IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 12, 2021, accepted August 24, 2021, date of publication September 3, 2021, date of current version September 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3110271

Coverage Aware Scheduling Strategies for 3D Wireless Video Sensor Nodes to Enhance Network Lifetime

KISHALAY BAIRAGI, (Graduate Student Member, IEEE), SULATA MITRA, AND UMA BHATTACHARYA

Department of Computer Science and Technology, IIEST Shibpur, Howrah 711103, India

Corresponding author: Kishalay Bairagi (kishalayb29@gmail.com)

ABSTRACT High-density wireless video sensor nodes (VSNs) having limited battery power are deployed randomly in the disaster-hit area for capturing visual data, but its local processing and transmission consume high energy. High deployment density of those VSNs results in a larger overlap in the coverage area across VSNs that can be utilized to cover the sensing region of some VSNs and shut off such VSNs to decrease energy consumption and increase network lifetime without losing much area coverage. Two advanced approaches (APP_5 and APP_6) with realistic 3D rectangular pyramid camera coverage of VSN monitoring 2D target area is proposed in this paper. These approaches reduce the number of active VSNs in the target area and energy consumption maintaining the overall coverage area above some threshold value ensuring network connectivity. The approaches are compared with the three state-of-the-art approaches EX_1, EX_2 and EX_3 in the same simulation setup. Observed that for 150 deployed VSNs over the target area of size 75 × 75 square meters, APP_5 and APP_6 reduce energy consumption by 6.98% and 18.6% respectively from the existing approach EX_3 (producing a better result among three existing approaches in terms of energy consumption). Reducing the number of active VSNs helps decrease energy consumption at the expense of reduced area coverage. For the same node density, both APP_5 and APP_6 lose a little amount of area coverage (i.e. 0.93% and 0.95%) than the existing approach EX_2 (producing a better result among three existing approaches in terms of percentage of area coverage). Additionally, both the proposed approaches (having the same communication overhead) establish superiority by 3.19%/7.83%/4.25% from EX_1/EX_2/(EX_3) in terms of communication overhead for 100 deployed VSNs on the same target area. Finally, APP_6 substantiates superiority in terms of reduced energy consumption (11.97%) than APP_5 losing a very little percentage (0.02%) of area coverage for 150 deployed VSNs.

INDEX TERMS 2D target area, 3D video sensor nodes, area coverage, backup set computation, energy consumption, network lifetime, random deployment.

I. INTRODUCTION

A wireless video sensor network (WVSN) consists of a set of video sensor nodes (VSNs) equipped with tiny miniature video cameras (associated with image and video capture functionality) known as CMOS cameras. Such sensors which have image and video capture functionality can be used in varieties of applications like monitoring in the disaster-hit area, environment monitoring, area surveillance, tracking etc. Such sensors being resource-constrained devices need high bandwidth and energy for the generation of audio and video streams and can run out of energy very quickly for continuous sensing and transmission of video streams. It decreases network lifetime along with monitoring quality.

The WVSN typically operates in an aggressive and hostile environment too, requiring the random deployment of VSNs with high deployment density which helps WVSN maintain its smooth working even if a few VSNs fail to operate. On the other hand, the high deployment density of VSNs results in a larger overlap in the coverage area across VSNs. Such overlapping coverage can be utilized to shut off some VSNs whose sensing regions are covered by the remaining VSNs in the coverage area.

The associate editor coordinating the review of this manuscript and approving it for publication was Ding Xu.

The backup set of VSN v is the set of VSNs that cover its sensing region. An eligible backup set is a backup set with overlapping coverage greater than or equal to a threshold value (Th_V). BS(v) constitutes the set of all the eligible backup sets of the VSN v sorted in non-increasing order of overlapping area coverage. VSN v can be shut off in presence of an eligible backup set in BS(v). An eligible backup set dies if any VSN in the set dies due to energy depletion. VSN v can remain still in shutoff condition in presence of any other active eligible backup set. In absence of any such backup set, the VSN v itself becomes active. Such methodology is formally known as duty cycling, which helps increase the overall network lifetime by reducing the number of active VSNs.

Many works [1]–[6] have already been reported to compute a backup set of wireless sensor nodes (WSN). A 2D coverage model of WSN is used in [1]–[6] and a 2D coverage model of VSN is used in [7]–[14] for computing backup sets. But such 2D modelling does not represent practical camera coverage. Each VSN sends an activity message to its neighbour VSNs to go into the sleep mode in [8]–[15]. But such transmission and reception of activity message by VSNs maintain no ordering which incurs huge message loss. For example, VSN v may have an eligible backup set which implies that VSN may go into sleep mode. But, due to message loss, the VSN v may not receive some activity messages from its neighbours belonging to its eligible backup set and is unable to go into sleep mode. As a result, a subset (of small size) of the set of eligible VSNs go into sleep mode and most of the VSNs remain active.

Minimization of the number of active VSNs thus becomes a concern in WVSN. But the reduction in the number of active VSNs leads to decrease area coverage which may hamper network connectivity. Thus, minimization of the number of active VSNs would be such that it must guarantee at least a threshold amount of area coverage ($Th_{coveraage}$) which is needed to maintain network connectivity.

In this paper, the proposed work considers the more realistic 3D modelling of VSNs deployed over a 2D target area. Each 3D VSN has a rectangular base. The camera of 3D VSN located at point P and also four points of the rectangular base (not shown in this figure) is in the 3D space. Projection of these base points over the 2D target area forms a trapezoid $D_1D_2D_3D_4$ (shown in Fig. 1) which is detailed in Section 3 while discussing the coverage model.

Therefore, when a large number of 3D VSNs are deployed to monitor a 2D target area, their trapezoidal sensing regions over the target area overlap with each other as shown in Fig. 2.

This overlapping coverage is utilized by a 3D VSN to compute its set of all the eligible backup sets. Such computation is challenging considering the trapezoidal sensing region of the VSN.

Two heuristic approaches (APP_5 and APP_6) with polynomial run time complexity are proposed in this paper. These works target to minimize the number of active VSNs while ensuring area coverage in the target area to remain above the threshold value. A base station is situated at the bank of the target area in APP_5 and the position of the base station is provided to all the VSNs before their deployment in the target area. In APP_6 a square-based target area is divided into four equal grids and each grid belongs to a quadrant. A base station is situated at the bank of each quadrant. The q^{th} $(1 \leq q \leq 4)$ base station is associated with the q^{th} quadrant of the target area. The position of the q^{th} base station is provided to all the VSNs in the q^{th} quadrant before their deployment. All the four base stations in APP_6 are connected with each other via high speed dedicated wired link. The base station/(q^{th} base station) is wirelessly connected to WVSN in the target area/(q^{th} quadrant) via some VSNs which are within the communication range of the base station/(q^{th} base station) in APP_5/(APP_6). In both the approaches, each VSN executes neighbour discovery phase, registration phase and scheduling phase sequentially. In the scheduling phase, each VSN has

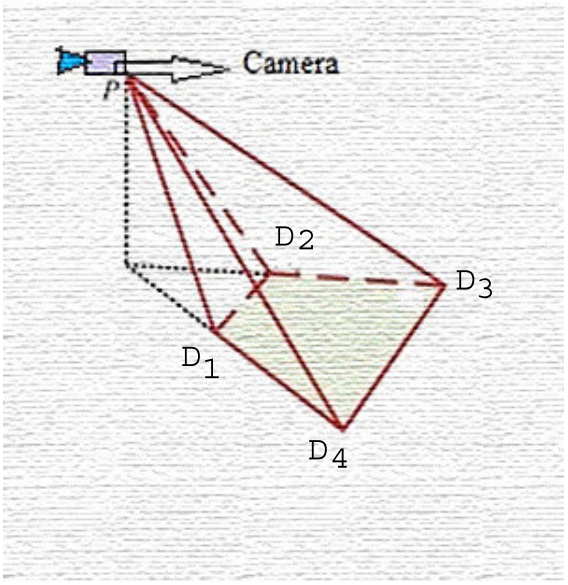


FIGURE 1. Projection of 3D VSN on a plane.

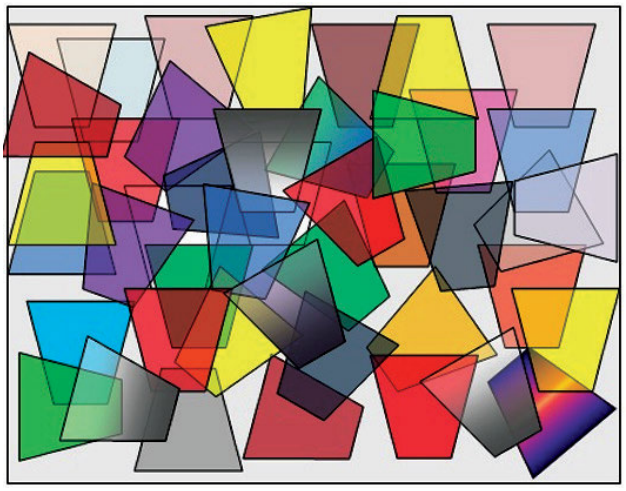


FIGURE 2. Randomly deployed 3D VSNs on a 2D plane.

to undergo two sub-phases - backup set computation and duty cycling. Both the approaches differ in the duty cycling sub-phase.

The major contributions of this paper are as follows:

(i) A realistic coverage model of VSNs has been described by adopting 3D VSNs projected on a 2D plane surface.

(ii) The backup set computation technique used for 2D VSNs in [7], [8] is upgraded for 3D VSNs in the proposed work, APP_5 and APP_6.

(iii) The performance of APP_5 and APP_6 proposed in this paper is studied qualitatively and quantitatively. The qualitative performance is evaluated in terms of communication, storage and computation overhead. Simulation experiments are conducted to observe the variation of the number of active VSNs, total energy consumption, residual energy and percentage of area coverage by the set of active VSNs with the density of VSNs in the target area as well as with the simulation time. The variation of network lifetime and delay in executing all the phases with the density of VSNs in the target area are also studied during the simulation. Both the qualitative and quantitative performance of APP_5 and APP_6 are compared with the three existing approaches EX_1 (upgraded 3D version of [7]), EX_2 (upgraded 3D version of [8]) and EX_3 (upgraded 3D version of [2]).

It has been observed during simulation that APP_6 performs much better in terms of energy consumption than APP_5, EX_1, EX_2 and EX_3. Both APP_5 and APP_6 reduce energy consumption, communication overhead and increase network lifetime compared to EX_1 and EX_2. APP_5 performs well in terms of energy consumption than EX_3 for a higher density of VSNs. Both APP_5 and APP_6 reduce communication overhead compared to EX_3.

In this paper, Section II covers some related works, Section III deals with coverage model, network model and some definitions. The energy consumption model is described in Section IV, Section V deals with proposed work and Section VI describes existing approaches (EX_1, EX_2 and EX_3). Section VII analyses the qualitative performance of the proposed approaches and the existing approaches, Section VIII describes simulation experiments and quantitative performance evaluation, Section IX documents observation related to the performance of APP_5 and APP_6 and Section X provides experimental analysis. Finally, Section XI concludes the paper suggesting the future scope followed by references.

II. RELATED WORK

Several works [1]–[3] to compute the backup set of WSN exist. A 2D omni-directional sensing model i.e. circular sensing model is used in [1]–[3] and in [4]–[6]. The backup sets in [1]–[3] are computed considering 2D modelling of the field of view (FoV) of WSN.

A 2D directional coverage model for VSN v is used in [7]–[14]. The sensing region or Field of View (FoV) of a VSN v is modelled as an isosceles triangle with an angle of view (AoV) as the vertex angle of the triangle and the depth of view (DoV) as the height of the triangle. Backup sets are computed in [7]–[14] considering 2D modelling of FoV of VSN. However, the 2D modelling of FoV does not represent practical camera coverage.

Two correlation-based sensor scheduling algorithms are proposed in [4]. The scheduling algorithms are cluster-based. The cluster formation and cluster head election is a complicated and time-consuming operation. The work in [5] investigates the area coverage configuration method based cooperative sensing model in WSNs (Tri-DCP). It can make active sensors build the triangle cell structure in the target area by self-organization. Tri-DCP is more efficient than some of the existing protocols concerning the reduction in energy consumption and prolonging network lifetime. A novel probabilistic method of sensor node scheduling is presented in [6] to increase network lifetime while maintaining the network coverage. But the problem is the (π-m) coverage problem where every cell in the target area is covered by (π-m) number of sensors [6]. Another novel sensor node scheduling algorithm is proposed in [2]. The sensor node scheduling algorithm in [2] is based on the redundancy of WSN. The algorithm for shutting off WSNs is hybrid (i.e. combination of distributed and centralized) and grid-based. But heuristics in [2], [4]–[6] are based on the 2D omni-directional sensing mode of WSNs which is not a practical camera coverage model and they also lack any collision handling mechanism to prevent message loss due to collision. A distributed duty cycling approach is proposed in [8]–[15]. Each VSN generates two activity messages by specifying its active/inactive status and sends these two activity messages to its neighbour VSNs. One activity message is sent by the VSN to its neighbours when it wants to decide whether to stay active or to go into inactive mode. The other activity message is sent by it after taking the decision of staying awake or going into sleep mode. But no order is maintained during such transmission and reception of message which causes huge message loss. In EX_1 two duty cycling approaches have been proposed where a mixture of large percentage (60%) of static all-time active VSN (ATVSN) and a small percentage (40%) of static active/inactive VSN (AIVSN) are deployed randomly in the target area. It is an improvement over the duty cycling approach as stated in EX_2 and in other approaches [9]–[15]. Only AIVSNs in EX_1 participate in the duty cycling approach. This reduces collision among messages and more VSNs go into sleep mode. Two drawbacks exist in EX_1. First is only 40% of total VSNs (AIVSNs) are eligible to go into sleep mode. Secondly, message loss due to collision has been avoided by creating fewer messages, but this has not been addressed from the physical level or MAC level of VSN. The physical level or MAC level protocols e.g. Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), Carrier Sense Multiple Access/Collision Detection (CSMA/CD) etc. help either to prevent from or to detect collision among a large number of messages during transmission. A differential coding based scheduling framework has been proposed in [16] for efficiently gathering visually correlated images.

This framework consists of two components including MinMax Degree Hub Location and Maximum Lifetime Scheduling. The proposed differential coding based scheduling can effectively enhance the network throughput and the energy efficiency of camera sensors. But, although, MinMax Degree Hub Location problem is NP-complete, the Maximum Lifetime Scheduling problem is NP-Hard. In [17] two problems have been addressed. One is camera scheduling i.e. the selection among available possibilities of a set of cameras for providing the desired coverage at each time instance and the second is energy allocation i.e. the distribution of total available energy among the camera sensor nodes. The energy allocation problem is formulated as a max-min optimization problem that aims to maximize the duration of coverage for the most critical part of the monitored region for which the available energy is the least. But max-min optimization problem is an NP-Hard problem which is solvable only for a small problem size. A priority-based real-time dynamic scheduling algorithm scheme for wireless multimedia sensor networks has been proposed in [18]. Although scheduling has been done only at the physical level of the sensor node, there is no mechanism of preventing message loss due to collision. Besides, the scheme in [18] doesn't have any application-level scheduling among VSNs and as a result, all VSNs stay awake. In [19] a prioritized scheduling algorithm has been proposed to enhance the lifetime of the network. But, here all VSNs are not stationary. Some VSNs especially camera sensors are movable sensors resulting in a lot of waste of energy. In [20] an optimal point of intelligence partitioning between the sensor node and the central base station has been chosen. Results in [20] show that sending compressed images after segmentation will result in a longer life for the sensor node. But as all VSNs in the target area stay awake both in [18], [20], there still exists a lot of wastage of energy and redundancy in visual data. A two-phase algorithm has been proposed in [21] as an algorithm based on Binary Integer Programming that can solve the optimal camera placement problem for a placement space larger than in the current study. This study solves the problem in three-dimensional space for a real-world structure. A binary particle swarm optimization algorithm to solve a homogeneous planned camera network placement problem has been provided in [23]. But both [21], [23] deal with planned deployment of VSNs in the target area which is not possible in the post-disaster scenario.

Two very recent papers [22], [24] consider non-static (movable) 3D VSNs deployed over 2D plane. Motivation in both the papers is to mainly enhance area coverage by the active sensor nodes (deployed over 2D plane) by spreading non-static VSNs which reduce overlapping coverage area and then shutting off some VSNs while maintaining the same area coverage. Objectives of the proposed approaches (APP_5 and APP_6) are different from that of [22], [24] but similar to that of EX_1, EX_2 and EX_3.

Significance of the Proposed Approaches APP_5 and APP_6:

The approaches APP_5 and APP_6 use the 3D coverage model of VSN which is a more practical camera coverage model than the 2D omni-directional (circular) coverage model of WSN as considered in [1]–[6] and 2D coverage model of VSN as considered in [7]–[14]. Unlike [4] the proposed approaches are centralized. This reduces complications associated with cluster formation, cluster head election and delay. The proposed approaches are based on a 1-coverage problem unlike [6].The proposed centralized duty cycling does scheduling at the application level of VSN as well as MAC level to take care of message loss due to collision unlike in [1]–[15], [18]. Such a hybrid type of scheduling reduces message loss and energy consumption, increases network lifetime than the existing approaches as observed during the simulation. The use of stationary VSN further reduces energy consumption due to the mobility of VSNs unlike [19], [22], [24]. Both APP_5 and APP_6 can be solved in polynomial time unlike [16], [17]. All the VSNs in the target area do not have to stay awake as in [18], [20]. Both APP_5 and APP_6 shut off a few VSNs in the duty cycling sub-phase without much change in area coverage and hence collect the same amount of visual data with reduced data redundancy unlike in [18], [20]. Finally, the random distribution of 3D VSNs unlike [21], [23] in the target area is suitable for the area where the human being can't reach fast. The list of acronyms used in this paper is displayed in TABLE 1. Several related recent works, as well as proposed works in this context, are summarized in TABLE 2.

III. COVERAGE MODEL, NETWORK MODEL AND SOME DEFINITIONS

A. COVERAGE MODEL

The 3D directional sensing model of VSN v is like a rectangular pyramid in shape as shown in Fig. 3 a. It has a rectangular base $V_1V_2V_3V_4$ where all the four points V_1, V_2, V_3 and V_4 are in 3D space. The apex point P(x,y,z) of the rectangular pyramid is the position of the camera of the VSN v which is in 3D space. Here, $\overrightarrow{D}$ is the line of sight or direction of the VSN v where $\overrightarrow{D}$ is perpendicular to the base $V_1V_2V_3V_4$ (Fig. 3 a) of the rectangular pyramid. 2α and 2β denote the horizontal and vertical AoV respectively whereas α and β are the respective horizontal and vertical offset angles [25], [26] of 3D sensing model of the VSN v. In 3D directional sensing model the projection of the points, V_1, V_2, V_3 and V_4 in 3D plane becomes the points D_1, D_2, D_3, D_4 in 2D plane respectively. The FoV of the VSN v on a 2D plane is a trapezoidal area $D_1D_2D_3D_4$ formed by the points D_1, D_2, D_3 and D_4 (Fig. 3 b). In Fig. 3 b, C is the intersecting point between the 2D target area that has been defined as $z = 0$ in the X-Y-Z coordinate system and the line of sight $\overrightarrow{D}$. The point C is the centroid of the sensing area ($D_1D_2D_3D_4$). The 3D directional model of the VSN v is shown in more detail in Fig. 3 c after the projection of the rectangular pyramid ($PV_1V_2V_3V_4$) in Fig. 3 a on the 2D plane. That's why in Fig. 3 c the points, V_1, V_2, V_3, V_4 and the line of sight $\overrightarrow{D}$ have not been shown.

TABLE 1. Nomenclature table.

Acronym	Description
2D	Two Dimensional
3D	Three Dimensional
AIVSN	Active/Inactive Video Sensor Node
ATVSN	All-Time Active Video Sensor Node
AoV	Angle of View
BT	Base Table
CMOS	Complementary Metal Oxide Semiconductor
CSMA/CA	Career Sense Multiple Access with Collision Avoidance
CSMA/CD	Career Sense Multiple Access with Collision Detection
DoV	Depth of View
FoV	Field of View
GPS	Global Positioning System
GPSR	Greedy Perimeter Stateless Routing
ICGA	Iterative Centralized Greedy Algorithm
id	identification
MAC	Media Access Control
RL	Request List
Size_AM	Size of an activity message
Size_D	Size of a dead message
Size_id	Size of id of a VSN
Size_Rec_NT	Size of a record in neighbour table
Size_RM	Size of a request message
Size_RS	Size of a request string
tunable MAC	Tunable Media Access Control
VSNs	Video Sensor Nodes
WSNs	Wireless Sensor Nodes
WVSN	Wireless Video Sensor Network

Let P′ is the projection of P(x,y,z)(which is the position of the VSN v) on z = 0. A 3D directional sensing model can be described as (P, Q_1, Q_2, $|\overrightarrow{P'Q_2}|$, $|\overrightarrow{P'Q_1}|$, γ, α, β) (Fig. 3 c) where Q_1, Q_2 are the intersecting points between the lines D_1D_2 and $|\overrightarrow{P'Q_1}|$, D_3D_4 and $|\overrightarrow{P'Q_2}|$ respectively. $|\overrightarrow{P'Q_2}|$ and $|\overrightarrow{P'Q_1}|$ are the line of sight of the VSN v. ($|\overrightarrow{P'Q_2}|$-$|\overrightarrow{P'Q_1}|$) = Q_2Q_1 is the DoV for the VSN v where $|\overrightarrow{P'Q_2}|$ is the DoV for the triangle $P'D_3D_4$ and $|\overrightarrow{P'Q_1}|$ is the DoV for the triangle $P'D_1D_2$. 2α and 2β denote the horizontal and vertical AoV respectively. γ called a tilt angle is an angle between ($\overrightarrow{PC}$) and PP′. Q_1 is the centroid of segment D_1D_2 and Q_2 is the centroid of segment D_3D_4 as shown in Fig. 3 c. Besides, each VSN is identified by a unique identification (id) number in the target area.

The x and y coordinates of the vertices (D_1, D_2, D_3 and D_4) of the projected trapezoidal area are calculated using Fig. 3 c.

$$|\overrightarrow{P'Q_1}| = zXtan(\gamma - \beta) \quad (1)$$

$$|\overrightarrow{P'Q_2}| = zXtan(\gamma + \beta) \quad (2)$$

$$d_1 = |\overrightarrow{P'D_1}| = |\overrightarrow{P'D_2}| = |\overrightarrow{P'Q_1}|/cos(\alpha) \quad (3)$$

$$d_2 = |\overrightarrow{P'D_3}| = |\overrightarrow{P'D_4}| = |\overrightarrow{P'Q_2}|/cos(\alpha) \quad (4)$$

From the equations (3) and (4) it can be said that

$$|\overrightarrow{P'D_4}| - |\overrightarrow{P'D_1}| = |D_1D_4| = |\overrightarrow{P'D_3}| - |\overrightarrow{P'D_2}| = |D_2D_3| \quad (5)$$

D_1.x, D_2.x, D_3.x, D_4.x are the X-coordinates and D_1.y, D_2.y, D_3.y, D_4.y are the Y-coordinates of the vertices D_1, D_2, D_3, D_4 respectively. Equation 5 reveals that the projected trapezoidal area, $D_1D_2D_3D_4$ is an isosceles trapezoid where $|D_1D_4| = |D_2D_3|$. D_1D_2 and D_3D_4 are the two bases of the trapezoid.

From equations 3 and 4 it is also clear that the triangles, $P'D_1D_2$ and $P'D_3D_4$ are isosceles triangles. The vertex angle of both the triangles is (2α) as shown in Fig. 3 c. $P'Q_1$ is the median line for the triangle $P'D_1D_2$ and $P'Q_2$ is the median line for the triangle $P'D_3D_4$. Now, a point (say M) (Fig. 4) is said to be inside the trapezoidal FoV $D_1D_2D_3D_4$ if the following two conditions are satisfied.

- $P'M_1 \leq P'M \leq P'M_2$, where M_1 is the intersection point between the line P′M and the line D_1D_2, M_2 is the intersection point between the line P′ M and the line D_3D_4.
- The angle between the line P′M and the line $P'Q_1$ or $P'Q_2$ must be within the range of $[-\alpha, +\alpha]$

Informally, a point (say M) is said to be inside the trapezoidal FoV $D_1D_2D_3D_4$ if the point M is inside the triangle $P'D_3D_4$, but not inside the triangle $P'D_1D_2$ (Fig. 4).

B. NETWORK MODEL

Both the proposed approaches and the existing approaches follow the same network model which is stated below in a step by step manner.

- The target area of the static 3D VSNs is a 2D square plane and the location of the camera of each VSN is on the same horizontal plane (x, y) where $z = k_1$, k_1 is a constant and is fixed after the initial deployment.
- Each VSN knows its position using GPS and can detect any event within its coverage area.
- APP_5 has one base station and APP_6 has four base stations corresponding to four quadrants. The base stations in APP_6 are connected by dedicated wired links.
- The position of the base station/q^{th} quadrant is at the bank of the target area/q^{th} quadrant for APP_5/APP_6

TABLE 2. Comparative study among related works and the proposed approaches.

Method	Sensor Type	Coverage Model	FoV Area	Duty Cycling Strategy	Deployment	Collision Handling Mechanism	Target Area or Space	Coverage Enhancement
K. Bairagi et al. [7]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	YES	2D	NO
K. Bairagi et al. [8]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
N. Bendimerad et al. [9]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
N. Bendimerad et al. [10]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
A. Salim et al. [11]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
A. Makhoul et al. [12]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
V. Ukani et al. [13]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
A. Benzerbadj et al. [14]	Static VSN	2D Directional	Isosceles Triangle	Distributed	Random	NO	2D	NO
V. Ukani et al. [15]	Static VSN	3D Directional	Rectangular Pyramid	Distributed	Random	NO	3D	NO
L. Zhang et al. [23]	Non Static VSN	3D Directional	Various 2D Geometrical Shapes	Centralized	Random	NO	2D	YES
Z. Jiao et al. [24]	Non Static VSN	3D Directional	Elliptical	Centralized	Random	NO	2D	YES
EX_1	Static VSN	3D Directional	Trapezoidal	Distributed	Random	YES	2D	NO
EX_2	Static VSN	3D Directional	Trapezoidal	Distributed	Random	NO	2D	NO
EX_3	Static VSN	3D Directional	Trapezoidal	Hybrid	Random	NO	2D	NO
APP_5	Static VSN	3D Directional	Trapezoidal	Centralized	Random	YES	2D	NO
APP_6	Static VSN	3D Directional	Trapezoidal	Centralized	Random	YES	2D	NO

• The base station/(q^{th} base station) in APP_5 / (APP_6) also knows its position using GPS and the dimension of the target area/(q^{th} quadrant).

• All 3D VSNs exchange their location information and the orientation with their neighbours.

• Two VSNs are said to be wirelessly connected if and only if they are within each other's communication range. The communication model in the present work follows the unit disk graph model [27].

• Communication range of the base station/(q^{th} base station) in APP_5/(APP_6) is equal to the communication range of each VSN.

• Communication range of each VSN has been considered greater than the sensing range.

• All 3D VSNs are homogeneous, which indicates that α, β and γ for each of them are the same. Each VSN has the same sensing (coverage area) and communication range.

• WVSN remains operational so long as the area coverage remains above some threshold value ($Th_{coverage}$).

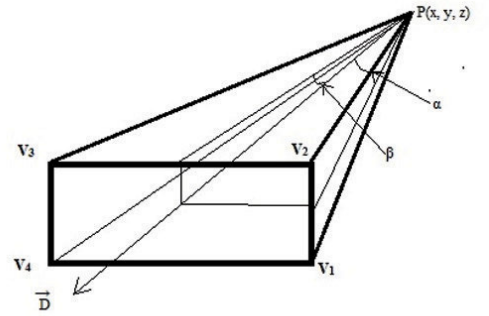
C. SOME DEFINITIONS

• N(v) is the set of neighbour VSNs of the VSN v.

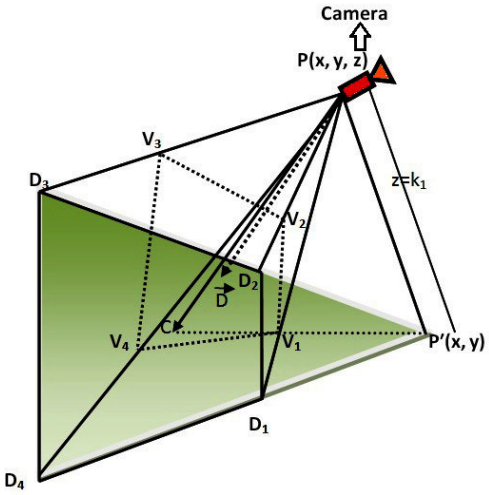
• $BS_i(v)$ is the i^{th} backup set of the VSN v associated with an overlapping coverage value. A subset of neighbour VSNs of VSN v that covers the FoV of VSN v forms $BS_i(v)$. So, $BS_i(v) \in BS(v)$. Size of $BS_i(v)$ is the number of neighbour VSNs of VSN v in that backup set.

• A backup set $BS_i(v)$ with an overlapping coverage (Ov_CoV_i) of greater than or equal to Th_V is termed as an eligible backup set. Ov_CoV_i is defined as (FoV area of v covered by VSNs in $BS_i(v)$/FoV area of v) * 100%

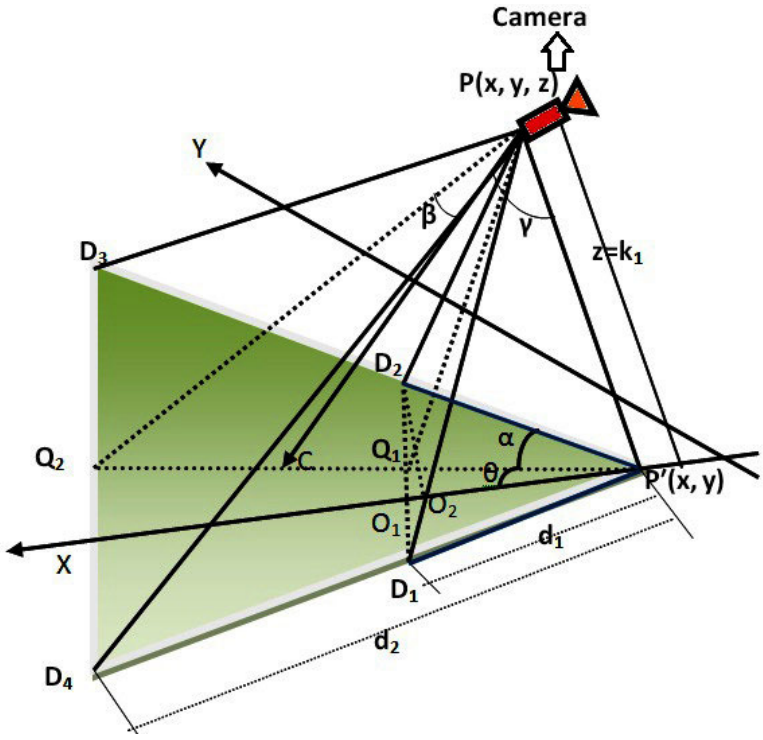
• BS(v) is the set of all the eligible backup sets of the VSN v, sorted already in terms of non-increasing overlapping coverage. Therefore, $BS(v) = \bigcup_i BS_i(v)$.



(a)



(b)



(c)

FIGURE 3. a. 3D directional sensing model. b. The projected trapezoidal area on the target 2D plane. c. Details of 3D directional sensing model.

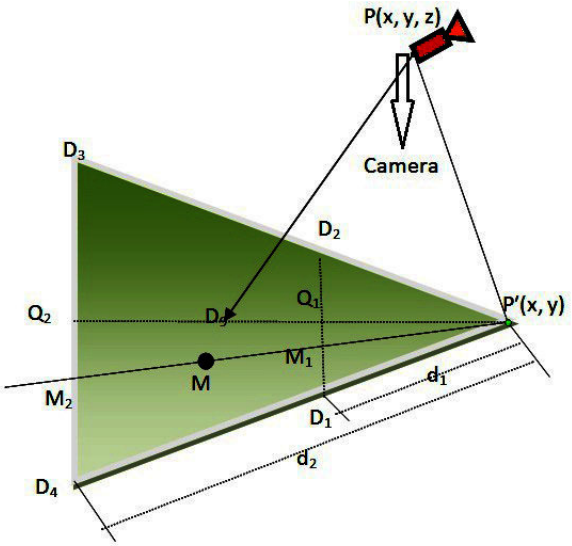


FIGURE 4. A point within the sensing area.

- Size of BS(v) is the number of backup sets inside BS(v).
- Best backup set is an eligible backup set with the highest overlapping coverage.

IV. ENERGY CONSUMPTION MODEL

The energy consumption model for VSN v in a WVSN includes two parts describing two states (active and inactive) of the VSN v.

A. ACTIVE MODE

Three energy consuming components are associated with each VSN v in active mode: (i) minimum energy consumption by the VSN ($E_{Baseline}$), (ii) energy consumption due to message exchange (i.e. radio transmission and reception) (E_{Radio}) and (iii) energy consumption due to processing (E_{App}). The total energy consumption by the VSN v (E_{va}) is calculated by using the equation

$$E_{\text{va}} = E_{\text{Baseline}} + E_{\text{App}} + E_{\text{Radio}} \tag{6}$$

Let t = Simulation period; and t_1 = ImageCaptureCount * (TimeForImageCapture + TimeForImageProcessing) where TimeForImageCapture is the time needed to capture one image and TimeForImageProcessing is the time needed to process one image. ImageCaptureCount is the number of images captured by the VSN v.

$E_{Baseline} = (t - t_1) * P_{Baseline}$ where $P_{Baseline}$ is the power consumption by the VSN v during the time period $(t - t_1)$, considering the power consumption remains the same throughout the period. $P_{Baseline}$ includes power consumption due to activity of microprocessor in the VSN, power consumption due to turning on the VSN and power consumption due to turning off the VSN.

E_{App} = (MeasuredEnergyPerImageCapture + MeasuredEnergyPerImageProcessing) * ImageCaptureCount where MeasuredEnergyPerImageCapture is energy spent due to one image capture and MeasuredEnergyPerImageProcessing is energy spent due to one image processing.

$E_{Radio} = E_{Idle} + E_{Transmitting} + E_{Receiving} + E_{Switching} + E_{CSMA/CA}$ where E_{Idle}, $E_{Transmitting}$ and $E_{Receiving}$ are the energy spent during the periods of the idle state, transmitting state and receiving state of the radio of the VSN v respectively. The radio of the VSN v can be in one of these three states during the time period t. Energy spent during the idle state of radio i.e. E_{Idle} is minimum among these three states. Switching energy ($E_{Switching}$) is expended when the VSN changes from receiving to transmission mode or vice versa. The CSMA/CA (Carrier sense multiple access/Collision avoidance) algorithm states that each time a VSN intends to transmit, it first checks if the channel is free from other transmission. If this happens then the VSN proceeds to transmit. $E_{CSMA/CA}$ is the energy consumed by the execution of CSMA/CA algorithm.

B. INACTIVE MODE

Let the VSN v is in inactive/sleep mode during simulation period (0-t). It is to be noted that in inactive/sleep state, VSNs do not actually transmit but can receive a limited amount of message; some parts of the sensor circuitry [e.g., microprocessor, memory, radio frequency components] are turned off; and the power consumption as well as the operational capabilities of the sensor decreases which makes energy consumption for the inactive VSN v (E_{vi}) almost equal to zero. Therefore,

$$E_{\text{vi}} \approx 0 \tag{7}$$

Calculation of Energy Consumption ($Energy_{Tot}$):

Let $Energy_{Tot}$ is defined as the total energy consumption by the total number of active VSNs (Active_VSN) in joule during the simulation time 0 to t seconds. $Energy_{Tot}$ is calculated for Active_VSN. Therefore, according to the energy consumption model stated above $Energy_{Tot}$ is calculated at simulation time t seconds as

$$Energy_{\text{Tot}} = E_{\text{va}} X Active\_VSN \tag{8}$$

assuming all VSNs are deployed at the same time i.e. at t = 0

V. PROPOSED WORK

In both APP_5 and APP_6, each VSN executes the neighbour discovery phase, registration phase and scheduling phase sequentially. In the scheduling phase, each VSN has to undergo two sub-phases - backup set computation and duty cycling. Both the approaches differ in the duty cycling sub-phase. In the neighbour discovery phase, each VSN knows the orientation and position of its neighbours by exchanging messages among themselves and inserts such neighbour information in a neighbour table both in APP_5 and APP_6. In the registration phase, each VSN in the target area/(q^{th} quadrant) in APP_5/(APP_6) sends its position and orientation to the base station/(q^{th} base station). Each VSN in APP_5/(APP_6) selects a route from itself to the base station/(q^{th} base station) using multi-hop based GPSR routing protocol [28]. The base

station/(q^{th} base station) inserts such information into a table called base table (BT). In the backup set computation sub-phase, each VSN computes a set of backup sets in the target area. In the duty cycling sub-phase, a VSN generates a request message if it likes to go into the sleep mode and sends the request message to the base station using GPSR routing protocol. The base station generates a request string from the received request message and inserts the request string in a request list (RL). In both approaches, the base station uses an iterative centralized greedy algorithm (ICGA) to process the request strings in the request list for identifying a set of requesting VSNs to be shut off and sends reply (sleep) message to all the identified VSNs. Such VSNs enter into sleep mode after receiving the sleep message.

The neighbour discovery phase, registration phase and the two sub-phases of the scheduling phase are elaborated in this section for both approaches.

A. NEIGHBOUR DISCOVERY PHASE

Each VSN exchanges its id and the FoV parameters with its neighbour VSNs and stores FoV parameters along with the id of its neighbour VSNs in a neighbour table. Such FoV parameters of VSN v include X-Y coordinates of three points (P, Q_2, Q_1 as shown in Fig. 3 c) of the trapezoidal FoV of VSN v, two DoVs, $|\overrightarrow{P'Q_1}|$ and $|\overrightarrow{P'Q_2}|$, two offset angles α and β and the tilt angle γ. The id of each VSN is an integer number of size (Size_id) 2 bytes. Each record in the neighbour table contains three X-coordinates corresponding to the points P, Q_1, Q_2 of type float, three Y-coordinates corresponding to the points P, Q_1, Q_2 of type float, two DoVs of type float, α, β and γ of type float and one id of type integer. The total number of parameters in each record (Tot_Param) is 12. So the size of each record in the neighbour table (Size_Rec_NT) is $(6\times4\times8+2\times4\times8+3\times4\times8+2*8) =$ 368 bits i.e. 46 bytes.

B. REGISTRATION PHASE

Each VSN in the target area/(q^{th} quadrant) in APP_5/(APP_6) sends its FoV parameters to the base station/(q^{th} base station) using GPSR routing protocol in this phase. The base station/(q^{th} base station) inserts these FoV parameters into the base table. In this way, all VSNs in the target area/(q^{th} quadrant) register themselves with the base station/(q^{th} base station). Each record in the base table contains Tot_Param parameters and two more Boolean variables, namely 'isVSNActive' and 'isVSNDead' for all the VSNs registering to the base station/(q^{th} base station). The value of 'isVSNActive' for a particular record is set to 1/(0) if the VSN corresponding to that record is active/(inactive). The value of 'isVSNDead' in the record is 1/(0) if the corresponding VSN is dead/(alive). A VSN exhausted out of energy is considered a dead one. An alive VSN is not dead but can either be active or inactive.

Initially, all VSNs are active, therefore, for all records, the values for 'isVSNActive' and 'is VSNDead' are initialized to 1 and 0 respectively by default. The size of each record in the base table having (Tot_Param + 2) parameters is (Size_Rec_NT + 2) i.e. 370 bits. The base station/(qth base station) in APP_5/(APP_6) calculates the total number of deployed VSNs (Tot_VSN) and % of area coverage by active VSNs (Per_ACoV) in the target area/(q^{th} quadrant) using [equation 9] as follows:

$$\text{Per\_ACoV} = (\text{area covered by active VSNs/total target area}) * 100\% \quad (9)$$

In APP_6, the q^{th} base station calculates Tot_VSN, and Per_ACoV by collaborating with the other base stations using wired connections. As all VSNs are active in this phase, the number of active VSNs (Active_VSN) is equal to Tot_VSN.

C. SCHEDULING PHASE

The two sub-phases of the scheduling phase for both the approaches are elaborated next for VSN v.

1) BACKUP SET COMPUTATION SUB-PHASE

In this sub-phase, VSN v computes backup sets. Probability of existence of small-sized backup set is reasonably high as each such backup set needs less number of VSNs to cover FoV of the VSN v. A small sized backup set also implies less number of VSNs to stay active to cover the FoV of the VSN v. Therefore, usage of small-sized backup set helps increase network lifetime as well as achieve more savings in energy consumption. Consideration of this fact limits sizes of the backup set as 1, 2 and 3 in this proposed work.

a: COMPUTATION OF BACKUP SETS OF SIZE 3

An example of FoV coverage of the VSN v is shown in Fig. 5. $D_1D_2D_3D_4$ is the FoV area of VSN v. D_9 is the centroid point of the trapezoidal FoV of VSN v. v_1, v_2, v_3 are the three neighbour VSNs of VSN v. The FoV of v_3 covers the points D_1 and D_2 in the FoV of VSN v.The FoV of v_1 covers the point D_9. The FoV of v_2 covers the points D_3 and D_4 in the FoV of VSN v. The VSN v computes three sets as D_{12}, D_{34},

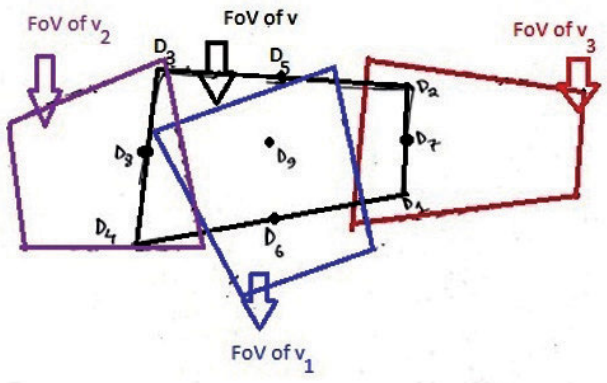


FIGURE 5. Computation of backup set of size 3.

D_{99}. So $D_{12} = \{v_3\}$, $D_{34} = \{v_2\}$ and $D_{99} = \{v_1\}$. D_{99} is the set of neighbour VSNs that have covered the centroid point D_9 of the FoV of VSN v. VSN v computes BS(v) by executing cartesian product among the sets D_{12}, D_{34}, D_{99} to cover all the five points D_1, D_2, D_3, D_4, D_9 in the FoV of VSN v. So (D_{12} X D_{34} X D_{99}) = $\{v_1, v_2, v_3\}$ and BS(v) = $\{\{v_1, v_2, v_3\}\}$. The number of backup sets inside BS(v) is one and the size of this backup set is 3.

So mathematically the FoV of the VSN v is covered by a set $BS_i(v) \in BS(v)$ if the following two conditions are satisfied.

1. For each $v' \in BS_i(v)$, v' must cover either (D_1 and D_2) or (D_3 and D_4) or D_9
2. D_1, D_2, D_3, D_4, D_9 are covered by the VSNs in $BS_i(v)$

This backup set is an eligible backup set if its overlapping coverage is greater than or equal to Th_V.

But if any set among D_{12}, D_{34}, D_{99} is null, the result of (D_{12} X D_{34} X D_{99}) is null. To solve this problem, a few more alternate points like D_5, D_6, D_7, D_8 are considered in the FoV of the VSN v (shown in Fig. 5). D_5, D_6, D_7, D_8 are the midpoints of the line joining D_2 and D_3, D_1 and D_4, D_1 and D_2, D_3 and D_4 respectively.

If D_{12} is null, the VSN v chooses the point D_7 as it is the nearest point from both D_1 and D_2 among all the alternate points and tries to form the set D_{77} to cover D_1D_2. If both the sets D_{12} and D_{77} are null, the VSN v uses either point D_5 or point D_6 as an alternate point. D_5 is the nearest point of D_2, and D_6 is the nearest point of D_1 as shown in Fig. 5. So the VSN v forms the two sets D_{55} and D_{66} simultaneously but uses one of the two sets depending upon their overlapping coverage as an alternative set for D_{12}.

Obviously, it is less probable that the set D_{55} will cover the point D_1. But if its Cartesian product with the other two sets i.e. (D_{55} X D_{99} X D_{34}) creates a backup set of size 3 with overlapping coverage greater than equal to Th_V, then the set D_{55} can be considered as an alternative set for D_{12}. Now, if the overlapping coverage of the best backup set belonging to the set (D_{66} X D_{99} X D_{34}) is greater than that of the best backup set in (D_{55} X D_{99} X D_{34}), then D_{66} is considered as an alternative set for D_{12}.

Similarly, if D_{34} is null, the VSN v chooses the point D_8 as it is the nearest point from both D_3 and D_4 and tries to form the set D_{88} to cover D_3D_4. If both the sets D_{34} and D_{88} are null, the VSN v uses either point D_5 or point D_6 as alternate points. D_5 is the nearest point of D_3 and D_6 is the nearest point of D_4 as shown in Fig. 5. So the VSN v forms two sets D_{55} and D_{66} simultaneously but uses one of the two sets depending upon their overlapping coverage as an alternative set for D_{34}. Obviously, it is less probable that the set D_{55} will cover the point D_4. But if its Cartesian product with the other two sets i.e. (D_{12} X D_{99} X D_{55}) creates a backup set of size 3 with overlapping coverage greater than or equal to Th_V, then the set D_{55} can be considered as an alternative set for D_{34}. Now, if overlapping coverage of the best backup set belonging to the set (D_{12} X D_{99} X D_{66}) is greater than the best backup set in (D_{12} X D_{99} X D_{55}), then D_{66} is considered as an alternative set for D_{34}. If both D_{12} and D_{34} are null sets then the VSN v uses D_{55} in place of D_{12} and D_{66} in place of D_{34}.

D_{99} is null if the centroid point, D_9 in the FoV of the VSN v is not covered by the FoV of its neighbour VSNs. In such a case the VSN v considers a circle of radius half the segment D_5D_9 or D_6D_9, centred around its centroid point, D_9. The VSN v generates a set of four points on the circumference of this circle. The neighbour VSN whose FoV contains any one of these four points is included in the set D_{99} because each of the four points is closer to the point D_9. Therefore each of the four points can be an alternative point of D_9. The four points (not shown in Fig. 5) are D_{10}, D_{11}, D_{12} and D_{13}. The calculation of x and y coordinates of these four points are shown below.

$D_{10}.x = (D_9.x + D_6.x)/2$
$D_{10}.y = (D_9.y + D_6.y)/2$
$D_{11}.x = D_9.x + (0.5*D_5D_9)$
$D_{11}.y = D_9.y$
$D_{12}.x = (D_9.x + D_5.x)/2$
$D_{12}.y = (D_9.y + D_5.y)/2$
$D_{13}.x = D_9.x - (0.5*D_5D_9)$
$D_{13}.y = D_9.y$

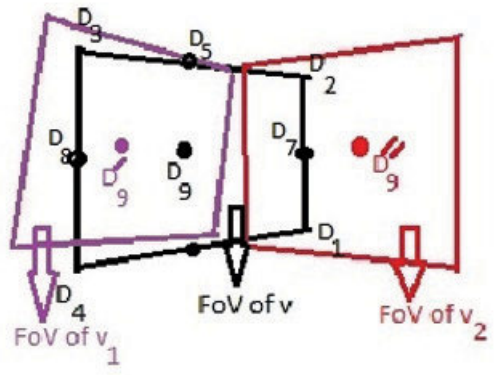
b: COMPUTATION OF BACKUP SET OF SIZE 2

D_9 is the centroid point of the trapezoidal FoV of the VSN v and D_{99} is the set of neighbour VSNs of the VSN v that covers D_9. On the other hand D'_{99} is the set of neighbour VSNs of VSN v whose centroid points are inside the FoV of VSN v. So $D_{99}D'_{99} = (\{D_{99} \bigcap D'_{99}\})$ is the set of neighbour VSNs of VSN v that covers D_9, as well as the set of neighbour VSNs of VSN v whose centroid points are inside the FoV of VSN v.

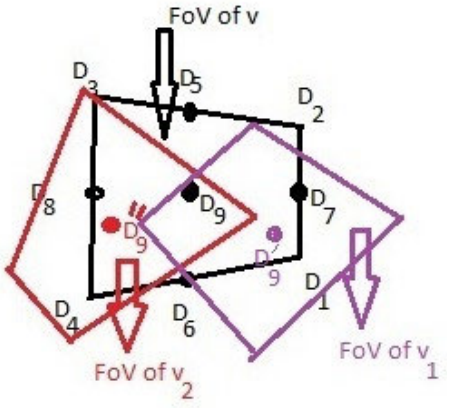
Case 1: It can be observed from Fig. 6 a that v_1 and v_2 are the two neighbour VSNs of VSN v. The FoV of v_1 and FoV of v_2 cover almost the entire FoV of VSN v. Moreover, v_1 belongs to the set $D_{99}D'_{99}$ i.e. $D_{99}D'_{99} = \{v_1\}$ whereas v_2 belongs to the set D_{12} i.e. $D_{12} = \{v_2\}$. The backup set of size 2 is generated by taking the Cartesian product between the sets $D_{99}D'_{99}$ and D_{12} and it is $\{v_1\}X\{v_2\}$ i.e. $\{\{v_1, v_2\}\}$. As the Cartesian product is the ordered pair of elements from each set, the maximum size of the element set inside the resultant set of sets will be two. The result also shows that size of the set is indeed two. But it can be observed from Fig. 6 a that D_{34} is null as there is no neighbour VSN of VSN v that covers the points D_3 and D_4 of the FoV of VSN v. So the set $D_{99}D'_{99}$ X D_{34} is null.

Case 2: It can be observed from Fig. 6 b that $D_{99}D'_{99} = \{v_1, v_2\}$ and D_{12} is null as there is no neighbour VSN of VSN v that covers the point D_1 and D_2 of the FoV of VSN v. So ($D_{99}D'_{99}$ X D_{12}) is null whereas ($D_{99}D'_{99}$ X D_{34}) is $\{v_1, v_2\}X\{v_2\} = \{\{v_1, v_2\}, \{v_2, v_2\}\} = \{\{v_1, v_2\}, \{v_2\}\}$. Hence, the maximum size of the element (set) belonging to the set of sets ($D_{99}D'_{99}$ X D_{34}) is two.

If both D_{12} and D_{34} are not null, then the union operation between the two resultant sets (($D_{99}D'_{99}$ X D_{34}) $\bigcup$ ($D_{99}D'_{99}$ X D_{12}) is taken to produce set of sets having a maximum size of the element (set) 2.



(a)



(b)

FIGURE 6. a. Computation of backup set of size 2: case 1. b. Computation of backup set of size 2: case 2.

c: COMPUTATION OF BACKUP SETS OF SIZE 1

It can be observed from Fig. 7 that the FoV of neighbour VSN v_1 has covered a significant portion of the FoV of the VSN v. The FoV of v_1 covers the centroid point D_9 of the VSN v i.e. $D_{99} = \{v_1\}$ and the FoV of v covers the centroid point D'_9 of neighbour VSN v_1 i.e. $D'_{99} = \{v_1\}$. Now the set $D_{99}D'_{99} = (D_{99} \bigcap D'_{99})$ is the set of neighbour VSNs of v that covers D_9 and also the set of neighbour VSNs of v whose centroid points are inside the FoV of v. Each element inside the set $D_{99}D'_{99}$ with significant overlapping coverage ($\geq$Th_V) can be the backup set of size 1 for the VSN v. As shown in Fig. 7, $D_{99}D'_{99} = (D_{99} \bigcap D'_{99}) = \{v_1\}$ is the backup set of size 1 if the overlapping coverage of the set $\{v_1\}$ is greater than or equal to Th_V.

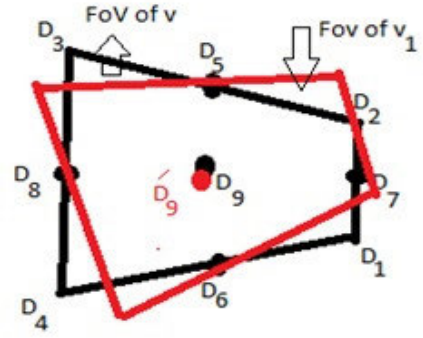


FIGURE 7. Computation of backup set of size 1.

BS(v) is obtained as a result of union operation of all the backup sets of the VSN v of different size. The VSN v discards the backup sets having overlapping coverage less than Th_V from BS(v) and arranges the remaining backup sets (if any) in BS(v) in non-increasing order of their overlapping coverage. The VSN v starts to execute the duty cycling sub-phase if BS(v) is not null.

2) DUTY CYCLING SUB-PHASE

In this sub-phase, the VSN v generates a request message if it wants to go into the sleep mode and locates its position in the target area using GPS in both the approaches. Let the VSN v is in q^{th} quadrant in APP_6. In APP_6 VSN v also checks whether the location of the neighbour VSNs corresponding to its best backup set in BS(v) is also in the q^{th} quadrant. If all these locations are in q^{th} quadrant, the criterion of generating the request message is fulfilled and the VSN v generates a request message. Otherwise, the VSN v repeats the same step with the next best backup set in BS(v). The VSN v does not generate any request message if all the backup sets in BS(v) fail to fulfil the criterion.

In both the approaches, the request message of VSN v contains three parameters. The first parameter is the id of the requesting VSN (VSN v) of size 2 bytes. The second parameter is the ids of VSNs which are present in the best backup set of VSN v. The second parameter contains at least/(at most) one/(three) id(s)(since the maximum size of the backup set is 3 and minimum size of the backup set is 1). So the size of the 2nd parameter varies from 2 to 6 bytes. The third parameter is the overlapping coverage area of the best backup set of VSN v. It is also an integer number of size 2 bytes. The size of the request message consisting of these three parameters (Size_RM) varies from 6 to 10 bytes.

The VSN v uses GPSR routing protocol to select a route for sending the request message to the base station/(qth base station) in APP_5/(APP_6). It also uses tunable MAC protocol not to lose the request message due to collision by exploiting CSMA/CA mechanism.

The base station/(q^{th} base station) in APP_5 /(APP_6) concatenates the three parameters of the request message of VSN v to generate the k^{th} request string. Let the k^{th} request string of VSN v is v/v_2-v_3-v_4/95. Here, v is the id of the requesting VSN. v_2-v_3-v_4, separated by "-" are the id of the VSNs in the best backup set of VSN v and so the size of the best backup set of VSN v is 3. 95 is the % of overlapping coverage of the best backup set of VSN v. The three parameters of the request message of VSN v are separated by "/" in the k^{th} request string. "/" and "-"are stored as the character of one byte each. Therefore, the size of the k^{th} request string (Size_RS) is 14 bytes.

Algorithm 1 ICGA

Input: request list (RL), |RL| the length of the request list, base table (BT), a list (L).
Output: VSN v can be shut off or not.
Step 1: Begin
/*Initialization*/
Step 2: Arrange the request strings in RL in the descending order of their overlapping coverage.
Step 3: Two consecutive request strings in RL having same overlapping coverage are sorted in terms of ascending order of the backup set size.
Step 4: j; count = 0; m = 1; |RL| = number of request strings in RL.
/*Process starts*/
Step 5: Read the id of the requesting VSN as v from k^{th} request string.
Step 6: j = k
/*Searching and storing of the request strings whose backup set contains the id of v*/
Step 7: **while** $j < |RL|$ **do**
  $j = j + 1$
  Search for the id of v in the backup set of j^{th} request string in RL.
  if *found* **then**
    $count = count + 1$
    Read the id of the requesting VSN from the j^{th} request string and store it in a list L[count].
  end
end
/*Shutting off the VSN v and updating RL*/
Step 8: **if** $count \leq 3$ **then**
  Send sleep message to VSN v and VSN v shuts off.
  Search the VSN v by id in BT and Updates the value of isVSNActive and isVSNDead of v to 0 and 0 respectively.
  while $m \leq count$ **do**
    Delete the request string from RL, whose VSN id matches with that present in L[m].
    $m = m + 1$
  end
  $|RL| = |RL| - count$
  Remove the request string for VSN v from RL.
  $|RL| = |RL| - 1$
  /*processing is completed for VSN v*/
  Delete all the ids from the list L.
else
  Remove the request string for VSN v from RL.
  $|RL| = |RL| - 1$
  Delete all the ids from the list L.
end
/*processing starts again with the next available v in RL*/
Step 9: **if** $RL > 1$ **then**
  go to Step 5.
else
  Send sleep message to VSN v and VSN v shuts off./*RL is exhausted*/
  Search the VSN v by id in BT.
  Update the value of isVSNActive and isVSNDead of v to 0 and 0 respectively.
  Remove the request string for VSN v from RL.
end
Step 10: End

v/v_2-v_3-$v_4/95$	v_2/v_5-v_6-$v/90$	v_3/v_5-v_7-$v_8/89$	v_4/v_3-v_2-$v_5/87$	v_5/v_{10}-v_{11}-$v_{12}/86$

FIGURE 8. A snapshot of request list at the base station.

v/v_5-$v_{20}/95$	v_{13}/v-v_3-$v_4/95$	v_3/v_5-v_7-$v_8/89$	v_4/v_3-v_2-$v_5/88$	v_5/v_{11}-$v_{12}/87$

FIGURE 9. A snapshot of request list at the base station.

The base station/(q^{th} base station) observes δ_k as the time delay between the generation of the k^{th} request string and the insertion of the (k-1)th request string in the request list. The base station/(q^{th} base station) computes the mean waiting time (μ_{k-1}) and standard deviation (σ_{k-1}) after inserting the (k-1)th request string in the request list. If $\delta_k \leq (\mu_{k-1} + C*\sigma_{k-1})$(value of C is 3 [29]) the base station/(q^{th} base station) inserts k^{th} request string in the request list, calculates μ_k as $\sum^k \delta_k$/(k-1), var_k as (δ_k - μ_k), σ_k as $\sqrt{\sum (var_k^2/(k-1))}$ and repeats the same steps of operation. Otherwise, the base station/(q^{th} base station) starts the execution of ICGA for the processing of the (k-1) number of request strings from the request list to identify a set of requesting VSNs in the target area/(q^{th} quadrant) to be shut off.

Processing of the k^{th} Request String in the Request List Using ICGA (Algorithm 1):

The processing of the k^{th} request string in the request list for identifying whether the VSN v can be shut off is elaborated in this section. Fig. 8, Fig. 9 and Fig. 10 show the three different request lists at the base station/(q^{th} base station) in APP_5/(APP_6). The k^{th} request string is in the

v/v_2-v_3-$v_4/95$	v_2/v_5-$v/94.5$	v_3/v_5-v_7-$v/94$	v_4/v_3-v_2-$v/93$	v_5/v-v_{11}-$v_2/92$

FIGURE 10. A snapshot of request list at the base station.

v/v_2-v_3-$v_4/95$	v_2/v_5-$v/94.5$	v_3/v_5-v_7-$v/94$	v_4/v_3-v_2-$v/93$	v_5/v_{20}-v_{11}-$v_2/92$

FIGURE 11. A snapshot of request list at the base station.

form v/v_2-v_3-$v_4/95$, v/v_5-$v_{20}/95$ and v/v_2-v_3-$v_4/95$ at the base station/(q^{th} base station) as shown in Fig. 8, Fig. 9 and Fig. 10 respectively.

ICGA at the base station/(q^{th} base station) arranges the request strings in the request list in the descending order of their overlapping coverage before starting the processing of the k^{th} request string. If two consecutive request strings in the list have same overlapping coverage then they are sorted in ascending order of the backup set size in the request string.

It is observed from Fig. 8 that the VSN v is present in the backup set of VSN v_2 and the overlapping coverage of the backup set of VSN v (=95%) is greater than the overlapping coverage of the backup set of VSN v_2 (=90% as the list is sorted). The base station/(q^{th} base station) sends reply (sleep) message to the VSN v so that the VSN v can shut itself off and updates the value of isVSNActive and isVSNDead Boolean variables in the record of VSN v in the base table to 0 and 0 respectively.

The sleep message consists of the id of the VSN v which is an integer number and hence the size of the sleep message (Size_id) is 2 bytes. As the VSN v goes into sleep mode, the backup set of VSN v_2 becomes invalid automatically and so the base station/(q^{th} base station) discards the request string of the VSN v_2 from the request list. The base station/(q^{th} base station) also removes the request string of the VSN v from the request list as its processing is over. The base station starts to process the request string of the VSN v_3.

It is observed from Fig. 9 that VSN v is present in the backup set of VSN v_{13}, the overlapping coverage of the backup set of VSN v is same as the overlapping coverage of the backup set of VSN v_{13} and the size of the backup set of VSN v is 2 whereas the size of the backup set of VSN v_{13} is 3.

The base station/(q^{th} base station) prefers the small-sized backup set and hence sends sleep message to the VSN v. The VSN v goes into sleep mode. The base station/(q^{th} base station) removes the request string of VSN v from the request list as its processing is over. As the VSN v goes into sleep mode the backup set of VSN v_{13} becomes invalid automatically and the base station/(q^{th} base station) discards the request string of VSN v_{13} from the request list. The base station starts to process the request string of the VSN v_3.

It has been assumed that the request message of VSN v will not be processed or the VSN v can't be shut off if VSN v is present in the backup set of more than three VSNs. It can be observed from Fig. 10 that the VSN v is present in the backup set of VSN v_2, v_3, v_4, v_5. Hence the base station/(q^{th} base station) removes the request string of VSN v from the request list without sending reply (sleep) message to it. The base station/(qth base station) starts to process the request string of v_2.

In other cases, if VSN v is present in the backup set of less than 4 number of VSNs, the base station/(q^{th} base station) sends sleep message to v and the VSN v goes into sleep mode. Fig. 11 expresses this scenario. VSN v is present in the backup set of v_2, v_3, v_4. In this case, the VSN v goes into sleep mode, the backup set of VSN v_2, v_3, v_4 becomes invalid automatically and so the base station/(q^{th} base station) discards the request string of the VSN v_2, v_3, v_4 from the request list. The base station/(q^{th} base station) also discards the request string of VSN v from the request list as its processing is over and starts to process the request string of v_5.

The base station/(q^{th} base station) stops execution of ICGA if the processing of request list is over. The duty cycling sub-phase ends here. Algorithm ICGA for the processing of the k^{th} request string is described below.

Post Scheduling Scenario:

A set of VSNs is in sleep mode in the target area when the base station/(all the four base stations) stop the execution of ICGA in APP_5/(APP_6). At this moment the base station/(q^{th} base station) calculates Tot_VSN, number of VSNs in inactive mode (say Υ) and Active_VSN (=(Tot_VSN - Υ)) for the target area from the base table. The base station/(q^{th} base station) also calculates the percentage of area coverage (Per_ACoV) by Active_VSN in the target area. The q^{th} base station in APP_6 measures these parameters for the target area by communicating with other base stations. If (Per_ACoV $< Th_{coverage}$, the threshold value of area coverage) the base station/(q^{th} base station) stops functioning of WVSN i.e. the base station stops collecting data from WVSN. Considering VSN v is in active mode and (Per_ACoV $\geq Th_{coverage}$) the VSN v starts monitoring the target area. Due to the constant dissipation of energy, it will die after a period of time. While its energy is reduced to the value which is a little more than zero, VSN v sends and routes (using GPSR) dead message to its neighbours and the base station/(q^{th} base station) respectively. The dead message consists of two parameters. The first parameter is the id of the VSN v of size 16 bits. The second parameter is a Boolean variable, isDead of size 1 bit. The value of the Boolean variable is set as true when the VSN v is dead, otherwise false. The size of the dead message consists of these two parameters

(Size_D) is (16 + 1) bits i.e. 17 bits. After getting the dead message from a VSN the base station/(q^{th} base station) searches the base table by the id of the VSN and updates the values, isVSNActive and isVSNDead to 0 and 1 respectively for the VSN v. Let a neighbour VSN of v (say v_1) in the set of Υ gets a dead message from the VSN v. Let the VSN v is also in the backup set of v_1. So, the backup set of v_1 becomes dead as the VSN v is dead and the VSN v_1 becomes awake. This phenomenon takes place for all dead VSNs and as a result, the sleeping VSNs belonging to Υ become awake. The base station/(q^{th} base station) now searches records of inactive VSNs (belonging to the set Υ) in the base table by observing the values of isVSNActive and isVSNDead set as 0 and 0 respectively. The base station/(q^{th} base station) updates the values of isVSNActive and isVSNDead of these VSNs to 1 and 0 respectively as they are active now.

When all the sleeping VSNs belonging to the set Υ become awake, the base station/(q^{th} base station) again computes Per_ACoV by (Tot_VSN - Active_VSN) number of VSNs. If Per_ACoV is less than $Th_{coverage}$ the base station will stop functioning of WVSN at this moment. Otherwise, Active_VSN will continue monitoring the target area until they die due to a shortage of energy.

VI. EXISTING WORKS EX_1, EX_2 AND EX_3 IN COMPARISON WITH APP_5 AND APP_6

In EX_1 duty cycling approach has been proposed where a mixture of large percentage (60%) of static all-time active VSN (ATVSN) and a small percentage (40%) of static active/inactive VSN (AIVSN) is deployed randomly in the target area. In all four approaches (APP_5, APP_6, EX_2, EX_3) each VSN/(in EX_1 each AIVSN) executes neighbour discovery phase, registration phase and scheduling phase sequentially. Each ATVSN in EX_1 executes only the neighbour discovery phase and the registration phase. In the scheduling phase of (APP_5, APP_6, EX_2/EX_1), each VSN/AIVSN has to undergo two sub-phases - backup set computation and duty cycling. In the scheduling phase of EX_3, each VSN has to undergo two sub-phases- redundancy judgment and duty-cycling. The neighbour discovery phase, registration phase and backup set computation sub-phase are same for EX_1, EX_2, APP_5 and APP_6 as in all the approaches same coverage model (3D directional coverage model) of VSN has been used. EX_1 and EX_2 differ with APP_5 and APP_6 in the duty cycling sub-phase. EX_3 differs from APP_5 and APP_6 in the registration phase and the scheduling phase while in the neighbor discovery phase, they are the same.

The duty cycling sub-phase of APP_5 and APP_6 is handled in a centralized manner whereas the duty cycling sub-phase of EX_1 and EX_2 is handled in a distributed manner. In the duty cycling sub-phase of EX_1 less number of activity messages of size (Size_AM), 17 bits are exchanged among AIVSNs. This reduces collision among messages in EX_1 and as a result, more VSNs go into sleep mode in EX_1 than in EX_2. Like base station(s) in APP_5 and APP_6, there exists a coordinator VSN in EX_1, EX_2 and EX_3. The coordinator VSN is a special kind of VSN in EX_1, EX_2 and EX_3 with a larger processing capability and storage space. The function of the coordinator VSN is same as the base station in all phases except at the duty cycling sub-phase. In the duty cycling sub-phase, the base station/(q^{th} base station) in APP_5/(APP_6) runs ICGA which decides whether VSN should go into sleep mode or stay awake in the target area/(q^{th} quadrant). On the other hand, in EX_1/EX_2 each AIVSN/VSN runs its own algorithm and decides independently whether to stay active or to go into sleep mode. Whenever an AIVSN/VSN in EX_1/EX_2 goes into sleep mode it sends an activity message to the coordinator VSN along with the neighbour VSNs to let the coordinator VSN know its status. The coordinator VSN updates the base table accordingly. At last in the post scheduling scenario when a VSN in both EX_1 and EX_2 goes out of energy, it sends a dead message to the coordinator VSN along with the neighbour VSNs, and the coordinator VSN updates its base table to reflect that the VSN is dead.

In EX_3 a hybrid (combination of distributed and centralized) duty-cycling mechanism based on grids is proposed. The entire target area is divided into equal size square grids. In EX_3, the target area of size (75m × 75m) is divided into 25 equal sizes square grids each of size (15m × 15m). In each grid, one of the VSNs is chosen as the group leader by the coordinator VSN. In [2] usually the sensor having the largest initial energy in a particular grid is selected as the group leader for that grid. But in the upgraded version of [2] i.e. in EX_3, VSN with the largest id in the grid is selected as a group leader as the initial energy of all the VSNs in EX_3 is considered as equal and sufficient enough to be a group leader. In the registration phase, each VSN in the target area sends its identification (id), position and orientation to the coordinator VSN. Simultaneously, each VSN in the q^{th} grid sends its identification, position, number of neighbors and residual energy to the group leader in the q^{th} grid. The coordinator VSN and the group leader of q^{th} grid insert such information in the base table and group leader table respectively. The size of each record in the group leader table (Size_Rec_GL) is $(2 \times 8 + 2 \times 4 \times 8 + 2 \times 8 + 4 * 8) =$ 128 bits i.e. 16 bytes. In the redundancy judgment sub-phase, each VSN calculates overlapping coverage by its neighbors as stated in [2]. If the overlapping coverage of the VSN is greater than and equal to Th_V, then the VSN is said to satisfy the condition of redundant coverage. In the duty-cycling sub-phase, the group leader in the q^{th} grid calculates a weight for each VSN including itself in the q^{th} grid based on the number of neighbor VSNs and residual energy of the VSN as described in equation 3.1 of [2]. Then the group leader in the q^{th} grid inserts the weight of each VSN (including itself) in the corresponding record of the VSN in the group leader table. It then sorts the records of the group leader table by the weights of VSNs in the descending order of their values. The VSN with the largest weight is the first to be judged its redundancy by the corresponding group leader.

If the weights of two VSNs are the same, the VSN with the larger id is judged for redundancy. After this, the VSN having the second-largest weight is judged in the same manner.The group leader in the q^{th} grid sends sleep messages to the two VSNs having the largest and second-largest weight in the same grid. The VSNs receive the sleep messages from the corresponding group leader and check whether they satisfy the condition of redundant coverage. If any one of them or both satisfy the condition of redundant coverage, the VSN or both the VSNs go into sleep mode and broadcast the SAM message [2] to their neighbors to indicate that the VSN(s) have gone into sleep mode. If the group leader in the q^{th} grid itself has the highest weight and also satisfies the condition of redundant coverage, then it goes into sleep mode, broadcast the SAM message to its neighbors and also sends a sleep message to the VSN having the second-largest weight. If a VSN including the group leader in the q^{th} grid doesn't meet the condition of redundant coverage then it stays awake.

VII. QUALITATIVE PERFORMANCE

The qualitative performance is evaluated in terms of communication overhead (COMM_OH), computation overhead (COMP_OH) and storage overhead (STO_OH) for the five schemes (APP_5, APP_6, EX_1, EX_2 and EX_3). In the worst case each VSN in the target area for APP_5, EX_1 and EX_2/(q^{th}) quadrant of the target area for APP_6)/q^{th} grid for EX_3 has (Tot_VSN-1) number of neighbour VSNs. The maximum number of elements in the backup set of a VSN/(an AIVSN) in APP_5, APP_6 and EX_2/(EX_1) is 3. The overheads are studied in the worst case for the target area/(per quadrant) by considering Size_RM and Size_RS as 80 bits and 112 bits respectively for APP_5 /(APP_6). The size of a SAM message [2] (Size_SAM) in EX_3 is 17 bits.

COMM_OH: It is the summation of the communication overhead in the neighbour discovery phase ($COMM\_OH_1$), registration phase ($COMM\_OH_2$), backup set computation sub-phase ($COMM\_OH_3$) and duty cycling sub-phase ($COMM\_OH_5$) for APP_5, APP_6, EX_1 and EX_2. It is the summation of $COMM\_OH_1$, $COMM\_OH_2$, communication overhead in redundancy judgment sub-phase ($COMM\_OH_4$) and $COMM\_OH_5$ for EX_3.

$COMM\_OH_1$: In APP_5, APP_6, EX_2, EX_3 each VSN sends a packet of size Size_Rec_NT bits whereas in EX_1 each VSN sends mustActive Boolean variable of size 1 bit along with the packet of size Size_Rec_NT bits to its (Tot_VSN-1) number of neighbours. Hence $COMM\_OH_1$ in EX_1 is ((Size_Rec_NT + 1) * Tot_VSN * (Tot_VSN-1)) bits whereas in each of (APP_5, APP_6, EX_2 and EX_3) the value is (Size_Rec_NT * Tot_VSN * (Tot_VSN-1)) bits.

$COMM\_OH_2$: In APP_5/(APP_6) each VSN in the target area/(q^{th} quadrant) sends a packet of size Size_Rec_NT bits to the base station/(q^{th} base station). In EX_2/EX_3 each VSN sends the packet of size Size_Rec_NT, whereas in EX_1 each VSN sends mustActive Boolean variable of size 1 bit along with the packet of size Size_Rec_NT bits to a coordinator VSN. In EX_3, each VSN in the q^{th} grid also sends a packet of size (Size_Rec_GL) bits to the group leader in the q^{th} grid. Hence $COMM\_OH_2$ in EX_1/(APP_5, APP_6, EX_2) is ((Size_Rec_NT + 1) * Tot_VSN)/(Size_Rec_NT * Tot_VSN) bits. Besides, $COMM\_OH_2$ in EX_3 is (Size_Rec_NT*Tot_VSN) + (Size_Rec_GL) * (Tot_VSN-1) bits.

$COMM\_OH_3$: In APP_5, APP_6, EX_1 and EX_2 no message communication takes place among VSNs and hence $COMM\_OH_3$ is 0 for all the four schemes.

$COMM\_OH_4$: In EX_3 no message communication takes place among VSNs and hence $COMM\_OH_4$ is 0 for EX_3.

$COMM\_OH_5$: In APP_5/(APP_6) each VSN sends request message of size Size_RM to the base station/(q^{th} base station). In response the base station/(q^{th} base station) sends sleep message of size Size_id. In EX_1/(EX_2) AIVSN/(VSN) sends the activity message of size Size_AM bits twice to its (Tot_VSN-1) number of neighbours. When an AIVSN/VSN goes into sleep mode it also sends an activity message to the coordinator VSN in EX_1/(EX_2). The group leader in the q^{th} grid in EX_3 sends two sleep messages to the VSNs having largest and second largest weight respectively and belonging to the same grid. The VSNs broadcast SAM message of Size_SAM to its neighbors if they satisfy the condition of redundant coverage. When a VSN goes into sleep mode it also sends a SAM message to the coordinator VSN in EX_3. Each VSN sends dead message of size Size_D to its (Tot_VSN-1) number of neighbours in APP_5, APP_6, EX_1, EX_2 and EX_3. Each VSN also sends dead message to the base station/(q^{th} base station) in APP_5/(APP_6) and to the coordinator VSN in EX_1, EX_2 and EX_3. Hence $COMM\_OH_5$ is (Size_RM) * (Tot_VSN) + (Size_id) * (Tot_VSN-1) + (Size_D) * Tot_VSN * (Tot_VSN-1) + (Size_D) * (Tot_VSN) for APP_5/(APP_6) and 2 * (Size_AM) * 0.4 * Tot_VSN * (Tot_VSN-1) + (Size_AM) * (0.4*Tot_VSN) + (Size_D) * Tot_VSN * (Tot_VSN-1) + (Size_D) * (Tot_VSN) / (2 * (Size_AM) * Tot_VSN * (Tot_VSN-1)) + (Size_AM) * (Tot_VSN-1) + (Size_D) * Tot_VSN * (Tot_VSN-1) + (Size_D) * (Tot_VSN)) bits for EX_1/(EX_2). $COMM\_OH_5$ is 2 * (Size_id) + (Size_SAM) * (Tot_VSN) * (Tot_VSN-1) + (Size_SAM) * (Tot_VSN-1) + (Size_D) * Tot_VSN* (Tot_VSN-1) + (Size_D) * (Tot_VSN) bits for EX_3.

STO_OH: It is the summation of the storage overhead in the neighbour discovery phase (STO_OH_1), registration phase (STO_OH_2), backup set computation sub-phase (STO_OH_3) and duty cycling sub-phase (STO_OH_5) for APP_5, APP_6, EX_1 and EX_2. It is the summation of STO_OH_1, STO_OH_2, storage overhead in redundancy judgment sub-phase (STO_OH_4) and STO_OH_5 for EX_3.

STO_OH_1: In EX_1/(APP_5, APP_6, EX_2, EX_3) each AIVSN/(VSN) stores Tot_VSN number of records each of size Size_Rec_NT + 1/(Size_Rec_NT) bits in its neighbour table. So STO_OH_1 in EX_1/(APP_5, APP_6, EX_2, EX_3) is (Size_Rec_ NT + 1) * 0.4 * Tot_VSN * Tot_VSN/(Size_Rec_NT * Tot_VSN * Tot_VSN) bits.

STO_OH_2: In APP_5/(APP_6) the base station/(q^{th} base station) stores (Tot_Param + 2) number of parameters of size (Size_Rec_NT + 2) bits in the base table. So STO_OH_2 in APP_5/(APP_6) is (Size_Rec_NT + 2) * Tot_VSN bits. In EX_1/(EX_2, EX_3) the coordinator VSN stores (Tot_Param + 3)/(Tot_Param + 2) number of parameters of size (Size_Rec_NT + 3)/(Size_Rec_NT + 2) bits in the base table. In EX_3 the group leader in the q^{th} grid stores four parameters of size 128 bits in the group leader table for each VSN. So STO_OH_2 in EX_1/(EX_2) is (Size_Rec_NT + 3) * Tot_VSN/(Size_Rec_NT + 2) * Tot_VSN bits. STO_OH_2 in EX_3 is (Size_Rec_NT + 2) * Tot_VSN + 128*Tot_VSN bits.

STO_OH_3: In EX_1/(APP_5, APP_6, EX_2) AIVSN v/(VSN v) stores all the backup sets in BS(v). Now BS(v) contains ${}^{Tot\_VSN-1}C_1$ number of sets of size 1, ${}^{Tot\_VSN-1}C_2$ number of sets of size 2 and ${}^{Tot\_VSN-1}C_3$ number of sets of size 3. ${}^{Tot\_VSN-1}C_1$, ${}^{Tot\_VSN-1}C_2$ and ${}^{Tot\_VSN-1}C_3$ number of sets contain χ (= ${}^{Tot\_VSN-1}C_1 + 2 * {}^{Tot\_VSN-1}C_2 + 3 * ({}^{Tot\_VSN-1}C_3)$ number of ids, each of size Size_id bits. So STO_OH_3 in EX_1/(APP_5, APP_6, EX_2) is Size_id * 0.4 * Tot_VSN * χ/(Size_id * Tot_VSN * χ) bits.

STO_OH_4: In EX_3 each VSN including the group leader in the q^{th} grid calculates the percentage of overlapping coverage by the neighbor VSNs and stores the overlapping coverage in a variable. Therefore, STO_OH_4 in EX_3 is 32*Tot_VSN bits.

STO_OH_5: In APP_5/(APP_6) the base station/(q^{th} base station) stores Tot_VSN number of request strings each of size Size_RS bits in the request list. So STO_OH_5 in APP_5/(APP_6) is Size_RS * Tot_VSN bits. In EX_1/(EX_2) each AIVSN/(VSN) stores the ids of 0.4 * Tot_VSN-1/(Tot_VSN-1) number of neighbour AIVSNs/(VSNs) in an Activity list. In EX_3 the group leader in the q^{th} grid calculates a weight for each VSN including itself in the grid and stores the weight (data type float) in the group leader table as the fifth parameter. So STO_OH_5 in EX_1/(EX_2) is Size_id * (0.4 * Tot_VSN-1) * 0.4 * Tot_VSN/(Size_id * Tot_VSN * (Tot_VSN-1)) bits and STO_OH_5 is 32*Tot_VSN bits in EX_3.

COMP_OH: It is the summation of the computation overhead in the neighbour discovery phase ($COMP\_OH_1$), registration phase ($COMP\_OH_2$), backup set computation sub-phase ($COMP\_OH_3$) and duty cycling sub-phase ($COMP\_OH_5$) for APP_5, APP_6, EX_1 and EX_2. It is the summation of $COMP\_OH_1$, $COMP\_OH_2$, computation overhead in redundancy judgment sub-phase ($COMP\_OH_4$) and $COMP\_OH_5$ for EX_3.

$COMP\_OH_1$: Each VSN inserts Tot_VSN number of records in its neighbour table in all the five schemes. In APP_5, APP_6, EX_2 and EX_3, each record contains Tot_Param number of parameters whereas in EX_1 each record contains mustActive Boolean variable of size 1 bit along with Tot_Param number of parameters. So $COMP\_OH_1$ in EX_1/(APP_5, APP_6, EX_2, EX_3) is O((Tot_Param + 1) * Tot_VSN) / (O(Tot_Param * Tot_VSN)) i.e. O(Tot_VSN).

$COMP\_OH_2$: The base station/(q^{th} base station) inserts Tot_VSN number of records in the base table in APP_5/(APP_6). The coordinator VSN also inserts Tot_VSN number of records in the base table in EX_1, EX_2 and EX_3. The group leader in the q^{th} grid inserts Tot_VSN number of records where each record consists of four parameters in the group leader table in EX_3. In APP_5, APP_6, EX_2 and EX_3 each record in the base table contains (Tot_Param + 2) number of parameters whereas in EX_1 each record contains mustActive Boolean variable of size 1 bit along with the (Tot_Param + 2) number of parameters. So $COMP\_OH_2$ in EX_1/(APP_5, APP_6, EX_2) is O((Tot_Param + 3) * Tot_VSN)/(O((Tot_Param + 2) * Tot_VSN)) i.e. O(Tot_VSN). $COMP\_OH_2$ in EX_3 is O((Tot_Param + 2) * Tot_VSN) + O(4 * Tot_VSN) i.e. O(Tot_VSN).

$COMP\_OH_3$: In EX_1/(APP_5, APP_6, EX_2) each AIVSN/(VSN) computes set of backup sets by computing (D_{12} X D_{34} X D_{99}). Each of the three sets has (Tot_VSN-1) number of ids of neighbour VSNs. So the computation overhead to compute (D_{12} X D_{34} X D_{99}) is O(Tot_VSN-1) * O(Tot_VSN-1) * O(Tot_VSN-1). Now the set of backup sets for each AIVSN/(VSN) in EX_1/(APP_5, APP_6, EX_2) contains (${}^{Tot\_VSN-1}C_1 + {}^{Tot\_VSN-1}C_2 + {}^{Tot\_VSN-1}C_1$) = (1/6) ($Tot\_VSN^3 + 5 * Tot\_VSN$) number of backup sets. The computation overhead to sort all these backup sets is $O((Tot\_VSN^3)^2)$. So $COMP\_OH_3$ for APP_5, APP_6, EX_1, EX_2 is ($O(Tot\_VSN^3) + O(Tot\_VSN^6)$) = $O(Tot\_VSN^6)$.

$COMP\_OH_4$: Each VSN in the q^{th} grid calculates overlapping coverage by its neighbor VSNs. Each VSN has (Tot_VSN-1) neighbor VSNs. The computation overhead to check whether a neighbor VSN occupies a portion of the FoV of the VSN is O(1). Therefore, the computation overhead to calculate the portion of the area covered by (Tot_VSN-1) neighbor VSNs is O(Tot_VSN).

$COMP\_OH_5$: In APP_5/(APP_6) the base station /(q^{th} base station) inserts Tot_VSN number of request strings in the request list. It calculates μ_{k-1}, var_{k-1}, σ_{k-1} and ($\mu_{k-1} + 3*\sigma_{k-1}$) with computation overhead of (O(1) + O(2) + ... + O(k-1)) i.e. $O(k^2)$ after inserting (k-1)th request string in the request list. The overhead to compute μ_{Tot_VSN-1}, var_{Tot_VSN-1}, σ_{Tot_VSN-1} and ($\mu_{Tot\_VSN-1} + 3 * \sigma_{Tot\_VSN-1}$) after the insertion of $(Tot\_VSN - 1)^{th}$ request string in the request list is $O(Tot\_VSN^2)$ and hence, the computation overhead to insert Tot_VSN number of request strings in the request list is $O(Tot\_VSN^2) * O(Tot\_VSN) = O(Tot\_VSN^3)$. ICGA runs bubble sort to sort Tot_VSN number of request strings with computation overhead $O(Tot\_VSN^2)$. ICGA reads the id of the requesting VSN from a request string and searches this id in the backup set of (Tot_VSN-1) number of request strings in the request list with computation overhead O(Tot_VSN-1) = O(Tot_VSN). Therefore, the computation overhead to process Tot_VSN number of request strings is O (Tot_VSN) * O(Tot_VSN) = $O(Tot\_VSN^2)$. So $COMP\_OH_5$

in APP_5/(APP_6) is (O(Tot_VSN^3) + O(Tot_VSN^2) + O(Tot_VSN^2)) = O(Tot_VSN^3).

In EX_1/(EX_2) the set of backup sets for each AIVSN/(VSN) contains ($^{Tot\_VSN-1}C_1$ + $^{Tot\_VSN-1}C_2$ + $^{Tot\_VSN-1}C_3$) number of sets sorted in the non-increasing order of overlapping coverage. The computation overhead to select the best backup set from these sets is O(1). In EX_1 each AIVSN reads the id of VSNs from its best backup set (maximum size 3) and determines the type of these VSNs by searching (Tot_VSN - 1) number of records in the neighbour table with computation overhead O(3 * Tot_VSN) i.e. O(Tot_VSN). If the type of these VSNs is AIVSN the Activity list consisting of (0.4 * Tot_VSN-1) number of neighbour AIVSNs is searched to know whether the ids of these AIVSNs are present in the Activity list or not with computation overhead O(Tot_VSN). In EX_2 each VSN reads the id of VSNs from its best backup set and searches the Activity list to know whether these ids are present in the Activity list or not with computation overhead O(Tot_VSN). The AIVSN/VSN v in EX_1/EX_2 tries with the next best backup set if the ids of these AIVSNs/VSNs in the current best backup set are not present in the Activity list. So $COMP\_OH_5$ in EX_1/(EX_2) is O(Tot_VSN^3) * {O(Tot_VSN) * O(Tot_VSN)}/O(Tot_VSN^3) * O(Tot_VSN) = O(Tot_VSN^5)/O(Tot_VSN^4).

In EX_3, the group leader in the qth grid reads the number of neighbor VSNs and the residual energy from each record in the group leader table to calculate a weight and to insert this weight as fifth attribute in the record. The computation overhead to read the number of neighbor VSNs and residual energy, to calculate weight and to insert weight in Tot_VSN number of records is O(1) * O(Tot_VSN) i.e. O(Tot_VSN). Lastly, the group leader sorts the group leader table by the value of weight parameter in the descending order with computation overhead of O(Tot_VSN^2).

COMP_OH in APP_5/(APP_6) is O(Tot_VSN) + O(Tot_VSN) + O(Tot_VSN^6) + O(Tot_VSN^3) = O(Tot_VSN^6). COMP_OH in EX_1 / EX_2 is O(Tot_VSN) + O(Tot_VSN) + O(Tot_VSN^6) + O(Tot_VSN^5)/O(Tot_VSN) + O(Tot_VSN) + O(Tot_VSN^6) + O(Tot_VSN^4) = O(Tot_VSN^6). COMP_OH in EX_3 is O(Tot_VSN) + O(Tot_VSN) + O(Tot_VSN) + O(Tot_VSN^2) = O(Tot_VSN^2).

Therefore, COMP_OH (computational overhead) is the same and O(Tot_VSN^6) for all the four schemes (APP_5, APP_6, EX_1 and EX_2). COMP_OH is least in EX_3.

COMM_OH and STO_OH for the five schemes are calculated and shown in TABLE 3 and TABLE 4 respectively when Tot_VSN is 70 and 100.

It is observed from TABLE 3 and TABLE 4 that COMM_OH is the same for APP_5 and APP_6 and highest in EX_2. COMM_OH of EX_1 is lesser than EX_2 and EX_3. COMM_OH is least in APP_5 and APP_6. It is also observed from TABLE 3 and TABLE 4 that STO_OH is the same for APP_5 and APP_6. STO_OH of EX_1 is lesser than that of APP_5, APP_6 and EX_2. STO_OH of APP_5 and APP_6 is slightly less than EX_2. STO_OH is the minimum in EX_3.

TABLE 3. Comparative study of COMM_OH and STO_OH among APP_5, APP_6, EX_1, EX_2 and EX_3 for Tot_VSN = 70.

	COMM_OH	STO_OH
APP_5	231.33 kilobytes	64.35 megabytes
APP_6	231.33 kilobytes	64.35 megabytes
EX_1	239.18 kilobytes	25.75 megabytes
EX_2	250.70 kilobytes	64.36 megabytes
EX_3	241.54 kilobytes	0.22 megabytes

TABLE 4. Comparative study of COMM_OH and STO_OH among APP_5, APP_6, EX_1, EX_2 and EX_3 for Tot_VSN = 100.

	COMM_OH	STO_OH
APP_5	471.62 kilobytes	92.05 megabytes
APP_6	471.62 kilobytes	92.05 megabytes
EX_1	488.18 kilobytes	36.83 megabytes
EX_2	511.73 kilobytes	92.07 megabytes
EX_3	492.58 kilobytes	0.45 megabytes

VIII. QUANTITATIVE PERFORMANCE

Both APP_5 and APP_6 are simulated using OMNET++ castalia simulator [30]. WVSN-v4 framework [31] is used to support the modelling of video sensor coverage. It contains a simulation model of WVSN. TABLE 5, TABLE 6, TABLE 7 and TABLE 8 summarize the basic simulation environment, a simulation environment for energy consumption, tunable MAC parameters and GPSR protocol parameters respectively for APP_5 and APP_6. The layout of VSN adopted in the present work is shown in Fig. 12.

TABLE 5. The basic experimental parameters for APP_5 and APP_6.

Parameter	Value
Target Area	75 m X 75 m
Number of VSNs	70-150
Deployment	Random
Horizontal Offset Angle α	36°
Vertical Offset Angle β	45°
Maximum Tilt Angle γ	65°
Height z=k_1	10 m
Sensing Range	23.8 m
Communication Range	30 m
Th_V	60% of the FoV area of VSN
$Th_{coverage}$	50% of the target area

TABLE 6. The experimental parameters for energy consumption in APP_5 and APP_6.

Parameter	Value
Initial Energy	50 J
BaselineNodePower	6 mW
Output Transmission Power	46.2 mW
MeasuredEnergyPerImageCapture	1 μJ
MeasuredEnergyPerImageProcessing	1 μJ
TimeForImageCapture	440 ms
TimeForImageProcessing	1512 ms

TABLE 7. Tunable MAC parameters for APP_5 and APP_6.

Parameter	Value
MACProtocolName	TunableMAC
DutyCycle	1 ms
ListenInterval	10 ms
RandomTxOffset	1
BackoffType	2

TABLE 8. GPSR protocol parameters for APP_5 and APP_6.

Parameter	Value
GPSRProtocolName	GPSR
HelloInterval	60000 ms
NetSetupTimeout	1000 ms

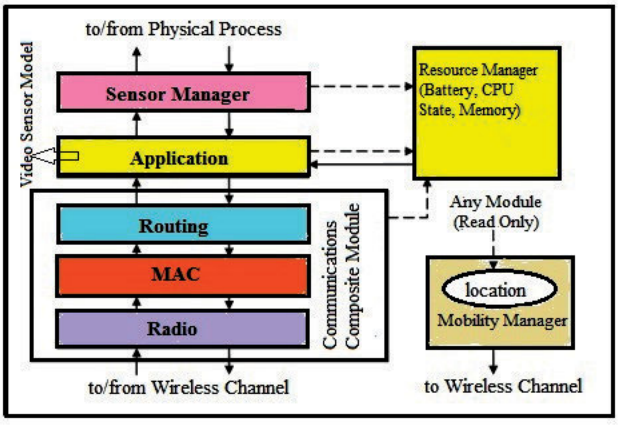


FIGURE 12. The layout of the VSN (source: [32]).

A. CONJECTURE

50% of total area coverage as well as 30 m. communication range in APP_5/(APP_6) ensures the connectivity of VSNs among themselves in APP_5/(APP_6).

Lemma 1 and Lemma 2 in [33] have proved that a set of Wireless Sensor Nodes (WSNs) in a finite area can be represented as a Poisson point process or simply Poisson process. Philips *et al.* [34] considered a geometric random graph model of a wireless packet radio network with a Poisson process of WSNs within a finite but large area of size A. It was also conjectured that for the communication range $R_c = \sqrt{((1+\epsilon)lnA/(\pi\lambda))}$ the network remains surely connected where λ is the number of WSNs (including the base station) per unit area and A is the size of the area.

It has been observed by extensive simulation that (i) in APP_5, minimum 25 VSNs each having 23.8 m sensing range need to stay active in the target area of size 75m $\times$ 75m to obtain 50% Per_ACoV of the target area and (ii) in APP_6 minimum 6 VSNs each having 23.8 m sensing range need to stay active in each quadrant of size 37.5m $\times$ 37.5m to get 50% Per_ACoV of the target area.

Considering the value of ϵ as 0.3, the communication range R_c is calculated both for APP_5 and APP_6. For APP_5, the minimum number of active VSNs is 26 (including base station) in the area of size 75 $\times$ 75 sq.m. and the value of λ is 26/(75 $\times$ 75) VSNs per sq. m. For APP_6, the minimum number of active VSNs is 7 (including the base station) in the area of size (37.5 $\times$ 37.5) sq.m. and the value of λ is 7/(37.5 $\times$ 37.5) VSNs per sq.m.

Putting the value of ϵ, A and λ in APP_5 and APP_6, the value of R_c is obtained as 27.78 m and 24.73 m respectively which ensures connectivity of the network.

In the proposed work, the value of $R_c = 30$ m, which obviously satisfies the connectivity among VSNs in the network both for APP_5 and APP_6 associated with 50% Per_ACoV.

B. SIMULATION METRIC

The quantitative performance of APP_5/(APP_6) is studied on the basis of Active_VSN, total energy consumption ($Energy_{Tot}$) in Joule, residual energy ($Energy_{Res}$) in Joule, network lifetime in seconds, Per_ACoV (in percentage) in the target area and Delay in seconds (delay in executing the neighbour discovery phase, registration phase and scheduling phase by all VSNs in the target area) versus node density and simulation time in seconds after random deployment of VSNs. Node density is the total number of nodes (Tot_VSN) already deployed in the target area. At any time t, $Energy_{Tot}$ is measured using equation 8 and $Energy_{Res}$ is the residual energy of Active_VSN and is measured as (Total Initial Energy of VSNs - $Energy_{Tot}$). Per_ACoV is measured using equation 9. Network lifetime is the time duration after which Per_ACoV falls below a predefined threshold value ($Th_{coverage}$) [35], [36] from the time of deployment of VSNs.

Ten simulation experiments are conducted for comparing the performance of APP_5 and APP_6 with EX_1, EX_2, EX_3 and Initial_Ran. Initial_Ran indicates the situation after the initial random deployment of VSNs when all VSNs are active. The first five simulation experiments have been conducted for the duration (0-700) s. The last five simulation experiments have been conducted for the duration (0-1500) s.

C. SIMULATION RESULT AND PERFORMANCE EVALUATION

In this subsection variation of Active_VSN, $Energy_{Tot}$, $Energy_{Res}$, Per_ACoV, Delay and network lifetime are

observed with increase in node density first and then the variation of Active_VSN, Per_ACoV, Energy$_{Tot}$, Energy$_{Res}$ are observed with increase in simulation time.

1) VARIATION OF ACTIVE_VSN WITH NODE DENSITY

Fig. 13 shows the plot of Active_VSN vs. node density for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3.

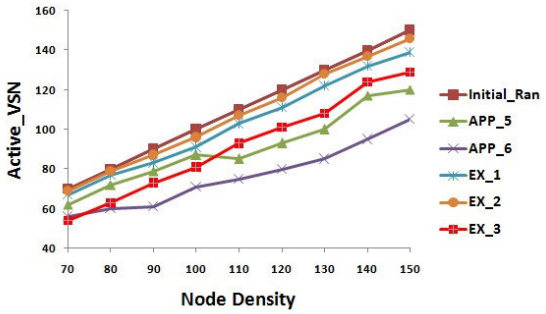


FIGURE 13. Active_VSN vs node density.

Observation From Fig. 13: Active_VSN increases with node density for all the six schemes (Initial_Ran, APP_5, APP_6, EX_1, EX_2, EX_3). Active_VSN is highest for Initial_Ran and least for APP_6. APP_5 produces a lesser value than that produced by both EX_1 and EX_2, whereas the value of Active_VSN in EX_1 is lesser than that of EX_2. Active_VSN for EX_3 is less compared to that for EX_1 and EX_2 but Active_VSN for EX_3 is always greater than that for APP_6. Again, Active_VSN for APP_5 is less compared to EX_3 when node density is greater than 100.

2) VARIATION OF ENERGY$_{Tot}$ WITH NODE DENSITY

Fig. 14 shows the variation of Energy$_{Tot}$ vs. node density for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3. Energy$_{Tot}$ in Fig. 14 is measured using basic simulation settings for the OMNET++ Castalia simulator and WVSN framework.

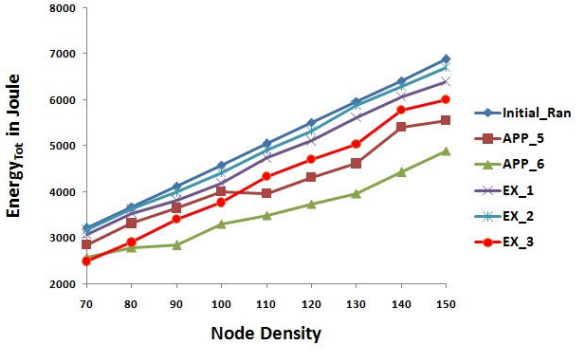


FIGURE 14. Energy$_{Tot}$ vs node density.

Observation From Fig. 14: Energy$_{Tot}$ increases with node density for all six schemes. Energy$_{Tot}$ is least in APP_6 and highest in Initial_Ran, and Energy$_{Tot}$ is lesser in APP_5 than both in EX_1 and EX_2. Energy$_{Tot}$ for EX_1 is lesser than for EX_2. Energy$_{Tot}$ is lesser in APP_5 than in EX_3 for the node density greater than 100 whereas Energy$_{Tot}$ in EX_3 is lesser than both in EX_1 and EX_2. The differences in the value of Active_VSN among the different schemes as observed from Fig. 13 causes such variation in Energy$_{Tot}$ among the six schemes.

3) VARIATION OF ENERGY$_{Res}$ WITH NODE DENSITY

Fig. 15 shows the plot of Energy$_{Res}$ vs. node density for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3.

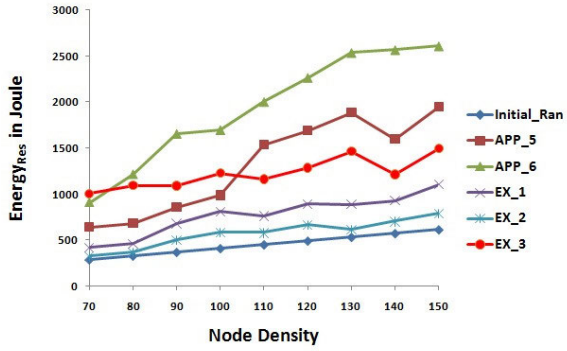


FIGURE 15. Energy$_{Res}$ vs node density.

Observation From Fig. 15: Energy$_{Res}$ increases with node density for all six schemes. Energy$_{Res}$ is least in Initial_Ran, highest in APP_6, less in EX_1 and EX_2 than in APP_5 and EX_3, less in EX_3 than in APP_5 for node density greater than 100 and less in EX_2 than in EX_1. The difference in Active_VSN as observed from Fig. 13 causes such variation in Energy$_{Res}$ among the six schemes.

4) VARIATION OF Per_ACoV WITH NODE DENSITY

Fig. 16 shows the plot of Per_ACoV vs. node density for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3.

Observation From Fig. 16: Per_ACoV increases with node density for all the six schemes. Per_ACoV is the highest for Initial_Ran as all VSNs in the target area are active and lowest in APP_6 due to the least value of Active_VSN associated with it. Per_ACoV is lesser in APP_5 than EX_1 and EX_2, lesser in APP_5 than EX_3 for node density greater than 100, lesser in EX_3 than EX_1 and EX_2, lesser in EX_1 than in EX_2 and also higher in APP_5 than in APP_6. The difference in Active_VSN as observed from Fig. 13 causes such variation in Per_ACoV among the six schemes.

5) VARIATION OF DELAY WITH NODE DENSITY

Fig. 17 shows the plot of Delay vs. node density for APP_5, APP_6, EX_1, EX_2 and EX_3.

Observation From Fig. 17: Delay is least in EX_1 and highest in APP_5, less in EX_2 than in APP_5 and APP_6, greater in EX_3 than in EX_1 and EX_2. This variation of Delay is explained in the experimental analysis [Section 10].

6) VARIATION OF NETWORK LIFETIME WITH NODE DENSITY

Fig. 18 shows the plot of network lifetime vs. node density for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3.

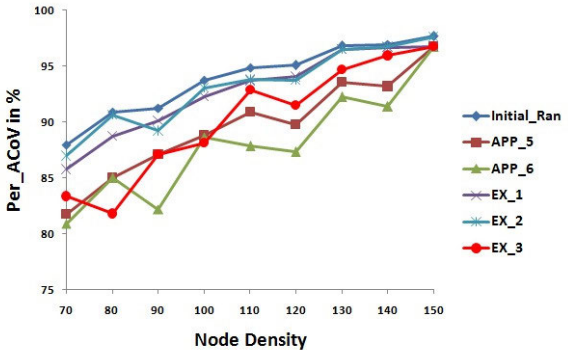


FIGURE 16. Per_ACoV vs node density.

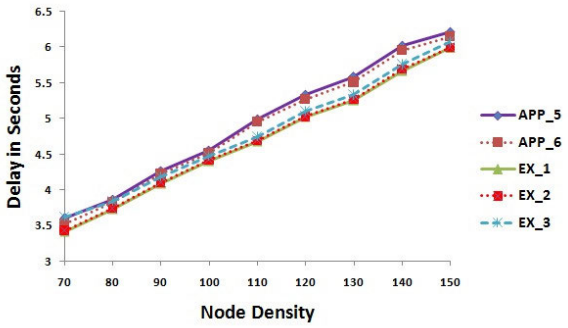


FIGURE 17. Delay vs node density.

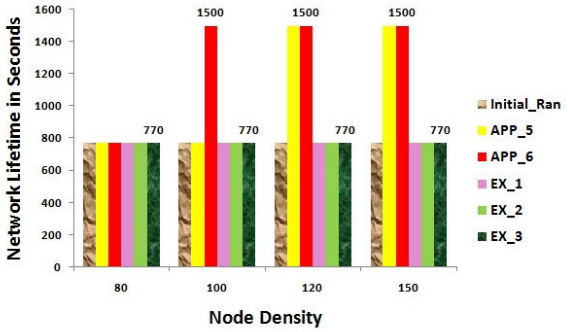


FIGURE 18. Network lifetime vs node density.

Observation From Fig. 18: Network lifetime is least in Initial_Ran, EX_1, EX_2 and EX_3 (770 s) for all node density and highest in APP_6 (1500 s) from node density 100. It is lesser in APP_5 than in APP_6 for all node densities except node density 120 and 150.

7) VARIATION OF ACTIVE_VSN WITH SIMULATION TIME

Fig. 19 and Fig. 20 show the plot of Active_VSN vs. simulation time for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3 when the node density is 120 and 150 respectively.

Observation From Fig. 19 and Fig. 20: Active_VSN is lowest in APP_6, maximum in Initial_Ran, less in APP_5, EX_3 than in both EX_1 and EX_2, less in APP_5 than EX_3 and less in EX_1 than in EX_2 during the simulation time 0 to 760 s when node density is 120 and 150 respectively.

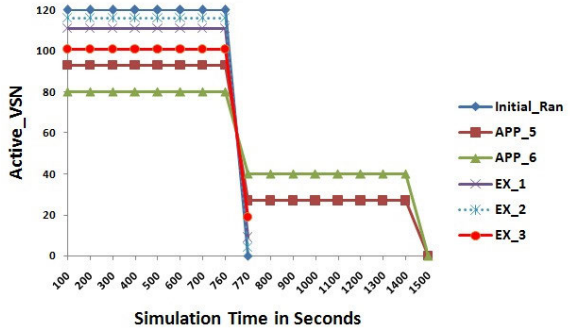


FIGURE 19. Active_VSN vs simulation time (node density = 120).

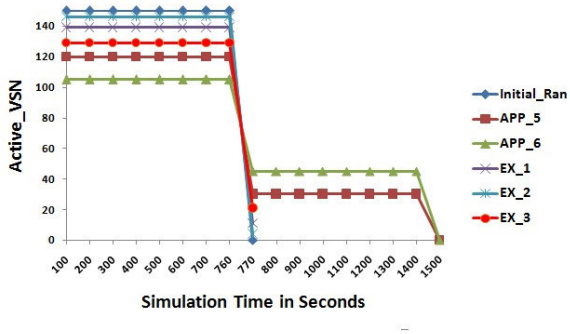


FIGURE 20. Active_VSN vs simulation time (node density = 150).

There is a sudden drop of Active_VSN at 770 s for both APP_5, APP_6 as observed from Fig. 19 and Fig. 20. This happens because the energy of all active VSNs gets exhausted at a time of 760 s. As soon as these active VSNs die, Active_VSN$_{new}$ (=Tot_VSN-Active_VSN) number of VSNs becomes active at that time which implies that the size of Active_VSN$_{new}$ is less than that of Active_VSN. From Fig. 19 and Fig. 20, it is also observed that Active_VSN$_{new}$ is highest in APP_6.

The base station(s) stops functioning of WVSN at simulation time 770 s for Initial_Ran, EX_1, EX_2 and EX_3 when node density is 120 (Fig. 19) and 150 (Fig. 20) respectively. The reason behind this is explored in our next investigation where the variation of Per_ACoV with simulation time has been observed. At simulation time 1500 s, no VSN is alive due to shortage of energy in both APP_5, APP_6 when node density is 120 (Fig. 19) and 150(Fig. 20) respectively.

8) VARIATION OF Per_ACoV WITH SIMULATION TIME

Fig. 21 and Fig. 22 show the plot of Per_ACoV vs. simulation time for Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3 when the node density is 120 and 150 respectively.

Observation From Fig. 21 and Fig. 22: Per_ACoV by Active_VSN is highest in Initial_Ran, lowest in APP_6, less in APP_5 than in EX_1, EX_2 and EX_3, less in EX_3 than in EX_1 and EX_2 and less in EX_1 than in EX_2 during the simulation time 0 to 760 s. It is also observed from Fig. 21 and Fig. 22 that Per_ACoV is highest in APP_6 when Active_VSN$_{new}$ becomes active. Per_ACoV

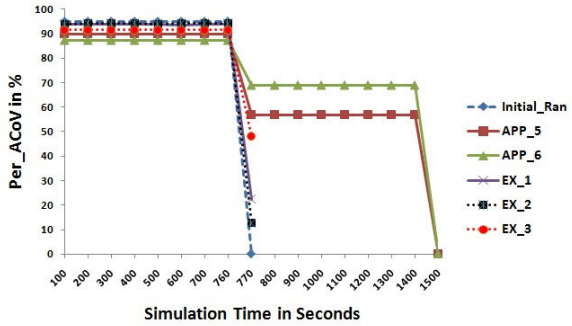


FIGURE 21. Per_ACoV vs simulation time (node density = 120).

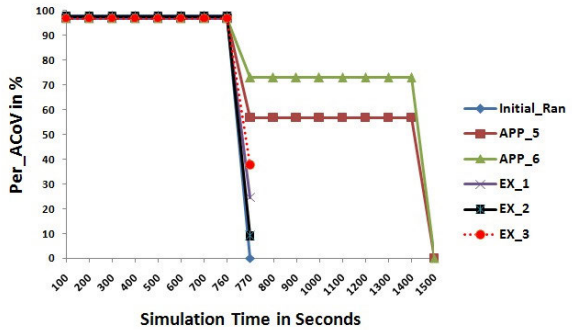


FIGURE 22. Per_ACoV vs simulation time (node density = 150).

by Active_VSN$_{new}$ becomes less than Th$_{coverage}$ at simulation time (t = 770 s) for Initial_Ran, EX_1, EX_2 and EX_3 when node density is 120 (Fig. 21) and 150 (Fig. 22) respectively. So, the network lifetime is 770 s for Initial_Ran, EX_1, EX_2 and EX_3 when node density is 120 (Fig. 21) and 150 (Fig. 22) respectively. A sudden drop in Per_ACoV is observed to happen at 770 s both for APP_5 and APP_6 in Fig. 21 and Fig. 22 which can be explained easily by using Fig. 19 and Fig. 20 respectively. At simulation time t = 1500 s, no VSN is alive and hence Per_ACoV becomes zero for both node densities. Therefore, network lifetime is (1500-0) s i.e. 1500 s for both APP_5 and APP_6 when node density is 120 (which is also visible in Fig. 19 and Fig. 20).

9) VARIATION OF ENERGY$_{Tot}$ WITH SIMULATION TIME

Fig. 23 and Fig. 24 show the plot of the variation of Energy$_{Tot}$ vs. simulation time while WVSN remains functioning for

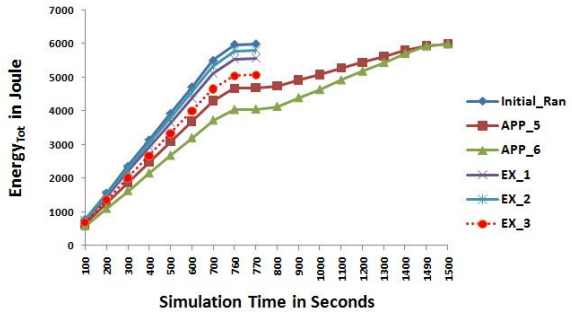


FIGURE 23. Energy$_{Tot}$ vs simulation time (node density = 120).

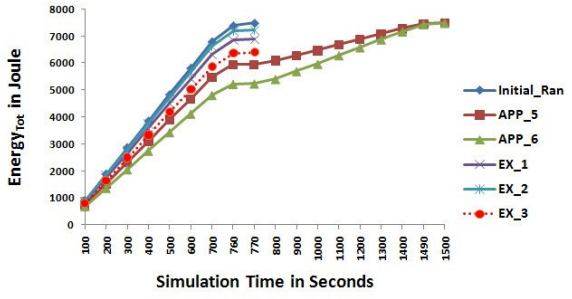


FIGURE 24. Energy$_{Tot}$ vs simulation time (node density = 150).

Initial_Ran, APP_5, APP_6, EX_1, EX_2 and EX_3 when the node density is 120 and 150 respectively.

Observation From Fig. 23 and Fig. 24: Energy$_{Tot}$ is highest in Initial_Ran, lowest in APP_6, less in APP_5 than EX_1, EX_2 and EX_3, less in EX_3 than EX_1, EX_2 and less in EX_1 than in EX_2 during the simulation time 0 to 760 s. At simulation time 770 s, WVSN stopped functioning for Initial_Ran, EX_1, EX_2 and EX_3 for node density 120 (Fig. 23) and 150 (Fig. 24) respectively since Per_ACoV becomes less than Th$_{coverage}$, as already observed in Fig. 21 and Fig. 22 respectively. At t = 1500 s all active VSNs die for both APP_5 and APP_6 at node density 120 (Fig. 23) and 150 (Fig. 24) respectively. It is also clear from Fig. 23 and Fig. 24 that Energy$_{Tot}$ in APP_6 is always less than that in APP_5 up to 1490 s respectively. At t = 1500 s, Energy$_{Tot}$ becomes identical for APP_5 and APP_6 (Fig. 23 and Fig. 24). The difference existing in Active_VSN as observed in Fig. 19 and Fig. 20 causes such variation of Energy$_{Tot}$. Energy$_{Tot}$ at any time t is calculated as the total energy consumed by Active_VSN from the time 0 to t. Energy$_{Tot}$ is not considered further for any scheme when WVSN stopped its functioning due to the absence of adequate value of Per_ACoV by Active_VSN.

10) VARIATION OF ENERGY$_{Res}$ WITH SIMULATION TIME

Fig. 25 and Fig. 26 show the plot for variation of Energy$_{Res}$ vs. simulation time for Initial_Ran, APP_5, APP_6, EX_1,

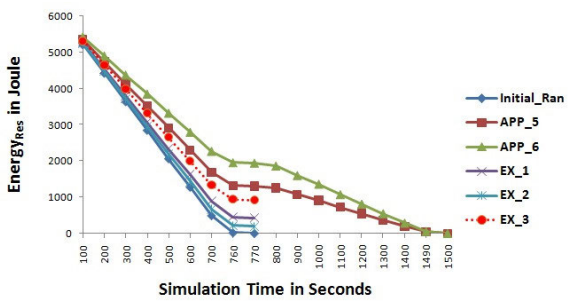


FIGURE 25. Energy$_{Res}$ vs simulation time (node density = 120).

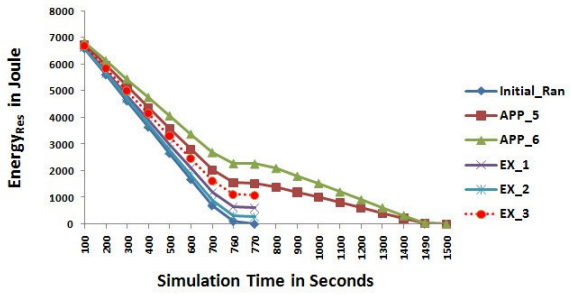


FIGURE 26. Energy$_{Res}$ vs simulation time (node density = 150).

EX_2 and EX_3 when the node density is 120 and 150 respectively.

Observation From Fig. 25 and Fig. 26: Energy$_{Res}$ is lowest in Initial_Ran, highest in APP_6, greater in APP_5 than EX_1, EX_2 and EX_3, greater in EX_3 than EX_1, EX_2 and greater in EX_1 than in EX_2 during the simulation time 0 to 760 s. From Fig. 25 and Fig. 26 it is also clear that Energy$_{Res}$ in APP_6 is greater than in APP_5 up to 1490 sec. At 1500 sec all active VSNs die both for APP_5 and APP_6 (Fig. 25, Fig. 26) and therefore, at that time Energy$_{Res}$ becomes zero for both the approaches (Fig. 25 and Fig. 26). The difference in Active_VSN as observed from Fig. 19 and Fig. 20 causes such variation of Energy$_{Res}$.

IX. PERFORMANCE OF APP_5 AND APP_6: OBSERVATION

It has been observed that EX_3 produces a better result (lesser) than that of (EX_1, EX_2) in terms of energy consumption. In terms of percentage of area coverage, EX_2 shows the best result among these three existing works. Between APP_5 and APP_6, Active_VSN is lesser in APP_6 compared to APP_5. APP_5 and APP_6 are able to reduce energy consumption by 6.98% and 18.6% respectively from the existing best approach EX_3 (in terms of energy consumption) for 150 deployed VSNs over the target area. Reducing the number of active VSNs helps decrease energy consumption at the expense of reduced area coverage. For the same node density, both APP_5 and APP_6 lose a little amount of area coverage (i.e. 0.93% and 0.95%) than the existing better approach EX_2 (in terms of percentage of area coverage). Both APP_5 and APP_6 have the same communication overhead. Both of them show better results by 3.19%/7.83%/(4.25%) from EX_1/EX_2/(EX_3) in terms of communication overhead for 100 deployed VSNs on the same target area. Finally, APP_6 clearly reveals its superiority in terms of reduced energy consumption (11.97%) than that of APP_5 while losing a very little percentage (0.02%) of area coverage.

X. EXPERIMENTAL ANALYSIS

In the duty cycling sub-phase, every VSN sends an activity message to its neighbour VSNs twice. The transmission and reception of such activity messages maintain no order. This results in a collision among messages and hence loss in messages. As a result, a lot of VSNs in the target area can't go into sleep mode despite having an eligible backup set. The communication overhead in the duty-cycling sub-phase is highest in EX_2 for all node densities. Therefore, message loss due to collision is highest in EX_2 and this also results in the highest Active_VSN (Fig. 13), Energy$_{Tot}$ (Fig. 14) and Per_ACoV (Fig. 16) and lowest Energy$_{Res}$ (Fig. 15) among all the five approaches (APP_5, APP_6, EX_1, EX_2 and EX_3). EX_2, EX_3 do not have any collision control mechanism whereas, in EX_1, collision among the activity messages have been controlled to some extent by generating a fewer number of activity messages.

In APP_5 and APP_6, the protocol tunable MAC employs CSMA/CA which helps reduce collision to a great extent among messages and hence loss in messages. As a result, the base station receives most of the request messages from VSNs and sends a sleep message for turning off a large number of VSNs having an eligible backup set. This results in a huge reduction of Active_VSN (Fig. 13) and Energy$_{Tot}$ (Fig. 14) resulting in a great enhancement of Energy$_{Res}$ (Fig. 15) and network lifetime (Fig. 18) both in APP_5 and APP_6 compared to EX_1 whereas EX_1 performs in the same manner in all those aspects with EX_2. In EX_3, the VSN with the largest weight and the VSN with the second-largest weight get a sleep message from the corresponding group leader in a particular grid. The VSNs broadcast SAM message to their corresponding neighbours after receiving sleep messages. Although sleep messages don't interfere with the other sleep messages across the grids in the target area, they interfere with the SAM messages. It results in a loss of sleep message. The message (sleep) loss due to collision increases as the number of VSNs increases in the target area. A lot of VSNs having the largest or second-largest weight in several grids don't get sleep messages from their corresponding group leader. As a result, those VSNs stay awake despite fulfilling the condition of redundant coverage. This increases Active_VSN (Fig. 13), Energy$_{Tot}$ (Fig. 14) and Per_ACoV (Fig. 16) and decreases Energy$_{Res}$ (Fig. 15) and network lifetime (Fig. 18) in EX_3 compared to APP_6 and APP_5 (for node density greater than 100). In APP_5, request messages from all the VSNs are handled by a single base station. Any message sending/receiving event with the base station happens in a multi-hop based path. Request messages from most of the VSNs may not reach the base station owing to traffic congestion as the number of routes is limited when node density is low. This increases Active_VSN (Fig. 13), Energy$_{Tot}$ (Fig. 14) and Per_ACoV (Fig. 16) and decreases Energy$_{Res}$ (Fig. 15) in APP_5 compared to EX_3 (for node density less than equal to 100). In EX_1, each AIVSN broadcasts activity messages to its neighbours. Unlike sleep messages, activity messages interfere with each other over the target area resulting in the loss of a lot of activity messages in comparison to the loss of sleep messages in EX_3. Besides, only AIVSNs (40% of Tot_VSN) participate in the duty cycling in EX_1 while the maximum of 50 VSNs (which is greater in

number than 40% of Tot_VSN up to node density 120) can be shut off in the target area divided into 25 grids in EX_3. These facts cause more reduction in Active_VSN (Fig. 13), $Energy_{Tot}$ (Fig. 14), Per_ACoV (Fig. 16) and increase in $Energy_{Res}$ (Fig. 15) in EX_3 than EX_1.

In APP_6, the VSN in the q^{th} quadrant sends the request message to the q^{th} base station where the length of the communication path from the VSN to the q^{th} base station is lesser than the length of the communication path from the VSN to the base station in APP_5. This further reduces message loss while transmitting in APP_6. More request messages in the q^{th} base station promote more VSNs to be shut off by the q^{th} base station. This reduces Active_VSN (Fig. 13) and hence $Energy_{Tot}$ by the Active_VSN (Fig. 14) in APP_6 than in APP_5. This also enhances $Energy_{Res}$ (Fig. 15) and network lifetime (Fig. 18) and reduces Per_ACoV (Fig. 16) in APP_6 more than APP_5.

Active_VSN for the approaches EX_1, EX_2, EX_3, APP_5, APP_6 die out of energy at the same time (770 s) for all node densities because they all have equal initial energy and have been deployed almost at the same time in the target area. Active_VSN for EX_1, EX_2 and EX_3 is very high and lower than Initial_Ran. As Active_VSN die at time 770 s, VSNs of size $Active\_VSN_{new}$ (=Tot_VSN-Active_VSN) wake up almost at the same time also. But, Per_ACoV by $Active\_VSN_{new}$ number of VSNs, (which is quite small enough in size in EX_1, EX_2 and EX_3) falls below the threshold value $Th_{coverage}$ at 770 s for all node densities. The network lifetime of both EX_1, EX_2 and EX_3 is equal to 770 s (same as for Initial_Ran) (Fig. 18). For smaller node density (<=100), Per_ACoV by $Active\_VSN_{new}$ for APP_5 also falls below the threshold value $Th_{coverage}$ at the same time and the network lifetime of APP_5 becomes 770 s (Fig. 18). As node density increases to 120, Per_ACoV value by $Active\_VSN_{new}$ for APP_5 becomes greater than $Th_{coverage}$ (Fig. 21 and Fig. 22) and $Active\_VSN_{new}$ number of VSNs remain active up to 1500 s (Fig. 19 and Fig. 20). The value of Per_ACoV by $Active\_VSN_{new}$ is higher in APP_6 compared to in APP_5 (Fig. 21 and Fig. 22) since a higher number of VSNs are active in APP_6 than in APP_5 (Fig. 19 and Fig. 20). $Active\_VSN_{new}$ both in APP_5 and APP_6 die at the same time 1500 s because they all have equal initial energy and are awake at the same time $t = 770$ s (Fig. 19 and Fig. 20).

Both in APP_5 and APP_6, VSNs send the request message to the base station through a multihop path in WVSN. Each VSN uses GPSR routing protocol with tunableMAC while routing request messages. But, (i) in EX_1 and EX_2, any message sending/receiving event is of a single hop in nature and the processing of messages for each VSN is based on some stored information in the VSN itself (local information). (ii) In EX_3, any message sending/receiving event among VSNs and between the group leader and the VSNs in a particular grid is single hop in nature. This reduces delay in EX_1, EX_2 and EX_3 than APP_5 and APP_6 as observed from Fig. 17. As four base stations work simultaneously in APP_6 and any communication from a VSN to the corresponding base station requires fewer hops than in APP_5, the delay incurred in APP_6 is less than that in APP_5 (Fig. 17). Delay in EX_3 is slightly larger than EX_1 and EX_2 as in the registration phase of EX_3, there is an additional communication between VSNs of each grid with the group leader of that grid (Fig. 17).

XI. CONCLUSION

Two advanced approaches, APP_5 and APP_6 have been proposed in this paper to reduce energy consumption by reducing the number of active VSNs while maintaining the area coverage above some threshold level so as to ensure network connectivity in a disaster-hit area with randomly deployed VSNs. Message loss due to collision in the duty cycling sub-phase within the target area keeps more number of VSNs awake. This collision among messages in APP_5 and APP_6 has been reduced by using tunable MAC protocol which employs carrier sense multiple accesses/collision avoidance techniques while routing messages to the base station. APP_5 and APP_6 are compared with three existing approaches EX_1, EX_2 and EX_3 which show clear superiority to them. APP_6 is most efficient in terms of energy consumption and network lifetime.

In future, a new methodology will be developed to reduce energy consumption and to increase the network lifetime of the WVSN in presence of 3D obstacles.

REFERENCES

[1] M. W. P. Maduranga, P. Saengudomlert, and H. M. N. D. Bandara, "Redundant node management in wireless sensor networks with multiple sensor types," in *Proc. Nat. Inf. Technol. Conf. (NITC)*, Oct. 2018, pp. 1–6, doi: 10.1109/NITC.2018.8550076.

[2] K. Zhang, J. Chen, C. Shen, Y. Chen, K. Long, and B. U. Aslam, "Node scheduling algorithm based on grid for wireless sensor networks," in *Proc. IEEE 19th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2019, pp. 987–990, doi: 10.1109/ICCT46805.2019.8947095.

[3] B. Shi, W. Wei, Y. Wang, and W. Shu, "A novel energy efficient topology control scheme based on a coverage-preserving and sleep scheduling model for sensor networks," *Sensors*, vol. 16, no. 10, p. 1702, Oct. 2016.

[4] G. Saad, H. Harb, C. A. Jaoude, and A. Jaber, "Correlation-based sensor activity scheduling mechanisms for wireless sensor networks," in *Proc. IEEE/ACS 16th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2019, pp. 1–8, doi: 10.1109/AICCSA47632.2019.9035254.

[5] Y. Tian, X. Yang, Y. Ou, and G. You, "An area coverage control protocol based on probabilistic coverage for wireless sensor networks," in *Proc. 7th Int. Conf. Intell. Control Inf. Process. (ICICIP)*, Dec. 2016, pp. 22–27, doi: 10.1109/ICICIP.2016.7885910.

[6] A. Vishwanath, R. Datta, and N. Marchang, "Probabilistic node scheduling in dense wireless sensor networks," in *Proc. IEEE Region Conf. (TENCON)*, Oct. 2019, pp. 1169–1173, doi: 10.1109/TENCON.2019.8929407.

[7] K. Bairagi, S. Mitra, and U. Bhattacharya, "Coverage aware dynamic scheduling strategies for wireless video sensor nodes to reduce energy consumption," in *Proc. 11th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Kharagpur, India, Jul. 2020, pp. 1–7, doi: 10.1109/ICCCNT49239.2020.9225516.

[8] K. Bairagi and U. Bhattacharya, "Resource constrained coverage model of a video sensor node to reduce energy consumption," in *Proc. IEEE Int. Conf. Electr., Comput. Commun. Technol. (ICECCT)*, Coimbatore, India, Feb. 2019, pp. 1–7, doi: 10.1109/ICECCT.2019.8869026.

[9] N. Bendimerad and B. Kechar, "Rotational wireless video sensor networks with obstacle avoidance capability for improving disaster area coverage," *J. Inf. Process. Syst.*, vol. 11, no. 4, pp. 509–527, 2015, doi: 10.3745/JIPS.03.0034.

[10] N. Bendimerad and B. Kechar, "Coverage enhancement with rotatable sensors in wireless video sensor networks for post-disaster management," in *Proc. 1st Int. Conf. Inf. Commun. Technol. Disaster Manage. (ICT-DM)*, Algiers, Algeria, Mar. 2014, pp. 1–7, doi: 10.1109/ICT-DM.2014.6918584.

[11] A. Salim, W. Osamy, and A. M. Khedr, "Effective scheduling strategy in wireless multimedia sensor networks for critical surveillance applications," *Appl. Math. Inf. Sci.*, vol. 12, no. 1, pp. 101–111, Jan. 2018.

[12] A. Makhoul and C. Pham, "Dynamic scheduling of cover-sets in randomly deployed wireless video sensor networks for surveillance applications," in *Proc. 2nd IFIP Wireless Days*, 2009, pp. 1–6, doi: 10.1109/WD.2009.5449700.

[13] V. Ukani, K. Patel, and T. Zaveri, "Computation of coverage backup set for wireless video sensor network," in *Proc. IEEE Region Symp.*, Ahmedabad, India, May 2015, pp. 37–40, doi: 10.1109/TENSYMP.2015.23.

[14] A. Benzerbadj and B. Kechar, "Redundancy and criticality based scheduling in wireless video sensor networks for monitoring critical areas," *Proc. Comput. Sci.*, vol. 21, pp. 234–241, Jan. 2013.

[15] V. Ukani, K. Patel, and T. Zaveri, "A realistic coverage model with backup set computation for wireless video sensor network," *NUJET*, vol. 3, pp. 1–5, 2015.

[16] P. Wang, R. Dai, and I. F. Akyildiz, "A differential coding-based scheduling framework for wireless multimedia sensor networks," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 684–697, Apr. 2013, doi: 10.1109/TMM.2012.2236304.

[17] C. Yu and G. Sharma, "Camera scheduling and energy allocation for lifetime maximization in user-centric visual sensor networks," *IEEE Trans. Image Process.*, vol. 19, no. 8, pp. 2042–2055, Aug. 2010, doi: 10.1109/TIP.2010.2046794.

[18] M. A. Jamshed, M. F. Khan, K. Rafique, M. I. Khan, K. Faheem, S. M. Shah, and A. Rahim, "An energy efficient priority based wireless multimedia sensor node dynamic scheduler," in *Proc. 12th Int. Conf. High-Capacity Opt. Netw. Enabling/Emerg. Technol. (HONET)*, Islamabad, Pakistan, Dec. 2015, pp. 1–4, doi: 10.1109/HONET.2015.7395435.

[19] V. D. Bhosale and R. A. Satao, "Lifetime maximization in mobile visual sensor network by priority assignment," in *Proc. Int. Conf. Control, Instrum., Commun. Comput. Technol. (ICCICCT)*, Kumaracoil, India, Dec. 2015, pp. 695–698, doi: 10.1109/ICCICCT.2015.7475368.

[20] K. Khursheed, M. Imran, M. O'Nils, and N. Lawal, "Exploration of local and central processing for a wireless camera based sensor node," in *Proc. Int. Conf. Signals Electron. Circuits (ICSES)*, Gliwice, Poland, 2010, pp. 147–150.

[21] J.-W. Ahn, T.-W. Chang, S.-H. Lee, and Y. W. Seo, "Two-phase algorithm for optimal camera placement," *Sci. Program.*, vol. 2016, Sep. 2016, Art. no. 4801784, doi: 10.1155/2016/4801784.

[22] Y.-G. Fu, J. Zhou, and L. Deng, "Surveillance of a 2D plane area with 3D deployed cameras," *Sensors*, vol. 14, no. 2, pp. 1988–2011, Jan. 2014, doi: 10.3390/s140201988.

[23] L. Zhang, T.-T. Liu, F.-Q. Wen, L. Hu, C. Hei, and K. Wang, "Differential evolution based regional coverage-enhancing algorithm for directional 3D wireless sensor networks," *IEEE Access*, vol. 7, pp. 93690–93700, 2019, doi: 10.1109/ACCESS.2019.2927805.

[24] Z. Jiao, L. Zhang, M. Xu, C. Cai, and J. Xiong, "Coverage control algorithm-based adaptive particle swarm optimization and node sleeping in wireless multimedia sensor networks," *IEEE Access*, vol. 7, pp. 170096–170105, 2019, doi: 10.1109/ACCESS.2019.2954356.

[25] C. Han, L. Sun, F. Xiao, J. Guo, and R. Wang, "A camera nodes correlation model based on 3D sensing in wireless multimedia sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 11, Nov. 2012, Art. no. 154602.

[26] J. Peng, J. Jingqi, W. Chengdong, and H. Nan, "A coverage detection and re-deployment algorithm in 3D directional sensor networks," in *Proc. 27th Chin. Control Decis. Conf. (CCDC)*, Qingdao, China, May 2015, pp. 1137–1142, doi: 10.1109/CCDC.2015.7162088.

[27] T. Nieberg, "Independent and dominating sets in wireless communication graphs," Ph.D. dissertation, Univ. Twente, Enschede, The Netherlands, 2006.

[28] *Greedy Perimeter Stateless Routing (GPSR)*. Accessed: Jul. 18, 2020. [Online]. Available: https://www.icir.org/bkarp/gpsr/gpsr.html

[29] *68-95-99.7 Rule (Empirical Rule)*. Accessed: Jul. 25, 2020. [Online]. Available: http://en.wikipedia.org/wiki/68-95-99.7_rule

[30] C. Pham. *A Video Sensor Simulation Model With OMNET++, Castalia Extension*. Accessed: Sep. 25, 2016. [Online]. Available: http://cpham.perso.univ-pau.fr/WSN-MODEL/wvsn-castalia.html

[31] C. Pham. *A Video Sensor Simulation Model With OMNET++*. Accessed: Sep. 25, 2016. [Online]. Available: http://cpham.perso.univ-pau.fr/WSN-MODEL/wvsn.html

[32] C. Pham. *Videosense: A Simulation Model of Image Sensors Under OMNET++/Castalia*. Accessed: Jun. 28, 2021. [Online]. Available: http://cpham.perso.univ-pau.fr/Paper/Talk-RESSACS2012.pdf

[33] W. Mo, D. Qiao, and Z. Wang, "Mostly-sleeping wireless sensor networks: Connectivity, k-coverage, and α-lifetime," in *Proc. 43rd Annu. Allerton Conf. Commun., Control Comput.*, 2005, pp. 1–10.

[34] T. K. Philips, S. S. Panwar, and A. N. Tantawi, "Connectivity properties of a packet radio network model," *IEEE Trans. Inf. Theory*, vol. 35, no. 5, pp. 1044–1047, Sep. 1989, doi: 10.1109/18.42219.

[35] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. H. Hanzo, "A survey of network lifetime maximization techniques in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 828–854, 2nd Quart., 2017, doi: 10.1109/COMST.2017.2650979.

[36] S. Mini, S. K. Udgata, and S. L. Sabat, "M-connected coverage problem in wireless sensor networks," *ISRN Sensor Netw.*, vol. 2012, Mar. 2012, Art. no. 858021, doi: 10.5402/2012/858021.

KISHALAY BAIRAGI (Graduate Student Member, IEEE) received the M.Tech. degree from the Department of Computer Science, RCC Institute of Information Technology, India, in 2014. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology (IIEST), Shibpur, Howrah, India. His research interests include pedagogical framework for distributed computing systems and scheduling strategies in wireless sensor networks.

SULATA MITRA received the B.E. degree from Bengal Engineering College, India, in 1986, and the Ph.D. degree in mobile computing from Bengal Engineering College (D.U), Shibpur, India, in 2005. She joined the Indian Institute of Technology, Kharagpur, in 1989, as a Senior Research Assistant, and moved to the Regional Institute of Technology, Jamshedpur, India, in 1991, as a Lecturer. She is currently with the Computer Science and Technology Department, Indian Institute of Engineering Science and Technology, Shibpur, as a Professor. She has published 50 technical articles in journals and international conference proceedings. Her current research interests include QoS issues in 3G/4G cellular networks, VANET, and multihomed mobile networks.

UMA BHATTACHARYA received the Ph.D. degree in computer science from the University of Calcutta, India, in 1995. She was with the University of Windsor, Canada, as a Postdoctoral Fellow, in 1998. She also spent six months at Sheffield Hallam University, U.K., as a Commonwealth Fellow, from 2002 to 2003. Being U.K.–India Networking Awardee, she was at Northumbria University, U.K., in 2005, funded by the Royal Society, U.K., and the Department of Science and Technology (DST), India. She is currently working as a Professor with the Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur (formerly Bengal Engineering and Science University, Shibpur), India. Her research interests include optical networks, interconnection networks, and fault–tolerance.