Malicious Node Detection in Wireless Sensor Networks Using an Autoregression Technique

Daniel-Ioan Curiac, Ovidiu Banias,
Florin Dragan, Constantin Volosencu
Politehnica University Timisoara, Romania
daniel.curiac@aut.upt.ro

Octavian Dranga
James Cook University
Townsville, Quensland, Australia
octavian.dranga@jcu.edu.au

Abstract

In this paper we propose a strategy based on past/present values provided by each sensor of a network for detecting their malicious activity. Basically, we will compare at each moment the sensor's output with its estimated value computed by an autoregressive predictor. In case the difference between the two values is higher then a chosen threshold, the sensor node becomes suspicious and a decision block is activated.

1. Introduction

Wireless sensor networks (WSN) are one of the most important and promising domains of the 21st century. Recent growth in technology (increased computational power, smaller chips and microprocessors with less power consumption) opened a new world for research in this field. A sensor network is a collection of small distributed devices called motes, using sensors for measurement (temperature, motion, pressure, sound) and for prediction (weather forecast, fire ignition, earthquakes, military attack, building safety). Some of the most important characteristics of wireless sensor networks are the environment free property and their capability of self organization during the entire life cycle.

Being deployed in any kind of environment, the motes are subject to attacks and without high security the information passing through the network could be listened to and also altered. By this, the wireless sensor network could be damaged and become useless for given purposes. Secure protocols are still under research, none was successful enough to be standardized. Although securing protocols is a very important issue in wireless sensor networks development, detecting anomalies and intruders is also a significant problem. After detection of intruders, the sensor network can take decisions to investigate, find, remove or rewrite malicious nodes if possible. If intruder detection is not made in appropriate time, captured nodes code could be read and rewritten for malicious purposes.

Depending on the chosen mote architecture (to be able, or not to be able to read and rewrite code), the wireless sensor network implementation and costs varies. If reading and rewriting of software is not needed for given network, than security for that network could be set very high using tamper proof hardware, leaving no chance of mote usage for malicious purposes in case of capturing by an attacker. Unfortunately this solution is not cost effective as long as the motes are dedicated for special use without possibility of rewriting software. On the other hand, being able to rewrite the software on motes leaves a wide-opened door for attacks. In this case, once an attacker captures one mote, he will read protocols, security keys, software code and will rewrite new software for malicious purposes. For these reasons, an important decision should be taken in order to choose the mote architecture, regarding the deployment environment and sensor network purposes. Avoiding corruption of the network through captured motes (by rewritten code) could find its solution in early detection of such motes with the immediate decision of expelling them from the network topology or in their recovery by rewriting original software. This paper presents such a methodology for malicious node detection by using an autoregression (AR) technique.

2. Intrusion detection in WSN

A wireless sensor network being a collection of motes deployed in an environment having the property of self organization is different from wired networks and even from elder brother ad hoc network. Intruder detection methods for wireless sensor networks should

Third International Conference on Networking and Services(ICNS'07)
0-7695-2858-9/07 $20.00 © 2007 IEEE

therefore take in consideration their restrictive architecture, with reference to the smaller computational power and energy consumption.

Solutions based on localization for anomaly detection using GPS receivers would visibly increase the cost per mote making it unviable [1]. Nevertheless GPS receivers planted on every mote will signal the changes in the motes position immediately to the rest of the network, making intruder detection fast and efficient. Unfortunately the costs are too high for such implementations. Intrusion detection in general splits into misuse detection and anomaly detection [2]. Misuse detection by identifying attacks by their known signatures doesn't need many resources, but has the drawback of unrecognizing unpublished attacks. Anomaly detection discovers published/unpublished attacks by comparing the known normal behavior of subject (sensor node) with its reported activities. If the subject's behavior differs from the expected or estimated one than the system has to report it. These kinds of models may need large resources for log and audit files, needing to be optimized for better usage with wireless sensor networks technology. By taken the decision to use base stations (laptop class devices) for anomaly detection these disadvantages can be minimized in order to provide an efficient solution for the problem like the one presented here based on AR prediction.

Intruder detection in wireless sensor networks should be signaled at the base station level (Fig. 1), where some decisions can be taken against it.

Even more malicious node detection techniques were proposed in the literature [3][4][5], but none of them could offer important results for different types of wireless network architectures and special cases. Previous work results prove that a general intrusion detection method for wireless networks is hard to implement due to the high restrictions, leaving space for dedicated methods with better application in particular cases. In this paper we propose an intruder detection method based on AR prediction technique.

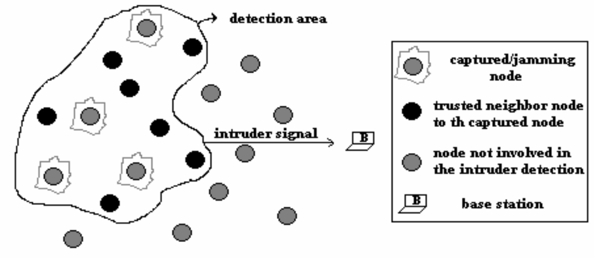
detection area
intruder signal
B
captured/jamming node
trusted neighbor node to th captured node
node not involved in the intruder detection
B
base station

Figure 1. Attacked sensor network

3. Malicious node detection strategy

Generally speaking, information security is an important emerging requirement that has to be considered in every phase of an IT&C product life-cycle: design, testing, deployment and operation. Because of the particularities of sensor networks, relying on security features developed during deployment and operation phases is not an option. Nodes may behave differently based on: final deployment position, remaining energy, proximity of other nodes, desired fidelity of sensor readings, commands from base stations. These factors are not predictable before deployment. Under these circumstances we have to build a strong secure architecture around sensor network starting from the designing phase of the network.

In order to assure a high grade of efficiency for our malicious node detection strategy we chose a topology for the sensor network having the following attributes:

a) The sensor network is static, i.e., sensor nodes are not mobile; each sensor node knows its own location [6] even if they were deployed via aerial scattering or by physical installation. If not, the nodes can obtain their own location through the location process described in [7]. Moreover, all the sensors passed a one-time authentication procedure done just after their deployment in the field.

b) The sensor nodes are similar in their computational and communication capabilities and power resources to the current generation sensor nodes, e.g. the Berkeley MICA2 motes. We assume that every node has space for storing up to hundreds of bytes of keying materials in order to secure the transfer of information through symmetric cryptography.

c) The base station, sometimes called access point, acting as a controller and as a key server, is assumed to be a laptop class device and supplied with long-lasting power. We also assume that the base station will not be compromised.

d) The measured values provided by each sensor present a strong deterministic component rather than a truly random (stochastic) one (e.g. wind speed or temperature measurements in different locations). In this case there exists a correlation between past values and the current one.

Because of their specific limitations, sensor networks are exposed to particular types of attacks, like eavesdropping, traffic analysis, spoofing, selective forwarding, sinkhole attack, wormhole attack, Sybil attack and Hello flood attack [5]. In order to remove the possibility to develop such types of attacks, we considered that our strategy has to rely on:

a) efficient secret-key cryptography with pre-distributed keys using Skipjack, RC5 or AES algorithms to encipher all data communications inside the sensor network [8][9]; All these three types of encryption algorithms have a common feature that makes them an attractive option in case of sensor networks: they are able to encrypt short or medium size messages, like the ones send by sensors and received by base stations, in the case of limited power consumption. By using such appropriate cryptographic techniques the harmful potential of the passive attacks (eavesdropping and traffic analysis) can be neglected.

b) an appropriate choosing of the sensor network topology based either on wireless cellular network (WCN) architecture [10] either, in case of large-scale sensor network, on SEnsor Network with Mobile Access (SENMA) architecture [11]. The main features of WCN and SENMA architectures are: nodes talk directly to base stations; there is no node-to-node communications and no multi-hop data transfer; sensor synchronism is not necessary; sensor do not listen, only transmit and only when polled for; complicated protocols are avoided; reliability of individual sensors is much less critical; system reconfiguration for mobile nodes is not necessary. These characteristics make attacks on routing protocols (spoofing, selective forwarding, sinkhole attack, wormhole attack, Sybil attack and Hello flood attack) almost impossible.

Probably the biggest threat for a wireless sensor network is node-capturing attack [12] where an adversary gains full control over sensor nodes through direct physical access. This type of attack is fundamentally different from the attacks already mentioned because it doesn’t rely on security holes in protocols, broadcasting, operating systems, etc. It is based on the geographic deployment of the sensor nodes in the field. Realistically, we cannot expect to control access to hundreds of nodes spread over several kilometers and, by this, we make a node capturing attack very possible. In addition, sensors are rarely tamper resistant, so an attacker can damage or replace sensors and computation hardware or extract sensitive material such as cryptographic keys to gain unrestricted access to higher levels of communication. Moreover, all sensors are usually assumed to run the same software, in particular, the same operating system. Finding an appropriate bug in the sensor network, through reverse engineering techniques applied to the captured sensor, allows the adversary to control the entire sensor network.

Our proposed countermeasure relies on the fact that a corrupted sensor node, even if it may still send authentic messages (e.g., it can use the cryptographic keys already stored in its memory), it may not work according to its original specifications sending erroneous readings to the base station. We will identify these sensors in the moment that they start to send wrong data by using a linear autoregressive predictor and will eliminate their malicious effect.

3.1. Autoregressive model

Our stratagem considers that an autoregressive (AR) model can efficiently approximate the time evolution of the measured values provided by each sensor. An autoregressive or AR model, also known as an infinite impulse response filter or all-pole model, describes the evolution of a variable measured over the same sample period as a linear function of only its past evolution. This kind of systems evolves due to its "memory", generating internal dynamics.

The AR model definition is as follows:

$$x(t) = a_1 \cdot x(t-1) + \ldots + a_n \cdot x(t-n) + \xi(t) \qquad (1)$$

where x(t) is the series under investigation (in our case is the series of values measured by the same sensor), a_i are the autoregression coefficients, n is the order of the autoregression and ξ is the noise which is almost always assumed to be a Gaussian white noise. By convention the time series x(t) is assumed to be zero mean. If not, another term (a_0) is added in the right member of equation (1).

If the a_i coefficients are time-varying the equation (1) can be rewritten as:

$$x(t) = a_1(t) \cdot x(t-1) + \ldots + a_n(t) \cdot x(t-n) + \xi(t). \qquad (2)$$

Based on model (2) we can either estimate the coefficients $a_i(t)$ in case the time series x(t),…, x(t-n) is known (recursive parameter estimation), either predict future value $\hat{x}(t)$ in case that $a_i(t)$ coefficients and past values x(t-1),…,x(t-n) are known (AR prediction).

3.2. Autoregressive prediction for malicious nodes detection

Our strategy uses the time series of measured data provided by each sensor and relies on an autoregressive predictor placed in base stations (Fig.2).

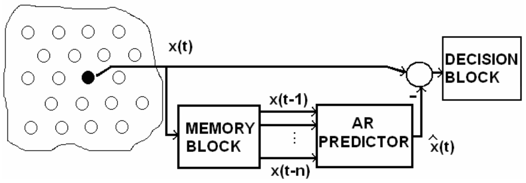

Figure 2. Malicious node detection strategy

The principle is the following: a malicious sensor node that will try to enter false information into the sensor network will be identified by comparing its output value $x(t)$ with the value $\hat{x}(t)$ predicted using past/present values provided by the same sensor. The proposed methodology is described as follows:

In the beginning we have to associate a threshold $\varepsilon > 0$ with every sensor node considered to be in the intrusion detection area (Fig.1). This threshold will be used to determine if a sensor acts normal or abnormal and its value depends on the type of the sensor and its specific functioning in-field conditions. For a specific sensor A, the threshold will be denoted by ε_A.

After this initialization, at every instant t we will compute the estimated value $\hat{x}_A(t)$ relying only on past values $x_A(t-1),\ldots, x_A(0)$ and we will use both parameter estimation and prediction as in the following steps:

First we will estimate the parameters $a_i(t)$ using a recursive parameter estimation method. There are a large number of methods for obtaining AR coefficients. The three main categories rely on: a)computing the autocorrelation estimates, where an important factor is the truncation threshold (maximum leg); b) calculating the partial autocorrelation (reflection) coefficients, where an important role is played by the specific definition of the reflection coefficient; and c) least-square matrix formulas. In our case we consider that a recursive least square method (RLS) is the most appropriate to solve this problem in an efficient manner since it produce the best spectral estimates. Taking into consideration that the basic RLS algorithm cannot be chosen due to its poor numerical properties and due to the demanding computational requirements, we decided to use a numerically robust RLS variant. This variant is known in literature as QRD-RLS and is based on orthogonal triangularization of the input data matrix via QR decomposition (QRD). QRD-RLS is developed around Givens transformations and can be implemented efficiently on the base stations level (laptop class device) [13].

Second, we will obtain the prediction value $\hat{x}(t)$ using the following equation:

$$\hat{x}_A(t) = a_1(t)\cdot x_A(t-1) + \ldots + a_n(t)\cdot x_A(t-n) + \xi(t). \quad (3)$$

After that, we will compare the present value $x_A(t)$ measured by the sensor node with its estimated value $\hat{x}_A(t)$ by computing the error:

$$e_A(t) = x_A(t) - \hat{x}_A(t). \quad (4)$$

If this error is higher than the threshold ε_A then the sensor A will be considered to be a potentially corrupted sensor and the decision block will be activated (Fig.2). Here, based on a data base containing the known attacks models, a knowledge-based system can take the decision to expel the malicious node from the network topology.

3.3. The selection of an appropriate AR model

There is no simple method to establish the correct model order in case of an AR model. In our case there are two parameters that influence our decision: the type of data measured by sensors and the computing limitations of the base stations. Because both of them are a priori known we propose the use of an off-line methodology presented in [14]. On the other hand, using the threshold ε_A in our methodology we minimize the risks of an inappropriate selection. Realistic values are between 3 and 6.

3.4. Autoregressive predictor as part of a complex intrusion detection system for WSN

The development of an efficient intrusion detection system for wireless sensor networks has to rely on different kinds of techniques in order to assure a desired level of security. From our point of view, a good choice is to combine the malicious node detection strategy presented below with an intrusion technique based on the values provided by neighboring sensors such as [15]. In this case, the decision taken on behalf of two criteria to eliminate the attacked sensors will be more accurate.

4. Case study

Let us consider the propagation of a temperature wave in a homogenous planar field where several sensor nodes $S_{i,j}$ with $i=1,\ldots,N$ and $j=1,\ldots,M$, being a part of a sensor network have been deployed. These sensors are measuring the local temperature $\theta[^{o}C]$. We also consider a possible malicious node denoted by S_A that has to be detected by using our strategy. For this reason we developed an autoregression model that estimates the temperature value provided by the sensor A, $\hat{x}_A(t) = \hat{\theta}_A(t)$ by taking into consideration the previous values of the data provided by sensor $x_A(t-1)$, $x_A(t-2),\ldots,x_A(t-n)$, with n chosen correlated to the consideration made above.

We presume the time distribution of the temperature θ through the homogenous medium in space to be:

$$\theta = \theta(z,t) \qquad (5)$$

where $\theta(z,t)$ is the temperature at the moment t, at distance z from the heat source. The heat conduction, when neglecting the heat loses in the environment, is described by the heat equation [16]:

$$c_\theta \frac{\partial^2}{\partial z^2}\theta(z,t) = \frac{\partial}{\partial t}\theta(z,t) \qquad (6)$$

where c_θ is the heat conductivity coefficient of the medium.

In order to investigate how our strategy works, we have discretised the function $\theta = \theta(z,t)$ into the aggregates $\theta_{j,k}$ (temperature value provided by $S_{j,k}$) situated at the distance $z_{j,k}$ from the origin. Our goal is to obtain the temperature θ_A measured by the corresponding sensor (S_A).

The energy conservation is governed for each point in the field by the following equation:

$$\frac{d}{dt}W_{j,k} = P_{in}^{j,k} - P_{out}^{j,k} \qquad (7)$$

where $W_{j,k}$ is the energy stored in point (j,k), $P_{in}^{j,k}$ is the input power in the point and $P_{out}^{j,k}$ is the output power from that point. The space model of the sensors deployed in the field with the heat sources is presented in Fig. 3. The sensor S_A measures the temperature θ_A in a point in this space. Let the heat capacity of each point be denoted C and the heat transfer coefficient between the points $K_i^{j,k}$. These give the equation in time of the heat diffusion:

$$\frac{d}{dt}C\theta^{j,k}(t) = \sum_{i1} K_{i1}^{j,k}[\theta_{i1}^{j,k}(t) - \theta^{j,k}(t)] - \sum_{i2} K_{i2}^{j,k}[\theta_{i2}^{j,k}(t) - \theta^{j,k}(t)] \qquad (8)$$

A discrete time equivalent equation of (8), with a chosen adequate sample period h is used. We consider, for example, that each cell of sensors may receive inputs from the around medium, from r sources with powers P_i, i=1,…,r, positioned around the network. The heat sources P_i are positioned in different points in the coordinate system xOy. Some coordinate transformations may be done and the sources may be moved in the adjacent points of the network.

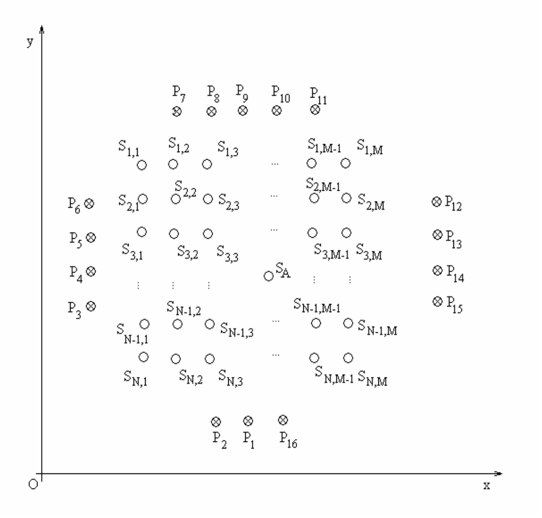
y
x
O
P_7 P_8 P_9 P_{10} P_{11}
$S_{1,1}$ $S_{1,2}$ $S_{1,3}$ $S_{1,M-1}$ $S_{1,M}$
P_6 $S_{2,1}$ $S_{2,2}$ $S_{2,3}$ $S_{2,M-1}$ $S_{2,M}$ P_{12}
P_5 $S_{3,1}$ $S_{3,2}$ $S_{3,3}$ $S_{3,M-1}$ $S_{3,M}$ P_{13}
P_4 S_A P_{14}
P_3 $S_{N-1,1}$ $S_{N-1,2}$ $S_{N-1,3}$ $S_{N-1,M-1}$ $S_{N-1,M}$ P_{15}
$S_{N,1}$ $S_{N,2}$ $S_{N,3}$ $S_{N,M-1}$ $S_{N,M}$
P_2 P_1 P_{16}

Figure 3. The deployment of the sensors

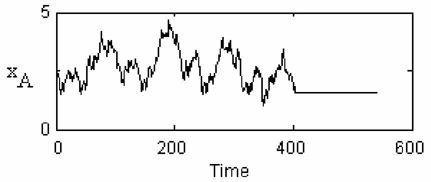

Figure 4. The sensor's output time series

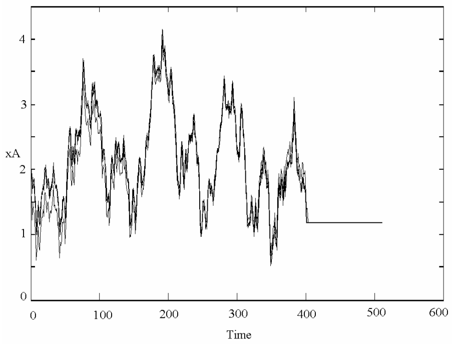

Figure 5. The two time series: the measured one and the predicted one

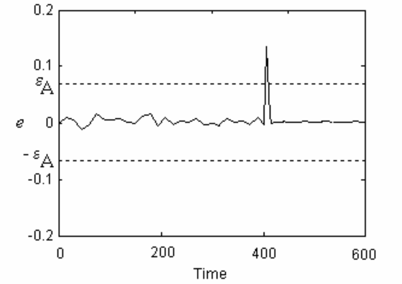

Figure 6. The time evolution of the error $e_A(t)$

0-7695-2858-9/07 $20.00 © 2007 IEEE

The procedure, used by us to determine a model of the dynamical system from observed input-output data involves the following ingredients:

1. The input data for estimation is a time series of the temperature sensor S_A, as the state of the heat diffusion model (8). This time series is obtained using a sum of the traveling temperature waves, for this example on the North side of the network, generated by the heat sources $P_7,\ldots,P_{11}$. The temperature is propagated through the sensor network to the sensor S_A.

2. At a specific time moment (t=400) we considered that the sensor was corrupted. The time series of the sensor output in this case is presented in Fig. 4.

3. Some different sets of candidate models for the model structure could be experimented. We have chosen a 4th order AR model.

This autoregressive estimation is applied for the sensor S_A by using the times series from Fig.4. The estimated temperature $\hat{x}_A(t) = \hat{\theta}_A(t)$ for the sensor S_A is presented in Fig.5, over the original time series $x_A(t)$ and in Fig.6 it is presented the error $e_A(t)$.

We may see that the error increases at the corruption time moment. The maximum error value is $e_{Max}(t)=e(402)=0.16$ being above the threshold value considered to be $\varepsilon_A=0.075$ will trigger the decision block to expel the sensor from the network.

5. Conclusions

One of the important problems that are related to the use of wireless sensor networks in harsh environments is the gap in their security. This paper provides a solution to discover any malicious nodes in wireless sensor networks using an autoregressive predictor based on past values obtained from the same nodes. This solution can be also a way to discover the malfunctioning nodes that were not a subject of an attack. Being localized on the base station level, our algorithm is suitable for large-scale sensor networks.

6. References

[1] W. Du, L. Fang and P. Ning, “LAD: Localization Anomaly Detection for Wireless Sensor Networks”, *Journal of Parallel and Distributed Computing (JPDC)*, Volume 66, Issue 7, July 2006, pp. 874-886.

[2] D.K. Kang, D. Fuller and V. Honavar, “Learning Classifers for Misuse and Anomaly Detection Using a Bag of System Calls Representation”, *Proceedings of the 2005 IEEE Workshop on Information Assurance and Security*, United States Military Academy, West Point, NY 2005, pp. 118-125

[3] S. Doumit and D.P. Agrawal, “Self-Organized Criticality & Stochastic Learning Based Intrusion Detection System for Wireless Sensor Networks”, MILCOM 2003, pp.609-614.

[4] P. Albers, O. Camp, J. Percher, B. Jouga, L. Me and R. Puttini, “Security in ad hoc Networks: A General Intrusion Detection Architecture Enhancing Trust Based Approaches”, *WIS’02 Proceedings*, April 2002, pp.1-12.

[5] C. Karlof and D. Wagner, “Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures”, *Elsevier's AdHoc Networks Journal*, 1(2-3), 2003, pp. 293-315.

[6] T. He, C. Huang, B.Blum, J.A. Stankovic, T. Abdelzaher, “Range-Free Localization Schemes in Large-Scale Sensor Networks”, *MOBICOM 2003*, San Diego, USA, pp. 81-95.

[7] A. Savvides, C.C. Han and M.B. Srivastava, “Dynamic fine-grained localization in ad-hoc networks of sensors”, *In Proc. 7th ACM MobiCom*, Rome, Italy, 2001, pp. 166-179.

[8] C. Karlof, N. Sastry and D.Wagner, “TinySec: A Link Layer Security Architecture forWireless Sensor Networks”, *ACM Sensys2004 Proc.*, Baltimore, USA, pp.162–175.

.

[9] A. Vitaletti and G.Palombizio, “Rijndael for sensor networks: is speed the main issue?”, *WCAN 2006*, Venice, Italy, July 2006.

[10] J. Feng, F. Koushanfar and M. Potkonjak, “System-Architectures for Sensor Networks Issues, Alternatives, and Directions”, *Proc. ICCD’02*, Freiburg, Germany, Sept. 2002, pp.226-231.

[11] L. Tong, Q. Zhao and S. Adireddy, “Sensor Networks with Mobile Agents", *Proceedings IEEE 2003 MILCOM*, Boston, USA, October 2003, pp.688-694.

[12] A. Becher, Z. Benenson and M. Dornseif, “Tampering with motes: Real-world physical attacks on wireless sensor networks”, *SPC Proc.*, York, UK, April 2006, pp.104-118.

[13] B. Haller, J. Gotze and J. Cavallaro, “Efficient Implementation of Rotation Operations for High Performance QRD-RLS Filtering”, *ASAP '97 Proc.*, 14-16 July 1997, Zurich, Switzerland, pp. 162-174.

[14] P. Stoica and Y. Selen, “Model-order selection: a review of information criterion rules”, *IEEE Signal Processing Mag.*, July 2004, pp.36-47.

[15] D.I. Curiac, C. Volosencu, A. Doboli, O. Dranga and T. Bednarz, “Discovery of Malicious Nodes in Wireless Sensor Networks using Neural Predictors”, *WSEAS Transactions on Computer Research*, Issue 1, Volume 2, January 2007, pp. 38-43.

[16] D. Basmadjian, “The Art of Modeling in Science and Engineering”, *Chapman & Hall, CRC*, Boca Raton, 1999.