Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 502082, 10 pages
http://dx.doi.org/10.1155/2013/502082

Research Article

Visual Scheme for the Detection of Mobile Attack on WSN Simulator

Young-Sik Jeong,[1] Hyun-Woo Kim,[1] and Jong Hyuk Park[2]

[1] Department of Multimedia Engineering, Dongguk University, 30 Pildong-ro 1 Gil, Jung-Gu, Seoul 100-715, Republic of Korea
[2] Department of Computer Science and Engineering, Seoul National University of Science and Technology,
Seoul 139-743, Republic of Korea

Correspondence should be addressed to Jong Hyuk Park; parkjonghyuk1@hotmail.com

Received 23 May 2013; Accepted 21 July 2013

Academic Editor: Michael H. Jeong

Copyright © 2013 Young-Sik Jeong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, wireless sensor networks (WSNs) technologies have been utilized in diverse domains. Areas where WSNs are applied have been expanded from industries, schools, and research institutes to all fields of the human society. However, WSNs should be fixed or flexible depending on the areas of application and monitoring situations. Furthermore, measures for the security of data sensed when sensor nodes communicate with each other are not perfect, so sensors are sometimes easily attacked, and the security measures cover neither cases where sensors receive wrong information nor cases where attacks on external sensor nodes are sensed. Therefore, this paper provides the GML that can be mapped on actual topography so that optimum coverage can be inferred and can set target areas for two situations: mobile sensor networks (MSNs) and fixed sensor networks (FSNs). Sensors can be efficiently arranged through this sensor node information, and when the sensors have been arranged, security simulation functions applied with data encryption for data transmission between sensor nodes are provided. This paper also proposes an external detection trace simulator (EDTS) that would make sensing data transmitted between sensors visually provide information on the sensing of external attacks.

1. Introduction

Recently, wireless sensor networks (WSNs) technologies have been utilized in diverse areas as application services. In addition, as ubiquitous paradigms have been expanded, areas where WSNs are applied have been expanded from industries, schools, and research institutes to all areas in the human society [1–3].

In general, sensor nodes transmit the sensed information to sink nodes through wireless networks, and sink nodes transmit the information to middleware or servers. These transmitted data are processed to be suitable for applications in diverse situations. WSNs technologies have been used in diverse areas such as medical devices, medical systems,such as those for the elderly persons, traffic control and safety, high-grade car systems, process control, energy saving, important social infrastructures, aviation software, weapon systems, distributed robots (robots processed by multiple computers), manufacturing, and communications [2, 4].

WSNs are divided into mobile sensor networks (MSNs) and fixed sensor networks (FSNs) from the aspect of the mobility of sensors. MSNs are usefully used in certain target areas that cannot be easily accessed by humans or other equipment or that must be always flexibly monitored. The sensed information by the sensors while they are moving such as heat, temperature, magnetic fields, and sounds is transmitted to application servers through communication protocols between sensors. Depending on the areas of application, sensor nodes establish FSNs which remain at fixed positions, and these FSNs are usefully used in target areas where diverse kinds of sensing information should be constantly observed and periodically monitored.

For establishing these MSNs and FSNs, the coverage of sensing by sensor nodes, the connectivity between sensor nodes, the optimum arrangements of sensor nodes in target areas, the number of sensors, and the data security between sensors are very important. Evaluation and judging whether sensors have been optimally arranged when establishing

MSNs or FSNs is very difficult. Actually, configuring MSNs or FSNs and testing them incurs quite large costs. For these reasons, simulators have been frequently used to apply newly studied network configurations and sensor arrangements and to verify and supplement the foregoing. As tools for sensor arrangements and communication protocol design and verification, diverse simulators have been developed such as GloMoSim [5], SNetSim [6], ATEMU [7], QualNet [8], NS2 [9], EmStar [10], TOSSIM [11], J-Sim [12], AVRORA [13], SWANS [14], and SENSE [15]. However, despite the fact that diverse simulators are available, the simulators are operated with limited sensor node information, and, thus, quite limited outcomes are obtained. Furthermore, measures for the security of data sensed when sensor nodes communicate with each other are not perfect, so sensors are sometimes easily attacked, and the security measures do not cover cases where sensors receive wrong information and cases where attacks on external sensor nodes are sensed. In addition, methods to visually provide information on situations where modified data or data that lost meaning are communicated between sensor nodes are not provided.

Therefore, in this paper, GML [16] can be mapped on actual topography so that optimum coverage can be inferred and sets target areas for two situations: MSNs and FSNs. So that diverse sensor nodes can be simulated instead of limited sensor node information for the set target areas, sensor node information is entered by users. Sensors can be efficiently arranged through this sensor node information, and when the sensors have been arranged, security simulation functions applied with data encryption for data transmission between sensor nodes are provided. This paper also proposes an external detection trace simulator (EDTS) that would make the sensing data transmitted between sensors visually provide information on the sensing of external attacks.

2. Related Work

The functions and characteristics of the existing simulators related to the present study are reviewed as shown in Table 1.

3. Detection of Attacks with Situations

EDTS, which is proposed in this paper, in order to enhance the usability of the simulator, functions for the recognition of situations of attack on WSNs were divided into the following three functions. First, it recognizes disconnection of communications to sensor nodes. The disconnection is the case where communications to a sensor node in an established WSN are disconnected despite the fact that the sensor node has 20% or more residual battery capacity. EDTS records the time of disconnection of communications to the sensor node and recognize the disconnection as an attack if the period is shorter than the value set by the user. For instance, if the user has set the communication disconnection period as 24 hours, cases where communications to one sensor node are disconnected per day are not recognized as situations of attack, but cases where communications to two more sensor nodes are disconnected within 24 hours are recognized as situations of attack. Actually, this function also includes functions for sensing sensor node failure (hardware/software failure).

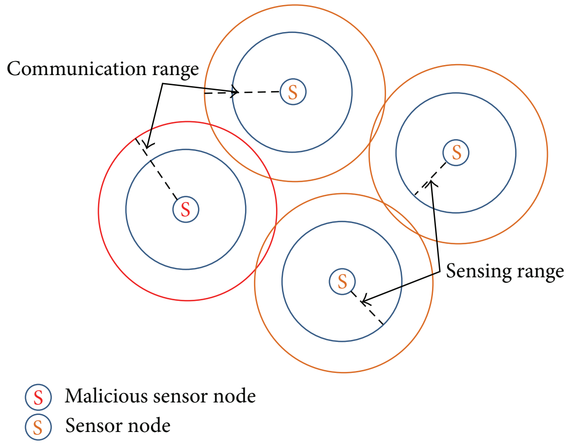


Figure 1: The analogy position of the attacked sensor node with communication range.

Second, EDTS recognizes cases where MSNs and FSNs have been established in the target area and the numbers of events of sensor nodes rapidly increase intensively in a place despite the fact that there is no factor for increases in the numbers of events of sensor nodes in the observed area as situations of potential attacks from the outside.

Finally, it recognizes cases where unknown data which are not sensing data generated during communications with sensor nodes are periodically received and energy consumption rates are increased due to interruptions in communications as situations of attack. In each situation of attacks, EDTS predicts the position of the attacking node based on the positions of the sensor nodes that have been attacked.

As shown in Figure 1, if a case that falls under one of the three situations for sensing attacks exists in relation to a malicious sensor node, the position of the attacking node is sensed based on the positions of sensor nodes.

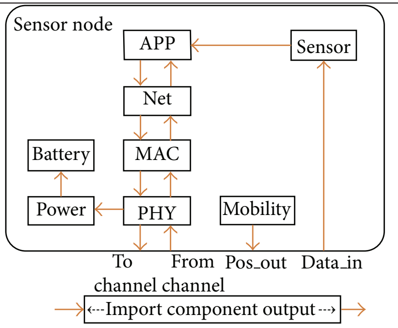
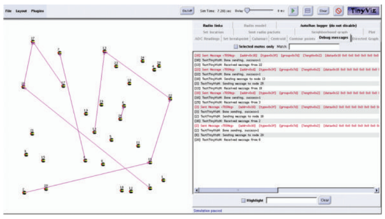
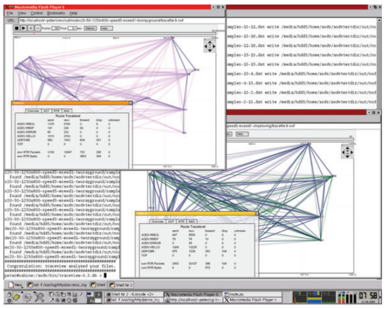
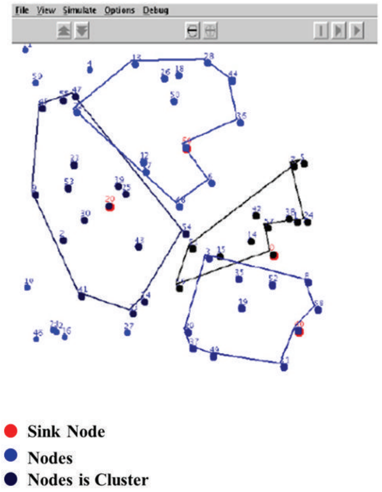
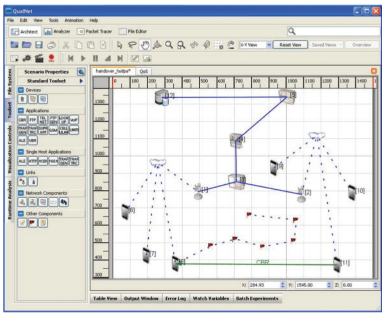
4. Design of EDTS

The design of EDTS includes the following functions to provide simulations wanted by applications and users. First, EDTS receives inputs of information on the topography necessary for simulations and sensor nodes and maximizes coverage for the target area of the sensor nodes inputted. Second, EDTS basically includes a function to sense attacks from the outside during communications between sensor nodes. Finally, EDTS provides users with visual information on sensed attacks from the outside and sensor node arrangements and mobility.

Basically, EDTS largely consists of a user interface, an interaction broker, a target area manager, a map manager, a map controller, a node manager, a detected manager, a coordinate converter, and a viewer as shown in Figure 2.

To be specific, the user interface component includes a map interface intended to receive information on the GML that can be mapped on actual topography. To predict the

Table 1: Functions and characteristics of the existing simulators.

Simulator	Functions and characteristics	Execution example
SENSE [15]	(i) C++ component-based design (ii) The speed of implementation of the simulation is low (iii) No function to recognize malicious attacks on sensor nodes	 Sensor node APP Sensor Net Battery MAC Power PHY Mobility To channel From channel Pos_out Data_in ←--Import component output--→
TOSSIM [11]	(i) TinyOS-based simulator developed by Berkeley University in the USA using open sources (ii) Major functions include packet loss rate measurement and CRC detection (iii) Low expandability since this can be applied to only Mica series (iv) Additional sensors other than LEDs cannot be used (v) No function to recognize situations of malicious attacks on sensor nodes during data transmission	
NS2 [9]	(i) Discrete event simulator (ii) Can simulate diverse network protocols (iii) Cannot be applied to large-scale systems that have large numbers of nodes and are complicated, this involving a high degree of unnecessary interdependence between modules (iv) No function to simulate sensing of situations of attack on established networks	
GloMoSim [5]	(i) Expanded simulation library intended to make wireless network systems using the C-based parallel simulation language PARSER (ii) Consists of several layers as with OSI 7 layer models (iii) No functions to express coverage and to sense situations of malicious attacks on sensor nodes during data transmission	 File View Simulate Options Debug ● Sink Node ● Nodes ● Nodes is Cluster
QualNet [8]	(i) This is the next version of GloMoSim and is a large wireless network simulator (ii) When modules of individual layers have been developed by different designers, the scenarios and models can be experimented, and the statistics of packet flows can be identified through results automatically collected in each layer (iii) Although functions for sensor networks have been designed, the functions support only RF analysis for coverage expression; therefore, this simulator has limitations in sensing coverage expression (iv) No function to sense situations of malicious attacks on sensor nodes	

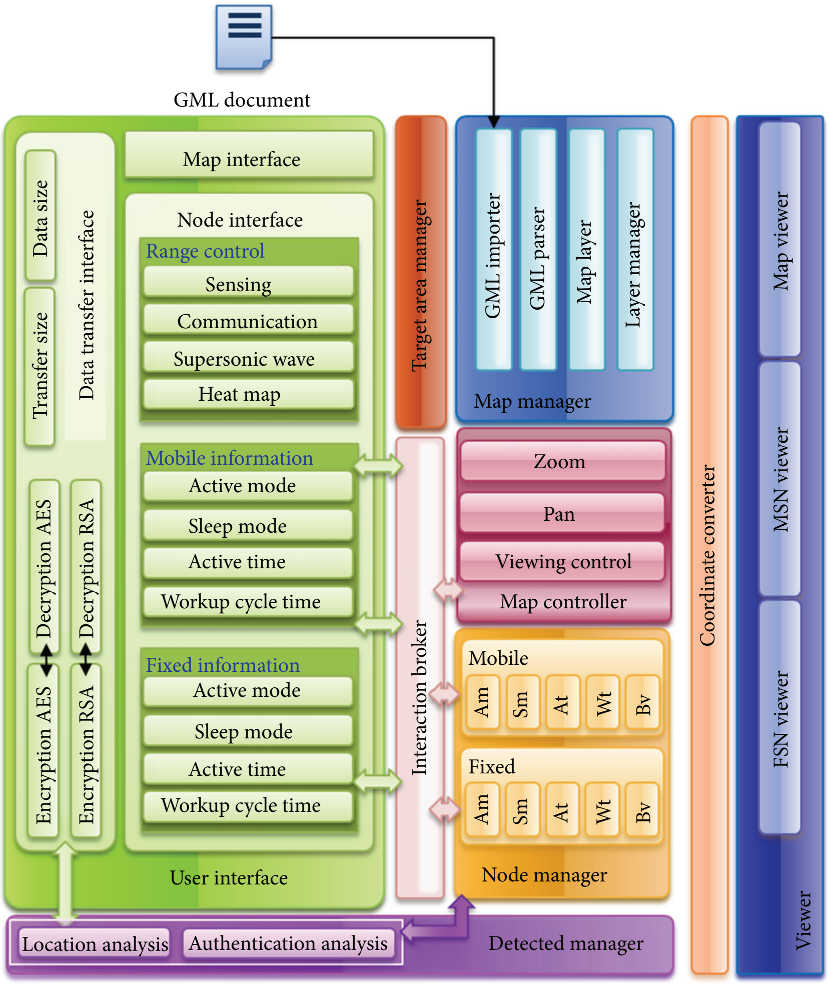


Figure 2: The architecture of EDTS.

lives of the mobile sensor node (MSN) and the fixed sensor node (FSN) based on energy consumption rates, the user interface also includes a node interface to receive inputs of the active mode, the sleep mode, the active time, and the workup cycle time from the user as a default setting. In addition, a data transfer interface is configured to receive inputs of basic sizes of data and data transmission size settings as encrypted settings, and selective encryption settings of RSA (Rivest-Shamir-Adleman) and AES (advanced encryption standard) from the user when transmitting data from the set sensor nodes.

The interaction broker component serves the role of a broker to transfer the information received by the user interface from the user to the map controller, the node manager, and the detected manager.

The map manager component reads the GML documents for the target topography through the GML importer and parses the documents through the GML parser. The map layer creates map objects for obstacles in the analyzed GML topography data and sends the information to the layer manager. The layer manager manages the analyzed topography information. The map controller component plays the role of controlling the user's inputs such as enlarging and reducing maps in the layer manager, enlarging areas, moving areas, and outputting the results to the viewer through the coordinate converter.

The target area manager component sets the target areas that should be observed based on the GML documents read through the GML importer and analyzed thereafter. The detected manager component applies algorithms based on encryption settings when data are transmitted to mobile sensor nodes and fixed sensor nodes and analyzes attacking nodes when they are found so that their positions can be inferred.

The node manager component applies and arranges information on mobile sensor nodes and fixed sensor nodes received by the user interface from the user and creates and operates sensor nodes where the obstacles defined by the map manager and the target area defined by the target area manager interact with each other. The node manager Component also provides and manages statistical information on residual battery capacities and expected lives of mobile sensor nodes and fixed sensor nodes and analyzes sensor nodes suspected

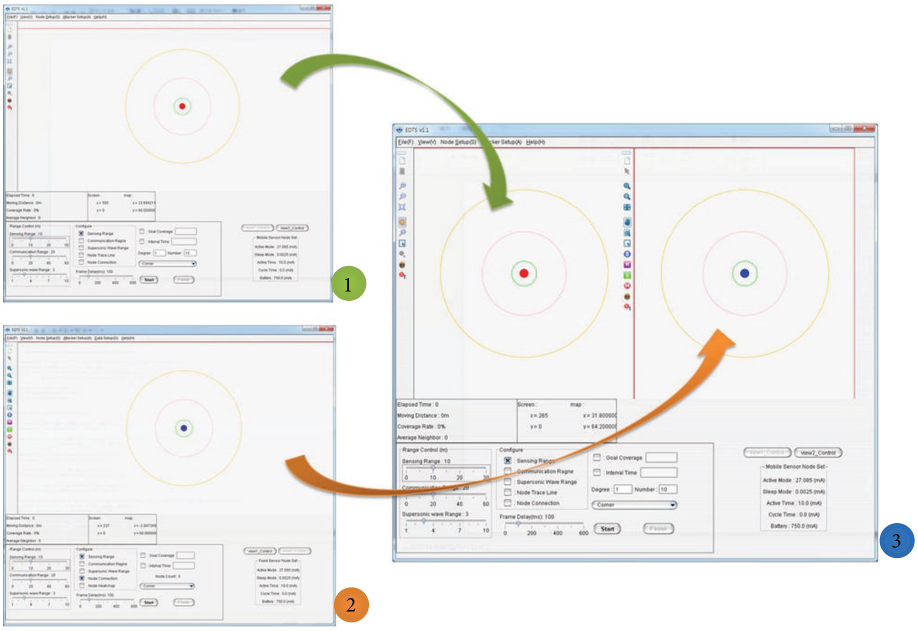
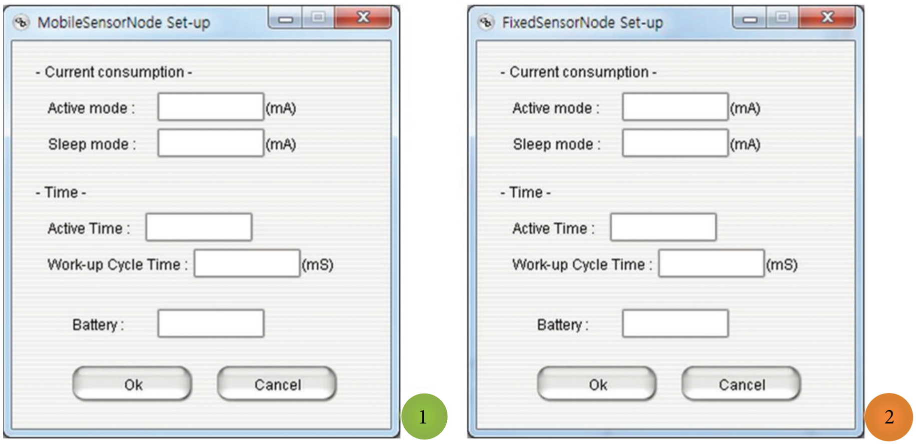


Figure 3: The initial execution of EDTS.


MobileSensorNode Set-up

- Current consumption -

Active mode : (mA)

Sleep mode : (mA)

- Time -

Active Time :

Work-up Cycle Time : (mS)

Battery :

FixedSensorNode Set-up

- Current consumption -

Active mode : (mA)

Sleep mode : (mA)

- Time -

Active Time :

Work-up Cycle Time : (mS)

Battery :

Figure 4: The setting of MSN and FSN.

as attacking nodes in accordance with encryption settings during data transmission based on user settings.

The coordinate converter component serves the function of sending data on the topography and the situations of sensor node operations to the viewer. The viewer component draws efficient arrangements of data received through the coordinate converter for the user in relation to mobile sensor nodes and fixed sensor nodes as well as visually providing information on detected situations of threatening nodes suspected as attacking nodes during sensor data communications.

5. Implementation of EDTS

Figure 3 is a visualized initial execution of EDTS that shows a view (①) of an MSN (mobile sensor node) and a view (②) of an FSN (fixed sensor node). The right side (③) of Figure 3 is a view of the execution when the user set the simulator to operate the MSN and the FSN simultaneously.

In the composition of EDTS exists the menu bar on the top to select the views of the current consumption values based on the states of operation of the sensor nodes in the MSN and the FSN, battery sizes, and operations and to select data values and data encryption settings when data are transmitted. The center of the screen consists of maps for the target topography, a Viewer visualized to recognize sensor node information, arrangement states, and attack situations, and a toolbarexists on the left side of the viewer for entering the maps of the target topography to be displayed on the viewer, enlarging and reducing the maps, sensor node addition, and additional arrangements of malicious nodes. An RC (range control) is provided for setting SR (sensing range), CR (communication range), and SwR (supersonic wave range) as internal settings of sensor nodes for each MSN

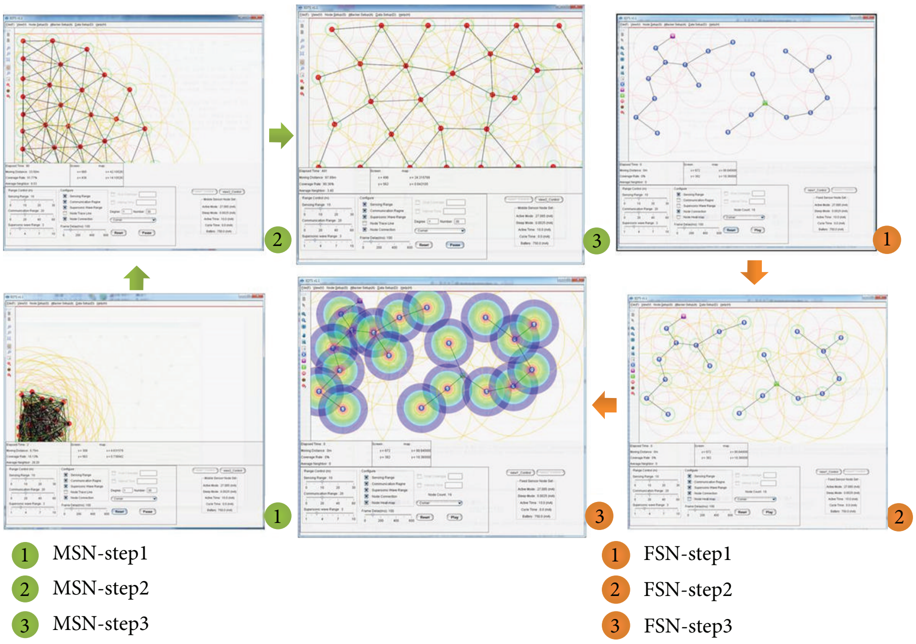
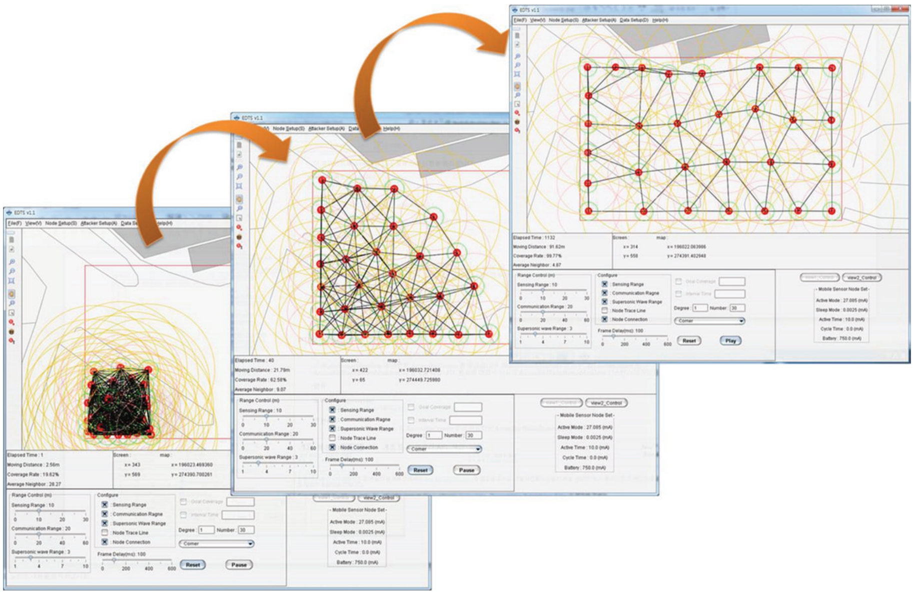


Figure 5: The activation of MSN and FSN.


Elapsed Time : 1
Moving Distance : 2.56m
Coverage Rate : 19.62%
Average Neighbor : 28.27
Screen : x = 343 y = 569
map : x = 196023.469360 y = 274390.700261

Elapsed Time : 40
Moving Distance : 21.79m
Coverage Rate : 62.58%
Average Neighbor : 9.07
Screen : x = 422 y = 65
map : x = 196032.721408 y = 274449.725980

Elapsed Time : 1132
Moving Distance : 91.62m
Coverage Rate : 99.77%
Average Neighbor : 4.87
Screen : x = 314 y = 558
map : x = 196022.063986 y = 274391.402948

Range Control (m)
Sensing Range : 10
Communication Range : 20
Supersonic wave Range : 3

Configure
☒ : Sensing Range
☒ : Communication Ragne
☒ : Supersonic Wave Range
☐ : Node Trace Line
☒ : Node Connection
☐ Goal Coverage
☐ Interval Time
Degree : 1 Number : 30
Corner
Frame Delay(ms): 100

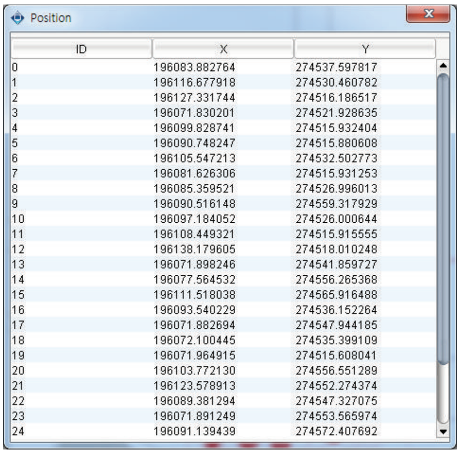
- Mobile Sensor Node Set -
Active Mode : 27.085 (mA)
Sleep Mode : 0.0025 (mA)
Active Time : 10.0 (mA)
Cycle Time : 0.0 (mA)
Battery : 750.0 (mA)

Figure 6: The execution of the MSN area.

viewer and FSN viewer. In addition, it also comprises the Configure section to receive inputs of individual ranges of sensor nodes to be displayed on the viewer, the control view at the bottom for setting Frame Delay to set simulator operation periods, and the Information view to provide information on sensor nodes entered through the menu bar.

Figure 4 shows cases where sensor node information is set through the menu bar on the top. In these cases, to predict battery lives of the MSN and the FSN, the basic battery capacities of sensor nodes, the amounts of current energy consumption in relation to the operation states, and operation periods should be entered. The available settings include the amounts of the current energy consumption in the cases of the active mode and the sleep mode, the active time for operating time, and the work-up cycle time to set the periods of the starting of the sensor node operation, and the user can set arbitrary sensors and operations by setting battery sizes not only uniform sensors.

Figure 5 shows a state where basic information on sensors is set and sensors are simulated without receiving the input of the target topography in order to observe the view of operations in relation to the sensor positions in an MSN and



Position

ID	X	Y
0	196083.882764	274537.597817
1	196116.677918	274530.460782
2	196127.331744	274516.186517
3	196071.830201	274521.928635
4	196099.828741	274515.932404
5	196090.748247	274515.880608
6	196105.547213	274532.502773
7	196081.626306	274515.931253
8	196085.359521	274526.996013
9	196090.516148	274559.317929
10	196097.184052	274526.000644
11	196108.449321	274515.915555
12	196138.179605	274518.010248
13	196071.898246	274541.859727
14	196077.564532	274556.265368
15	196111.518038	274565.916488
16	196093.540229	274536.152264
17	196071.882694	274547.944185
18	196072.100445	274535.399109
19	196071.964915	274515.608041
20	196103.772130	274556.551289
21	196123.578913	274552.274374
22	196089.381294	274547.327075
23	196071.891249	274553.565974
24	196091.139439	274572.407692

FIGURE 7: real coordinate of the target area for the sensor node.

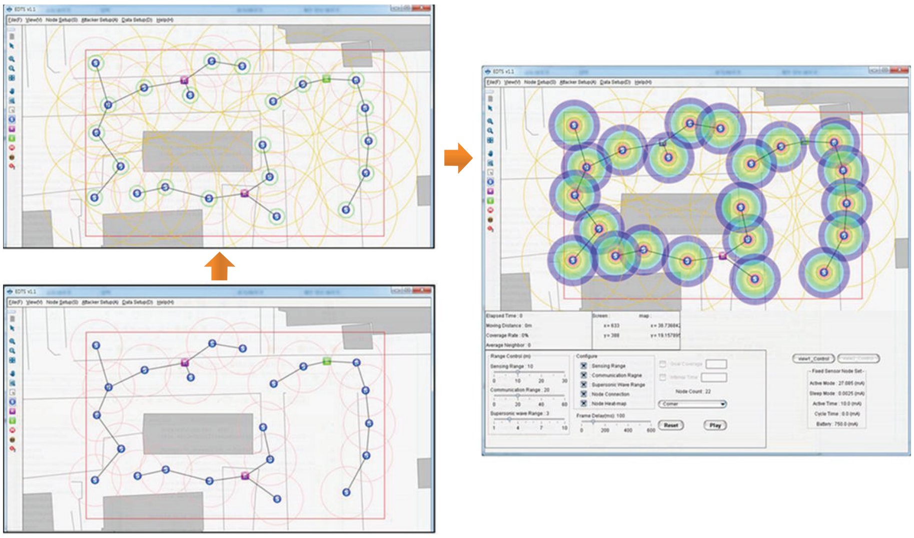
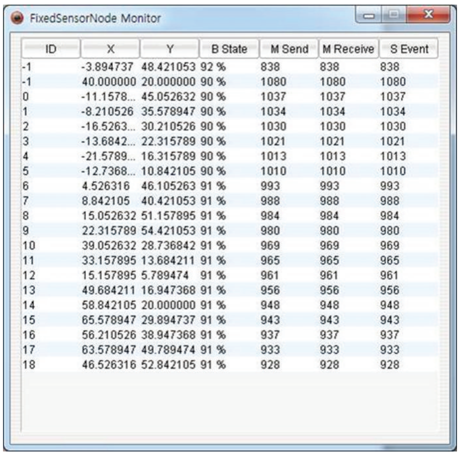


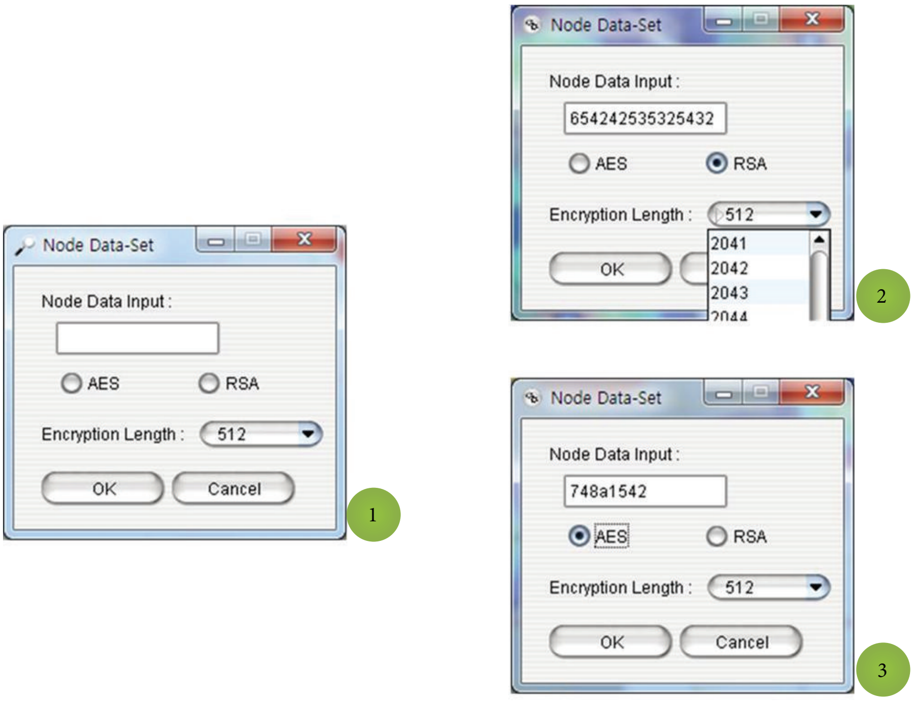
FIGURE 8: The simulation of FSN with the target area.



FixedSensorNode Monitor

ID	X	Y	B State	M Send	M Receive	S Event
-1	-3.894737	48.421053	92 %	838	838	838
-1	40.000000	20.000000	90 %	1080	1080	1080
0	-11.1578...	45.052632	90 %	1037	1037	1037
1	-8.210526	35.578947	90 %	1034	1034	1034
2	-16.5263...	30.210526	90 %	1030	1030	1030
3	-13.6842...	22.315789	90 %	1021	1021	1021
4	-21.5789...	16.315789	90 %	1013	1013	1013
5	-12.7368...	10.842105	90 %	1010	1010	1010
6	4.526316	46.105263	91 %	993	993	993
7	8.842105	40.421053	91 %	988	988	988
8	15.052632	51.157895	91 %	984	984	984
9	22.315789	54.421053	91 %	980	980	980
10	39.052632	28.736842	91 %	969	969	969
11	33.157895	13.684211	91 %	965	965	965
12	15.157895	5.789474	91 %	961	961	961
13	49.684211	16.947368	91 %	956	956	956
14	58.842105	20.000000	91 %	948	948	948
15	65.578947	29.894737	91 %	943	943	943
16	56.210526	38.947368	91 %	937	937	937
17	63.578947	49.789474	91 %	933	933	933
18	46.526316	52.842105	91 %	928	928	928

FIGURE 9: The status of the sensor node with the consumption ratio and The event number.



Node Data-Set

Node Data Input:

○ AES ○ RSA

Encryption Length: 512

OK Cancel

1

Node Data-Set

Node Data Input: 654242535325432

○ AES ● RSA

Encryption Length: 512

2041
2042
2043
2044

OK

2

Node Data-Set

Node Data Input: 748a1542

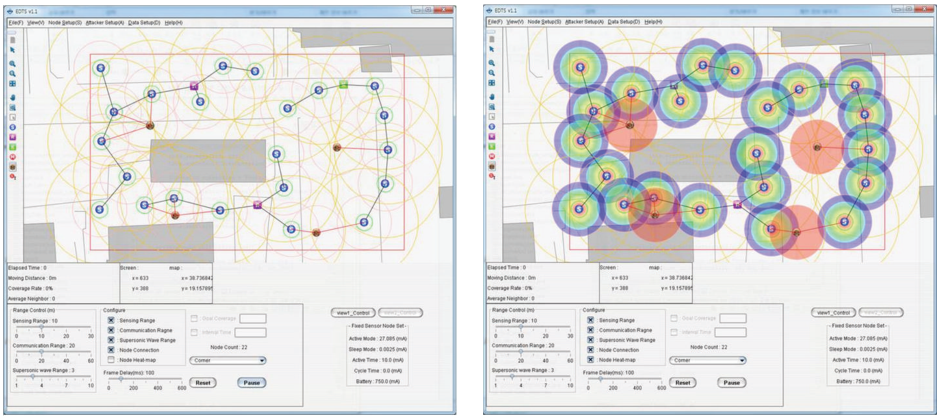
● AES ○ RSA

Encryption Length: 512

OK Cancel

3

Figure 10: The setting of the data and the encryption of the sense node.



(a)

(b)

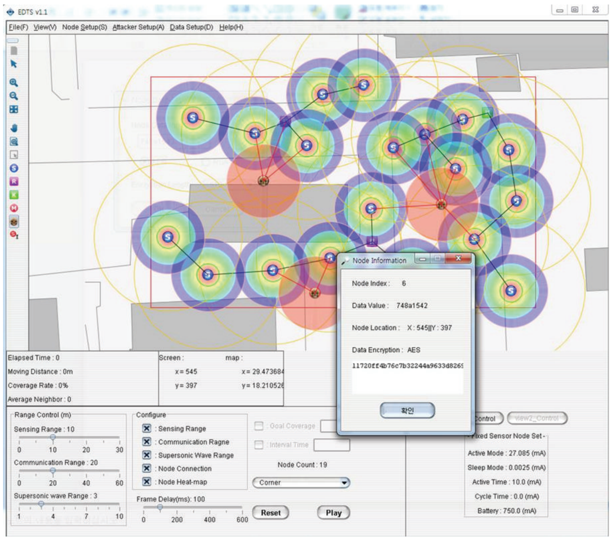
Figure 11: The detection of the attack situation for the sensor node.

an FSN and the sensor operations progress in the order of step 1-step 2-step 3 over operating time.

Figure 6 shows an execution of a case where sensor nodes were set after receiving the GML information which is the information on topography and the sensor nodes were simulated. This shows the process of the sensor nodes' movements over operating time. Figure 7 exhibits the topographic coordinates of the sensor nodes in the process of movements shown in Figure 6. Through the topographic coordinates, information on positions where the actual sensor node will be arranged can be selected.

Figure 8 shows also information on sensor nodes in an FSN arranged and simulated firsthand by the user. Individual steps visually provide sensor node ranges and sensor nodes' energy consumption.

Figure 9 explains the coordinates of the sensor nodes arranged in Figure 8, the rates of the residual battery capacities, the sensor node's sensing, and the numbers of communication events between the sensors. Based on the default settings of the sensor nodes set by the user, the speed of energy consumption can be predicted in relation to energy consumption rates to judge and monitor whether the sensor nodes have been efficiently set. In addition, by grasping the overall degree of energy consumption while monitoring energy consumption rates of different sensor nodes, the sensor nodes can be optimally arranged.


EDTS v1.1
File(F) View(V) Node Setup(S) Attacker Setup(A) Data Setup(D) Help(H)
Node Information
확인
Reset
Play
Control
view2_Control
Corner

Node Index : 6

Data Value : 748a1542

Node Location : X : 545||Y : 397

Data Encryption : AES

11720ff4b76c7b32244a9633d826

Elapsed Time : 0
Moving Distance : 0m
Coverage Rate : 0%
Average Neighbor : 0

Screen : x = 545 y = 397
map : x = 29.473684 y = 18.210526

Range Control (m)
Sensing Range : 10
Communication Range : 20
Supersonic wave Range : 3

Configure
☒ : Sensing Range
☒ : Communication Ragne
☒ : Supersonic Wave Range
☒ : Node Connection
☒ : Node Heat-map

Frame Delay(ms): 100

□ : Goal Coverage
□ : Interval Time
Node Count : 19

- Fixed Sensor Node Set -
Active Mode : 27.085 (mA)
Sleep Mode : 0.0025 (mA)
Active Time : 10.0 (mA)
Cycle Time : 0.0 (mA)
Battery : 750.0 (mA)

Figure 12: Information of the sensor node.

As shown in Figure 10, data to be transmitted between sensors can be set for the established MSN and FSN, and when data are transmitted, encryption algorithms can be set by the user. Since experimenting actually encrypted sensor nodes requires a large number of sensor nodes, and related budgets, sensor nodes were prepared as a class. Figure 10 ① is the initial state of the setting, and ② is a case where the user arbitrarily set the data to be transmitted as "654242535325432" and selected RSA as an encryption setting; ③ is a case where an arbitrary data value was set as "748a1542" and AES was selected as an encryption setting.

Figure 11 exhibits an example of monitoring the state of sensing a situation where attacks on sensor nodes occurred. On the left side of Figure 11, is shown the sensing of a situation where malicious sensor nodes were set and arranged and the malicious sensor nodes attacked sensor nodes in short communication distances. In this case, the information on the sensor nodes attacked by the malicious sensor nodes is visually provided by red lines. The execution on the right side of Figure 11 provides sensing distances more visually than that on the left side and shows visualized battery consumption rates related to the damage caused by the attacking nodes. As shown in Figure 12, the index, the data value, the position, and the encrypted data of each sensor node can be monitored by using the mouse on the sensor node in order to monitor detailed information. Therefore, the user can observe damage according to expected positions of attacking nodes so that the user can actively respond to attacks.

6. Conclusion

EDTS implemented in this paper used information on the target topography from the user and the GML that can be mapped simultaneously to provide simulation environments similar to actual topography. Through the EDTS, visualized individuals therebymaximize the coverage to provide efficient methods of sensor arrangements and simulation functions. EDTS provided also views of mobile sensor nodes and user's autonomous inputs of position arrangements and enabled the judgment of whether sensor nodes have been efficiently arranged by checking battery lives of WSNs of sensor nodes configured based on the input of sensor information and the residual amounts of battery capacities in relation to events. By providing views of FSN, it drew sensor position arrangements efficient to users. In addition, users were enabled to set inputs of data during communications between sensor nodes. It also enabled more diverse simulations by allowing the selection of encryption and the setting of the length of encryptionto be encryption settings of entered data. If there are malicious sensor nodes among arranged sensor nodes, the damage can be observed. If WSNs are composed of the same sensor nodes, the geographical positions of malicious sensor nodes can be inferred based on communications so that the user can actively respond.

For the future research, when configuring WSNs based on EDTS, the precision of the grasping of the positions of malicious sensor nodes will be enhanced, and the functions will be added so that communication protocols currently in use can be applied for experiments under more diverse conditions. In addition, environments to determine the optimum protocol for the target topography under the applied communication protocol will be provided.

Acknowledgments

This paper was supported by the Global Science Experimental Data Hub Center/KISTI in 2013 and was also supported by the IT R&D Program of MKE/KEIT (10035708, "the development of cyber-physical systems (CPS) core technologies for high confidential autonomic control software").

References

[1] R. Arkin and K. Ali, "Integration of reactive and telerobotic control in multi-agent robotic systems," in *Proceedings of the 3rd International Conference on Simulation of Adaptive Behavior*, pp. 473–478, August 1994.

[2] X. Zhou, Y. Ge, X. Chen, Y. Jing, and W. Sun, "A distributed cache based reliable service execution and recovery approach in MANETs," *Journal of Convergence*, vol. 3, no. 1, pp. 5–12, 2012.

[3] A. U. Bandaranayake, V. Pandit, and D. P. Agrawal, "Indoor link quality comparison of IEEE 802.11a channels in a multi-radio mesh network testbed," *Journal of Information Processing Systems*, vol. 8, no. 1, pp. 1–20, 2012.

[4] S. Silas, K. Ezra, and E. B. Rajsingh, "A novel fault tolerant service selection framework for pervasive computing," *Human-Centric Computing and Information Sciences*, vol. 2, no. 5, pp. 1–14, 2012.

[5] GloMoSim, http://pcl.cs.ucla.edu/projects/glomosim.

[6] SNetSim, http://www.softpedia.com/get/Science-CAD/SNet-Sim.shtml.

[7] J. Polley, D. Blazakis, J. McGee, D. Rusk, and J. S. Baras, "ATEMU: a fine-grained sensor network simulator," in *Proceedings of the 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON '04)*, pp. 145–152, October 2004.

[8] Qualnet, http://web.scalable-networks.com/content/qualnet.

[9] "The Network Simulator—ns—2," http://www.isi.edu/nsnam/ns/.

[10] L. Girod, J. Elson, A. Cerpa, T. Stathopoulos, M. Lukac, and D. Estrin, "EmStar: a software environment for developing and deploying wireless sensor networks," in *Proceedings of the USENIX Technical Conference*, 2004.

[11] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: accurate and scalable simulation of entire TinyOS applications," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 126–137, November 2003.

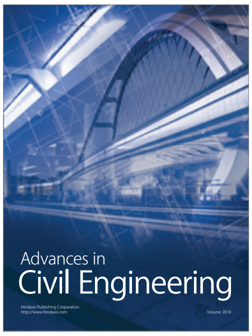
[12] "J-Sim: a simulation and emulation environment for wireless sensor networks," http://icserv.kjist.ac.kr/mis/publications/data/2006/01678171.pdf.

[13] B. L. Titzer, D. K. Lee, and J. Palsberg, "Avrora: scalable sensor network simulation with precise timing," in *Proceedings of the 4th International Conference on Information Processing in Sensor Networks (IPSN '05)*, pp. 477–482, Los Angeles, Calif, USA, April 2005.

[14] "Java in simulation time/scalable wireless Ad Hoc network simulator," http://jist.ece.cornell.edu/.

[15] G. Chen, J. Branch, M. J. Pflug, L. Zhu, and B. K. Szymanski, "SENSE: a wireless sensor network simulator," http://www.ita.cs.rpi.edu/publications/sense-book-chapter.pdf.

[16] OpenGIS Consortium, Inc., "Geography Markup Language [GML]," 07-36_Geography_Markup_ Language_GML_V3.2.1.pdf, http://portal.opengeospatial.org/files/?artifact_id=20509.



Hindawi

Submit your manuscripts at
http://www.hindawi.com