# Markov Modeling of Fault-Tolerant Wireless Sensor Networks

Arslan Munir and Ann Gordon-Ross
Department of Electrical and Computer Engineering
University of Florida, Gainesville, Florida, USA
e-mail: {amunir@ufl.edu, ann@ece.ufl.edu}

***Abstract*—Technological advancements in communications and embedded systems have led to the proliferation of wireless sensor networks (WSNs) in a wide variety of application domains. One commonality across all WSN application domains is the need to meet application requirements (e.g., lifetime, reliability, etc.). Many application domains require that sensor nodes be deployed in harsh environments (e.g., ocean floor, active volcanoes), making these sensor nodes more prone to failures. Unfortunately, sensor node failures can be catastrophic for critical or safety related systems. To improve reliability in such systems, we propose a fault-tolerant sensor node model for applications with high reliability requirements. We develop Markov models for characterizing WSN reliability and MTTF (Mean Time to Failure) to facilitate WSN application-specific design. Results show that our proposed fault-tolerant model can result in as high as a 100% MTTF increase and approximately a 350% improvement in reliability over a non-fault-tolerant WSN. Results also highlight the significance of a robust fault detection algorithm to leverage the benefits of fault-tolerant WSNs.**

***Index Terms*—Fault-Tolerance, reliability, Markov modeling, wireless sensor networks**

## I. Introduction and Motivation

Wireless sensor networks (WSNs) consist of spatially distributed autonomous sensor nodes that collaborate with each other to perform an application task. WSN sensor nodes are typically mass produced and are often deployed in unattended and hostile environments making them more susceptible to failures than other systems [1]. Additionally, manual inspection of faulty sensor nodes after deployment is typically impractical. Nevertheless, many WSN applications are mission-critical, requiring continuous operation. Thus, in order to meet application requirements reliably, WSNs require fault detection and fault-tolerance (FT) mechanisms.

Fault detection encompasses distributed fault detection (DFD) algorithms which identify faulty sensor readings that indicate faulty sensors. DFD algorithms typically use existing network traffic to identify sensor failures and therefore do not incur any additional transmission cost. A fault detection algorithm's *accuracy* signifies the algorithm's ability to accurately identify faults. Though fault detection helps in isolating faulty sensors, WSNs require FT to reliably accomplish application tasks.

One of the most prominent FT techniques is to add hardware and/or software redundancy to the system [2]. However, WSNs are different from other systems as they have stringent constraints and the added redundancy for FT must justify the additional cost. Studies indicate that sensors (e.g., temperature and humidity sensors) in a sensor node have comparatively higher fault rates than other components (e.g., processors, transceivers) [3][4]. Fortunately, sensors are cheap and adding spare sensors contribute little to the individual sensor node's cost.

Even though FT is a well studied research field [5][6][7][8], fault detection and FT for WSNs are relatively unstudied. Additionally, fault detection and FT for WSNs have added complexities due to varying FT requirements across different applications. For instance, mission critical applications (e.g., security and defense systems) have very high reliability requirements whereas non-mission critical applications (e.g., ambient conditions monitoring applications) typically have relatively low reliability requirements. To the best of our knowledge there exists no sensor node model to provide better reliability for such critical applications. Furthermore, applications are designed to operate reliably for a certain period of time (i.e., WSN applications typically have specific lifetime requirements). Unfortunately, literature provides no rigorous mathematical model with insights into WSN reliability and lifetime. Finally, fault detection and FT have been studied in isolation and their synergistic relationship has not been investigated in the context of WSNs.

Our main contributions in this paper are:

- We investigate the synergy of fault detection and FT for WSNs and propose an FT sensor node model consisting of duplex sensors (i.e., one active sensor and one inactive spare sensor), which exploits this synergy between fault detection and FT. Whereas sensors may employ N-modular redundancy (e.g., triple modular redundancy (TMR) is a special case of N-modular redundancy) [2], we propose a duplex sensor model to minimize the additional cost for our FT model.
- To the best of our knowledge, we for the first time develop a Markov model for characterizing WSN reliability and MTTF. Our Markov modeling facilitates WSN design by enabling WSN designers to determine the exact number of sensor nodes required to meet the application's lifetime and reliability requirements. Our Markov modeling provides an insight on the type of sensor nodes (duplex or simplex) feasible for an application to meet the application's requirements.

## II. Related Work

Although general FT is a well-studied research field [5][6][7][8], little work exists in WSN-specific fault detection and FT. Jiang [9] proposed a DFD scheme that detected faulty sensor nodes by exchanging data and mutually testing among neighboring nodes. Jian-Liang et al. [10] proposed a weighted median fault detection scheme (WMFDS) that used spatial correlations among the sensor measurements (e.g., temperature, humidity). Lee et al. [11] presented a DFD algorithm that identified faulty sensor nodes based on comparisons between neighboring sensor nodes' data. The DFD algorithm used time redundancy to tolerate transient faults in sensing and communication. Khilar et al. [12] proposed a probabilistic approach to diagnose intermittent faults in WSNs. The simulation results indicated that the accuracy of the DFD algorithm increased as the number of diagnostic rounds increased (each round comprised of exchanging measurements with the neighboring nodes).

Further work exists in WSN fault detection. Ding et al. [13] proposed two algorithms: faulty sensor identification and fault-tolerant event boundary detection. Their algorithms considered that both the faulty sensors and sensor in the event region could generate abnormal readings (readings that deviate from a typical application-specific range). Krishnamachari et al. [14] proposed a distributed, Bayesian algorithm for sensor fault detection and correction that exploited the notion that measurement errors due to faulty equipment are likely to be uncorrelated. Wu et al. [15] presented a fault detection scheme in which the fusion center (the node that aggregated data from different nodes) attempted to identify faulty sensor nodes through temporal sequences of received local decisions using a majority voting technique.

In the area of FT for WSNs, Koushanfar et al. [4] proposed an FT scheme that provided back up for one type of sensor using another type of sensor. However, they did not propose any FT model. Clouqueur et al. [16] presented algorithms for collaborative target detection in the presence of faulty sensors. Chiang et al. [17] built and evaluated system-level test interfaces for remote testing, repair, and software upgrades for sensor nodes. They added a test interface module (TIM) to provide the testing function and experimental results indicated that the TIM with double, triple, and quadruple redundancy increased the WSN's availability.

Even though DFD algorithms were proposed in literature for detecting sensor faults, the fault detection was not leveraged to provide FT. Additionally, there does not exist any model for FT sensor nodes, nor does there exist any model for characterizing WSN FT metrics such as reliability and MTTF.

## III. Fault-Tolerant Markov Models

In this section, we present our proposed Markov models for FT WSNs. Our Markov models are comprehensive and encompass the sensor node, a WSN cluster (a group of sensor nodes), and the overall WSN.

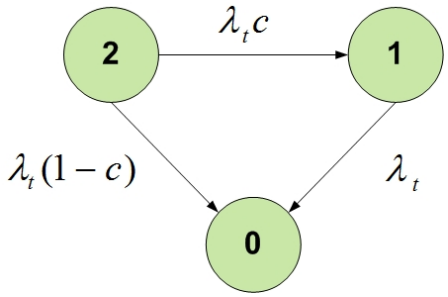


Fig. 1. Sensor node Markov model.

### A. Fault-Tolerance Parameters

The FT parameters leveraged in our Markov model are *coverage factor* and *sensor failure probability*. The coverage factor $c$ is defined as the probability that the faulty active sensor is correctly diagnosed, disconnected, and replaced by a good inactive spare sensor. The $c$ estimation is critical in an FT WSN model and can be determined by:

$$c = c_k - c_c \tag{1}$$

where $c_k$ denotes the accuracy of the fault detection algorithm in diagnosing faulty sensors and $c_c$ denotes the probability of an unsuccessful replacement of the identified faulty sensor with the good spare sensor. $c_c$ depends upon the sensor switching circuitry and is usually a constant and $c_k$ depends upon the average number of sensor node neighbors $k$ and the probability of sensor failure $p$ [9][10][13][14].

The sensor failure probability $p$ can be represented using an exponential distribution with failure rate $\lambda_s$ over the period $t_s$ (the period $t_s$ signifies the time over which the sensor failure probability $p$ is specified) [18]. Thus, we can write:

$$p = 1 - exp(-\lambda_s t_s) \tag{2}$$

### B. Fault-Tolerant Sensor Node Model

We propose an FT duplex sensor node model consisting of one active sensor (such as a temperature sensor) and one inactive spare sensor. The inactive sensor becomes active only once the active sensor is declared faulty by the fault detection algorithm. Fig 1 shows the Markov model for our proposed FT sensor node. The states in the Markov model represent the number of good sensors. The differential equations describing the sensor node duplex Markov model are:

$$\begin{aligned} P_2^{'}(t) &= -\lambda_t P_2(t) \\ P_1^{'}(t) &= \lambda_t c P_2(t) - \lambda_t P_1(t) \\ P_0^{'}(t) &= \lambda_t (1-c) P_2(t) + \lambda_t P_1(t) \end{aligned} \tag{3}$$

where $P_i(t)$ denotes the probability that the sensor node will be in state $i$ at time $t$ and $P_i^{'}(t)$ represents the first order derivative of $P_i(t)$. $\lambda_t$ represents the failure rate of an active temperature sensor and the rate at which recoverable failure occurs is $c\lambda_t$. The probability that the sensor failure cannot be recovered is $(1-c)$, and the rate at which unrecoverable failure occurs is $(1-c)\lambda_t$.

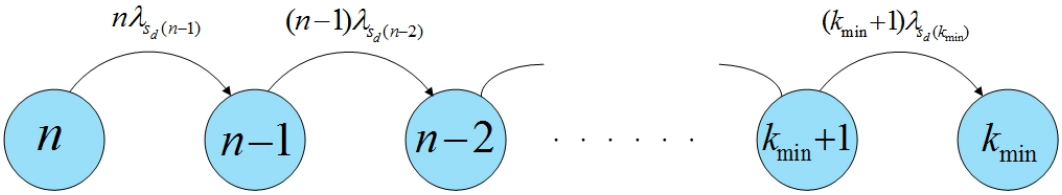


Fig. 2. WSN cluster Markov model.

Solving (3) with the initial conditions $P_2(0) = 1$, $P_1(0) = 0$, and $P_0(0) = 0$, the reliability of the duplex sensor node is given by:

$$\begin{aligned} R_{s_d}(t) &= 1 - P_0(t) \\ &= e^{-\lambda_t t} + c\lambda_t t e^{-\lambda_t t} \end{aligned} \tag{4}$$

The MTTF of the duplex sensor system is

$$\begin{aligned} \text{MTTF}_{s_d} &= \int_0^\infty R_{s_d}(t)\, dt \\ &= \frac{1}{\lambda_t} + \frac{c}{\lambda_t} \end{aligned} \tag{5}$$

The average failure rate of the duplex sensor system depends on $k$ (since the fault detection algorithm's accuracy depends on $k$ (Section III-A)) and is given by:

$$\lambda_{s_d(k)} = \frac{1}{\text{MTTF}_{s_d(k)}} \tag{6}$$

### C. Fault-Tolerant WSN Cluster Model

A typical WSN consists of many clusters and we assume for our model that all nodes in a cluster are neighbors to each other. If the average number of nodes in a cluster is $n$, then the average number of neighbor nodes per sensor node is $k = n-1$. Fig. 2 depicts our Markov model for a WSN cluster. We assume that a cluster fails (i.e., fails to perform its assigned application task) if the number of alive (non-faulty) sensor nodes in the cluster reduces to $k_{min}$. The differential equations describing the WSN cluster Markov model are:

$$\begin{aligned} P_n^{'}(t) &= -n\lambda_{s_d(n-1)} P_n(t) \\ P_{n-1}^{'}(t) &= n\lambda_{s_d(n-1)} P_n(t) - (n-1)\lambda_{s_d(n-2)} P_{n-1}(t) \\ &\vdots \\ P_{k_{min}}^{'}(t) &= (k_{min}+1)\lambda_{s_d(k_{min})} P_{k_{min}+1}(t) \end{aligned} \tag{7}$$

where $\lambda_{s_d(n-1)}$, $\lambda_{s_d(n-2)}$, and $\lambda_{s_d(k_{min})}$ represent the duplex sensor node failure rate (6) when the average number of neighbor sensor nodes are $n-1$, $n-2$, and $k_{min}$, respectively. For mathematical tractability and closed form solution, we analyze a special (simple) case of the above WSN cluster Markov model where $n = k_{min}+2$, which reduces the Markov model to three states as shown in Fig. 3.

Solving (7) for $n = k_{min} + 2$ with the initial conditions $P_{k_{min}+2}(0) = 1$, $P_{k_{min}+1}(0) = 0$, and $P_{k_{min}}(0) = 0$, the

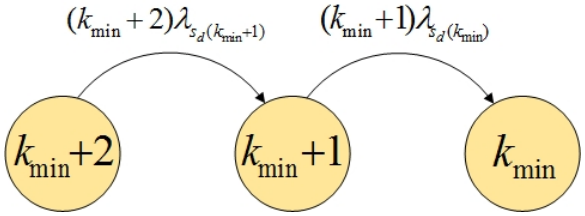


Fig. 3. WSN cluster Markov model with three states.

WSN cluster reliability is given as:

$$\begin{aligned} R_c(t) &= 1 - P_{k_{min}}(t) \\ &= e^{-(k_{min}+2)\lambda_{s_d(k_{min}+1)}t} + \\ &\frac{(k_{min}+2)\lambda_{s_d(k_{min}+1)} e^{-(k_{min}+2)\lambda_{s_d(k_{min}+1)}t}}{(k_{min}+1)\lambda_{s_d(k_{min})} - (k_{min}+2)\lambda_{s_d(k_{min}+1)}} + \\ &\frac{(k_{min}+2)\lambda_{s_d(k_{min}+1)} e^{-(k_{min}+1)\lambda_{s_d(k_{min})}t}}{(k_{min}+2)\lambda_{s_d(k_{min}+1)} - (k_{min}+1)\lambda_{s_d(k_{min})}} \end{aligned} \tag{8}$$

The MTTF of the WSN cluster is:

$$\begin{aligned} \text{MTTF}_c &= \int_0^\infty R_c(t)\, dt \\ &= \frac{1}{(k_m+2)\lambda_{s_d(k_m+1)}} + \\ &\frac{1}{(k_m+1)\lambda_{s_d(k_m)} - (k_m+2)\lambda_{s_d(k_m+1)}} + \\ &\frac{(k_m+2)\lambda_{s_d(k_m+1)}}{(k_m+2)(k_m+2)\lambda_{s_d(k_m)}\lambda_{s_d(k_m+1)} - (k_m+1)^2\lambda_{s_d(k_m)}^2} \end{aligned} \tag{9}$$

where we denote $k_{min}$ by $k_m$ in (9) for conciseness. The average failure rate of the cluster $\lambda_c(n)$ depends on the average number of nodes in the cluster $n$ at deployment time and is given by:

$$\lambda_{c(n)} = \frac{1}{\text{MTTF}_{c(n)}} \tag{10}$$

### D. Fault-Tolerant WSN Model

A typical WSN consists of $N = n_s/n$ clusters where $n_s$ denotes the total number of sensor nodes in the WSN and $n$ denotes the average number of nodes in a cluster. Fig. 4 depicts our WSN Markov model. We assume that the WSN fails to perform its assigned task when the number of alive clusters reduces to $N_{min}$. The differential equations describing

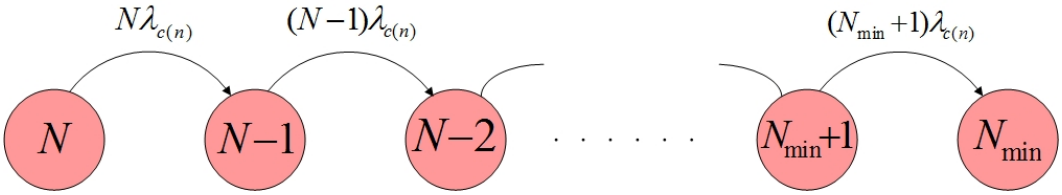


Fig. 4. WSN Markov model.

the WSN Markov model are:

$$\begin{aligned} P_N^{'}(t) &= -N\lambda_{c(n)} \\ P_{N-1}^{'}(t) &= N\lambda_{c(n)}P_N(t) - (N-1)\lambda_{c(n)}P_{N-1}(t) \\ &\vdots \\ P_{N_{min}}^{'}(t) &= (N_{min}+1)\lambda_{c(n)}P_{N_{min}+1}(t) \end{aligned} \quad (11)$$

where $\lambda_{c(n)}$ represents the average cluster failure rate (10) when the cluster contains $n$ sensor nodes at deployment time.

Solving (11) for $N = N_{min} + 2$ with the initial conditions $P_{N_{min}+2}(0) = 1$, $P_{N_{min}+1}(0) = 0$, and $P_{N_{min}}(0) = 0$, the WSN reliability is given as:

$$\begin{aligned} R_{wsn}(t) &= 1 - P_{N_{min}}(t) \\ &= e^{-(N_{min}+2)\lambda_{c(n)}t} + (N_{min}+2)\lambda_{c(n)} \times \\ &\quad \left[e^{-(N_{min}+1)\lambda_{c(n)}t} - e^{-(N_{min}+2)\lambda_{c(n)}t}\right] \end{aligned} \quad (12)$$

where $\lambda_{c(n)}$ represents the average cluster failure rate (10) when the cluster contains $n$ sensor nodes at deployment time. The WSN MTTF when $N = N_{min} + 2$ is:

$$\begin{aligned} \text{MTTF}_{wsn} &= \int_0^\infty R_{wsn}(t)\,dt \\ &= \frac{1}{(N_{min}+2)\lambda_{c(n)}} + \frac{N_{min}+2}{N_{min}+1} - 1 \end{aligned} \quad (13)$$

## IV. Results

We use the SHARPE Software Package [19] to obtain our FT sensor node, WSN cluster, and WSN model results. We assume $c_c = 0$ in (1) (i.e., once a faulty sensor is identified, the faulty sensor is replaced by a good spare sensor perfectly, and thus $c = c_k$ in (1)). We use typical $c_k$ values for our analysis that represent $c_k$ for different fault detection algorithms [9][10][13][14]. We compare the MTTF for FT and non-FT (NFT) sensor node, WSN cluster, and WSN models. The MTTF also reflects the system reliability (i.e., a greater MTTF implies a more reliable system).

Fig. 5 depicts the MTTF for an NFT and FT sensor node (based on our sensor node duplex model Section III-B) for $k$ values of 5, 10, and 15 versus the sensor failure probability $p$ when $t_s$ in (2) is 100 days [13][14]. The FT results are obtained for different $k$ because a fault detection algorithm's accuracy, and thus $c$, depends upon $k$. The results show that the MTTF for an FT sensor node improves with increasing $k$. However, the MTTF shows negligible improvement when $k = 15$ over $k = 10$ as the fault detection algorithm's accuracy improvement gradient (slope) decreases between

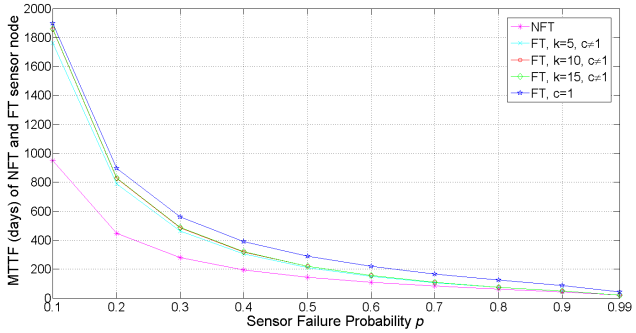


Fig. 5. MTTF (days) for an FT and a non-FT (NFT) sensor node.

large $k$ values. Fig. 5 also compares the MTTF for an FT sensor node when $c = 1\ \forall\ k, p$ representing the ideal case (i.e., the fault detection algorithm is perfect and the faulty sensor is identified and replaced perfectly for any number of neighbors and sensor failure probability). Whereas $c \neq 1$ for existing fault detection algorithms, however, comparison with $c = 1$ provides insight into how the fault detection algorithm's accuracy affects the sensor node's MTTF. Fig. 5 shows that the MTTF for an FT sensor node with $c = 1$ is always greater than the FT sensor node with $c \neq 1$. We observe that the MTTF for both the NFT and FT sensor node decreases as $p$ increases, however, the FT sensor node maintains better MTTF than the NFT sensor node for all $p$ values.

We calculated the percentage MTTF improvement gained by an FT sensor node over an NFT sensor node for different values of $p$. We observed that the MTTF percentage improvement for an FT sensor node decreases as $p$ increases when $c \neq 1$. The percentage MTTF improvement for an FT sensor node with $k = 5$ and $k = 10$ are 86% and 96%, respectively, for $p = 0.1$. The MTTF percentage improvement drops to 0.9% and 1.3%, respectively, for $p = 0.99$. The MTTF percentage improvement for an FT sensor node over an NFT sensor node is 100% on average when $c = 1$, thus highlighting the importance of a robust fault detection algorithm.

Fig. 6 depicts the MTTF for NFT and FT WNS clusters versus $p$ when $k_{min} = 4$ (we observed similar trends for other $k_{min}$ values). The FT WSN cluster consists of sensor nodes with duplex sensors (Section III-B) and the NFT WSN cluster consists of NFT non-duplex sensor nodes. The figure shows the results for two WSN clusters that contain on average $n = k_{min} + 2$ and $n = k_{min} + 5$ sensor nodes at deployment time. The figure reveals that the FT WSN cluster's MTTF is considerably greater than the NFT WSN cluster's MTTF for

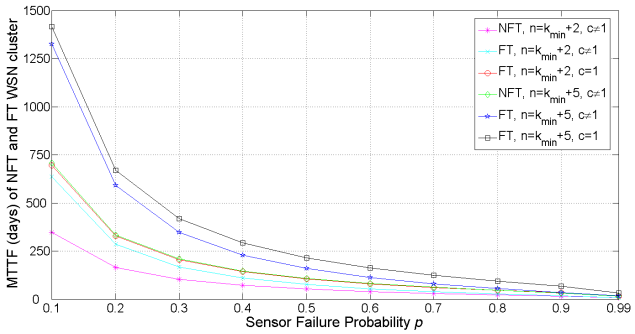


Fig. 6. MTTF (days) for the FT and non-FT (NFT) WSN clusters with $k_{min} = 4$.

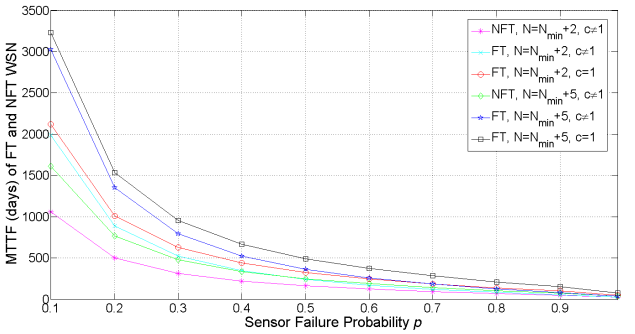


Fig. 7. MTTF (days) for the FT and non-FT (NFT) WSNs with $N_{min} = 0$.

both cluster systems ($n = k_{min} + 2$ and $n = k_{min} + 5$). Fig. 6 also compares the MTTF for FT WSN clusters when $c = 1$ with $c \neq 1$ and shows that the MTTF for FT WSN clusters with $c = 1$ is always better than the FT WSN clusters with $c \neq 1$. We point out that both the NFT and FT WSN clusters with $n > k_{min}$ have redundant sensor nodes and can inherently tolerate $n - k_{min}$ sensor node failures. The WSN cluster with $n = k_{min} + 5$ has more redundant sensor nodes than the WSN cluster with $n = k_{min} + 2$ and thus has a comparatively greater MTTF.

We observed the percentage MTTF improvement of FT WSN clusters as compared to NFT WSN clusters for two cluster systems containing $n = k_{min} + 2$ and $n = k_{min} + 5$ sensor nodes. The MTTF percentage improvement for the FT WSN cluster with $n = k_{min} + 2$, $c \neq 1$, is 83% for $p = 0.1$ and drops to 2.3% for $p = 0.99$. Similarly, the percentage MTTF improvement for the FT WSN cluster with $n = k_{min} + 5$, $c \neq 1$, is 88% for $p = 0.1$ and drops to 2.5% for $p = 0.99$. The percentage MTTF improvement for the two cluster systems is 100% on average when $c = 1$. We observed that the MTTF percentage improvement for the FT WSN cluster with $n = k_{min} + 5$ over $n = k_{min} + 2$ is 103% on average.

Fig. 7 depicts the MTTF for two WSNs containing on average $N = N_{min} + 2$ and $N = N_{min} + 5$ clusters at deployment time and each WSN fails when there are no more active clusters (i.e., $N = N_{min} = 0$). The FT WSN contains sensor nodes with duplex sensors (Section III-B) and the NFT WSN contains NFT non-duplex sensor nodes. We assume that both WSNs contain clusters with $n = k_{min} + 5$ where $k_{min} = 4$ (Section III-C). The figure reveals that the FT WSN improves the MTTF considerably over the NFT WSN for both cases ($N = N_{min} + 2$ and $N = N_{min} + 5$). Fig. 7 also shows that the MTTF for FT WSNs when $c = 1$ is always greater than the MTTF for FT WSNs when $c \neq 1$. We observe that as $p \rightarrow 1$, the MTTF for the FT WSN drops close to the NFT WSN, thus leading to an important observation that to build a more reliable FT WSN, it is crucial to have low failure probability sensors. We observe that the MTTF for WSNs with $N = N_{min} + 5$ is always greater than the MTTF for WSNs with $N = N_{min} + 2$. This observation is intuitive because WSNs with $N = N_{min} + 5$ have more redundant WSN clusters (and sensor nodes) and can survive more cluster failures before reaching the failed state ($N = 0$) as compared to WSNs with $N = N_{min} + 2$.

We observed the percentage MTTF improvement for FT WSNs over NFT WSNs for two cases where $N = N_{min} + 2$ and $N = N_{min} + 5$. The MTTF percentage improvement for FT WSNs with $N = N_{min} + 2$, $c \neq 1$, is 88% for $p = 0.1$ and drops to 3.3% for $p = 0.99$. Similarly, the MTTF percentage improvement for FT WSNs with $N = N_{min} + 5$, $c \neq 1$, is 88% for $p = 0.1$ and drops to 3.3% for $p = 0.99$. We observe that the MTTF improvement for FT WSNs with $c = 1$ is 100% on average for all $p$ values and is greater than the FT WSNs with $c \neq 1$. The MTTF percentage improvement for FT WSNs with $N = N_{min} + 5$ over FT WSNs with $N = N_{min} + 2$ is 52% on average.

We present example reliability calculations using our Markov models. For an NFT sensor node reliability calculation, sensor failure rate $\lambda_t = (-1/100)\ln(1 - 0.05) = 5.13 \times 10^{-4}$ failures/day). SHARPE gives $P_1(t) = e^{-5.13\times10^{-4}t}$ and sensor node reliability $R_s(t) = P_1(t)$. Evaluating $R_s(t)$ at $t = 100$ gives $R_s(t)|_{t=100} = e^{-5.13\times10^{-4}\times100} = 0.94999$.

For an FT sensor node reliability calculation when $c \neq 1$, different reliability results are obtained for different $k$ because the fault detection algorithm's accuracy and coverage factor $c$ depends on $k$. For $k = 5$, $c = 0.979$, SHARPE gives $P_2(t) = e^{-5.13\times10^{-4}t}$ and $P_1(t) = 5.0223 \times 10^{-4}te^{-5.13\times10^{-4}t}$. The reliability $R_s(t) = P_2(t) + P_1(t) = e^{-5.13\times10^{-4}t} + 5.0223 \times 10^{-4}te^{-5.13\times10^{-4}t}$ and $R_s(t)|_{t=100} = e^{-5.13\times10^{-4}\times100} + 5.0223\times10^{-4}\times100\times e^{-5.13\times10^{-4}\times100} = 0.94999+0.04771 = 0.99770$.

Similarly, we performed reliability calculations for an NFT and an FT WSN cluster and a complete WSN. Based on these reliability calculations, Table I shows the reliability for an NFT WSN and an FT WSN evaluated at $t = 100$ days when $N = N_{min}+2$ ($N_{min} = 0$) for clusters with nine sensor nodes on average (though similar calculations can be performed for WSN clusters containing a different number of sensor nodes on average). We observe similar trends as with sensor node reliability and WSN cluster reliability where reliability for both an NFT WSN and an FT WSN decreases as $p$ increases (i.e., reliability $R_{wsn} \rightarrow 0 \iff p \rightarrow 1$) because a WSN contains clusters of sensor nodes and decreased individual

TABLE I
RELIABILITY FOR AN NFT WSN AND AN FT WSN WHEN $N = N_{min} + 2$ ($N_{min} = 0$).

| $p$ | NFT | FT $(c \neq 1)$ | FT $(c = 1)$ |
|---|---|---|---|
| 0.05 | 0.99557 | 0.99883 | 0.99885 |
| 0.1 | 0.98261 | 0.99474 | 0.99534 |
| 0.2 | 0.93321 | 0.97583 | 0.98084 |
| 0.3 | 0.85557 | 0.93775 | 0.95482 |
| 0.4 | 0.75408 | 0.87466 | 0.91611 |
| 0.5 | 0.63536 | 0.78202 | 0.86218 |
| 0.6 | 0.51166 | 0.65121 | 0.78948 |
| 0.7 | 0.36303 | 0.49093 | 0.69527 |
| 0.8 | 0.20933 | 0.30328 | 0.55494 |
| 0.9 | 0.08807 | 0.11792 | 0.39647 |
| 0.99 | $4.054 \times 10^{-3}$ | $4.952 \times 10^{-3}$ | 0.08807 |

sensor node reliability with increasing $p$ decreases both WSN cluster and WSN reliability. Table I shows that an FT WSN with $c = 1$ outperforms an FT WSN with $c \neq 1$ and an NFT WSN for all $p$ values. For example, the percentage improvement in reliability for an FT WSN with $c = 1$ over an NFT WSN and an FT WSN with $c \neq 1$ is 5% and 0.5% for $p = 0.2$ and 350% and 236% for $p = 0.9$, respectively. These results show that the percentage improvement in reliability attained by an FT WSN increases as $p$ increases because the fault detection algorithm's accuracy and $c$ decreases as $p$ increases.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed an FT duplex sensor node model based on the novel concept of determining the coverage factor using sensor fault detection algorithm accuracy. We developed a comprehensive Markov model for WSNs consisting of sensor node clusters to compare the MTTF for FT and NFT WSNs. Our Markov model helps compare and evaluate the MTTF (and/or reliability) of WSNs, which is vital in WSN design given that different WSN applications require different reliability and MTTF.

We observed that the fault detection algorithm's accuracy plays a crucial role in FT WSNs. Results indicated that our proposed FT sensor node duplex model can provide on average 100% MTTF improvement with a perfect fault detection algorithm whereas the MTTF improvement varied from 96% to 1.3% due to a fault detection algorithm's typically poor performance at high sensor failure rates. We also observed that the redundancy in WSNs plays an important role in improving WSN MTTF. Our results revealed that just three redundant sensor nodes in a WSN cluster resulted in an MTTF improvement of 103% on average. Similarly, redundancy in WSN clusters contributes to the MTTF improvement and the results indicated that three redundant WSN clusters can improve MTTF by 52% on average. We observed that the percentage improvement in reliability for an FT WSN with $c = 1$ over an NFT WSN and an FT WSN with $c \neq 1$ is 350% and 236%, respectively, for $p = 0.9$.

Our results motivate the development of robust distributed fault detection algorithms and are the focus of our future work. We plan to develop a WSN performability model to capture both the performance and availability (and/or reliability) simultaneously. We also plan to investigate FT in sensed data aggregation (fusion) in WSNs.

## ACKNOWLEDGMENTS

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSERC.

## REFERENCES

[1] G. Werner-Allen and et al., "Deploying a Wireless Sensor Network on an Active Volcano," *IEEE Internet Computing*, vol. 10, no. 2, pp. 18–25, March 2006.

[2] I. Koren and M. Krishna, *Fault-Tolerant Systems*. Morgan Kaufmann Publishers, 2007.

[3] A. Sharma, L. Golubchik, and R. Govindan, "On the Prevalence of Sensor Faults in Real-World Deployments," in *Proc. of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, San Diego, California, June 2007.

[4] F. Koushanfar, M. Potkonjak, and A. Sangiovanni-Vincentelli, "Fault Tolerance Techniques for Wireless Ad Hoc Sensor Networks," in *Proc. of the IEEE Sensors*, Orlando, Florida, June 2002.

[5] A. Hopkins, T. B. Smith, and J. Lala, "FTMP - A Highly Reliable Fault-Tolerant Multiprocessor for Aircraft," *Proc. of the IEEE*, vol. 66, no. 10, pp. 1221–1239, October 1978.

[6] J. Wensley and et al., "SIFT: Design and Analysis of a Fault-Tolerant Computer for Aircraft Control," *Proc. of the IEEE*, vol. 66, no. 10, pp. 1240–1255, October 1978.

[7] A. Avizienis and J. Laprie, "Dependable Computing: From Concepts to Design Diversity," *Proc. of the IEEE*, vol. 74, no. 5, pp. 629–638, May 1986.

[8] A. Somani and N. Vaidya, "Understanding Fault Tolerance and Reliability," *IEEE Computer*, vol. 30, no. 4, pp. 45–50, April 1997.

[9] P. Jiang, "A New Method for Node Fault Detection in Wireless Sensor Networks," *Sensors*, vol. 9, no. 2, pp. 1282–1294, May 2009.

[10] G. Jian-Liang, X. Yong-Jun, and L. Xiao-Wei, "Weighted-Median based Distributed Fault Detection for Wireless Sensor Networks," *J. of Software*, vol. 18, no. 5, pp. 1208–1217, May 2007.

[11] M. Lee and Y. Choi, "Fault Detection of Wireless Sensor Networks," *Elsevier Computer Communications*, vol. 31, no. 14, pp. 3469–3475, September 2008.

[12] P. Khilar and S. Mahapatra, "Intermittent Fault Diagnosis in Wireless Sensor Networks," in *Proc. of IEEE 10th International Conference on Information Technology (ICIT)*, Rourkela, India, December 2007.

[13] M. Ding, D. Chen, K. Xing, and X. Cheng, "Localized Fault-Tolerant Event Boundary Detection in Sensor Networks," in *Proc. of IEEE INFOCOM*, Miami, Florida, March 2005.

[14] B. Krishnamachari and S. Iyengar, "Distributed Bayesian Algorithms for Fault-Tolerant Event Region Detection in Wireless Sensor Networks," *IEEE Trans. on Computers*, vol. 53, no. 3, pp. 241–250, March 2004.

[15] J. Wu, D. Duh, T. Wang, and L. Chang, "On-Line Sensor Fault Detection Based on Majority Voting in Wireless Sensor Networks," in *Proc. of 24th Workshop on Combinatorial Mathematics and Computation Theory (ALGO)*, Eilat, Israel, October 2007.

[16] T. Clouqueur, K. Saluja, and P. Ramanathan, "Fault Tolerance in Collaborative Sensor Networks for Target Detection," *IEEE Trans. on Computers*, vol. 53, no. 3, pp. 320–333, March 2004.

[17] M. Chiang, Z. Zilic, J. Chenard, and K. Radecka, "Architectures of Increased Availability Wireless Sensor Network Nodes," in *Proc. of IEEE International Test Conference (ITC)*, Washington, DC, October 2004.

[18] N. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*. John Wiley and Sons, Inc., 1994.

[19] R. Sahner, K. Trivedi, and A. Puliafito, *Performance and Reliability Analysis of Computer Systems: An Example-Based Approach Using the SHARPE Software Package*. Kluwer Academic Publishers, 1996.