KNOWLEDGE BASED SYSTEM FOR RELIABLE PERIMETER PROTECTION USING SENSOR NETWORKS

Constantin Voloşencu, Daniel-Ioan Curiac, Ovidiu Banias
Automatics and Applied Informatics Department, "Politehnica" University of Timisoara
Bd. V. Parvan nr. 2, 300223 Timisoara, Romania

Alex Doboli
Electrical and Computer Engineering Department, State University of New York, NY 11794-2350, USA

Octavian Dranga
School of Engineering, James Cook University, Townsville, QLD 4811, Australia

Keywords: Perimeter protection, wireless sensor networks, redundancy, knowledge based systems, neural networks, perceptron.

Abstract: This paper provides a strategy for perimeter protection using sensor networks with hardware and analytical redundancy. The sensor network reliability is augmented using a knowledge-based system, which implicates the analysis of the trustworthiness of each sensor. For this, we used two stratagems: one that relies on hardware redundancy based on the Confidence Weighted Voting Algorithm and one that relies on analytical redundancy based on a neural perceptron predictor that uses past and present values obtained from neighbouring nodes. This solution can be also a way to discover the malfunctioning nodes that were subjects of an attack and it is localized at the base station level being suitable even for large-scale sensor networks.

1 INTRODUCTION

Sensor networks have proved their huge viability in the real world in a variety of domains. Advances in miniaturization, decreasing of their cost and power and improvements in wireless networking and micro-electro-mechanical systems have led to research for large-scale deployment of wireless sensor networks and formation of a new computing domain. In the last years the deployment of small-scale sensor networks in support of a growing array of applications has become possible (Akyildiz, 2002), (Pottie, 2000). A lot of applications, including seismic disturbances, contaminant flows and other ecological or environmental disasters, battlefield control, disaster management and emergency response, which involve sensor networks, will also be possible in the near future. Detecting targets moving inside a field of interest is one of the applications of wireless sensor networks (Li, 2002), (Cao, 2005). These networks consist of hundreds or thousands of heterogeneous disposable sensor nodes, capable of sensing their environment and communicating with each other via wireless channels, coordinating and monitoring large areas. Individually nodes possess properties such as functionality and inter-node cooperation, under limited energy reserves and technological limitations. There are applications where the sensors were generally bulky devices wired to a central control unit whose role was to collect, process, and act upon the data gathered by individual sensors. A network of sensors could be developed with small motion detectors, metal detectors, pressure detectors, and vibration detectors, deployed around a valuable asset. When the sensors were able to classify "intruders", a human reasoning to decide what to do in response was necessary. The vision of the smart dust program of wireless sensor network research was to make machines with self-contained sensing, computing, transmitting, and powering capabilities so small and inexpensive that they could be released into the environment in massive numbers. Sensor

Voloşencu C., Curiac D., Banias O., Doboli A. and Dranga O. (2007).
KNOWLEDGE BASED SYSTEM FOR RELIABLE PERIMETER PROTECTION USING SENSOR NETWORKS.
In *Proceedings of the Second International Conference on Wireless Information Networks and Systems*, pages 51-56
DOI: 10.5220/0002146500510056

networks are expected to play an important role in hybrid protection infrastructures when combined with robots and human decision makers. In such cases a knowledge-based system is a powerful way of solving the problems. Redundancy in sensor networks (hardware and analytical) can provide higher monitoring quality (Gao, 2003) by employing the adjacent nodes in order to discern the rightness of local data. When a sensor malfunction appears and the hardware redundancy is lost, the problems can be solved using the analytical redundancy. Redundancy increases data accuracy, system reliability and sensor network security to provide protection against service interruptions.

The rest of the paper is organized as follows. Section 2 presents the related work in the domain. Section 3 contains our strategy for perimeter protection. Section 4 describes a case study for our security strategy. Section 5 presents the conclusions.

2 RELATED WORK

There is relatively little work in the area of securing sensor networks based on redundancy. A useful survey for initiation in the domain of wireless sensor network is presented in (Akyildiz, 2002).

In (Nowak, 2003) a technique for edge detection of a phenomenon within a wireless sensor network is proposed. The approach involves a hierarchical processing strategy, where nodes collaborate, into a non-uniform rectangle, adapted to the phenomenon partition of the sensor field.

Research into authentication and confidentiality mechanisms of sensor network protocols have been started in order to identify the problems and to propose technical solutions (Avancha, 2003), (Intanagonwiwat, 2000). Some threats to these applications were identified and a security model operating on the base station level was proposed. The application mentioned requires mitigation against traffic analysis, relying solely on broadcasts of end-to-end encrypted packets. Nodes adjacent to the base station are utilised as intermediary hops. The model corrects some classes of aberrant node behaviour.

Using wireless sensors networks for tracking moving objects is discussed in (Cao, 2005), where an analysis of their performances is done. The authors provide analytic formulae for the mean delay until a target is detected, when moving on a straight line at a constant speed. The authors consider a system model where sensors are randomly distributed within a field of interest, with each sensor having identical sensing areas that follow the unit disk model.

In (Clouqueur, 2002) the authors propose collaborative detection models, where sensors collectively arrive at a consensus about the presence of a target. Sensors are assumed to be randomly deployed within the field of interest and the sensing capability of each sensor is assumed to decay with distance, with all sensors having identical sensing areas. They formulate the target detection problem as an unauthorized traversal problem and propose deployment strategies for minimizing the cost of the network that achieves the desired target detection probability.

These highly localized results of redundancy in sensor networks can be aggregated by methods such as (Xu, 2001) to provide higher data reliability for requesting applications such as event/target detection (Li, 2002), (Brooks, 2003).

In (Aslam, 2005), a network with binary sensors is used for tracking a moving object. This is an elementary case for our solution of using a perceptron as the model for a binary sensor network.

3 PROPOSED STRATEGY

3.1 Sensor Network Assumptions

We make the following assumptions related to the sensor network:

a) The sensor network is static, i.e., sensor nodes are not mobile; each sensor node knows its own location.

b) The sensor nodes are similar in their computational and communication capabilities and power resources to the current generation sensor nodes. Moreover, because they have to sense if an intruder is in their proximity, they can provide only two values, which we assumed to be 0 (for inexistence of an intruder in their proximity) and 1 (for existence of an intruder in their proximity).

c) We rely on wireless cellular network architecture (Feng, 2002). In this architecture, a number of base stations have already been deployed within the field. Each base station forms a cell around itself that covers part of the area. Mobile wireless nodes and other appliances can communicate wirelessly, as long as they are within the area covered by one cell.

d) 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.

With the purpose of solving the problem of a reliable perimeter protection, we rely on two very important properties: a) inherent redundancy, which is an important natural feature of sensor networks; and b) the determinism of the measured values provided by sensors related to their past recordings.

3.2 Redundancy in Sensor Networks and Its Benefits

One important natural feature of the sensor networks employed by our strategy is the inherent redundancy. We use both hardware and analytical redundancy in order to increase the reliability of our perimeter protection approach.

Hardware (physical) redundancy ensures the reliability in sensor networks (Gao, 2003), (Clouqueur, 2001) and implies the use of supplementary sensors (deployed in the field, due to the necessity of covering the area in case of malfunctioning of some sensor nodes) and selection of data that appears similarly on the majority of sensors.

Analytical (functional) redundancy is based on the determinism of the measured values provided by sensors. The information from different sensors is built on the fact that actual sensor value is related with past values provided by the same sensor. The use of analytical redundancy is done through a process of comparison between the actual sensor value and the expected/estimated sensor value. This approach is based on a mathematical model that can predict the value of one sensor by taking into consideration the past and present values of neighbouring sensors or of the implied sensor itself. The computation implied in this approach is done at the base station level (laptop class device), where all requirements are satisfied.

3.3 Knowledge Based System for Reliability Improvement

Our strategy to improve the reliability of the data provided by the perimeter protection sensor network relies on the knowledge-based system (KBS) presented in figure 1, which contains four components: a) Confidence Weighted Voting (CWV) Block, b) Neural Network Block, c) Knowledge Base Block, and d) Decision Block.

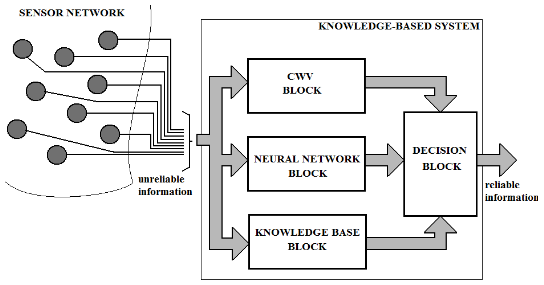

Figure 1: Knowledge-based system architecture.

3.3.1 Confidence Weighted Voting Block

This component relies on hardware redundancy and is based on a variant of Majority Voting algorithm (MV) known as Confidence Weighted Voting (CWV) (Sun, 2005). This algorithm gives higher weights to those sensors that are more likely to be correct (i.e. with higher confidence of correctness). The confidence value of each sensor can be determined in a distributed manner by comparing its sensing results with its sensing neighbours that share overlapping coverage area. The confidence value of sensor *i* is then defined as:

$$conf(i) = \frac{\sum_{j=1}^{m} \Delta_{ij} \cdot A_{ij}}{\sum_{j=1}^{m} A_{ij}} \tag{1}$$

where m represents the total number of the sensors within the sensor network,

$$\Delta_{ij} = \begin{cases} 0; \text{ if sensor i and j report different results} \\ 1; \text{if sensor i and j report the same result} \end{cases} \tag{2}$$

and

$$A_{ij} = \begin{cases} 0; \text{if the coverage of sensor i and j is not overlappec} \\ 1; \text{if the coverage of sensor i and j is overlapped} \end{cases} \tag{3}$$

The reliable value, obtained using CWV algorithm for sensor S, having the in-field position represented in Cartesian coordinates by the pair (x,y), is the value $k \in \{0;1\}$ corresponding to:

$$\mathrm{CWV}(x, y) = \max_{k} \sum_{j=1}^{m} \mathrm{conf}(j) \delta_{kj} C_j(x, y) \tag{4}$$

where

$$\delta_{kj} = \begin{cases} 0; \text{ if the report value from sensor j is not k} \\ 1; \text{if the report value from sensor j is k} \end{cases} \tag{5}$$

$$C_j(x, y) = \begin{cases} 0; \text{ if point (x, y) isn't covered by sensor j} \\ 1; \text{ if point (x, y) is covered by sensor j} \end{cases} \tag{6}$$

This CWV Block is an active block in our strategy only for sensors included in the coverage zones of other neighbouring sensors, for example sensor B from figure 2.

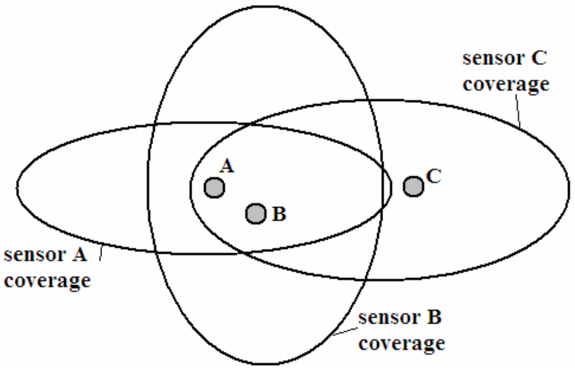

Figure 2: Sensor coverage diagram.

3.3.2 Neural Network Block

In order to assure a higher reliability for the information provided by the sensor network, even in the case of low hardware redundancy, we developed a neural network structure that provides an estimated value for each sensor, based on the past values provided by adjacent sensors. This estimate is compared with the actual sensor value deciding if this actual value is reliable or not. The neural network is based on a perceptron, with a number of binary neurons equal to the number of the network sensors. The sensor network is perceived in a static and also a dynamical way. Each neuron is considered as a binary model for a sensor. It receives at the sampling moments the weighted and biased iterative values of the adjacent sensors (neurons) and computes the estimates. A relevant architecture of this block is depicted in section 4.

3.3.3 Knowledge Base Block

Based on some assumptions and on past in-field data concretised in valuable rules, a knowledge base is established. This knowledge base includes information like: a) possible values of intruder's speed in the sense that detecting an intruder with a speed higher than a limit value must not be considered; b) the impossibility for an intruder to be detected by an inside sensor until the intruder's detection by an outside sensor has been reported. This knowledge base is used only for validation of the results provided by CWV and Neural Network Block.

3.3.4 Decision Block

The Decision Block is implemented in our strategy by the following pseudo-code:

```
For (every moment t and every sensor S)
do
{
   /* follows the implementation of
   /* Neural Network Block
   Result1(S,t)=ComputeNNB(past values
               of neighbouring sensors)
   If (hardware redundancy is present)
   then
     {
      /* follows the implementation of
      /* CWV Block
      Result2(S,t)=ComputeCWV(actual
              values from the sensors)
      ReliableResult(S,t)=Validate1(
           Result1(S,t),Result2(S,t),
           rules from Knowledge Base)
     }
   else
    {
     ReliableResult(S,t)=Validate2(
        Result1(S,t),
        rules from Knowledge Base)
     }
}
```

4 CASE STUDY

In this section, static and dynamical models for the sensor network are proposed, based on the possible trajectories of a strange object between sensors. A basic structure of the perceptron implementing the static and dynamical models of the sensor network is developed, trained and tested.

Let us consider a field of interest with NxM binary sensors for perimeter protection. Each sensor S_A from the field has other 8 adjacent sensors $S_{adj,i}$, i=1,…,8, as it is illustrated in figure 3. A cell with 9 sensors is taken into consideration.

A strange object could pass through the cell by many different directions, each with two senses: Di,j, i=1,…16, j=1, 2 and combinations of them.

The static model for the sensor network is illustrated as follows. If an object is situated in a point P_i, at an intersection of two directions, a set of sensor value results: $S_{Pi}=[S_{adj1}, S_{adj2}, S_{adj3}, S_{adj4}, S_A, S_{adj5}, S_{adj6}, S_{adj7}, S_{adj8}]_i$. For example, if the object is in the point of intersection between directions D3 and D6 the set of sensor values is [0, 0, 0, 1, 1, 0, 1, 1, 0]. At each intersection of two directions four adjacent sensors of the intersection are activated, based on the hardware redundancy. A table of sensor value sets is created for all points of intersections.

The dynamical model of the sensor network is illustrated as follows. The values of the sensors are available at the sample times S(kh), where h is the sample period. When a strange object passes trough the network the sensors are activated one after another. So, for a dynamical description of object movement between sensors, a train of impulses results. As an example, in figure 4 we represented the impulse train for trajectory D3,1-D7,1-D4,2.

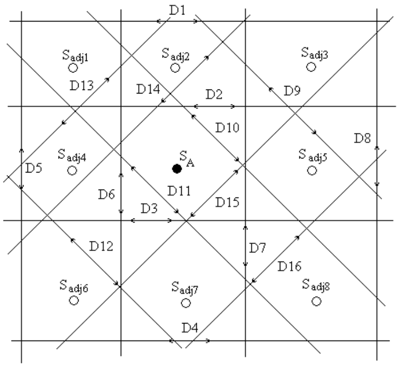

Figure 3: The cell structure.

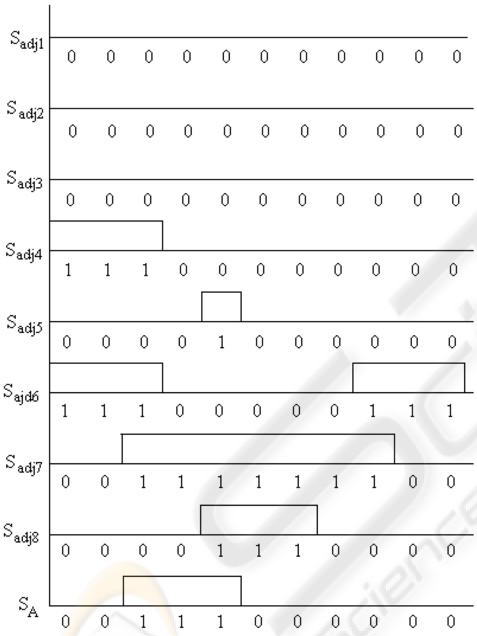

Figure 4: Examples of sensor impulse trains.

For a binary sensor modelling we use a neuron with two values 0 and 1. The neuron is trained to learn the impulse trains for all the possible trajectories between sensors. The sensor network may be modelled as a perceptron with N×M binary neurons, applying at the neuron inputs the measured values from the adjacent sensors.

The structure of the neuron for the dynamical model is presented in figure 5,

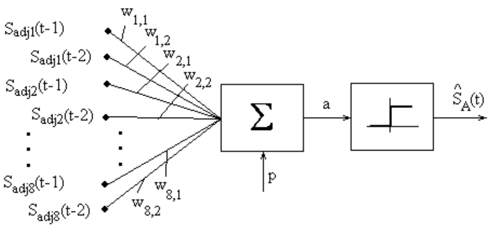

Figure 5: Neuron structure for the dynamical model.

where:

$$\hat{S}_A = f_A(a)$$

$$a = \sum_{i=1}^{8}\sum_{j=1}^{2} w_{i,j}S_{adj-i}(t-j) + p \qquad (7)$$

For the static position the following relation defines the neural model:

$$a = \sum_{i=1}^{8} w_i S_{adj-i} + p \qquad (8)$$

The neuron is using the hard-limit transfer function $f_A(a)$, which returns 0 or 1. Each input $S_{adj-i}(t-j)$ is weighted with an appropriate weight $w_{i,j}$, i=1,…8, j=1, 2. The sum *a* of the weighted inputs is sent to the hard-limit transfer function $f_A(a)$, which also has an input with a value equal to 1, biased by p. The neuron produces a result, based on the measured values provided by its adjacent sensors. The hard-limit transfer function gives the perceptron the ability to classify input vectors by dividing the input space into regions. Specifically, outputs will be 0 if the net input *a* is less than 0, or 1 if the net input *a* is 0 or greater.

We can estimate the value of the sensor S_A at the moment t, based on the measured values of the adjacent sensors at the previous two time moments (t-1) and (t-2).

A supervised learning rule is used as a procedure to modify the appropriate values of the weights w and bias p of the perceptron (Hagan, 1996). The training of the perceptron is made on all possible trajectories through the sensor network, the behaviour being summarized by a set of input-output pairs (u;y) = ($S_{adj,1}$, … , $S_{adj,8}$; $S_A(t)$). The corresponding target y of the perceptron is formed by the values of the sensor S_A. The objective of the neural network training is to reduce the error ε, which is the difference between the target vector and the neuron response (the estimate):

$$\varepsilon = S_A - \hat{S}_A \qquad (9)$$

The desired changes to the perceptron's weights Δw and bias Δp are calculated, given an input vector u and the associated training error ε:

$$w^{new} = w^{old} + \varepsilon u$$
$$p^{new} = p^{old} + \varepsilon \qquad (10)$$

The above perceptron rule is proven to converge on a solution in a finite number of iterations.

The error obtained after iterative trainings is presented in figure 6.

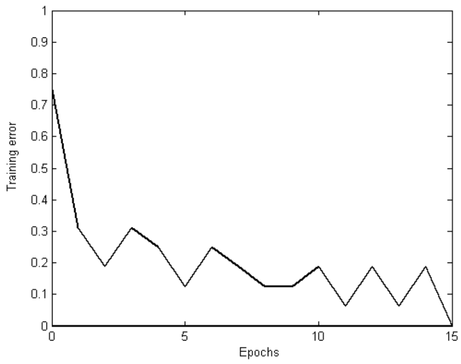

Figure 6: The training error.

The neural network was tested with impulse trains as test sets. The output accurately estimates the impulse trains for simulated trajectories.

An important result is that the neural network could be generalized for different possible trajectories. If the sensor node A is attacked, it is possible for its output value S_{Ac} to be different from the estimate. So, the sensor's estimated value, predicted by the neural network, differs from the actual value of the malicious sensor A, proving that something wrong happened to sensor A. In these circumstances, the decision block will exclude the sensor A from the network.

5 CONCLUSION

The goal of our research was to design a secure architecture for a sensor network used for perimeter protection. For this, we used a knowledge-based system based on hardware and analytical redundancy. Considering the detection of anomalies and intruders in binary sensor networks to be a very important issue, we relied on two coupled stratagems: a) a CWV based algorithm; and b) a perceptron predictor based on the past values of neighbouring sensors to solve this problem. After detection, the sensor network can take decisions to investigate, find and remove malicious nodes if possible. Being localized on a base station level, with a reduced amount of computation our method is suitable even for large-scale sensor networks.

REFERENCES

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002. Wireless Sensor Networks: A Survey. In *Computer Networks*, 38(4), March.

Aslam, J., Butler, Z., Constantin, F., Crespi, V., Cybenko, G., Rus, D., 2005. Tracking a Moving Object with a Binary Sensor Network. In *Proceeding of the 1st International Conference on Embedded Networked Sensor Systems*.

Avancha, S., Undercoffer, J., Joshi, A., Pinkston, J., 2003. Secure sensor networks for perimeter protection. In *Computer Networks* 43, Elsevier Press.

Brooks, R., Ramanathan, P., Sayeed, A., 2003. Distributed target classification and tracking in sensor networks, In *Proceedings of the IEEE*, vol. 91, no. 8.

Cao, Q., Yan, T., Abdelzaher, T., Stankovic, J., 2005. Analysis of Target Detection Performance for Wireless Sensor Networks, In *Proceeding of the International Conference on Distributed Computing in Sensor Networks*, CA.

Clouqueur, T., Ramanathan, P, Saluja, K.K., Wang., K.C., 2001. Value fusion versus decision-fusion for fault tolerance in collaborative target detection in sensor networks, In *Proceedings of Fusion 2001*, Montreal.

Gao, Y., Wu, K., Li, F., 2003. Analysis on the Redundancy of Wireless Sensor Networks, In *ACM WSNA Proceedings*, San Diego, USA.

Feng J., Koushanfar F., Potkonjak M., 2002, System-Architectures for Sensor Networks Issues, Alternatives, and Directions, In *Proceedings. of the 2002 IEEE International Conference on Computer Design (ICCD'02)*, Freiburg, Germany.

Hagan, M.T.; Demuth, H.B.; Beale, M.H., 1996. *Neural Network Design*, PWS Publishing, Boston.

Intanagonwiwat, C., Govindan, R., Estrin, D., 2000. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In *ACM International Conference on Mobile Computing and Networking* (MOBICOM'00).

Li, D., Wong, K., Hu, Y.H., Sayeed, A.M., 2002. Detection, classification, and tracking of targets in distributed sensor networks. In *Signal Processing Magazine,* IEEE, Vol. 19, No. 2.

Nowak, R., Mitra, U., 2003. Boundary Estimation in Sensor Networks: Theory and Methods. In *Proceedings of the First International Workshop on Information Processing in Sensor Networks*.

Pottie G.J., W.J. Kaiser, W.J., 2000. Wireless Integrated Network Sensors. In *Communications of the ACM*, vol. 43.

Sun, T., Chen, L.-J., Han, C-C., Gerla, M., 2005, *Reliable Sensor Networks for Planet Exploration*, ICNSC.

Xu, Y., Heidemann J., Estrin, D., 2001. Georgraph-informed Energy Conservation for Ad Hoc Routing, In *Proceeding of ACM Mobicom*, Rome, Italy.