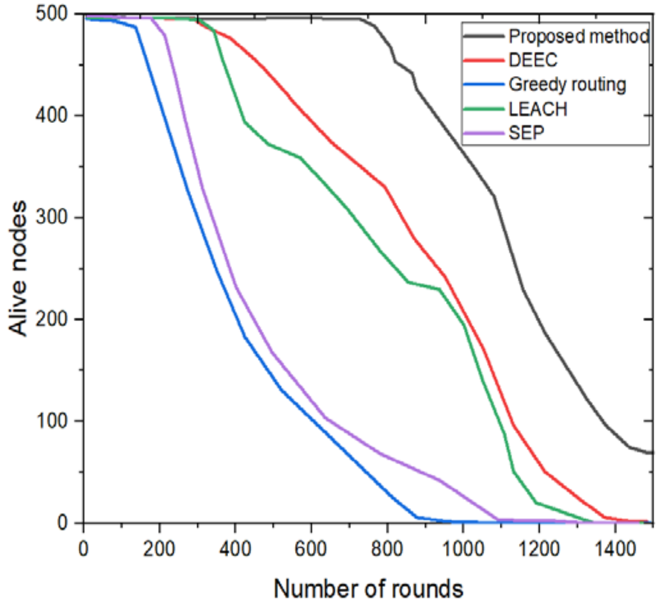

Fig. 17. Number of alive nodes against number of rounds.

The number of alive nodes with respect to the number of rounds is depicted in **Fig. 17**. The alive node number for the proposed method remains constant at approx 800 rounds as compared to other methods like DEEC, Greedy routing, LEACH, and SEP. Almost all other approaches' alive nodes reach zero after the 1400th round. But for the proposed scheme, it is maintained at 70 nodes with increasing rounds.

Fig. 18 shows the total energy consumption of the network. The proposed method improves the energy efficiency by an average of 87.5% when compared to other methods like DASF, CDSWS, DDS, GCKN. The reason behind the improved efficiency is the use of both duty cycle scheduling and greedy routing algorithm, which saves the energy of the wireless network. The energy consumption ranges from 0.2 J to 0.6 J in the proposed method, which is relatively lower compared to other cross-layer optimized models.

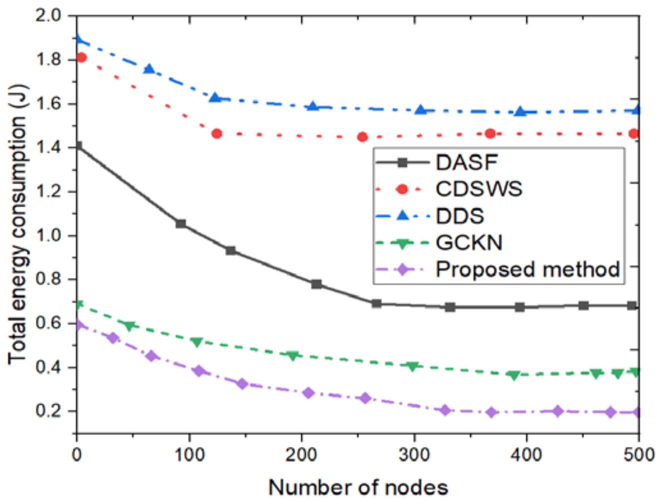

Fig. 18. Total energy consumption in the network with the increase in the number of nodes.

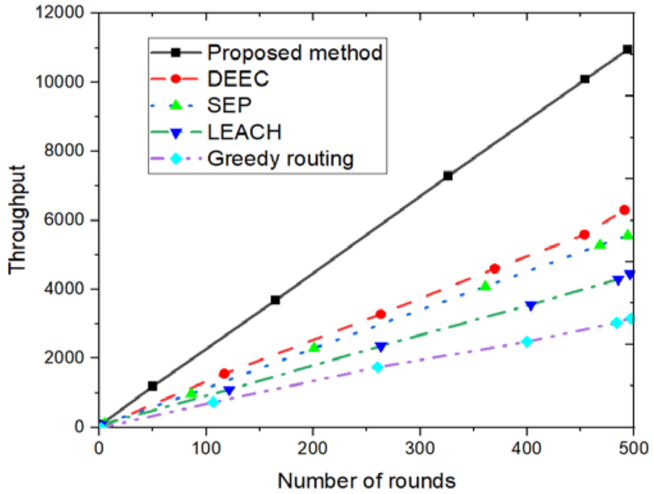

Fig. 19. The network throughput versus number of rounds.

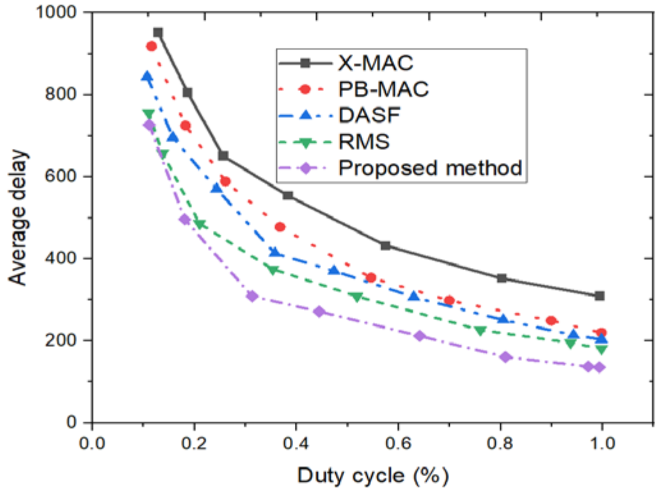

Fig. 20. Average delay (ms) of the system with increase in the percentage of duty cycle.

Fig. 19 is the throughput analysis of the proposed method and other approaches like DEEC, SEP, LEACH, Greedy algorithm. The throughput of the proposed method records 77.27% enhancement when compared to the method with the lowest throughput. The throughput for the proposed method reaches 11000, while for others, it is lower than 7000. This is because the proposed method uses scheduling efficiently, which enhances the overall system throughput. The average delay for the proposed method mitigates 50%, as shown in **Fig. 20**. The proposed method ranges between 130 milliseconds to 720 milliseconds of average delay. Hence, it surpasses other existing models in terms of delay since the optimized cluster head positions and the greedy routing make the network work faster.

Fig. 21 gives an end-to-end delay analysis for CL-CFMPR, FMRP, and the proposed method. It can be observed that the proposed method shows a 61% reduction in the network's end-to-end delay and proves itself superior to the other two methods. The highest end-to-end delay noted by CL-CFMPR is 0.102 s. **Table 3**, **Table 4**, and **Table 5** showcase the comparison of existing models with our proposed scheme on the grounds of energy utilization, goodput, delay, etc. **Table 6** showcase overall comparision with latest existing schemes.

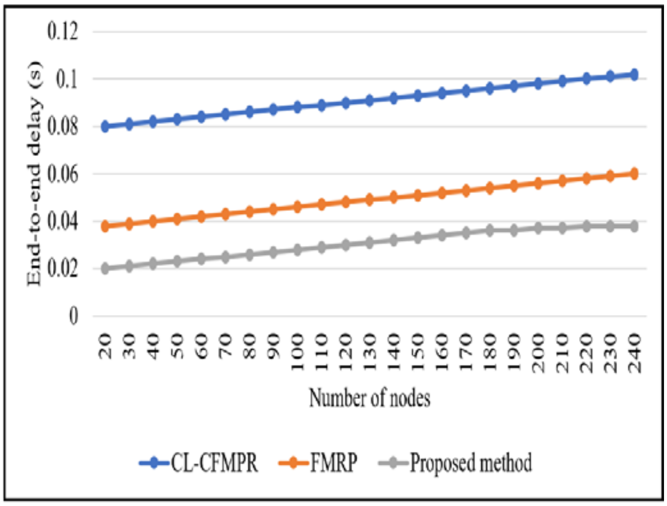

Fig. 21. End-to end delay with respect to the number of nodes.

Table 3. Comparison of routing protocols with proposed model

Metrics	**Protocols**				
	LEACH	DEEC	SEP	Greedy routing	Proposed
Throughput (rounds)	4100	5100	4700	3000	11000
Alive nodes till rounds (rounds)	1320	1390	1500	880	3000

Table 4. Comparison of delay aware duty-cycled scheduling methods with the proposed scheme

Metrics	**Protocols**					
	CL-CFMPR	FMRP	DASF	CDSWS	DDS	Proposed
Delay(s)	0.1	0.06	0.23	-	-	0.039
Goodput (Kbps)	325	375	-	-	-	400
Total energy consumption (J)	0.78	0.97	0.7	1.5	1.6	0.2

Table 5. Comparison of scheduling schemes with the proposed schemes

Metrics	**Protocols**				
	GCKN	X-MAC	PB-MAC	RMS	Proposed
Delay(s)	-	0.32	0.24	0.22	0.039
Total energy consumption (J)	0.4	0.3	-	0.8	0.2

Table 6. Comparison chart of similar existing relevant works and the proposed approach.

Reference	**Techniques**	**Objective**	**Layers involved**	**Outcome**	**Applications**
Xu et al. [24], 2018	Cross layer optimized opportunistic (COOR) routing	Improvement in network life and reliability in data transmission	Network and MAC layer	Delay reduced by 21.09%, energy efficiency by 86.9% and reliability increased by 36.6%	Delay sensitive applications

Kalaikumar et al. [25], 2018	Fuzzy and ABC based	Congestion control and network connectivity issues	MAC and Network layers	79% packet delivery ratio, 63% network life enhancement, 60% of energy efficiency	WSN
Kaur et al. [26],2020	PSO based unequal and fault tolerant clustering, multiobjective ACO	To minimize the trade-off between energy dissipation and delay	MAC and network layer	6.8 to 10.5% more energy efficient, 19.1 to 25.3% more delay efficient	Event based data traffic application
Lahane et al. [27],2020	Integrated Mothflame and dragonfly approach	Enhancement in lifespan of network	Physical and network layer	70-85%higher network lifetime than other hybrid methods	Dense WSN
Shanmugam et al. [31],2020	K-medoids, adaptive Harris Hawk optimization, variable weighted stacked autoencoder with adaptive sunflower optimization	QoS enhancement	Physical layer, link layer, network layer	Packet delivery ratio=99.5%, throughput=0.98 Mbps, energy consumption=30mJ, packet delay=0.9s	WSN
Patil et al.[32], 2020	Adaptive energy sensor selection model and hibernation with AODV	Mitigation in energy consumption	MAC and Network layer	Average of 45% reduction in delay	Monitoring
Mahajan et al.[33],2021	Bacterial foraging optimization	Raising the lifetime and energy efficiency	Physical, network and MAC	Energy efficiency increased by 243 rounds, decrease in delay by 0.72 ms.	Smart farming
Proposed model	KM-PSO clustering, KF based node scheduling based greedy routing	Improvement in energy efficiency and latency using cross-layer model	Physical, MAC, Network layer	improved energy efficiency and latency by 87.5% and 61% respectively.	Event or environment monitoring

7. Conclusion

This paper considers the problem of maintaining various parametric issues at different layers of the WSN protocol stack through a real-life application. The application-based IoT framework in the proposed work has used some protocols and techniques like routing at the network layer, clustering at the node level, estimation at the data level, and scheduling at the data link layer. The coordination between the methods working at different layers is achieved by a cross-layer architecture shown in **Fig. 6**. This cross-layer data transmission scheme is using duty-cycled scheduling at the MAC layer and optimized the network layer parameters. The results reveal that the proposed method came out to be efficient in terms of throughput, sleep scheduling rates, end-to-end delay, energy consumption, and the number of alive nodes. It marks an 87.5% and 77.27% hike in the energy efficiency and the overall network throughput, respectively. A 61% degradation in the system's end-to-end delay is noted compared to other state-of-the-art methods. As future work, the task of affected node scheduling can be made more effective by keeping the load balancing issue in view when the number of nodes becomes high. Also, an alternate, less complex method can be proposed for such applications where the accuracy matters most.

References

[1] A. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal and Q. Z. Sheng, "IoT Middleware: A Survey on Issues and Enabling Technologies," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 1-20, Feb. 2017. Article (CrossRef Link)

[2] K. Kumar, S. Kumar, O. Kaiwartya, Y. Cao, J. Lloret, and N. Aslam, "Cross-Layer Energy Optimization for IoT Environments: Technical Advances and Opportunities," *Energies*, vol. 10, no. 12, p. 2073, Dec. 2017. Article (CrossRef Link)

[3] I. Jemili, D. Ghrab, A. Belghith, and M. Mosbah, "Cross-layer adaptive multi-path routing for multimedia Wireless Sensor Networks under duty cycle mode," *Ad Hoc Networks*, vol. 109, p.102292, 2020. Article (CrossRef Link)

[4] S. Pandit and G. Singh, "Framework for Cross-Layer Optimization in Cognitive Radio Network," *Spectrum Sharing in Cognitive Radio Networks*, pp. 225-251, 2017. Article (CrossRef Link)

[5] D. K. Sah and T. Amgoth, "Parametric survey on cross-layer designs for wireless sensor networks," *Computer Science Review*, vol. 27, pp.112-134, 2018. Article (CrossRef Link)

[6] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. of 4th international conference on Embedded networked sensor systems (SenSys' 06)*, New York, NY, USA, pp. 307–320, 2006. Article (CrossRef Link)

[7] Z.T. LI, G.M. ZHU, Z.Q. WANG, T.R. PEI, G.F. PAN, "Low duty cycle and low collision asynchronous MAC protocol for wireless sensor network," *Journal on Communications*, vol. 34(10), pp. 9-16, 2013. Article (CrossRef Link)

[8] P. Shi, C. Gu, C. Ge and Z. Jing, "QoS Aware Routing Protocol Through Cross-layer Approach in Asynchronous Duty-Cycled WSNs," *IEEE Access*, vol. 7, pp. 57574-57591, 2019. Article (CrossRef Link)

[9] M. N. Jambli, M. I. Bandan, K. S. Pillay and S. M. Suhaili, "An Analytical Study of LEACH Routing Protocol for Wireless Sensor Network," in *Proc. of IEEE Conference on Wireless Sensors* (ICWiSe), pp. 44-49, 2018. Article (CrossRef Link)

[10] M. M. P. Reddy and S. V. Rajan, "DEEC Protocol for WSNs," *India Advances in Wireless and Mobile Communications*, vol. 10, no. 1, pp. 51–63, 2017. Article (CrossRefLink)

[11] G. Smaragdakis, I. Matta, A. Bestavros, "SEP: A stable election protocol for clustered heterogeneous wireless sensor networks," in *Proc. of Second international workshop on Sensor and Actor Network Protocols and Applications (SANPA 2004)*, Vol. 3, pp. 1-11, 2004. Article (CrossRefLink)

[12] I. Jemili, G. Tekaya and A. Belghith, "A Fast Multipath Routing Protocol for wireless sensor networks," in *Proc. of IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA)*, pp. 747-754, 2014. Article (CrossRef Link)

[13] G. Nan, G. Shi, Z.Mao, M. Li, "CDSWS: coverage-guaranteed distributed sleep/wake scheduling for wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, pp. 1-14, Dec. 2012. Article (CrossRef Link)

[14] L. Shu, L. Wang, J. Niu, C. Zhu and M. Mukherjee, "Releasing Network Isolation Problem in Group-Based Industrial Wireless Sensor Networks," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1340-1350, Sept. 2017. Article (CrossRef Link)

[15] G. Han, Y. Dong, H. Guo, L. Shu, and D. Wu, "Cross-layer optimized routing in wireless sensor networks with duty cycle and energy harvesting," *Wireless communications and mobile computing*, vol. 15(16), pp.1957-1981, 2015. Article (CrossRef Link).

[16] X. Lv, P. Sun, X. Jia and J. Li, "A Threshold-Based Kalman Filter with Recursive Covariance Estimation," in *Proc. of International Conference on Network and Information Systems for Computers (ICNISC)*, pp. 332-336, 2016. Article (CrossRef Link)

[17] H. Wang, Y. Yang, M. Ma, J. He and X. Wang, "Network lifetime maximization with cross-layer design in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 10, pp. 3759-3768, October 2008. Article (CrossRef Link)

[18] P. Sarwesh, N.S. Shet, K. Chandrasekaran, "ETRT–Cross layer model for optimizing transmission range of nodes in low power wireless networks–An Internet of Things Perspective," *Physical Communication*, vol. 29, pp. 307-318, Aug. 2018. Article (CrossRef Link)

[19] C. Chandravathi, K. Mahadevan, "Web Based Cross Layer Optimization Technique for Energy Efficient WSN," *Wireless Personal Communications*, vol. 117, no. 4, pp. 2781-2792, Apr. 2021. Article (CrossRef Link)

[20] T.-N. Dao, S. Yoon, and J. Kim, "A Deadline-Aware Scheduling and Forwarding Scheme in Wireless Sensor Networks," *Sensors*, vol. 16, no. 1, p. 59, Jan. 2016. Article (CrossRef Link)

[21] D.-S. Vu, T.-N. Dao, and S. Yoon, "DDS: A Delay-Constrained Duty-Cycle Scheduling Algorithm in Wireless Sensor Networks," *Electronics*, vol. 7, no. 11, p. 306, Nov. 2018. Article (CrossRef Link)

[22] Yongle Cao, S. Guo and T. He, "Robust multi-pipeline scheduling in low-duty-cycle wireless sensor networks," in *Proc. of IEEE INFOCOM*, pp. 361-369, 2012. Article (CrossRef Link)

[23] J. Tan, A. Liu, M. Zhao, H. Shen and M. Ma, "Cross-layer design for reducing delay and maximizing lifetime in industrial wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, pp. 1-26, Dec. 2018. Article (CrossRef Link)

[24] Xu, Xin, Minjiao Yuan, Xiao Liu, Anfeng Liu, Neal N. Xiong, Zhiping Cai, and Tian Wang, "A cross-layer optimized opportunistic routing scheme for loss-and-delay sensitive WSNs," *Sensors*, vol. 18, no. 5, p. 1422, 2018. Article (CrossRef Link)

[25] Kalaikumar, K., and E. Baburaj, "FABC-MACRD: Fuzzy and artificial Bee colony based implementation of MAC, clustering, routing and data delivery by cross-layer approach in WSN," *Wireless Personal Communications*, vol. 103, no. 2, pp. 1633-1655, 2018. Article (CrossRef Link)

[26] Kaur, Tarunpreet, and Dilip Kumar, "MACO-QCR: multi-objective ACO-based QoS-aware cross-layer routing protocols in WSN," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6775-6783, 2021. Article (CrossRef Link)

[27] Lahane, Shivaji R., and Krupa N. Jariwala, "Secured cross-layer cross-domain routing in dense wireless sensor network: A new hybrid based clustering approach," *International Journal of Intelligent Systems*, vol. 36, no. 8, pp. 3789-3812, 2021. Article (CrossRef Link)

[28] V. Nkemeni, F. Mieyeville, and P. Tsafack, "A distributed computing solution based on distributed kalman filter for leak detection in wsn-based water pipeline monitoring," *Sensors (Switzerland)*, vol. 20, no. 18, pp. 1–38, 2020. Article (CrossRef Link)

[29] P. Park, S. C. Ergen, C. Fischione, and A. S.Vincentelli, "Duty-cycle optimization for IEEE 802.15.4 wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 1, pp. 1-32, Dec. 2013. Article (CrossRef Link)

[30] Q. Q. Shi, H. Huo, T. Fang, and D. R. Li, "A Modified Greedy Distance Routing Algorithm for Wireless Sensor Networks," in *Proc. of China-Japan Joint Microwave Conference IEEE*, pp. 197-200, 2008. Article (CrossRef Link)

[31] Shanmugam, Ramalingam, and Baskaran Kaliaperumal, "An energy-efficient clustering and cross-layer-based opportunistic routing protocol (CORP) for wireless sensor network," *International Journal of Communication Systems*, vol. 34, no. 7, p. e4752, 2021. Article (CrossRef Link)

[32] Patil, Urmila, Anju V. Kulkarni, Radhika Menon, and Mithra Venkatesan, "A novel AEB-AODV based AADITHYA cross layer design hibernation algorithm for energy optimization in WSN," *Wireless Personal Communications*, vol. 117, no. 2, pp. 1419-1439, 2021. Article (CrossRef Link)
[33] Mahajan, Hemant B., and Anil Badarla, "Cross-layer protocol for WSN-assisted IoT smart farming applications using nature inspired algorithm," *Wireless Personal Communications*, vol. 121, no. 4, pp. 3125-3149, 2021. Article (CrossRef Link)

Ghanshyam Singh received a B.E. degree in Electronics and Communications from University of Rajasthan, Jaipur, India, in 2009. He pursued M.Tech in Digital Communication from Rajasthan Technical University, Kota, India, in 2012. He is currently pursuing Ph.D. in Electronics and Communications from the National Institute of Technology, Raipur, India. He has published across 8 research articles in Journals and international conferences His areas of interest include developing energy-efficient cross-layer models for WSN and IoT networks.

Pallavi Joshi received the B.E degree in Electronics and Communications from Rajiv Gandhi Technical University Bhopal, India in 2011. She pursued MTech in Embedded Systems from VIT University, Vellore, India in 2016. She is currently pursuing PhD in Electronics and Communications from National Institute of Technology, Raipur, India. She has published across 10 research articles in Journals and international conferences and achieved the best paper award in MCCS-2020 international conference. Her areas of interests focus on model development of data transmission in IoT enabled WSN applications, resource optimization in data aggregation in sensor networks using machine learning approaches.

Ajay Singh Raghuvanshi received his B.Tech. degree in Electronics and Communication Engineering from the North Eastern Regional Institute of Science and Technology, North Eastern Hill University, India in 1993. He has obtained his PhD degree at the Department of Electronics and Communication Engineering, Motilal Nehru National Institute of Technology, Allahabad, India in 2012. He is currently working as an assistant professor at NIT Raipur, India. He has published more than 20 research articles in the journals and international conferences. His research interests are in the area of artificial intelligence and machine learning-based approaches in data aggregation and intrusion detection in wireless sensor networks and Internet of Things.