DF-SWin: Sliding Windows for Multi-Sensor Data Fusion in Wireless Sensor Networks

Huy. Duong-Viet
Ph.D student of Data Communication and Computer Networks
University of Engineering and Technology, VNU, Hanoi
Hanoi, Vietnam
huy.duongviet@gmail.com

Viet. Nguyen-Dinh
University of Engineering and Technology
Vietnam National University, Hanoi
Hanoi, Vietnam
vietnd@vnu.edu.vn

***Abstract*— When using multiple sensor nodes in wireless sensor networks (WSNs) for monitoring (or measuring) parameters of the target and sending the result to the base station (BS), data redundancy is an inevitable problem. The measured data often contains the same information, and sending redundant data to BS causes the waste of energy of sensor nodes and the risk of congestion. Multi-sensor data fusion in WSNs is a technology of gathering and processing data applied from node to BS. It improves the performance of surveillance systems by allowing the obtained sensed information from multiple sensor nodes aggregated to one unified format data packet to send to BS to make decision. In this paper, we propose a solution namely DF-SWin for sampling sensor node in the cluster to optimize the energy conservation of sensor nodes in clusters and cluster head.**

***Keywords*— DF-SWin, sliding window, data fusion, WSNs**

I. INTRODUCTION

The WSNs typically, sensor nodes are powered by batteries with limited capacity. Sensor nodes in WSNs for monitoring parameters of the target and sending the result to the BS. When using multiple sensor nodes for monitoring a target, sending the same of information about a measure parameter to destination will cause two problems: redundancy of data and the risk of network congestion. Transmit packets with the same information will cause useless wasted residual energy of sensor node. Thus, data fusion (DF) multi-sensor is implemented on the routing transmission from the first sensor node (directly tracks the target) to the BS the solution must be implemented. There are many studies in DF such as [1, 2, 3, 4, 5]. Most of DF studies have applied to WSNs with clustering, each cluster has a sensor node selected as the cluster head (CH), it processes data from all member nodes in this cluster and sends data fusion result to BS.

In this paper, we propose a solution namely DF-SWin *(Data Fusion Sliding Window)* is setup to CH for using the sliding window to sampling some sensor node in the cluster based on the semantics properties of the sensor node (such as distances between sensor nodes and CH, residual energy, residual of packets to transmit, etc.). based on the semantic conclusion, CH will select the sensor node and using the value of measurement parameters for the data. Instead of data fusion all the data from the nodes in the cluster, CH only data fusion from some nodes that satisfy the condition.

The paper consists of three main parts: Some concepts for systematizing semantic properties data of sensor nodes into full data table; proposed solution DF-SWin method uses a sliding window solution to sampling sensor nodes according to the actual semantic data at the time of review; presents analysis of efficiency and simulation evaluation of the proposed solution using NS-2 network simulator.

II. CONCEPTS

A. Sensing data system

At the time DF, the semantic properties data of *n* sensor nodes in the cluster can be considered as an information system of the cluster, denoted by the IS represented by *4* elements [6]:

$$IS = \langle U, Q, V, f \rangle \quad (1)$$

Where, *U* is a finite set of sensor node; *Q* is a finite set of semantic properties; *V* is a finite set of semantic properties values; *f* is the value of a semantic properties corresponding a sensor node. At the time for review, *IS* is a table with *n* rows (number of sensor node) and *m* columns (number of semantic properties) as shown in Table 1.

TABLE 1. SENSING SYSTEM AT THE TIME DATA FUSION

U	Set of semantic properties (Q)			
	A_1	A_2	......	A_m
S_1	$V_{A1.S1}$	$V_{A2.S1}$	......	$V_{Am.S1}$
S_2	$V_{A1.S2}$	$V_{A2.S2}$	......	$V_{Am.S2}$
S_2	$V_{A1.S2}$	$V_{A2.S2}$	......	$V_{Am.S2}$
....	......	......	......	......
S_n	$V_{A1.Sn}$	$V_{A2.Sn}$	......	$V_{Am.Sn}$

TABLE 2. NUMBER OF LEVEL AND VALUE OF PROPERTIES

V	Set of semantic properties (Q)			
	A_1	A_2	......	A_m
X_1	$X_{1.A1}$	$X_{1.A2}$	......	$X_{1.Am}$
X_2	$X_{2.A1}$	$X_{2.A2}$	......	$X_{2.Am}$
....	......	......	......	......
	$X_{l.A1}$	......	......	......
X_l		......	......	$X_{l.Am}$
		$X_{l.A2}$	......	

Set $f(S_i, A_j)$ caller value of sensor node S_i at semantic properties A_j $(1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m)$, $f(S_i, A_j) = V_{Si.Aj}$. With each semantic properties, number of level and value of l be different as shown in Table 2 depends on the quantification of method to ensure smooth and asymptotic with the measurement levels integrated into the sensor node by the manufacturer.

B. Condition properties

Sensor nodes are characterized by a set of values of their respective properties and used as a condition to establish a decision about that sensor node, which may be: *properties of sensing data of sensor node* such as the number of packets to transmit to complete the sensory information, signal strength, noise, etc.; *properties of sensor node* such as residual energy, distances between sensor nodes and CH (or neighboring node) slot time TDMA index, etc.

Dependence of condition properties [6]: *IS* with $Q=C\cup D$ if and only if D depends on C, denoted by $C \rightarrow D$. D depends on C in a degree k $(0 \leq k \leq 1)$, denoted by $k = \gamma(C, D)$ (2)

If $k = 1$ we say that D *depends totally on* C, and if $k < 1$, we say that *D depends partially in a degree k on C.*

C. Importance of properties

Information Systems $IS = <U, C\cup D, V, f>$, if C is a condition properties, D is a decision properties and $C \rightarrow D$, importance of a properties (for example A) of *IS* is evaluated by the coefficient effect of removing $A \in C$ in the positive region of D, can be defined as follows [6]:

$$\mu_{(C,\ D)}(A) = \frac{(\gamma(C, D) - \gamma(C - \{A\}, D))}{\gamma(C, D)}$$

$$= 1 - \frac{\gamma(C - \{A\}, D)}{\gamma(C, D)} \qquad (3)$$

$0 \leq \mu_{(C,\ D)}(A) \leq 1$, if $\mu_{(C,\ D)}(A) = 1$ then we say that A is a very important properties for D. For example, residual energy, distances between sensor nodes and CH are important properties but not very important, mean $\mu_{(C,\ D)}(A) < 1$.

D. Measurement parameters

Measurement parameter of sensor node, denoted by P_{mes} is a finite set of measurement parameter, for example: Temperature, humidity, wind speed, etc.

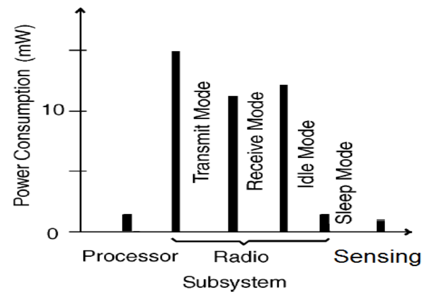
Sensing value of cluster: Set N_S is the number of sensor nodes in cluster, the data table consists of N_S row and P_{mes} column called measurement parameter value of cluster. This data table is used by the CH to data fusion and sent to BS.

E. Using and saving energy

According to the statistics [7], the energy consumed by transceiver radio signals is many times greater than the energy consumes to process other tasks of sensor node, including the calculation on the sensor node. The Figure 1 is the chart of energy consumption when the sensor node is operating.

The relationship between energy consumption E_{TX} when sending *k bit* with distance d and E_{RX} when receiving *k bit* have been explained by [8]: $E_{TX} = E_{elec}*k+E_{amp}*d^2$ and $E_{RX}=E_{elec}* k$, where E_{elec} is energy consumption of sensor node to send or receive *1 bit*, E_{amp} is energy consumption to send *1 bit/m²* by radio signals of sensor node.

Figure 1. Rate of energy consumption during sensor operation [7]



Thus, the energy of the electromagnetic waves transmitted from the sensor node data will decrease exponentially with the distance between sender node and receiver node. To ensure the packet to its destination, the sensor node must manually adjust (amplifier) power of transmitter with the square of the distance [8]. For this reason, research groups focus to reduce the amount of data sent from sensor node.

F. Sliding window

The sliding window is a matrix that is limited by two dimensions: The column is the number of condition properties (as defined in part II, A, B), the row is the number of sensor node selected by algorithm.

Sliding window size: Set H is the height of the window and N_S is the number of nodes of cluster for selecting window, W is width of the window and Q_A is number of condition properties. Thus, sliding window size is a set of (H, W), where $H \leq N_S$, $W \leq Q_A$. (4)

At any time the window is used, size of the window may vary depending on the status of the number of nodes in cluster, the residual energy of node, the distance from nodes in the cluster to CH, residual of packets of nodes to transmit, etc.

III. DF-SWIN SOLUTION

A. Sliding Window DF-SWin

The sliding window is used to select sensor nodes based on conditional properties. The Importance of conditional properties is define as (3), determined only once during the DF-SWin algorithm, fixed to the table and sort by decreasing level of importance. For DF-SWin, we select three conditional properties and sort as follows: *Distances* between sensor nodes and CH, *residual energy* of node, *residual packets* to transmit to complete the sensory. DF-SWin applies to cases of $N_S \geq 3$ sensors, $Q_A \geq 2$ conditional properties.

B. Properties data table

Table size is $(N_S \times Q_A)$: N_s a number of sensor node of cluster does not include the CH node, Q_A is a number of conditional properties as defined follows (2), (3) and the order of priority data: Distances sensor nodes and CH, residual energy, residual package. For example $N_S = 6$, $Q_A = 3$ as follows Table 3 data has not been sorted, Table 4 data has been sorted.

TABLE 3. DATA HAS NOT BEEN SORTED

Sensor	Properties condition		
	Distances sensor nodes and CH	*Residual energy*	*Residual package*
S_1	1	3	4
S_2	3	2	4
S_3	1	6	3
S_4	4	5	2
S_5	4	5	1
S_6	4	2	2

TABLE 4. DATA HAS BEEN SORTED

Sensor	Properties condition		
	Distances sensor nodes and CH	*Residual energy*	*Residual package*
S_3	1	6	3
S_1	1	3	4
S_2	3	2	4
S_5	4	5	1
S_4	4	5	2
S_6	4	2	2

C. Sliding window

Set £ is coefficient sliding of sliding window (H_{sw} is height, W_{sw} is width). H_{sw} varies by £ as defined in two cases follows:

If there are half (or more) of Ns nodes have more residuary energy than the average residual energy of N_S sensor nodes in cluster then:

$$£ = N_S/2 \text{ if } N_S \text{ is even or } £ = (N_S+1)/2 \text{ if } N_S \text{ is odd} \quad (5)$$

If less than half of Ns nodes have more residuary energy than the average residual energy of N_S sensor in cluster then:

- Select the nodes whose residuary energy is greater than the average energy of N_S node.
- Determine average distance between sensor node (has selected) and CH.
- £ = number of sensor node with a distance to CH are less than or equal to the average distance. (6)

D. Data for fusion

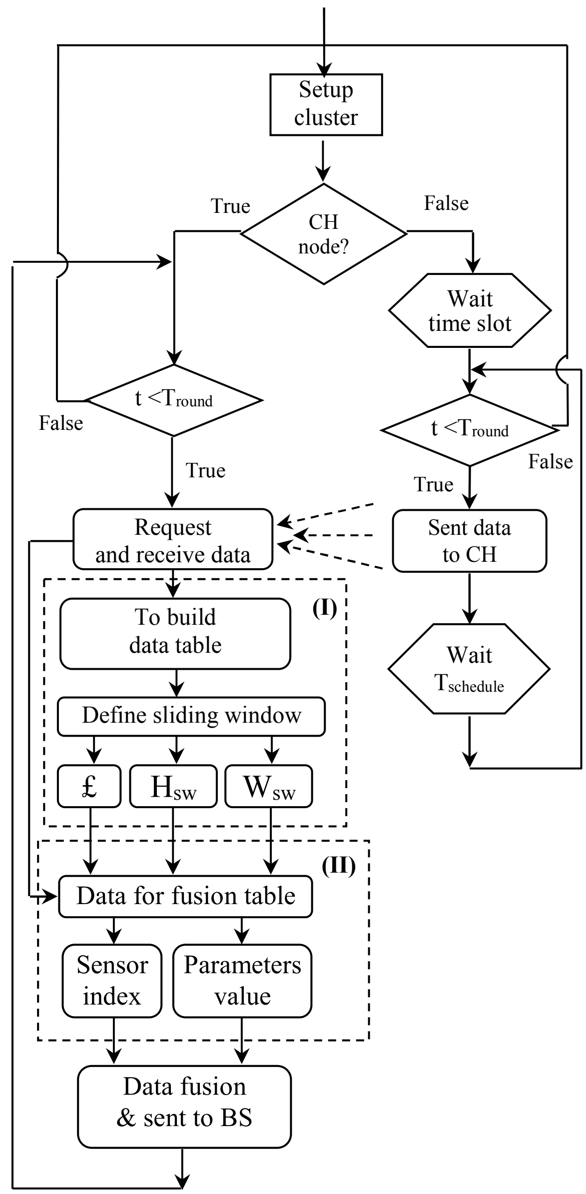
Based on £ to select sensor node. The table measure data for data fusion with £ row and P_{mes} column, where P_{mes} is measurement parameter of sensor node as defined follows II D part. The DF-SWin algorithm, some of sensor node in the sliding window with £ are selected and the measurement data of those nodes will be sent to CH. In this paper, we have not mentioned the fusion algorithm for this data table at CH node.

E. DF-SWin algorithm

1) Flow chart

Flow data chart of DF-SWin as follows Figure 2. Cluster establishment and CH election are inherited by LEACH algorithm. CH node is a node of the cluster, which is selected by probability [7]. The parameters: £ , H_{sw}, W_{sw} are defined in Section III, C.

Figure 2. Flow chart of DF – SWin algorithm



After the cluster is established, if the sensor node is CH and is not finished round (T_{round}), it will wait for receiver header data from sensor node in the cluster to built value of conditional properties of data table. Sensor nodes transmits data to the CH in round and frame based on CDMA *(Code Division Multiple Access)*, TDMA (*Time Division Multiple Access)* and only transmit data into the corresponding time slot specified by the CH. Sensor in standby with $T_{schedule}$, if it reaches to its time slot in frame and has not been finishing round, then transmits data to CH until the end of the time slot and wait for next frame time. Transmission data of sensor node in cluster to CH until the end of the frame and the end of the round.

After each round, WSNs is able to re-configure the network itself, re-divide the WSN to cluster and vote CH for cluster. DF-SWin algorithm consists of two main processing phases as shown on Figure 2. *Phase 1,* based on table of properties data with sensor header information (such as distances between sensor node and CH, residual energy of sensor node, residual package of sensor node) to define sliding window consists of *£, H_{sw}, W_{sw}*. *Phase 2,* based on the sliding window to define the sensor nodes and the measurement parameter values for data fusion at CH node.

In This paper, we have not yet mentioned the calculation and fusion of data to determine the final result before sending the BS. Therefore, the final result of the algorithm is a set of measurement data from several sensor nodes in the cluster, this data for data fusion at CH node.

2) DF-Swin algorithm

a) Convention

Sign	***Signification***
N_s	*Number of sensor of the cluster (not CH)*
S_i	*Sensor i^{th} of cluster ($1 \le i \le n$)*
Q_A	*Number of condition properties*
£	*Coefficent of sliding window*
W_{sw}	*Width of sliding window*
H_{sw}	*Height of sliding window*
Set_{Swin}	*Set of sensor was selected by sliding window*
Set_1, Set_2	*Set of sensor was selected*
P_{mes}	*Number of measurements parameter*
$dist.S_i$	*Distances between Si and CH*
$res.E_{Si}$	*Residual energy of Si*
$res.Pk_{Si}$	*Residual package of Si*
AtoZ	*Sort with order smallest to largest*
ZtoA	*Sort with order largest to smallest*

b) Algorithm

1. *Set $n = N_S$; $m = Q_A$; $Set_{Swin} = Ø$; $Set_1 = Ø$; $Set_2 = Ø$*
2. *Sort_cond_pro (important, left-right)*
3. *For {set i 1} {$i <= $n } {incr i}*
   For {set j 1} {$j <= $m } {incr j}
4. *CH_read_and_write_header_value (S_i, $cond\_pro_j$)*
5. *Return table_full_data (n, m)*
6. *Sort_table ($dist.S_i$_AtoZ,$res.E_{Si}$_ZtoA,$res.Pk_{Si}$_AtoZ)*
7. *Set n_1 = Count ($res.E_{Si} \ge$ Avg ($res.E_{Si}$, n))*
8. *If {$n_1 \ge$ (n/2 where n is even)} or {$n_1 \ge$ ((n+1)/2 where n is odd)}*
9. *Then (£ = n/2 where n is even) or (£ = (n+1)/2 where n is even)*
10. *If {$n_1 <$ (n/2 where n is even)} or {$n_1 <$ ((n+1)/2 where n is odd)}*
11. *Then Select_Si ($res.E_{Si} \ge$ Avg ($res.E_{Si}$, n) to Set_1*
12. *Select_Si_of_Set1 ($dist.S_i \le$Avg($dist.S_i$, n_1) to Set_2*
13. *Set £ = number_of_sensor_node_of_Set_2*
14. *Return_sliding_window (£, m)*
15. *Set_{Swin} = Select (Si, £)*
16. *End.*

Line 1, set initial values *n, m, Set_{Swin} , Set_1, Set_2*; *line 2,* sort the conditional properties according to the importance level decreasing from left to right; *line 3,* loop for each sensor node and each conditional property; *line 4,* CH read and write in the data table header values from sensor nodes in the cluster as conditional properties; *line 5,* return table *n* rows and *m* columns filled with data, *n* is number of sensor node, *m* is number of semantic properties (conditional properties); *line 6,* order of priority data: Distances sensor nodes and CH, residual energy, residual package; *line 7:* Set n_1 is number of sensors with residual energy is not less than the average residual energy of number of sensor node of the cluster; *line 8, line 9,* determinate *£* is based on *n* and n_1 of the cluster as defined follows (5); *line 10 to line 13*, is the case of determinate *£* if there are less than half of nodes of cluster whose residual energy is less than the average residual energy of the nodes of the cluster as defined follows (6); *line 14 to line 16*, return *£* and CH selects *£* sensor node and requests the sensor nodes (which have been selected) to send measurement parameters values, end of algorithm.

IV. SIMULATION

A. Configure simulation

We use NS2 simulation software, version 2.34 installed on Ubuntu 12.04 operating system and source code from MIT *(Massachusetts Institute of Technology).* The parameters of DF-SWin simulation are in Table 5:

TABLE 5. THE MAIN PARAMETERS

Parameter	***Value***
Number of sensor nodes (n)	*100*
Coordinates node in the (100m x 100m)	*Random*
Number of BS, BS coordinates	*1, option*
The min and max number of clusters	*1 → 10*
The number of clusters desired	*5*
Initial residual energy of sensor nodes	*2 J*
Energy to receive 1 bit	*5 nJ*
Energy consumption to send 1 bit	*50 nJ*
Amplification factor radio transmissions	*$10pJ/bit/m^2$*
Capacity of node while Idle or Sleep	*0 W*
Residual energy of BS	*No limit*
Speed of radio transmissions	*1 Mbps*
Header size (hdr_size)	*25 Byte*
Sensing data size (sig_size)	*500 Byte*
Time per round/ data fusion (T)	*20 s (option)*
Number of sensor nodes in cluster	*Random*
Number of condition properties (Q_A)	*3 (option)*
Number of measurements parameter (P_{mes})	*4 (option)*

B. Analysis and evaluate efficient

Number of clusters and nodes in each cluster during simulation in Table 6.

Every *20* seconds, the network automatically divides into clusters, each cluster has a corresponding number of sensors. For example, at *240^{th}* second, number of active nodes of the network are *98* nodes (*2* nodes are die) and is divided into *4* clusters. The number of nodes of the cluster (including CH), are *16* nodes, *39* nodes, *14* nodes and *29* nodes.

TABLE 6. NUMBER OF CLUSTER, SENSOR

Time (secth)	Total alive node	CLUSTER							
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
20	***100***	14	10	9	5	26	26	10	
40	***100***	26	16	23	10	25			
60	***100***	18	11	44	27				
80	***100***	48	25	11	16				
100	***100***	11	15	18	37	19			
120	***100***	17	10	10	17	17	14	15	
140	***100***	28	52	20					
160	***100***	17	8	10	7	11	19	14	14
180	***99***	17	24	16	31	11			
200	***99***	27	10	18	25	19			
220	***99***	26	29	5	39				
240	***98***	16	39	14	29				
260	***98***	14	31	30	23				
280	***97***	28	53	11	5				
300	***94***	9	13	27	12	14	19		
320	***94***	22	20	21	31				
340	***92***	48	4	26	14				
360	***89***	17	19	5	4	20	24		
380	***73***	22	30	21					
400	***68***	6	13	8	5	16	5	8	7
420	***50***	5	9	15	17	4			

Apply the algorithm when residual energy of sensor node is quite full (compared to the *2J* at initialization) for example at the *80th* second and when most of energy of sensor node has consumed, for example at *320th* seconds, the calculation result of coefficient *£, H_{sw}, W_{sw}* as shown in Table 7.

At *80th* second, for example analysis number of node of cluster *1* are *48* nodes, $N_S = 47$ because it does not include CH. After receiving and filtering header data *(hdr_size = 25 Bytes)* from the sensor nodes of cluster to fill the table. Then, apply DF-SWin, the number of nodes satisfy the condition: $res.E_{Si} \geq Avg\ (res.E_{Si},\ n)$ are *40* nodes, because the network just works, most of sensor nodes have a lot of residual energy compared to the *2J* at the time of initialization, *£ = 24* because N_S is odd. Based on analyzing the header data of nodes in the cluster and *£, CH* only chooses measurements parameter data of nodes of the sliding window with the total capacity (of those nodes) has sent to CH are *179* data instead of *474* data of LEACH, in this case had saved *295* data *(62.23%)*. With a data capacity is *sig_size = 500 Byte*, the energy savings for the cluster are: *(295 sig_size * 500 Byte/sig_size * 8bit / Byte * 50nJ / bit) = 59.000.000 nJ = 0.059 J.*

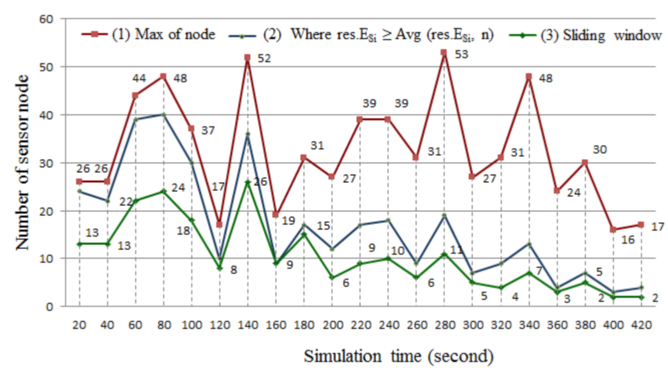
Similarly, at *320th* second, for example: analysis number of node of cluster *1* are *22* nodes, $N_S = 21$ because of $n_1 = 9$ that means only *9* sensors satisfy the condition $res.E_{Si} \geq Avg\ (res.E_{Si},\ n)$ and $n_1 < (N_S + 1)/2 = 11$ sensors, Set_1 will include this *11* sensors. Average distance of these *11* sensors to CH is $Avg\ (dist.S_i,\ n_1) = 25$. Only *6* sensors satisfy the condition: $dist.S_i \leq Avg\ (dist.S_i\ ,\ n_1)$ and set to Set_2, *£ = 6*. Total number of data packets DF-SWin saving compared to LEACH are *234 sig_size,* equivalent to *72.44%.* The energy saving amount of *21* sensors of cluster (means *15* sensors not send data to CH) is: *(234 sig_size * 500 Byte/sig_size * 8bit / Byte * 50nJ / bit) = 46.800.000 nJ = 0.0468 J.*

Table 8 is effective survey of DF-SWin to clusters that have the most sensor nodes of rounds *(T = 20s).*

The simulation results demonstrate that, DF-SWin algorithm tends to promote energy efficiency when residual energy of sensor node decreases and the network has few alive sensor nodes. For example, at *420th* second, only *50* sensors alive, they are divided into *5* clusters, the most of node of cluster are *17* (including CH), when using DF-SWin has saved *93.06%* of the cluster energy.

Comparing three cases: *first,* the number of sensor nodes of the cluster with the greatest number of nodes; *second,* the number of nodes whose residual energy levels are not lower than the average residual energy of the cluster; *third,* the number of sensor nodes in the sliding window. Test result in *420th* second as shown in Figure 3.

Figure 3. Test result in 420th second of 3 cases



Compare DF-SWin and LEACH about energy consumption of a sensor node (apply DF-SWin algorithm) that has been operating the during at *20th* to *420th* second (such as node *26*) and the average residual energy of *50* alive sensor nodes as shown in Figure 4.

Figure 4. Efficiency of residual energy between DF-SWin and LEACH

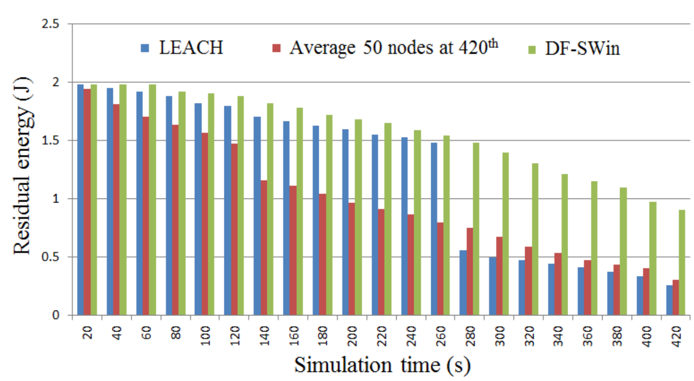


TABLE 7. RESULT AT 80TH SECOND, 320TH SECOND

Time at (second)	Cluster Index	Number sensor of cluster	Avg (res.E_{Si}, n) (J)	n_1 {res.E_{Si} ≥ Avg (res.E_{Si}, n)}	£ (Hws)	Capacity data in CH for DF (sig_size)		Energy savings (J)
						DF-SWin	LEACH	
80	*1*	48	1.73834295	40	24	179	474	0.059
	2	25	1.69793760	20	12	75	177	0.0204
	3	11	1.76963819	10	5	65	126	0.0122
	4	16	1.40296564	8	8	72	139	0.0134
320	*1*	22	0.25857710	9	6	89	323	0.0468
	2	20	0.49958504	7	3	54	374	0.064
	3	21	0.45585774	5	3	39	286	0.0494
	4	31	0.41489440	9	4	67	420	0.0706

TABLE 8. EVALUATE ENERGY EFFICIENCY BY EXAMINING THE MOST SENSOR NODES OF ROUNDS

Time (s^{th})	20	40	60	80	100	120	140	160	180	200	220
DF-SWin	120	223	226	179	90	143	318	101	270	50	138
LEACH	246	477	528	474	334	256	705	206	543	212	522
Efficent (%)	51.22	53.25	57.20	62.24	73.05	44.14	54.89	50.97	50.28	76.42	73.56
Time (s^{th})	**240**	**260**	**280**	**300**	**320**	**340**	**360**	**380**	**400**	**420**	
DF-SWin	78	90	102	65	67	72	15	54	18	15	
LEACH	301	391	474	258	420	499	115	315	135	216	
Efficent (%)	74.09	76.98	78.48	74.81	84.05	85.57	86.96	82.86	86.67	93.06	

The simulation results show that, for LEACH, every *20* seconds, all node in cluster sent data to their CH. For DF-SWin, every *20* seconds, the sensor node consumes energy to send header data to CH, each header packet *(hrd_size)* are *25* Bytes, energy consumption is about *10,000 nJ,* and it is possible that consume more energy to send data to CH if that sensor node belongs in sliding window. In addition, at any/a cluster in a round, if the sensor node is selected for CH, the consumption energy of that node (is CH) will increase for sending and receiving information between CH and sensor node in cluster. For example, at *260th* second, node *26* was selected as CH, number of node of cluster are *30* nodes, $n_1 = 9$ nodes, for DF-SWin only select *6* nodes with total data transmit to CH are *90* data (LEACH are *391* data). When the sensor node is CH and sending - receiving data with sensor nodes in the cluster and sending data to the BS, energy of node *26* is abruptly decreases *(1.482531708J to 0.562692387J)* as shown in Figure 4.

V. CONCLUSION

The idea of the DF-SWin method is based on a combination of some concepts of Rough Set Theory, Data mining and sliding window in TCP/IP protocol. The paper proposes using the sliding window to sampling some sensor node in the cluster based on the semantics properties of the sensor node, then select measurements parameter value of sensor node for data fusion in CH node.

To save energy, CH may to require nodes in the cluster to send header data with small capacity as a conditional properties value for adjusting sliding window size and select sensor to send data to CH.

The DF-SWin method is suitable for wireless sensor networks consists of sensors node with low residual energy and unequal energy levels.

REFERENCES

[1] Shio Kumar Singh, M P Singh, D K Singh, *"A Survey of Energy-Efficient Hierarchical Cluster-Based Routing in Wireless Sensor Networks"*, Int. J. of Advanced Networking and Applications, Volume: 02, Issue: 02, Pages: 570-580 (2010).

[2] Martin E. Liggins, David L. Hall, James Llinas, *"Handbook of Multisensor Data Fusion Theory and Practic"*, 2nd Edition, Sep 2008.

[3] Li Li, Fan Bai, *"Analysis of Data Fusion in Wireless Sensor Networks"*, Electronics, Communications and Control (ICECC), 2011 International Conference on, pp. 2547 – 2549.

[4] Avni Kaushik, *"A review on Routing Techniques in Wireless Sensor Networks"*, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 6, June 2014, ISSN (Online) : 2278-1021, pp 7221-7223.

[5] Geetika Dhand,S.S.Tyagi, *"Data aggregation techniques in WSN: Survey"*, ICCC-2016, Published by Elsevier B.V. Procedia Computer Science 92 (2016), pp 378 – 384.

[6] Zdzisaw Pawlak, *"Rough sets"*, International Journal of Computer and Information Sciences, 11, 341-356, 1982.

[7] W. Heinzelman, A.P. Chandrakasan and H. Balakrishnan,*"Energy-Efficient Communication Protocol for Wireless Microsensor Networks"*, IEEE Proceedings of the Hawaii International Conference on System Sciences, January 4-7, 2000, Maui, Hawaii.

[8] Vaibhav Pandey, Amarjeet Kaur and Narottam Chand, *"A review on data aggregation techniques in wireless sensor network"*, Journal of Electronic and Electrical Engineering, Vol. 1, Issue 2, 2010, pp.01-08.