A self-healing mobile wireless sensor network using predictive reasoning

Matthew Coles, Djamel Azzi and Barry Haynes
Electronic and Computer Engineering, University of Portsmouth, Portsmouth, UK

Abstract

Purpose – The paper aims to investigate performance benefits associated with adopting a mobile wireless sensor network (WSN). Sensor nodes are generally energy constrained due to the latter being acquired from onboard battery cells. If one or more sensor nodes fail, possible coverage holes may be created which could invariantly lead to a reduced network lifetime. The paper proposes that instead of rendering the entire WSN inoperative, sensor nodes should physically change position within the region of interest thus adaptively altering the WSN topology with a view of recovering from failures. This type of motion will be referred to as "self healing".

Design/methodology/approach – This paper presents a mobility scheme based on Bayesian networks for predictive reasoning (BayesMob) which is essentially a distributed self healing algorithm for coordinating physical relocation of sensor nodes. Using the algorithm, sensor nodes can predict the performance of the WSN in terms of coverage given that the node moves in a given direction. The evidence for this hypothesis is acquired from local neighborhood information.

Findings – The paper compares BayesMob with an alternative algorithm – Coverage Fidelity Algorithm – and shows that BayesMob maintains a higher level WSN coverage for a greater percentage of failures, thus increasing the useful lifetime of the WSN.

Research limitations/implications – The physical relocation of sensor nodes will incur energy overhead, therefore the tradeoffs between all application criteria should be investigated before implementation.

Originality/value – This paper presents a Bayesian network based motion coordination algorithm for WSN which repairs coverage holes caused by energy exhaustion and/or abrupt node failures.

Keywords Wireless, Sensors, Radio networks, Predictive process, Programming and algorithm theory

Paper type Research paper

1. Introduction

Advances in sensor technology (in terms of size, power consumption, wireless communication and manufacturing costs) have enabled the prospect of deploying large quantities of sensor nodes to form a wireless sensor network (WSN). These networks are created by distributing large quantities of usually small, inexpensive sensor nodes over a geographical region of interest with a view to collect data relating to one or more variables. These nodes are primarily equipped with the means to sense, process and communicate data to other nodes and ultimately to a remote user(s). Sensor nodes may cooperate with their neighbors (within communication range) to form an *ad hoc* network. WSN topologies are generally dynamic and decentralized. Sensor nodes can also have mobility capabilities which enable them to physically relocate with relation to neighboring nodes and the environment in which they are situated. WSNs have a wide range of applications including military, environmental monitoring, health, home, space exploration, chemical processing, and disaster relief (Akyildiz *et al.*, 2002).

The proposed application environments may be dynamic and the network designer may have limited knowledge of the region of interest. Also some of the proposed applications environment may be unmanned/unexplored terrain. In most cases the network designer would have little control over the exact deployment configuration. Such scenarios may include deployment via air drop.

The WSN deployment configuration is crucial to the network satisfying the performance criteria and operational lifetime. Even if the sensor nodes are deployed uniformly across the region of interest as time passes, sensor nodes may fail randomly due to energy exhaustion, malfunction or malicious destruction. Non-uniform traffic distribution and edge effects will directly influence the energy usage of the sensor nodes. The cumulative result of these factors may cause coverage holes and possibly detach a segment of the WSN. The implication of these failures may result in the WSN performance deteriorating thus preventing the performance criteria from being met. The net result of these failures is a reduced useful lifetime.

A proactive method of efficiently balancing the sensor nodes energy discharge rate (EDR) is to place nodes and assign tasks such that coverage holes are never formed in the WSN. This method may give an optimal solution but this approach to deployment is impractical for WSNs, as the

The current issue and full text archive of this journal is available at
www.emeraldinsight.com/0260-2288.htm

Sensor Review
28/4 (2008) 326–333

[DOI 10.1108/02602280810902631]

network designer would require a comprehensive knowledge of the application environment (Ganeriwal *et al.*, 2004). Generally when considering the application environment this information is unavailable. Many of the foreseen applications are within regions where human intervention is not always possible. In such situations deployment is random and the network designer has limited influence over the exact node placement.

If coverage holes appear the WSN can be rendered inoperative and the remaining active sensor nodes would be wasted. We propose that when such holes are created the WSN topology is reorganized via physically relocating surrounding sensor nodes to repair the coverage hole. Therefore, the remaining resources will be utilized via a motion control algorithm and thus extend the useful lifetime of the WSN. This motion algorithm is deemed as self healing. Figure 1 shows the self healing mechanism, where sensor nodes physically move to repair coverage holes.

In this paper, the performance criterion considered is sensing coverage, which is defined as the fraction of the total intended area actually covered by the WSN (Meguerdichian *et al.*, 2001). We propose an algorithm referred to as Bayesian network mobility (BayesMob) that uses mobility as an adaptive actuation facility for automated deployment and repair of the WSN with the sole objective of salvaging lost coverage. BayesMob incorporates a discrete Bayesian network which enables a sensor node to reason about the WSN coverage and actions required to maintain the required level of performance. It is a distributed mobility control algorithm which enables each sensor node to compute their optimum direction of movement with a view to maintain or increase the WSN coverage. The BayesMob algorithm predicts the probability of coverage increasing given that a node moves in a particular direction. Therefore, a sensor node may behave altruistically and expend energy on moving with a view to enhance the overall coverage of the WSN.

These predictions are derived from local neighborhood information. The accuracy of these predictions is dependant on the reconfiguration rate. The reconfiguration is the process of communicating with neighboring nodes to perform localization, navigation, neighbor discovery, synchronization and possibly generating routing tables. The reconfiguration process carries energy overheads due to the communication. Therefore, a trade-off exists between the reconfiguration rate and the WSN lifetime. The physical relocation of the sensor nodes also carries energy overheads due to the energy expended driving the motors and servos.

This approach to extend the network lifetime has already been proposed, by creating the Coverage Fidelity (CoFi) algorithm (Ganeriwal *et al.*, 2004), (see related work). BayesMob differs in that sensor nodes predict the performance implications using a discrete Bayesian network. Also each sensor node coordinates their own motion and do not rely on the dying sensor node(s) for instructions.

In Section 2 we outline the related work that considers mobile WSN, and the inherent performance implications and benefits. Section 3 presents the BayesMob algorithm, and discusses the pros and cons when implementing the algorithm. The results of simulations are presented in Section 4, and finally Section 5 concludes the paper.

2. Related work

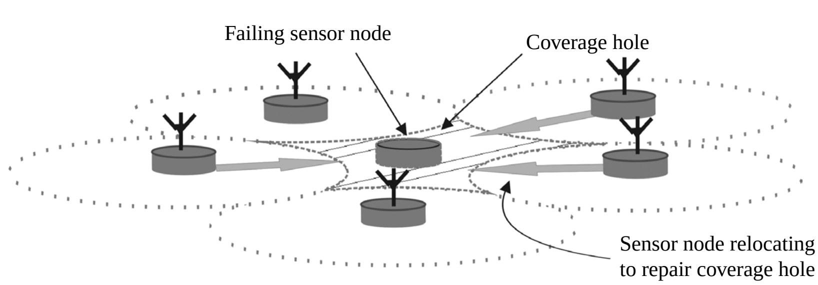
Mobility as a control primitive for self-deployment of WSNs has been investigated. For example, Wang *et al.* (2006) proposed a distributed self-deployment protocol which uses Voronoi diagrams to discover coverage holes caused by non-uniform deployment. The paper proposes three movement-assisted sensor deployment protocols which essentially relocate sensor nodes from densely deployed regions to areas with sparse coverage. Miao *et al.* (2006) proposed a self-deployment protocol for heterogeneous WSNs.

Using mobility as a control primitive to extend the network lifetime by balancing the EDR between all sensor nodes has also been investigated. For example, Rao and Biswas (2005) proposed a biologically inspired mobility model for balancing the energy overhead related to communication. The algorithm adopts a preventative approach to the creation of coverage holes due to node energy exhaustion. The mechanism was inspired by observing the natural grouping behavior of Emperor penguin communities in the Antarctic regions. The scheme however, does not consider node failure due to malfunction or malicious destruction.

Mobility as a control primitive for improving network coverage has also been investigated. For example, Ganeriwal *et al.* (2004) proposed a distributed CoFi algorithm that controls the relocation of a sensor node in order to repair coverage holes which are assumed to be a consequence of node failure.

Sekhar *et al.* (2005) proposed a dynamic coverage maintenance (DCM) scheme that also exploits the limited mobility of sensor nodes for active fault repair of the WSN. Four distributed rule-based DCM algorithms are presented which rely on local neighborhood topology information for coordinating the sensor relocation. The proposed DCM algorithms only relocate one hop neighbors of the dying sensor node, therefore, jeopardizing the effectiveness of the DCM scheme when limited redundancy is available within the vicinity of the dying node.

Figure 1 Use of node mobility to salvage performance and counteract coverage holes



Sensor Review
Volume 28 · Number 4 · 2008 · 326–333

Butler and Rus (2003) proposed an event-based mobility scheme that coordinates the relocation of sensor nodes to areas that require a higher sensing resolution due to environment, application and topology (nodes failing or moving) changes. Two distributed algorithms are proposed which use a history and history-free technique. The trade-offs between computation, memory and accuracy of the node's positions is also given. Using single dimensional mobility for improving sensing resolution and overcoming unpredictable environmental influences has also been investigated. Kansal *et al.* (2004b) presented a low complexity single dimension mobility strategy which has low energy actuation primitives. The nodes move along a single dimension to counteract a loss in coverage caused by environmental influences such as the presence of obstacles.

Low complexity mobility was also investigated by Kansal *et al.* (2004a) and Pon *et al.* (2005a, b) through the development of the Network InfoMechancial System (NIMS). NIMS's integrate distributed, embedded sensing and computing systems with infrastructure – supported mobility. The papers suggest that the NIMS's motion capability enables the network to adapt to environment, application and topology changes.

3. Bayesian self healing algorithm

3.1 Motivation

The lifetime of a WSN is directly influenced by the ability of the network to satisfy the application criteria. The latter would generally define an acceptable level of coverage and connectivity that the WSN should maintain. Sensor nodes in the WSN do not die simultaneously for a variety of reasons. Failure may occur due to energy exhaustion, malicious destruction, or malfunction. The phenomena that the WSN is detecting may itself be non-uniform. For example, a traffic monitoring application may yield varying traffic densities in alternative areas of the region of interest. The sensor nodes closer to the base station are likely to die faster as they would forward greater number of data packets than nodes on the outer periphery (this phenomenon is referred to as an edge effect).

These networks are also generally deployed in inhospitable environments where nodes may have to tolerate extreme environmental conditions. Therefore, abrupt sensor node failure may occur. These sensor node failures may impair the WSN coverage and connectivity, via the creation of coverage holes. A coverage hole is defined as an area of the region not covered by the WSN, or a sub-section of the WSN being disconnected due to the node failure. When these coverage holes are created the WSN may fail to satisfy the application criteria and therefore the remaining energy within the WSN would be wasted.

We propose to utilize the remaining resources by relocating the energy proficient sensor nodes to repair the coverage holes. Therefore, the nodes may give up their current position and expend energy to relocate, effectively resulting in the WSN healing itself. However, the physical relocation of the sensor nodes will reduce their energy reserve, due to the energy necessary to drive motors and servos. Hence, a method is required to determine whether the energy expended due to physical relocation would yield performance benefits, to assist with the decision-making process.

The existing self-healing algorithms outlined in the related work section are centralized, where the failing sensor nodes coordinate the relocation of the neighboring nodes. Therefore, these algorithms only consider failures due to energy exhaustion where nodes have sufficient time and energy to coordinate the relocation of neighboring nodes. Also the algorithms only consider the relocation of one hop neighbors therefore, the recovery from a node failure may be jeopardized when considering limited local redundancy.

3.2 Bayesian mobility (BayesMob)

The BayesMob algorithm coordinates the sensor node relocation to maintain coverage in the event of node failures which cause coverage holes. BayesMob provides a distributed approach to motion control by adopting a Bayesian network, which uses probabilistic reasoning to determine the optimum node motion direction (in terms of coverage). The node effectively predicts the performance benefits and implications of moving in a given direction. A Bayesian network is a graphical structure that describes how each sensor node evaluates the uncertainty within the WSN (Korb and Nicholson, 2004).

BayesMob incorporates a discrete Bayesian network which predicts the probability of the WSN coverage increasing or remaining constant/unchanged given the WSN topology and sensor node's motion direction. The variables used in BayesMob are outlined in Table I.

Netica application software from Norsys (Netica, 2007) was used to create and validate the Bayesian networks. The Bayesian networks are used for predictive reasoning. BayesMob only considers discrete variables so the relationships between connected nodes are represented by a conditional probability table (CPT), (Coles, 2007). The CPT values are generally specified using statistical data from the system, and/or Bayes' theorem. The values within BayesMob have been specified by the designer with a view to achieve the desired response from the Bayesian networks.

The information used as evidence fed into the Bayesian networks is acquired from the local neighborhood. The predictions are based on a possible move in one of the cardinal directions north, south, east, or west.

BayesMob calculates the conditional probability of an increase in WSN coverage if the sensor node moves in one of the cardinal directions given evidence based on neighbors'

Table I Definition of variables

Symbol	Definition
T	True
F	False
N	Cardinal direction north
S	Cardinal direction south
E	Cardinal direction east
W	Cardinal direction west
i	Cardinal direction index (N, S, E, W)
C_r	Sensor node communication range
C_i	Coverage increase in cardinal direction $i \in$ (true, false)
N_i	Need to move in cardinal direction $i \in$ (true, false)
A_i	Neighbor in cardinal direction $i \in$ (true, false)
d_i	Neighbor distance $< C_r$ in cardinal direction $i \in$ (true, false)
Θ	Motion angle of sensor node

Sensor Review
Volume 28 · Number 4 · 2008 · 326–333

positions. The evidence is expressed as the probability of neighbors lying in each one of the cardinal directions and the probability that the closest neighbor's distance in each direction is less than the communication range (C_r). The BayesMob structure is shown in Figure 2. The joint probability function for BayesMob is given in equation (1):

$$P(C_i, \theta, N_N, N_S, N_E, N_W) = P(\theta)P(N_N)P(N_S)P(N_E)P(N_W) \times P(C_i|\theta, N_N, N_S, N_E, N_W) \quad (1)$$

The probability of coverage increasing given that the node moves in direction i ($P(C_i = T|\theta = i)$) is calculated by using joint probability theory and marginalization, which is given in equation (2):

$$P(C_i = T|\theta = i) = \frac{P(C_i = T, \theta = i)}{P(\theta = i)} = \sum_{N_N, N_S, N_E, N_W \in \{T,F\}} P(N_N)P(N_S)P(N_E)P(N_W)P(\theta = i) \times P(C_i = T|\theta = i, N_N, N_S, N_E, N_W) \quad (2)$$

where $P(N_N)$ to $P(N_W)$ (probability of the need to move into each of the cardinal directions) is calculated using equation (3):

$$P(N_i = T) = \sum_{A_i, d_i \in \{T,F\}} P(A_i)P(d_i)P(N_i = T|A_i, d_i) \quad (3)$$

The belief in the evidence A_i and d_i are calculated from a contacts information table (CIT) local to each sensor node. The CIT stores all known information about neighboring nodes and also sensor nodes that route data packets through the node destined for the commander node. An example of this table is shown in Figure 3. The sensor nodes represents each of it neighbors' positions (motion) with two Gaussian distributions. Therefore, the CIT stores a mean (μ) and a standard deviation (σ) for each of its neighbors' x and y coordinates. The CIT also contains additional information such as a timestamp associated with last communication and a relay count associated with the number of communication hops to the contact. The motion characteristics such as motion direction (θ) and speed are also stored.

A sensor node must reconfigure at regular intervals to maintain a valid contacts table. The reconfiguration allows sensor nodes to exchange information with neighboring nodes for WSN maintenance purposes. This carries a communication overhead resulting in the need to extend the time between reconfigurations; this in turn, leads to increased uncertainty. Sensor nodes handle this uncertainty by representing contacts locations as Gaussian distributions which vary in time. This expresses the fact that information becomes less accurate the longer a node goes without being updated with contact information. In addition, the assumed Gaussian distributions provide a mechanism for quantifying the interdependencies between sensor nodes. When the contacts table is updated during reconfiguration μ_x, μ_y store the contacts' current x and y coordinates, which represent its neighbor's mean position coordinates. Also the standard deviations associated with x and y $\sigma_x = \sigma_y = 1\,\text{m}$ (arbitrarily initial value) which allows for localization errors.

Figure 2 Coverage increase Bayesian network structure

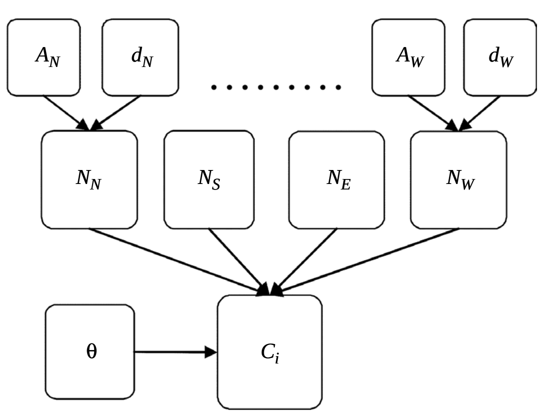


Figure 3 CIT local to sensor nodes

Node ID	μ_x (m)	σ_x (m)	μ_y (m)	σ_y (m)	Time-stamp	Relay Count	Spead (m/s^{-1})	θ
1								
....								
i								

During the period between reconfigurations and assuming the sensor node has not been in contact with its neighbors, it periodically increases the σ_x, σ_y associated with each contact entry thus increasing the uncertainty associated with the contacts' coordinates. The standard deviation increase is calculated using equation (4):

$$\sigma = \sigma + v \cdot t_{\text{update}} \quad (4)$$

where v is the node speed and t_{update} table update duty cycle time.

3.3 Probability of neighbor distance being less than the communication range

The Gaussian distributions associated with each of the contacts' coordinates are used to calculate the probability that neighbor's distance is less than the communication range ($P(d_i(j) = T)$ where $j = 1 \ldots n$, and n equals the total number of neighbors from the contacts table). If x and y are independent Gaussian random variables with nonzero means then the distance $z = \sqrt{x^2 + y^2}$ has a probability density function $f_z(z)$ which can be represented by a Rician distribution, which is given in equation (5):

$$f_z(z) = \frac{ze^{-(z^2+\mu^2)/2\sigma^2}}{\sigma^2} Io\left(\frac{z\mu}{\sigma^2}\right) \quad (5)$$

where:

$$\mu = \sqrt{\mu_x^2 + \mu_y^2}, \quad \theta = \tan^{-1}\left(\frac{y}{x}\right), \quad \sigma = \sigma_x = \sigma_y,$$

$$\mu_x = \mu\cos\phi, \quad \mu_y = \mu\sin\phi$$

and:

$$Io(\eta) \stackrel{\Delta}{=} \frac{1}{2\pi}\int_0^{2\pi} e^{\eta\cos(\theta-\phi)}\mathrm{d}\theta = \frac{1}{\pi}\int_0^{\pi} e^{\eta\cos\theta}\mathrm{d}\theta \quad (6)$$

Sensor Review
Volume 28 · Number 4 · 2008 · 326–333

Equation (6) is the modified Bessel function of the first kind and zeroth order.

$P(d_i(j) = T)$ in the cardinal direction i is approximated by numerically integrating the Rician distribution between zero and the C_r.

3.4 Probability of a neighbor lying in each of the cardinal directions

$P(A_i(j)),\ \ j = 1\ldots n$ is approximated using Algorithm 1 because the p.d.f $f_\theta(\theta)$ of the angle $\theta = \tan^{-1}(y/x)$ is intractable when x and y are independent Gaussian random variables with nonzero means:

The probabilities $P(d_i(j) = T)$ and $P(A_i(j) = T)$ are aggregated to approximate the evidence for BayesMob. This is shown in equations (7) and (8). $P(d_i(j) = T)$ is scaled with respect to the belief that the neighbor is in direction i:

$$P(A_i = T) = 1 - \prod_{j=1}^{n} \overline{P(A_i(j) = i)} \qquad (7)$$

$$P(d_i = T) = 1 - \prod_{j=1}^{n} \overline{P(d_i(j)) \times P(A_i(j) = i)} \qquad (8)$$

Algorithm 1

Approximate $P(A_i(j) = T)$ x, y = sensor node's coordinates, and $\sigma = \sigma_x = \sigma_y$:

$$\mu_{\text{distance}} = \sqrt{(x - x_j)^2 + (y - y_j)^2}$$

$$= \text{mean distance from neighbour}$$

$$\text{if } \ \sigma < (4 \times \mu_{\text{distance}})$$

Assume the $f_\theta(\theta)$ p.d.f is represented by a Gaussian distribution with:

$$\sigma_\theta(\sigma) = 100 - 100e^{-(\sigma/\mu_{\text{distance}})}$$

$$\mu_\theta = \tan^{-1}\left(\frac{\mu_y}{\mu_x}\right)$$

/*Probability of direction is approximated by integrating between
45 to 135 and 405 to 495 degrees for north
135 to 225 and -225 to -135 degrees for west
225 to 315 and -135 to -45 degrees for south
315 to 360 and 0 to 45 and -45 to 0 and 360 to 405 degrees for east */
else
Assume $f_\theta(\theta)$ is represented by a uniform distribution
/*assume a uniformed distribution between 0 and 360 degrees */

3.5 Selecting the motion status and direction

The sensor node selects the motion direction by evaluating probability of coverage increase $P(C_i)$ and determining which direction yields the maximum probability of the coverage being maintained or increased and also evaluating the adjacent cardinal direction probabilities. If $P(C_i)$ is greater than a specified threshold the node moves in the direction determined by Algorithm 2, else it will remain static. The threshold defines the motion status of the sensor node by setting the sensitivity of the BayesMob algorithm. If this value is too low the sensor node will continuously move and thus exhaust their limited energy reserve. Alternatively setting the threshold too high will prevent the sensor node from moving to repair coverage holes. Therefore, this value specifies how responsive a sensor node is to a coverage hole. The threshold was set to 35 per cent through trial and error testing.

Algorithm 2

Selecting motion direction
Calculate the difference between adjacent cardinal directions probabilities $P(C_i \pm 1)$
if difference <2 per cent and difference >-2 per cent
motion direction = direction which yields maximum probability
else
motion direction = direction which yields maximum probability + difference between adjacent cardinal direction probabilities
end

4. Simulation and results

4.1 Simulation set up

All simulations have been generated using a custom built Matlab based WSN simulator. The geographical region of interest was set to a 100 × 100 m area. Every sensor node is equipped with motion capabilities. Table II details the settings of the simulation parameters. The sensor nodes were configured to generate and transmit data packets destined for the commander node in a duty period of 2 min.

The reconfiguration duty time defines the period between neighbor communications used to update localization information within the CIT. The former was set to 2 min. We assumed a perfect medium access control (MAC) protocol, therefore the practical implications associated with communication were not considered. Having said that, in a practical implementation the repeat mechanism built into the MAC protocol would ensure successful transmissions at the cost of negligible packet delay. The latter is of the order of hundreds of milliseconds (upper estimate) which is negligible when compared with the duty time of the system at hand.

Two types of deployment strategies have been tested; fixed and random deployments. Under the fixed approach the sensor nodes would be placed manually, thus ensuring a uniform distribution of the sensor nodes and maximum coverage. Alternatively the random approach positions the nodes following a uniform distribution.

Simulations have been carried out for a range of sensor node densities (64, 81 and 100 nodes for the 100 × 100 m^2). The latter values were selected to provide a uniform spatial distribution across the region of interest (which is square shaped), whilst adopting a fixed deployment approach. For example, a deployment of 64 nodes which are evenly

Table II Simulation parameters

Parameter	Value	Rationale
Communication range	20 m	Typical ranges observed in external environments with Berkeley motes
Sensing range	10 m	Sensing range is half the communication range
Mobility cost	17.8758 J/m	X4e robot platforms (2006)
Total initial energy	87,480 J	Capacity of 6 v alkaline battery

distributed across the region of interest yields 8 × 8 node grid. The simulations evaluate the WSN coverage for the CoFi algorithm (Ganeriwal *et al.*, 2004) and whilst adopting BayesMob. We present results that compare both approaches in terms of coverage loss and motion energy overheads. The node failures are induced via a uniform distribution and occur at 30 min intervals which attempts to emulate abrupt failures under volatile application environment conditions.

4.2 Fixed deployment

First we consider a fixed deployment scenario, under which the sensor nodes are uniformly distributed over the region of interest to provide 100 per cent coverage. Figures 4-6 show the respective coverage loss plots for a deployment of 64, 81, and 100 nodes. These results show that the physical relocation of the sensor nodes under the BayesMob algorithm will sustain the

Figure 4 Coverage loss plot for a fixed deployment of 64 nodes

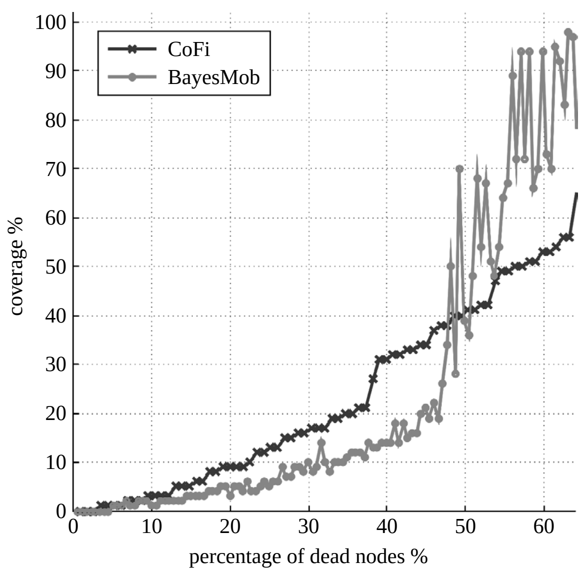


Figure 5 Coverage loss plot for a fixed deployment of 81 nodes

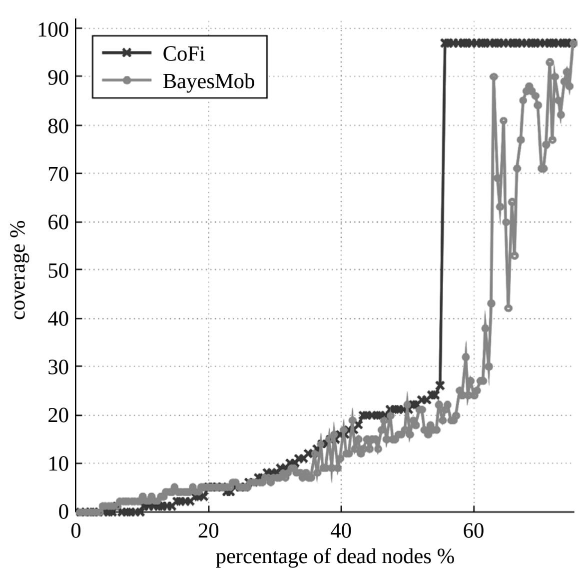
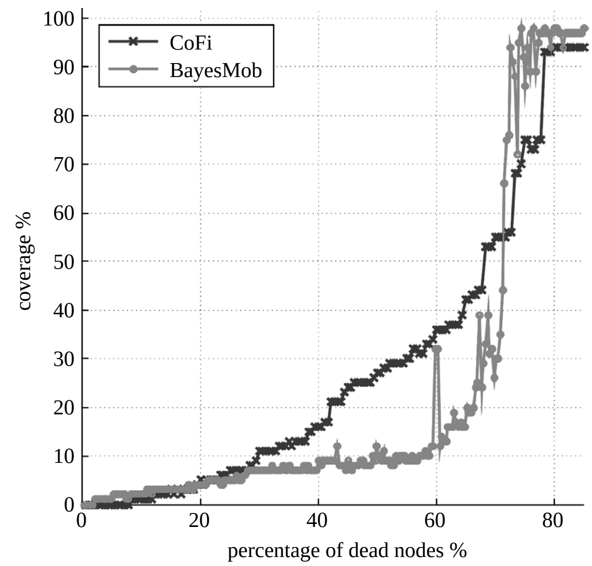


Figure 6 Coverage loss plot for a fixed deployment of 100 nodes



WSN coverage for an increased percentage of node failures. This observation becomes more apparent as the sensor node density increases. For example, if the application criterion specifies a maximum coverage loss of 20 per cent, with CoFi the WSN could tolerate 35, 43, and 44 per cent of nodes failing for 64, 81, and 100 nodes deployed, respectively. Using the BayesMob algorithm the WSN tolerates 45, 58, and 66 per cent of nodes failing for the same node densities.

The sharp increases in coverage loss observed in all figures are due to a subsection of the WSN becoming disconnected (loss in connectivity). The BayesMob fails to recover the coverage loss when the number of sensor nodes deployed is not sufficient to cover the region of interest.

4.3 Random deployment

Here, the sensor nodes are deployed randomly according to a uniform distribution over the region of interest. Figures 7-9 show the respective coverage loss plots. The results of the simulations show that the BayesMob algorithm tolerates 50, 60 and 68 per cent of nodes failing (for 64, 81 and 100 nodes deployed, respectively) before the coverage loss criterion is exceeded. Under the same conditions, the CoFi tolerates only 40, 43 and 57 per cent node failure before the coverage criterion is breached.

4.4 Motion energy overheads

The percentage of total energy available to the WSN used for motion for both strategies (BayesMob and CoFi) has been evaluated and is represented in Figure 10 as a function of the percentage of dead nodes. The total energy is calculated by multiplying the total number of sensor nodes by their individual energy reserves (total battery capacity) throughout the length of the simulation. The motion energy costs associated with the BayesMob exceed those of CoFi. When 80 per cent of the WSN has failed the motion energy for BayesMob algorithm is approximately 1.7 per cent of the total energy. The corresponding motion energy for the CoFi algorithm under the same conditions is approximately 0.025 per cent.

The motion energy for BayesMob rapidly increases as the percentage of nodes fail. This is due to sensor nodes moving

Figure 7 Coverage loss plot for a random deployment of 64 nodes

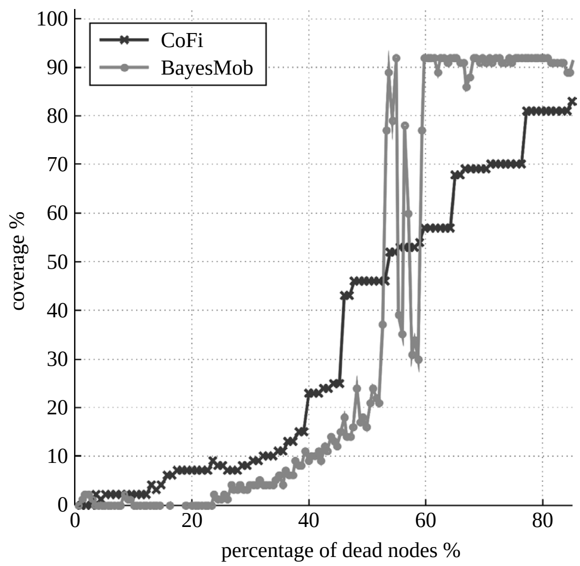
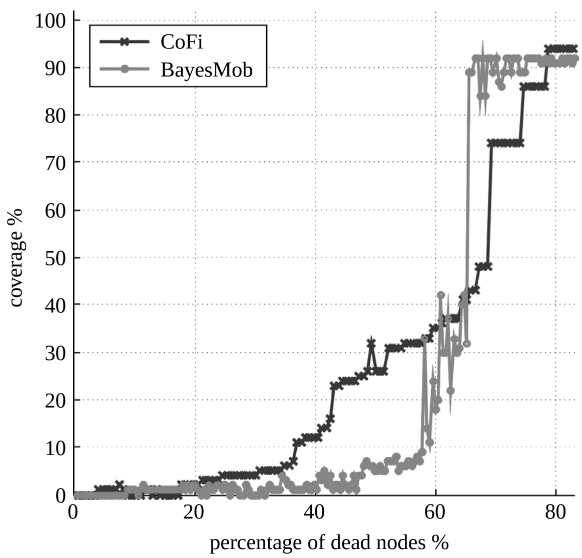


Figure 8 Coverage loss plot for a random deployment of 81 nodes



a greater distance to repair the coverage holes. The CoFi algorithm will only attempt to repair coverage holes via relocating one hop neighbors. Therefore, the motion distance is restricted at the cost of coverage. The decentralized nature of BayesMob means that it can also accommodate abrupt failures due to malicious destruction and or malfunction.

5. Conclusion

This paper has shown how, through the use of a Bayesian network based mobility scheme, a WSN can be imparted self healing properties with regards to coverage losses induced by energy exhaustion and abrupt node failures (such as those caused by malicious interference). The paper has described how sensor nodes operating according to BayesMob can predict WSN coverage variations using only local information and compute a direction of motion with a view to maximize or

Figure 9 Coverage loss plot for a random deployment of 100 nodes

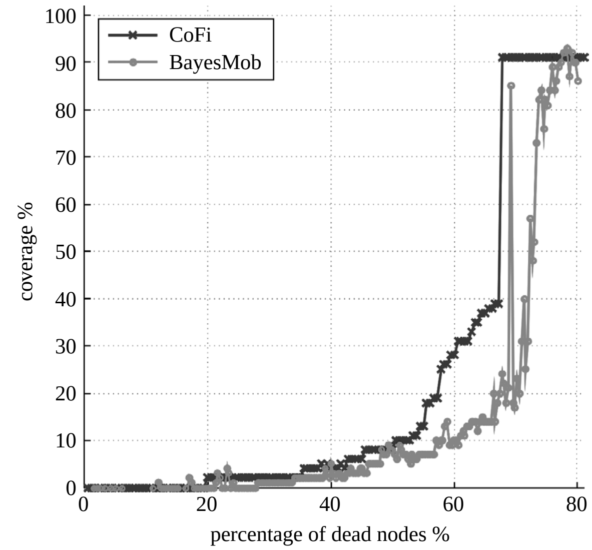
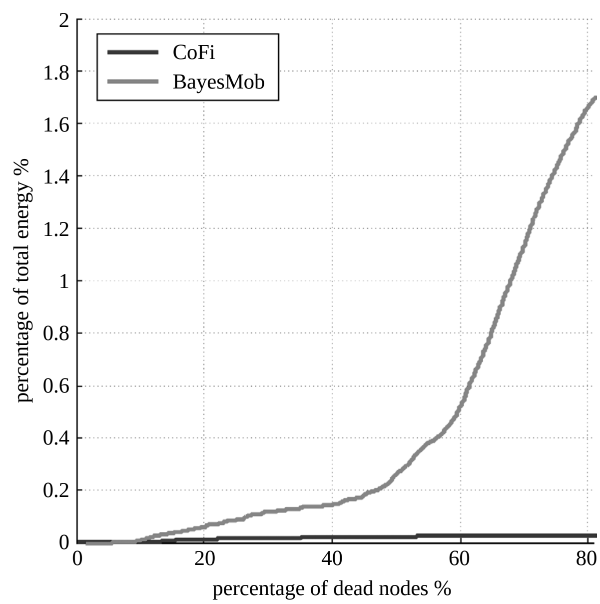


Figure 10 Motion energy overheads for BayesMob vs CoFi



maintain the coverage of the network. The results of comparing BayesMob with the CoFi algorithm have been presented and show that BayesMob maintains coverage for a greater percentage of dead nodes, albeit at the expense of increased energy overheads – which may be acceptable given the application coverage requirements.

BayesMob is a promising technique which is currently being extended to coordinate the motion of nodes within a mobile WSN to improve additional performance criteria such as lifetime (for example edge effect minimisation where a sensor node would base their next move on the estimated EDR of neighboring nodes). It is also planned to create mobility strategies which would provide varying degrees of sensing resolution across the region of interest, dependent on varying application criteria and operating conditions.

References

Akyildiz, I.F., Su, W., Sankarasubramanian, Y. and Cayirci, E. (2002), "Wireless sensor networks: a survey", *Elsevier Computer Networks*, Vol. 38, pp. 393-422.

Butler, Z. and Rus, D. (2003), "Event-based motion control for mobile sensor networks", *IEEE Pervasive Computing*, Vol. 2 No. 4, pp. 34-42.

Coles, M.D. (2007), "CPT for BayesMob", available at: http://userweb.port.ac.uk/ ~ colesm/index_files/page0002.htm

Ganeriwal, S., Kansal, A. and Srivastava, M.B. (2004), "Self aware actuation for fault repair in sensor networks", *Proceedings of IEEE International Conference on Robotics and Automation ICRA '04*, Vol. 5, pp. 5244-9.

Kansal, A., Somasundara, A., Jea, D.D., Srivastava, M.B. and Estrin, D. (2004a), "Intelligent fluid infrastructure for embedded networks", *Proceedings of ACM MobiSYS'04 Conference*, 6-9 June.

Kansal, A., Yuen, E., Kaiser, W., Pottie, G. and Srivastava, M.B. (2004b), "Sensing uncertainty reduction using low complexity actuation", *ACM Information Proceedings on Sensor Networks (ISPN)*.

Korb, K.B. and Nicholson, A.E. (2004), *Bayesian Artificial Intelligence*, Chapman & Hall, Boca Raton, FL.

Meguerdichian, S., Farinaz, K., Miodrag, P. and Srivastava, M.B. (2001), "Coverage problems in wireless ad-hoc sensor networks", *Proceedings of INFOCOM, Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 3, pp. 1380-7.

Miao, L., Qi, H. and Wang, F. (2006), "Biologically inspired self-deployable heterogeneous mobile sensor networks", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS 2005)*, 2-6 August, pp. 2363-8.

Netica (2007), application software and instructions, available at: www.norsys.com

Pon, R., Batalin, M.A., Gordon, J., Kansal, A., Liu, D., Rahimi, M., Shirachi, L., Yu, Y., Hansen, M., Kaiser, W.J., Srivastava, M.B., Sukhatme, G. and Estrin, D. (2005a), "Networked infomechanical systems: a mobile embedded networked sensor platform", *Information Proceeding in Sensor Networks IPSN'05*, 15 April, pp. 376-81.

Pon, R., Batalin, M.A., Chen, V., Kansal, A., Liu, D., Rahimi, M., Shirachi, L., Somasundra, A., Yu, Y., Hansen, M., Kaiser, W.J., Srivastava, M.B., Sukhatme, G. and Estrin, D. (2005b), "Coordinated static and mobile sensing for environmental monitoring", *Proceedings of IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), LNCS 3560*, pp. 403-5.

Rao, J. and Biswas, S. (2005), "Biologically inspired mobility models for energy conservation in sensor networks", *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*.

Sekhar, A., Manoj, B.S. and Siva Ram Murthy, C. (2005), "Dynamic coverage maintenance algorithm for sensor networks with limited mobility", *Proceedings of 3rd IEEE International Conference on Pervasive Computing and Communications (PerCom'05)*, pp. 51-60.

Wang, G., Cao, G. and La Porta, T. (2006), "Movement-assisted sensor deployment", *IEEE Trans. Mobile Computing*, Vol. 5, pp. 640-52.

X4e robot platforms (2006), Supplied by Total Robots, available at: www.totalrobots.com/program_files/x4e_rover.htm

Further reading

Jiang, J. and Dou, W. (2004), "A coverage-preserving density control algorithm for wireless sensor networks", *Proceedings of ADHOC-NOW 2004, LNCS 3158*, pp. 42-55.

Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R. and Gill, C. (2003), "Integrated coverage and connectivity configuration in wireless sensor networks", *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA*, pp. 28-39.

Corresponding author

Matthew Coles can be contacted at: matt.coles@port.ac.uk