# The Time-Free Comparison Model for Fault Diagnosis in Wireless Ad hoc Networks

**Hazim Jarrah[1,*], G. G. Md. Nawaz Ali[2], Arun Kumar[3], Peter H. J. Chong[1], Nurul I. Sarkar[4], and Jairo Gutierrez[4]**

[1] Department of Electrical and Electronic Engineering, Auckland University of Technology, Auckland, New Zealand; hjarrah@aut.ac.nz, peter.chong@aut.ac.nz
[2] Department of Applied Computer Science, University of Charleston, WV, USA; ggmdnawazali@ucwv.edu
[3] Department of Computer Science & Engineering, National Institute of Technology, Odisha, India; kumararun@nitrkl.ac.in
[4] Department of Information Technology and Software Engineering, Auckland University of Technology, Auckland, New Zealand; Nurul.sarkar@aut.ac.nz, Jairo.gutierrez@aut.ac.nz

**Abstract.** This paper describes a new comparison-based model for fault diagnosis in wireless ad hoc networks. Fault diagnosis is crucial for ensuring the dependability of systems. Wireless ad hoc networks are highly prone to faults as consequence of their dynamical conditions. The comparison approach is a practical diagnosis model that has been used to develop self-diagnosis systems in wired and wireless networks. This approach can detect and diagnose hard and soft faults in systems. The traditional fault diagnostic models were designed for static networks. Thus, they cannot provide complete and correct fault diagnosis in mobile wireless networks. In this paper, we introduce a time-free self-diagnosis model that respects the design requirements of mobile wireless networks. That is, it adapts to the topology's changes, it imposes no known bounds on time delays, and it requires limited network information. Further, we develop a fault diagnosis protocol that can correctly diagnose faulty nodes undergoing static and dynamic faults in mobile ad-hoc networks (MANETs). Both an analytical model and a simulation study have been used to prove and evaluate the efficiency of our protocol under various scenarios. Furthermore, the performance of our protocol is compared with related protocols. The results show that our proposed protocol is efficient in terms of communication and time complexity.

**Keywords** Dependability. Wireless ad-hoc networks. Dynamic faults. Fault diagnosis. Self-diagnosis.

## 1. Introduction

The design of dependable mobile wireless networks has emerged in response to the rapid proliferation of applications and services provided by such networks. These networks include mobile ad-hoc networks (MANETs), vehicular ad hoc networks (VANETs), wireless sensor networks (WSNs), and wireless mesh networks (WMNs), to name but a few [1-4]. They have been deployed and operated in critical and risky environments, such as battlefield operations, emergency rescues, and healthcare [5, 6]. Hence, undependable networks may endanger peoples' lives and wealth. Considering their characteristics and deployment environments, mobile wireless networks are more vulnerable to dependability's impairments, i.e., faults, errors and failures [7-9]. Therefore, it is necessary to deal with these impairments, and ensure that the system can deliver the expected services. A fault is a defect in the system's design or operation. The fault may cause an error, i.e., an incorrect output of the system. The error may lead to a failure in the system, and hence the system cannot operate as expected. It is clear that faults are the sources of other impairments. Thus, fault diagnosis may stop or reduce the occurrence of errors and failures [9].

Fault diagnosis is the process of detecting, identifying, and locating faults in a system. The literature studies various fault diagnosis techniques proposed for mobile wireless networks [10-14]. In particular, the system-level fault diagnosis has been investigated extensively to build self-diagnosable systems and procure the dependability of wire and wireless networks [14, 15]. The system-level diagnosis problem aims at automatically diagnosing faulty nodes. Many models have been proposed to solve this problem. The most popular one is the so-called comparison model [16]. This model has reasonable constraints on the system considered, and hence it has been recognised as an effective diagnosis model [17]. In this model, informally, a node appoints a task to different nodes, and the output results from the nodes are compared. The node, then, can identify faulty nodes depending on the comparison's outcome. The idea behind this model is that fault-free nodes executing the same task will generate the same results. Faulty nodes, however, may generate either different results like the asymmetric model [16] or inconsistent results like the symmetric model [18]. The comparison outcome could be either 0 or 1, where 0 indicates identical results and 1 indicates different results. Formally, self-diagnosis protocols are characterised by two properties: (1) Correct and (2) Complete. A self-diagnosis protocol is correct if no fault-free nodes are erroneously classified as faulty, and it is complete when all faulty nodes are correctly identified [17, 19].

In the literature, the Chessa and Santi model [20] is the seminal work that launched the development of comparison-based self-diagnosis protocols for ad-hoc wireless networks. Their model utilises the broadcast communication nature of ad-hoc networks to reduce the diagnosis overhead. More specifically, this model exploits the one-to-many communication paradigm to exchange diagnosis messages. However, this model cannot tolerate topology changes. To overcome this problem, the Chessa and Santi model was adapted for time-varying topologies [21]. In this modification, when a node replies, it includes the test task along with the test result; therefore, any nodes receive the reply message can diagnose the fault. However, the characterizations of diagnosable systems under these models are not suitable for mobile ad hoc networks. These models are timer-based models assuming the time delay is bounded and known. Additionally, they assume that nodes know the whole system's memberships and the maximum number of faults. These assumptions are intolerable in mobile wireless networks where communication delays fluctuate, and a global knowledge about the system is impractical. The diagnosis approach should respect the intrinsic characteristics of mobile wireless networks such as unsteady topologies, asynchronous communications, and limited knowledge about the network. In dynamical environments, systems should be distributed self-diagnosable (DSD), i.e., each fault-free node takes part in a diagnosis process and correctly identifies the status of all nodes in the system [7]. Therefore, this paper considers the system-level fault diagnosis problem in mobile wireless networks.

This paper provides three main contributions. First, it introduces a novel comparison-based diagnostic model for mobile wireless networks. The early idea of this model has been presented in [22]. This model is inspired by asynchronous implementations of failure detectors [23-25], and the computation model for dynamic systems [26]. The proposed model exploits message exchange patterns rather than timers to identify the faulty status of nodes and thus it is suitable for asynchronous systems and mobile wireless networks. The time-free diagnostic model has the following advantages: (1) time delays are unknown but finite, (2) adaptive with nodes' movement; (3) no global knowledge is required. Second, it characterizes the class of systems that are diagnosable under this model. Particularly, it describes the essential assumptions that a diagnosable system must hold. Third, it presents a diagnosis protocol as a proof-of-concept of our proposed model. The time-free fault diagnosis protocol can identify dynamic faults in MANETs. The protocol's proof of correctness, analytical analysis, and performance evaluation has been described in this paper.

The remainder of the paper is organised as follows. Section 2 illustrates the system model as well as the fault model. Section 3 presents the time-free diagnostic model for mobile wireless networks. Section 4 studies existent models showing the main contributions of our proposed model. Section 5 presents a time-free fault diagnosis protocol and Section 6 shows its correctness proof and complexity analysis. Section 7 presents the performance evaluation, and the analytical and simulation results obtained. Section 8 interprets and describes this research 's significance, implications and limitations. Finally, a brief discussion in Section 9 concludes the paper.

## 2. System and Fault Models

### 2.1. System Model

This research considers dynamic systems such as MANETs, WMNs, and WSNs. The system has nodes communicating via packet radio networks. This system is an asynchronous system. In this sense, it has no strong constraint on time. Notably, there are no known upper bounds on the speed of nodes; on message transmission delays; or on node's computation time. There is no global clock known by the nodes. However, we use the set of natural numbers like a clock's tick to represent the system's lifespan, $\mathcal{T} \subseteq \mathbb{N}$. It is worth pointing out that $\mathcal{T}$ is only introduced to ease the demonstration of the system's properties and proofs.

The system includes infinitely many nodes [27]. However, each run consists of a finite set $\prod$, where, $|\prod| \geq 3$, namely, $\prod = \{v_1, v_2, \ldots\ldots, v_n\}$, where $v_i$ represents node $i$. The nodes may join and leave the system anytime. Each node has a unique identifier $ID$, and it is reasonable to assume that a node may know the $ID$s of its neighbours using the communication medium.

We represent the mobile wireless network by a communication graph $\mathcal{G}$ with a dynamic topology. In this sense, the connections appear among nodes over the time $\mathcal{T}$. We describe the system's dynamics as a time-varying graph $\mathcal{G} = (V, E, \mathcal{T}, \rho, \varsigma)$, where $V = \prod$ is the set of mobile nodes; $E \subseteq V \times V$ represents the set of links between nodes; $\rho\colon E \times \mathcal{T} \to \{0, 1\}$ is the presence function, which indicates whether a given link is available at a given time; $\varsigma\colon E \times \mathcal{T} \to \mathbb{N}$ is the latency function, which indicates the time to pass a given link. Nonetheless, $\varsigma$ is unknown because the system is asynchronous [1].

Assume $TR_v$ denotes the transmission range of a node $v, v \in V$. It is assumed that the transmission ranges of nodes are equal and perfectly circular. Every node within $TR_v$ belongs to $v$'s neighbour set $N_v$ at time $t \in \mathcal{T}$. Furthermore, there is a two-way connection between neighbour nodes; a node $u \in \mathrm{N_v}$ iff $(v,u) \in E_v$, hence, $\rho\,((v,u),t) = 1$. The degree of $v$ is, $\deg(v) = |E_v|$. The neighbour nodes may vary frequently since they are mobile and prone to faults.

Assume a fixed graph $G = (V, E)$, a footprint of $\mathcal{G}$, describes the system at time $t \in \mathcal{T}$. The graph $G$ illustrates the pair of nodes that have relations at the time $t$. A graph $G' = (V', E')$, a subgraph of $G$, describes the relations between fault-free nodes in $G$ at time $t$. We assume that $G'$ complies with the connectivity assumption, namely Assumption 1. This assumption ensures that, in spite of changes in the topology of $G'$, $V'$ is connected over time. The Assumption 1 is a crucial condition to maintain the properties of fault self-diagnosis protocols, i.e., correctness and completeness. Let $D_G$ denotes the diameter of $G$, $\delta_G$ and $\Delta_G$ respectively denote the maximum degree of vertex and the minimum degree of vertex over the diagnostic session.

***Assumption 1. Connectivity over Time****: Let $G' \subseteq G$ be a subgraph that contains the fault-free nodes in $G$ at time $t$. Then, there must be at least one path between every two nodes $u, v \in G'$. That is, $\forall\, u, v \in V'$, $u \rightsquigarrow v$.*

We assume that nodes in $\prod$ are mobile. Thus, the nodes may continually move and pause. The neighbourhood of a mobile node $v$ may change once $v$ moves. In addition, we assume that $v$ follows the *passive mobility model [23]*, which means $v$ does not know that it is moving. Hence, it cannot inform its neighbours about its mobility. As a consequence, the neighbours of $v$ cannot distinguish whether $v$ has migrated out or is undergoing a fault.

### *2.2. Fault Model*

Traditionally, a fault model describes two aspects of the faults that may appear in the system so that the faulty nodes can be identified: the type of faults; and the maximum number of faults [28, 29]. In this subsection, we define the fault model considered in this research.

We assume that links are reliable and non-faulty; they do not create, alter, or lose messages. On the other hand, nodes are subject to hard and soft faults. A hard fault (e.g., fail-stop, fail silence, and crash) disrupts communications between a faulty node and other nodes. A hard fault is a result of dead battery conditions or a node crash in wireless networks. On the contrary, a soft fault alters the behaviour of a node without interrupting the communications with other nodes. Faults may happen during the diagnostic session, and these are referred to as dynamic faults [7]. More specifically, the fault is diagnosable if there are neighbour nodes that have not started their diagnosis process when the fault occurs. It is assumed that faults are benign; hence, byzantine and malicious faults are a matter for future works.

In the literature, a system is $\sigma$-diagnosable if $\sigma$ is the maximum number of faulty nodes that a system can guarantee to diagnose [30]. The diagnosability of a system,$\sigma$, is bounded from above by the minimum vertex degree of the graph $G$, denoted by $\delta_G$, i.e., $\sigma \leq \delta_G - 1$. The logic behind this bound is that if the number of faulty nodes, $\sigma$ , exceeds $\delta_G - 1$, then the system will be disconnected; hence, the system's diagnosability is not guaranteed. However, such global knowledge about the diagnosability of the system is improbable for mobile wireless networks. Therefore, this research considers a local fault model that has been introduced in the literature as a suitable strategy for dynamic environments. In the local fault model, bounds on the maximum number of local faults are defined so messages could be reliably delivered. That is, faults are locally bounded [31, 32]. We consider that $\sigma_v$ is the maximum number of faulty nodes in $v$'s neighbourhood. The $\sigma_v$ is bounded by the degree of the node $v$, $deg\ (v)$, i.e., $deg(v) > \sigma_v$. In case of the reliable broadcast model [33, 34], the bound should be, $deg(v) > 2\sigma_v$. This is because, it is not possible to achieve reliable broadcast if half or more of the nodes are faulty [33, 34].

***Definition 1. Local Diagnosability****: A mobile wireless network is locally $\sigma$-diagnosable at node v if each fault-free node can unambiguously identify the fault status of all nodes given that the number of neighbour faulty nodes does not exceed $\sigma_v$.*

Each fault-free node, either mobile or stable should be able to reply to $\sigma + 1$ test requests within the first $\alpha$ replies. This assumption implies that every fault-free node will be correctly diagnosed by at least one fault-free node. That is, fault-free nodes are winning nodes and achieve Assumption 2.

***Assumption 2. Winning Nodes****: Each fault-free node, $v$ has a set, $Q_v \subset N_v$ and $Q_v \neq \emptyset$, which can communicate with $v$ faster than with the other nodes.*

In other words, this assumption states that there is a faultless node, $v$ and a set $Q$ of $\sigma_v + 1$ neighbour nodes so that after a time, $t$ each node $u \in Q$ receives a winning reply from $v$ or $v$ is faulty. This behavioural assumption of the system should hold to ensure the required diagnosis. This assumption has been considered in time-free models instead of the synchrony assumptions in timer-based models. It places a constraint on the logical time of message delivery among nodes and ensures a stability behaviour that should be satisfied for sufficient time to perform the diagnosis process. Unstable situations caused by fast node mobility, short period of connections and numerous joins and leaves may prevent any useful diagnosis. However, mobile nodes tend to satisfy these assumptions, and these assumptions should be held only during the diagnosis session.

## 3. Time-Free Comparison-based Diagnostic Model

The time-free diagnostic model is a comparison-based model that has no constraints on the delay time, and it uses no timers. We believe that the proposed model is suitable for mobile wireless networks. In this section, we describe the proposed diagnostic model.

The proposed diagnostic model uses the comparison approach to identify the faulty status of nodes. Nonetheless, the model takes into consideration the following issues that may arise in mobile wireless networks: (1) nodes are mobile, and the topology may change; (2) communications are asynchronous, and (3) fault timing is unpredictable.

We assume that tasks are complete in the sense that they have perfect fault coverage. This strong assumption could be relaxed employing a probabilistic fault coverage, and this is a matter for future work. It is noteworthy that the design of such tasks is beyond this paper's scope and could be a matter for future research. We assume, additionally, that fault-free nodes assigned the same task always produce identical outputs and faulty nodes always produce different outputs, i.e., asymmetric model. Moreover, every fault-free node can compare the outputs and generate a comparison outcome. The comparison outcome could be 0 if the outputs are the same and 1 otherwise. On the other hand, a comparison outcome produced by a faulty node could be 0 or 1, irrespective of the outputs generated by tested nodes. The time-free diagnostic model employs the asymmetric invalidation model shown in Table 1 [35].

**Table 1** Comparison Outcomes for the gMM Model [35]

| **Nodes under comparison** | **Node Performing the comparison** | |
|---|---|---|
| | **Fault-Free** | **Faulty** |
| None is faulty | 0 | 0 or 1 |
| One is faulty | 1 | 0 or 1 |
| Both are faulty | 1 | 0 or 1 |

Even though the network topology varies during the diagnosis session, and the set of neighbour nodes of a node $u$ may change over time; $N_u\,(t) \neq N_u\,(t')$, where $t' > t$, fault-free nodes satisfy Assumption 1. Otherwise, complete and correct self-diagnosis is endangered.

A node experiencing dynamic fault has uncertain status during the diagnosis session. Therefore, tester nodes diagnosing the node at a different time will have contrasting comparison outcomes about the same node. Note that it is crucial for correct self-diagnosis to determine the order in which the outcomes are generated. However, since the nodes in the system do not have synchronised clocks, it is assumed that each node maintains a logical clock that tracks the order of events rather than ticking as with real-time clocks. Accordingly, each tester node associates its comparison outcomes with its current logical time $ct$. This time, however, is a local logical time at a node and can be used to determine the order in which outcomes have been generated [36]. It is noteworthy that this time is not related to the actual time.

The proposed model does not use timers to stop waiting in perpetuity for responses. Instead, it exploits messages' exchanging patterns. Each node $u$ waits for $\alpha_u$ distinct replies. The $\alpha_u$ is a local parameter, and its value should be chosen carefully to reflect the expected number of nodes communicating with $u$, despite faulty and moving nodes. The value of $\alpha_u$ is crucial for our model because it stops the node $u$ waiting forever. The value of $\alpha_u$ depends on the density of neighbourhood of $u$ and the maximum number of faults in $u$'s neighbourhood. That is, $\alpha_u = |N_u| - \sigma_u$. Knowing that $|N_u| > 2\sigma_u$, and $\alpha_u \geq \sigma_u + 1$ [23, 34, 37]. Indeed, the value of $\alpha_u$ depends on the network type and the current topology of the network. Thus, it can be computed on the run considering the underlying network behavior.

The time-free diagnostic model relies on the following time-free comparison protocol to perform comparisons:

- **Generate a test request:** A node $u$ creates a test request $T_i$, where $i$ is an integer number depicting the test number. Next, it broadcasts the test request message, $m = (TEST, T_i)$, where $TEST$ indicates the message type and $T_i$ represents a test task. Afterwards, $u$ waits for responses from $\alpha_u$ nodes. Noticeably, $u$ uses no timers.
- **Receive a test request:** Once a node $v$ receives the test request message $m$ from $u$, it produces the result $R_v{}^u$ of the task $T_i$. Then it broadcasts the test response message, $m' = (RESPONSE,\ T_i, R_v^u)$. After that, the node $v$ starts its diagnosis session by generating its test request message, if it has not sent a request yet. As we consider a dynamic topology, $v$ could be a non-neighbor node, i.e., $v \notin N_u$, that moved in $u$'s transmission range.
- **Receive a test response:** Consider a node $w \in V$. Upon receiving responses from $\alpha_w$ nodes, $w$ stops waiting. Then, $w$ takes either of the two following actions for every $v \in \alpha_w$:
  - *Case 1:* $w$ knows the expected result of the task $T_i$; it compares them. If $R = R_v^u$ , then $w$ can conclude that $v$ is fault-free; hence, $v$ will be added with an associated timestamp to the list of fault-free nodes diagnosed by $w$, i.e., $FF_w = FF_w \cup \{v, ct\}$. Otherwise, $v$ is added to the list of faulty nodes with a timestamp $ct$, i.e., $F_w = F_w \cup \{v, ct\}$.
  - *Case 2*: $w$ does not know the expected result of the task $T_u$. Hence, $w$ executes the task $T_u$ first and then compares the result with $R_v^u$. If the comparison outcome is 0 then, it will add $v$, with a timestamp $ct$, to the fault-free list, $FF_w = FF_w \cup \{v, ct\}$. Otherwise, $v$ will be added to the faulty nodes list, $F_w = F_w \cup \{v, ct\}$.

**Theorem 1.** *Let $G = (\mathcal{V}, \mathcal{E})$ be a graph that represents a locally $\sigma$-diagnosable mobile wireless network at a time t when the diagnosis session is started. Then, the following statements are correct when a faultless node, $u \in \mathcal{V}$ performs the time-free comparison protocol:*

- *The node u correctly diagnoses the status of $\alpha_u$nodes in a finite time.*
- *The latest status of the node $u \in \mathcal{V}$ is correctly diagnosed and associated with the greatest timestamps by at least one non-faulty neighbour node, $v \in \mathcal{V}$.*

**Proof**. Based on the time-free comparison, once the diagnosis session starts at time $t$, then a faultless node,$u$ sends a test request message at a time $t' \geq t$. The test request message stimulates the neighbour nodes, $N_u$ to start their own diagnosis and to send their test response messages. First, since the network is locally $\sigma$-diagnosable, there are at least $\alpha_u$ neighbour nodes that can send a test response within a finite unknown time. Therefore, at a time $t'' > t' + T_\alpha$, where $T_\alpha$ is the time required to collect $\alpha_u$ distinct replies, $u$ correctly diagnoses the status of the $\alpha_u$ nodes, given that $u$ is fault-free node. Thus, the first part of this theory holds. Notice that, because of the topology changes, a mobile node $v \in (N_u(t'') - N_u(t'))$ replied within the first $\alpha_u$ responses; hence, it was diagnosed correctly by $u$. On the other hand, a mobile node $w \in (N_u(t') - N_u(t''))$ moved away; hence, it did not reply within the first $\alpha_u$ responses and, as a consequence, was added erroneously to $u$'s faulty list, $F_u$. Second, because the neighbour nodes start their own diagnosis by sending a test request message, the test responses, which $u$ send, will be received by at least $\sigma_u + 1$ neighbour nodes. This is because of the fact that $u$ is a winning node based on Assumption 2. This means that the correct status of $u$ will be diagnosed by one fault-free node at worst. Notice that, in the case of dynamic faults, a node $v$ which is diagnosed as fault-free at time $t''$ by a node $u$ could be diagnosed as faulty at time $t''' > t''$ by another node $w$. Here, two nodes have different views about the same node. Since each node timestamps its partial view, then the latest decision is held by one fault-free node with higher timestamps. Thus, the second part of the theorem holds.

Section 5 describes, as a proof of concept, the capability to design and develop fault diagnosis protocols for mobile ad-hoc networks using the time-free comparison protocol.

## 4. Comparison with Related Models

To date, several comparison-based diagnosis models have been introduced [16, 18, 20, 21, 35]. Their implementations are applied to identify faulty nodes in many applications, particularly wire and wireless networks [17]. However, these models have some drawbacks when it comes to mobile wireless networks. This section discusses the advantages and the disadvantages of our proposed model compared to existing models. It also describes how the proposed model satisfies the requirements of mobile wireless networks.

The existing diagnosis models use fixed timeouts to diagnose hard faulty nodes. Considering the dynamic nature of mobile wireless networks, the main drawback is how to choose the timeout value. If the timeout time is too short, fault-free nodes may not be able to reply within this time due to unreliable wireless links or long computation times; hence, they will be diagnosed as faulty. On the other hand, if the time is too long, the diagnosis latency will be too long, and the status of nodes as well as the topology of the network may be changed during this time; hence, the diagnosis decisions may not be accurate. It is clear that this is a complex problem in terms of implementation complexity. Thus, the existing diagnosis models impose constraints on communication and computation delays and topology changes. These constraints assure that the nodes, which did not reply before the timer expiration, can be safely diagnosed as faulty. In addition, these models assume that nodes have some global information about the whole network such as the number of nodes, the connectivity of the network and the maximum number of allowable faults. However, these assumptions are impractical for mobile wireless networks where topology intrinsically varies, communications are asynchronous and faults are inevitable.

On the other hand, the proposed diagnosis model uses message exchange patterns instead of timers. That is, each node, $u$ waits until receiving a number, $\alpha_u$ of responses rather than using a timer. This parameter represents a logical deadline when $u$ stops waiting for responses. We believe that this mechanism suits asynchronous systems and mobile networks. Even though the number of responses that a node should wait for, $\alpha_u$ is crucial for implementation, its value could be defined on the fly. Moreover, this model depends on local information to diagnose the system, which resembles the distributed computation. That is, it proposes a local fault model instead of a global fault model; hence no global information is needed. In addition, it tolerates topology changes.

In addition, various implicit and explicit assumptions should be maintained in order to leverage the time-free diagnosis model. In particular, Assumptions 1 and 2 presented in Section 2 are necessary. The diagnosis models in the literature have imposed Assumption 1. It is inevitable since the lack of connectivity prevents the exchange of diagnosis messages among nodes; hence the completeness and the correctness properties are not guaranteed. Assumption 2 is also necessary in order to overcome the absence of synchrony assumptions and to ensure the correctness of the diagnosis process. Assumption 2 places constraints on the logical time of the message's delivery among nodes. However, it depends on the value of $\alpha_u$ that can be defined locally and on the fly by each node. In practice, these assumptions are to be held only during the diagnosis process. A range of networks that may satisfy these assumptions include WMNs, WSNs, VANETs and dense MANETs.

The proposed model considers, as do the most relevant models, a test task that provides complete fault coverage. While this assumption is difficult to satisfy, comprehensive tasks that consider specific faults might attain this assumption. Relaxing this assumption by considering a probabilistic fault model is of interest because it, too, helps with handling intermittent faults. This issue is a matter of ongoing research. **Error! Reference source not found.**2 summarises the comparison among our proposed diagnosis model and the two most relevant diagnosis models in the literature.

To sum up, mobile wireless networks experience intrinsically dynamic topology, unreliable links and change of connectivity. Unlike existing diagnosis models, the proposed time-free diagnosis model tolerates these characteristics and adapts with the mobile networks; therefore, we believe that it should be used to design and implement efficient fault diagnosis protocols for mobile wireless networks. The following section presents a fault diagnosis protocol, which implements the time-free diagnosis model and diagnoses static and dynamic faults in mobile wireless networks.

**Table 2** Comparison between our proposed diagnosis model and the most related models

| | **Chessa & Santi Model [20]** | **Elhadef et.al, Model [21]** | **Time-Free Model (Our model)** |
|---|---|---|---|
| **Diagnosis approach** | Comparison approach | Comparison approach | Comparison approach |
| **Network topology** | Fixed | Fixed and dynamic | Fixed and dynamic |
| **Asynchronous communications** | Intolerable | Intolerable | Tolerable |
| **Timers** | One timer | Two timers | No timers |
| **Fault types** | Permanent, soft and hard | Permanent, soft and hard | Permanent, soft and hard |
| **Fault time** | Static | Static | Static and dynamic |
| **Node knowledge** | Global | Global | Local |
| **Test Task** | Complete | Complete | Complete |

## 5. A Fault Self-Diagnosis Protocol for MANETs

This section introduces a distributed self-diagnosis diagnosis protocol (DSDP) for MANETs with the presence of dynamic faults, called Time_Free-DSDP. This new diagnosis protocol provides a complete and correct diagnosis for MANETs. Fig. 1 shows the main phases of the proposed Time-Free-DSDP protocol at a node $u$.

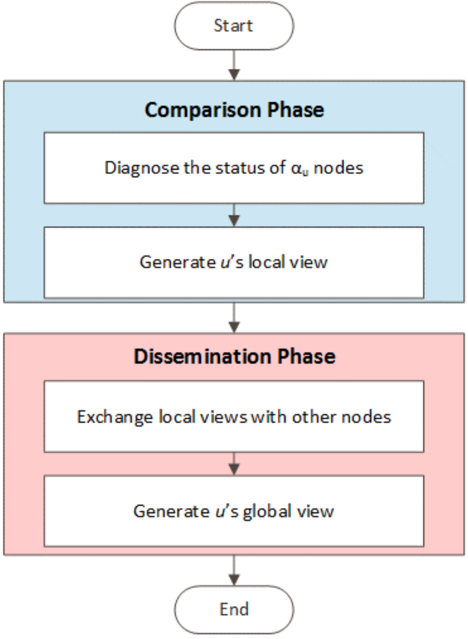
Start
Comparison Phase
Diagnose the status of $\alpha_u$ nodes
Generate $u$'s local view
Dissemination Phase
Exchange local views with other nodes
Generate $u$'s global view
End

**Fig. 1** The main phases of the Time-Free-DSDP protocol at node *u*.

The diagnosis session starts when a fault-free node executes its diagnostic protocol and ends when every fault-free node terminates the execution of its diagnostic protocol. This protocol satisfies the time-free diagnostic model and its specifications. Either every fault-free node executes the protocol systemically or once it detects an abnormal event. The protocol consists of two phases, namely, a comparison phase and a dissemination phase. During the comparison phase, fault-free nodes diagnose their neighbour nodes using the time-free comparison protocol stated in Section 3, and then they produce their local views, which include the status of adjacent nodes along with their timestamps. Subsequently, they exchange these views with the other nodes by using a flooding mechanism. Therefore, fault-free nodes share a correct and complete view of the network after the dissemination phase.

### *5.1. The Comparison Phase*

A diagnosis session at a node $u$ commences with a generation of a test task $T$. $u$ then broadcasts a test request message, $m = (TEST, T_u)$. One-hop nodes $(N_u)$ at that time send back their test response messages, $m' = (RESPONSE, T_u, R_v^u)$ and start their diagnosis session if they have not started yet. Note that the set of neighbour nodes $N_u$ changes over time due to the network dynamics. Thus, a new migrant node may receive this message while a previous neighbor node may not receive the message due to its mobility. The node $u$ performs the comparisons based on the time-free comparison protocol to identify the faulty status of its neighbours at that time. That is, if the node $u$ knows the expected result, then it compares that result with the received result from a node $v$. Matching results mean that the node $v$ is fault-free. Otherwise, it is faulty. Observe that as long as the node $u$ has two or more responses for the same task; then there is no need to execute the task to know the status of these nodes. On the other hand, unexplored tasks need to be executed and compared as shown before. Immediately upon receiving replies from $\alpha_u$ nodes, the node $u$ maintains two sets, $FF_u$ *and* $F_u$, including fault-free nodes and faulty nodes, respectively. Each element in these sets contains the node $ID$ with its associated timestamp $ct$ on the form $(ID, ct)$. Nonetheless, neighbour nodes that do not belong to $\alpha_u$ will be added to the faulty set $F_u$.

The main contribution here is that the protocol uses the number of replies $\alpha_u$to stop waiting forever, i.e., the protocol relies on the messages' pattern rather than a timer to identify the status of nodes. This mechanism eases the implementation of this protocol in asynchronous dynamic systems. As a consequence of that, slow fault-free neighbour nodes might be diagnosed as faulty. Eventually, this decision will be reformed with a higher timestamp by another fault-free node. The reason is that we assume every fault-free node is a winning node and achieves Assumption 2 stated in Section 2.

### *5.2. The Dissemination Phase*

After generating the local view, $d_u = (LOCALVIEW, F_u, FF_u)$, $u$ propagates $d_u$ and collects others' dissemination messages in order to produce a complete view of the network. Upon receiving a dissemination message from a fault-free node, $u$ performs the following actions: (1) it updates its faulty and fault-free sets taking into account the decision having the highest timestamp *ct* for every known node. Unknown nodes, however, are appended with their timestamps; and (2) it propagates this message to its one-hop neighbours. Dissemination messages having no new information or generated by faulty nodes will be discarded.

Topology dynamics may result in false decisions. For example, a mobile node $u$ may receive a test request message from a neighbour node $v \in N_u$ and reply to $w \notin N_u$. That is, $v$ is no longer a one-hop away from $u$ while $w$ is within the transmission range of $u$. Thus, $v$ will consider $u$ as a faulty node. However, eventually, the decision of $v$ will be rectified when $w$ disseminates its local view about $u$ with a higher associated timestamp.

Dynamic faults induce the production of contradictory views about the same node. This is not because any of them is mistaken, but because the state of the node changes. Here, knowing which decision was made later is crucial in determining the last status of a node experiencing a dynamic fault. If the final state of a node is diagnosed correctly by $\sigma_u$ fault-free nodes, all fault-free nodes will hold the correct decision after the dissemination phase because all of them will consider the latest decision.

## 6. Protocol Correctness and Analysis

This section first presents the correctness proof for the Time_Free-DSDP. It then shows the communication and time complexity of the Time_Free-DSDP protocol.

### *6.1. Proof of Correctness*

In this subsection, we prove that the Time_Free-DSDP protocol satisfies both the completeness and the correctness properties of distributed self-diagnosis protocols. The proof correctness rests on a couple of properties: (1) Partial correctness: the final faulty status of any node is diagnosed by at least one fault-free node; and (2) Complete correctness: every fault-free node will correctly receive the local view of the other fault-free nodes in the system. In the following, we prove that the Time_Free-DSDP satisfies the partial correctness at the end of the comparison phase and the complete correctness at the end of the dissemination phase.

**Lemma 1.** *Suppose that a diagnosis session is commenced at time t, then, the last node will receive the first diagnostic message and generate its test request in at most* $t + D.T_g$ *time.*

**Proof.** Let $T_g$ be the time required by a node to generate its test task after receiving the first diagnostic message. Since $D$ is the maximum diameter of $G$, it follows that in at most $D.T_g$ time, every fault-free node will generate its test request.

**Lemma 2.** *(Partial correctness) If a diagnosis session has been commenced, then the latest faulty state of each node will be correctly diagnosed by a single fault-free node at worst.*

**Proof.** The lemma follows directly from Lemma 1 and Theorem 1. That is, eventually all fault-free nodes will generate a test request and execute the time-free comparison protocol within a finite time, by Lemma 1. Then, the partial correctness is straightforward from Theorem 1.

**Lemma 3.** *(Complete correctness) Every fault-free node correctly receives the partial view generated by each fault-free node in a finite time.*

**Proof.** By Assumption 1, there is a path $P$ in the time passing through fault-free nodes between each of the non-faulty nodes, $u$ and $v$. We need to prove that once $u$ propagates its dissemination message $d$, then $v$ correctly receives $d$ in a finite time. By induction on $P$'s length, $l(P)$, if $l(P) = 1$, then $v$ and $u$ are neighbours. Therefore, it will correctly deliver $d$ to $v$, and then $v$ can update its lists considering the most up-to-date information if $u$ has been diagnosed as fault-free. So, the claim is valid. In the case where $v$ does not know the state of $u$, then it stores $d$ until recognising the status of $u$. Observe that $u$ is a fault-free node and eventually will be correctly diagnosed by a fault-free node according to Assumption 2. Eventually, $u$ will be diagnosed as fault-free and $d$ will be adequately considered by $v$, and the claim holds. if $l(P) = h, h >= 2$; That is, $v$ and $u$ are not neighbours. By the induction step, the claim is valid for node $w$ at the $h - 1$ position. So, $w$ will update its lists and broadcast $d$ to its neighbours. $v$ is a neighbour to $w;$ hence by the inductive hypothesis, $v$ will correctly collect $d$ in a finite time, and the lemma holds.

**Theorem 2.** *Each fault-free node correctly has the most up-to-date information about the faulty status of nodes in the system in a finite time.*

**Proof.** At the end of the comparison phase, the most recent information about the nodes in the system is held by at least one non-faulty node, and that follows from Lemma 2. Later on, nodes share their partial views and keep the most up-to-date information in a finite time by Lemma 3. Thus, the theorem holds.

### *6.2. Complexity Analysis*

This subsection studies the performance of the protocol in terms of communication and time complexity. The first concept points out the number of one-hop broadcasts disseminated during the diagnostic session while the second concept indicates the duration of that session.

**Theorem 3.** *The communication complexity of the Time_Free-DSDP protocol is* $O(n^2)$*, where n is the number of nodes in the network.*

**Proof.** Each fault-free node generates solely one test request message, $m$. As a result, this message triggers no more than $\Delta_G$ test response messages, $m'$, where $\Delta_G$ denotes the maximum vertex degree. Further, each fault-free node generates only one dissemination message, $d$. However, every fault-free node propagates others' $d$ once at most in the worst case. Hence, the overall number of one-hop broadcasts is as follows: $n$ test request messages, $n.\Delta_G$ test response messages and $(n(n-1)+n)$ dissemination messages. Thus, the total is $n\,(n + \Delta_G + 1)$ and the communication complexity is $O(n^2)$.

In the following, we express the time complexity in terms of (1) $T_g$ is the time interval between receiving the first diagnostic message and producing the test request message, $m$. (2) $T_d$ is the time required to disseminate a message. (3) $T_\alpha$ is the time required to collect $\alpha$ responses.

**Theorem 4**. *The time complexity of the Time_Free-DSDP protocol is* $O(D(T_g + T_d) + T_\alpha))$.

**Proof.** For analysis purposes, the last node will generate its test request message at most in $T_g$. It follows that all fault-free nodes will generate their test request messages in at most $D.T_g$. Each fault-free diagnoses $\alpha$ nodes and generates its local view in at most $T_\alpha$. Thus, the last dissemination message is generated by the time $D.T_d + T_\alpha$. Hence, the last node to receive this message will do so on no more than $D.T_d$. Thus, the total time needed is $D(T_g + T_d) + T_\alpha$, and the theorem holds.

## 7. Performance Evaluation

In this section, we evaluate the performance of the proposed Time_Free-DSDP protocol by extensive simulations using OMNeT ++ simulator [38]. OMNeT++ has been used due to its availability and credibility [39, 40]. We compare the performance of the Time_Free-DSDP against the Static-DSDP and the Mobile-DSDP protocols. These three protocols use simple flooding mechanisms to propagate the local view of nodes. To perform a fair comparative study, we have subjected the Static-DSDP and the Mobile-DSDP protocols to the same set of experiments. It is noteworthy that both the Static-DSDP and the Mobile-DSDP cannot correctly diagnose dynamic faults. Moreover, unlike the Mobile-DSDP and the Time_Free-DSDP, the Static-DSDP protocol does not tolerate topology changes. In addition, we validate the analytical model via simulations.

### *7.1. Performance Metrics*

To evaluate the performance of the proposed Time_Free-DSD protocol, we consider two metrics, namely, the number of diagnosis packets (communication complexity) and the diagnosis latency (time complexity).

The former metric represents the total number of diagnosis messages that has been exchanged during the diagnosis session including test request messages, test response messages and local view messages. The latter metric refers to the time between the beginning and the end of the diagnosis session. Undoubtedly, the lower the number of packets exchanged during the diagnosis session, the more efficient the protocol is. Likewise, a shorter diagnosis latency means better performance.

### *7.2. Description of Scenarios*

We designed three network scenarios to evaluate the performance of the protocols under different circumstances as follows. Table 3 summarises the configurations of the scenarios.

- **Scenario 1**: The primary objective of this scenario is to evaluate the efficiency and scalability of the protocols for various network sizes. Hence, we studied 10 networks having a number of nodes which varied from 10 to 100. Nodes deployment follows the random distribution. In each network, 10% of nodes are faulty. All three protocols (including Time_Free-DSDP) were subjected to this scenario.
- **Scenario 2:** The aim here is to measure the efficiency for detecting both static and dynamic faults. Therefore, a network consists of 80 nodes exposed to some faults, ranging from two to 30 and increased by four faults each time. The network has a fixed topology with connectivity of 36. In the case of the Time_Free-DSDP, we examined the performance for both static and dynamic faults. Nonetheless, the performances of the other two protocols were studied solely on static faults.
- **Scenario 3**: In this scenario, we studied the impact of a dynamic topology on the efficiency of the Time_Free-DSDP and the Mobile-DSDP protocols. That is, we examined a network comprising 50 nodes including some mobile nodes, ranging from 2 to 10. The movement of the mobile nodes led to topology changes. A range of static and dynamic faults from 1 to 5 has been considered.

- In the simulations, a node having a static fault may not participate in the diagnosis session, while a node experiencing a dynamic fault might have replied previously (it had responded to some test requests) before being faulty (it stopped responding, or its answer had an incorrect result). Moreover, we developed a simple mobility model that moves the required nodes and changes the network topology during the diagnosis session. In addition, it maintains the constraints described in the system model above. Considering the Time_Free-DSDP, each node may have different range densities and a local number of faults. Thus, the value of α can be locally computed on the fly by each node as previously demonstrated. The simulations were carried out and repeated 10 times with different random seed numbers to provide a confidence interval of 95%. Hence, the simulation results, reported in the next subsection, represent the average value for every measurement and the confidence interval. Note that in some figures the confidence intervals' bars may not be clearly visible due to the very insignificant error levels.

**Table 3** Simulation Scenarios

| **Notation** | **Scenario 1** | **Scenario 2** | **Scenario 3** |
|---|---|---|---|
| Protocols | Static-DSDP, Mobile-DSDP,<br>Time-Free-DSDP | Static-DSDP, Mobile-DSDP,<br>Time-Free-DSDP | Mobile-DSDP<br>Time-Free-DSDP |
| # of Nodes | 10 - 100 | 80 | 50 |
| # of Mobile nodes | 0 | 0 | 2 - 10 |
| Network topology | Fixed | Fixed | Dynamic |
| Fault types | Static | Static & Dynamic | Static & Dynamic |
| # of Faults | 1 - 10 | 2 - 30 | 1 - 5 |
| Area size | 600m×600m | 300m×300m | 300m×300m |
| Transmission range | 150m | 150m | 150m |

### *7.3. Simulation Results*

This subsection presents the results obtained from the scenarios explained in the previous subsection. In what follow, we describe and discuss the simulation results. Table 4 summarises the keys of figures' plots.

**Table 4** Figure plot keys

| The Key | Description |
|---|---|
| Static-DSDP | The Static Distributed Self-Diagnosis Protocol Proposed in [20] |
| Mobile-DSDP | The Mobile Distributed Self-Diagnosis Protocol Proposed in [21] |
| Time-Free-DSDP | The Time_Free Distributed Self-Diagnosis Protocol proposed in this paper |
| Time-Free-DSDP (Ana) | The analytical model of the Time_Free Distributed Self-Diagnosis Protocol proposed in this paper |
| Time-Free-DSDP (Static) | The Time_Free Distributed Self-Diagnosis Protocol proposed in this paper subjects to Static Faults |
| Time-Free-DSDP (Dynamic) | The Time_Free Distributed Self-Diagnosis Protocol proposed in this paper subjects to Dynamic Faults |

### Results of Scenario 1

Fig. 2 first compares the analytical and simulation results of the proposed Time-Free-DSDP protocol. The simulation results do not deviate more than 5% from the analytical result, which validates the correctness of this proposed protocol. Fig. 2 also compares the communication complexity of the Time_Free-DSDP, the Static-DSDP and the Mobile-DSDP protocols. It is clear that the whole number of packets exchanged increases with respect to the increasing number of nodes. That is, the communication complexity of all the protocols increases with the increasing network size. These results could be explained by the fact that these protocols employ flooding-based mechanisms during the dissemination phases, which worsens with the increasing network size. The result shows that the Mobile-DSDP has a slightly lower order than the Static-DSDP. In fact, in the Mobile-DSDP, each node only replies to σ test requests, while in the Static-DSDP, each node responses to all test requests. Not surprisingly, the Time_Free-DSDP protocol has the lowest communication complexity among the three protocols. The reason is that in the Time_Free-DSDP, a local view message will not be broadcasted unless it includes new information. Thus, this protocol can outperform others, particularly in high-density networks. Moreover, having a lower communication complexity is crucial in energy-constrained networks, because there is a direct correlation between the energy consumed and the traffic generated by a node. In this sense, we would say that the proposed protocol is more energy-efficient than the others are.

Fig. 3 compares the time complexity of the protocols. The result shows that the time complexity increases for all the protocols with the increasing number of nodes in the network. However, as the Time_Free-DSDP protocol performs the diagnosis by exchanging messages using the time-free technique, it requires fewer message exchanges, which results in lesser time complexity compared to other protocols.

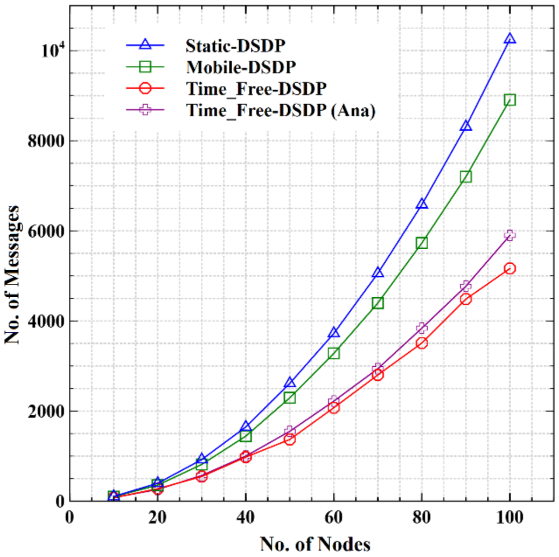

**Fig. 2** Communication complexity under Scenario 1.

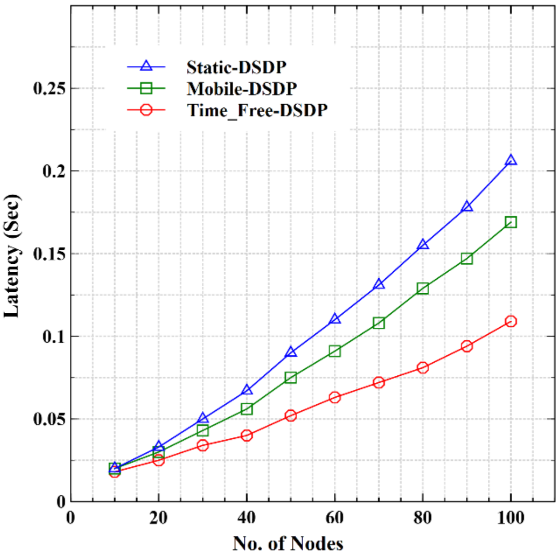

**Fig. 3** Time complexity under Scenario 1.

### Results of Scenario 2

Fig. 4 presents the communication complexity of the Time_Free-DSDP for detecting static and dynamic faults, denoted as Time_Free-DSDP (Static) and Time_Free-DSDP (Dynamic), respectively. In addition, it shows the communication complexity of the Static-DSDP and the Mobile-DSDP. Noticeably, the increase in the number of faults corresponds to the decrease in the communication complexity in all the protocols. This is because increasing faulty nodes means a fewer number of nodes participating in the message dissemination phase, and hence reduce the number of local view messages that will be propagated through the fault-free nodes. Thus, the total number of packets exchanged decreases. The results, also, show that the Mobile-DSDP outperforms the Static-DSDP. The reason again is the limited number of tests, $\sigma$ that each node should reply to in the Mobile-DSDP. Note that, the Time_Free-DSDP outperforms other protocols even when dynamic faults are considered. Again, this is because nodes discard local view messages that add no new information. However, the result shows that the Time_Free-DSDP (Dynamic) has a slightly higher communication complexity than the Time_Free-DSDP (Static). It is reasonable because nodes experiencing dynamic faults have been involved in the diagnosis session before they became faulty. The comparison among the protocols in terms of diagnosis latency is shown in Fig. 5, which exhibits the similar pattern of performance of Fig. 4.

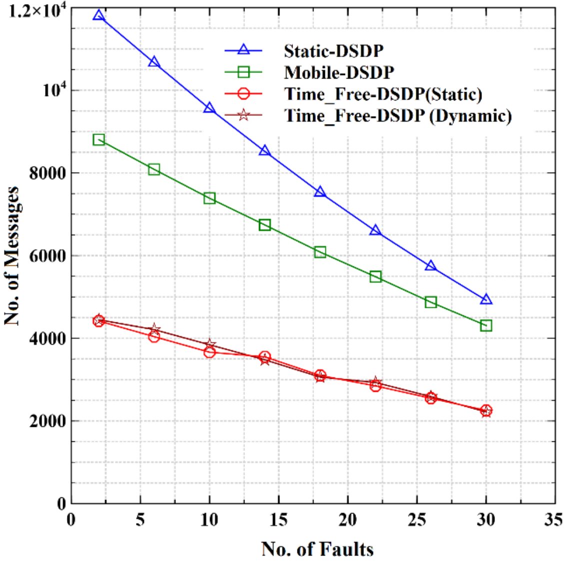

**Fig. 4** Communication complexity under Scenario 2.

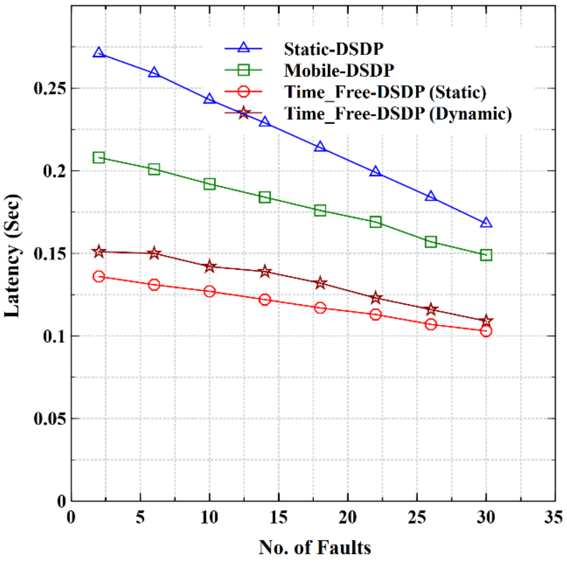

**Fig. 5** Time complexity under Scenario 2.

**Results of Scenario 3**

Fig. 6 illustrates the communication complexity of the Time_Free-DSDP under a different number of mobile nodes. Further, Fig. 7 compares their time efficiency.

Apparently, the increase in mobile nodes results in a decrease in the communication complexity. The node movement in this scenario has been designed such that it changes the topology and maintains the assumptions stated previously. That is, the movement of mobile nodes led to a denser graph. Thus, the number of local view messages decreases. On the other hand, it shows that topology changes have no considerable impact on the communication complexity of the Mobile-DSDP. These results could be attributed to the fact that the execution of the Mobile-DSDP protocol is insensitive to topology changes while a topology of a network has a significant impact on the performance of the Time_Free-DSDP. However, the Time_Free-DSDP shows better results in this scenario as shown in Fig. 6. Clearly, Time_Free-DSDP outperforms Mobile-DSDP in terms of time and communication complexity in dynamic environments.

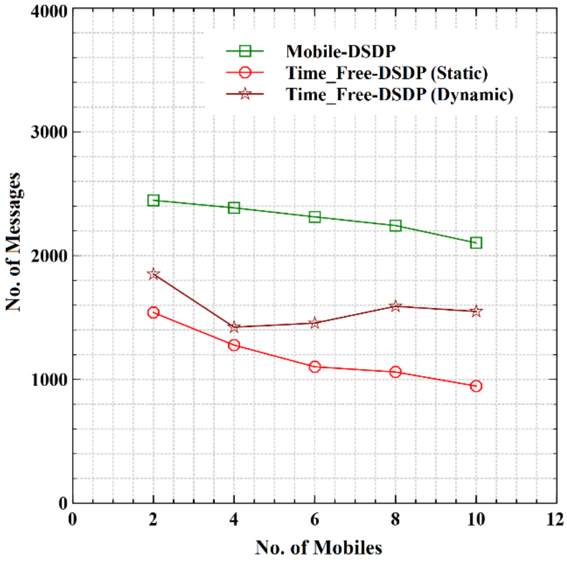

**Fig. 6** Communication complexity under Scenario 3.

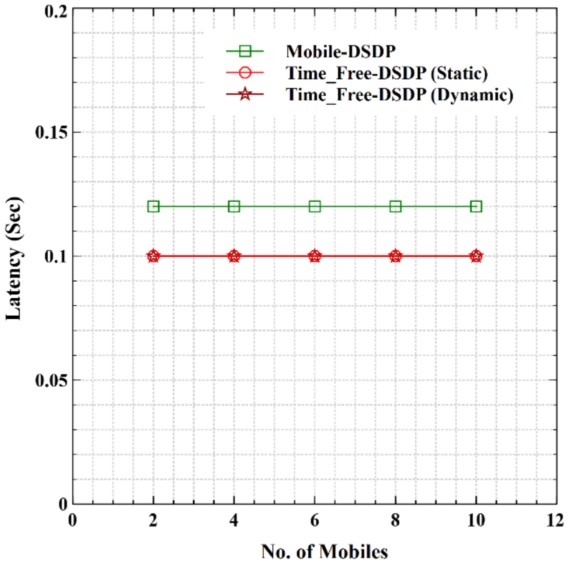

**Fig. 7** Time complexity under Scenario 3.

## 8. The Time_Free-DSDP protocol: Implications and Limitations

This section discusses the implications and the limitations of our proposed fault diagnosis protocol, the Time_Free-DSDP. It also shows the significance of our proposed protocol for mobile networks compared to other related protocols.

The simulation results in the previous section showed that the Time_Free-DSDP outperformed other protocols in terms of communication overhead and diagnosis time. The various simulations investigated the performance of these protocols under different scenarios. In these scenarios, several network sizes, mobility and topology changes, and fault types and numbers were considered.

The diagnosis protocol presented in Section 5 aims mainly to prove that the time-free diagnosis model can be leveraged to develop diagnosis protocols for mobile networks. It is clear that the proposed diagnosis protocol outperforms other protocols in terms of communication and time complexity. Hence, it is more scalable and energy efficient. However, since this protocol employs a flooding mechanism to disseminate the local view of nodes, its scalability might be insufficient for large-scale networks. Therefore, the next chapter presents a more efficient fault diagnosis protocol for mobile networks.

It is of interest to highlight some advantages that the Time_Free-DSDP has inherited from the time-free diagnosis model. In particular, unlike other considered protocols, the proposed protocol can diagnose dynamic faults that may appear during the diagnosis session. This advantage is of importance in real-life implementations; the diagnosis session might take a long time as a consequence of having comprehensive test tasks, and faults may occur during this time. Another advantage derives from using the $\alpha_u$ parameter instead of timers. It is worth mentioning that the value of $\alpha_u$is calculated using information available locally for each node, and nodes can determine the value on the fly. This advantage gives the protocol the ability to succeed in mobile networks where the links are unreliable. Even though the other protocols are implemented on top of a data link layer protocol that helps them to handle communication issues, they cannot tolerate significant transmission delays. For example, in these protocols, if the transmission delays are inconsiderable longer than the timeouts, then they cannot offer a complete or correct diagnosis. On the other hand, our proposed protocol is able to diagnose the system correctly and completely, tolerating these intrinsic characteristics of mobile networks. Even though the transmission delays increase the diagnosis time, they have no impact on the correctness of our proposed protocol.

To sum up, the Time_Free-DSDP fault diagnosis protocol outperforms other protocols quantitatively, where it shows significant lower communication and time overhead, and qualitatively, where it supports dynamic faults, asynchronous communications and limited knowledge about the network. Hence, it is of interest for mobile networks. **Error! Reference source not found.**5 summarises the comparison between our proposed fault diagnosis protocol and the most related fault diagnosis protocols in the literature.

**Table 5** Comparison between the Static-DSDP, Mobile-DSDP and Time_Free-DSDP

| | **Static-DSDP** | **Mobile-DSDP** | **Time_Free-DSDP** |
|---|---|---|---|
| **Fault Types** | Soft and hard | Soft and hard | Soft and hard |
| **Fault Time** | Static | Static | Static and Dynamic |
| **Network Topology** | Fixed | Fixed and Dynamic | Fixed and Dynamic |
| **Dissemination** | Flooding | Flooding | Flooding |
| **Timers** | One timer | Two timers | No timers |
| **Asynchronous** | No | No | Yes |
| **Transmission Delays** | Intolerable | Intolerable | Tolerable |
| **Communication complexity** | $O(n\ (n + d_{max} + 1))$ | $O(n\ (n + \sigma + 1))$ | $O(n\ (n + \alpha_u + 1))$ |
| **Time complexity** | $O(\Delta(T_{gen} + T_f) + T_{out}))$ | $O(\Delta'(T_{gen} + T_f) + T_{out}))$ | $O(\Delta(T_{gen} + T_f) + T_\alpha))$ |

## 9. Conclusion

We have presented a robust time-free comparison model for wireless ad hoc networks suitable for diagnosing soft and hard faults. Undoubtedly, this model opens the way to design dependable mobile wireless networks. We have also presented a fault diagnosis protocol that implements our time-free comparison model. This protocol provides a complete and correct diagnosis for both static and dynamic faults in MANETs. Extensive simulations evaluate the performance of the proposed system. The results obtained have shown that our diagnosis protocol is more efficient in terms of communication and time complexity. The identification of intermittent and transient faults in mobile wireless networks are suggested as future work. Further, we plan to investigate the performance of our proposed protocol under error-prone links in wireless ad hoc networks. The development of a fault diagnosis protocol for integrating routing protocols in vehicular networks is our ongoing research.

## References

1. Casteigts, A., et al. *Time-varying graphs and dynamic networks*. in *International Conference on Ad-Hoc Networks and Wireless*. 2011. Springer.
2. Rasheed, A., et al. *Vehicular Ad Hoc Network (VANET): A Survey, Challenges, and Applications*. 2017. Singapore: Springer Singapore.
3. Qiu, T., et al., *Heterogeneous ad hoc networks: Architectures, advances and challenges.* Ad Hoc Networks, 2017. **55**: p. 143-152.
4. Yick, J., B. Mukherjee, and D. Ghosal, *Wireless sensor network survey.* Computer Networks, 2008. **52**(12): p. 2292-2330.
5. Miranda, K., A. Molinaro, and T. Razafindralambo, *A survey on rapidly deployable solutions for post-disaster networks.* IEEE Communications Magazine, 2016. **54**(4): p. 117-123.
6. Sarkar, S.K., T.G. Basavaraju, and C. Puttamadappa, *Ad hoc mobile wireless networks: principles, protocols, and applications*. 2016: CRC Press.
7. Jarrah, H., N.I. Sarkar, and J. Gutierrez, *Comparison-based system-level fault diagnosis protocols for mobile ad-hoc*

*networks: A survey.* Journal of Network and Computer Applications, 2016. **60**: p. 68-81.

8. Azar, H.N. and H.T. Malazi, *Decentralized detection of hybrid faults in mobile sensor nodes.* Simulation Modelling Practice and Theory, 2018. **87**: p. 210-225.

9. Avizienis, A., et al., *Basic concepts and taxonomy of dependable and secure computing.* IEEE Transactions on Dependable and Secure Computing, 2004. **1**(1): p. 11-33.

10. Muhammed, T. and R.A. Shaikh, *An analysis of fault detection strategies in wireless sensor networks.* Journal of Network and Computer Applications, 2017. **78**: p. 267-287.

11. Zhang, Z., et al., *A survey on fault diagnosis in wireless sensor networks.* IEEE Access, 2018. **6**: p. 11349-11364.

12. Novotny, P., B.J. Ko, and A.L. Wolf, *Locating Faults in MANET-Hosted Software Systems.* IEEE Transactions on Dependable and Secure Computing, 2018(3): p. 452-465.

13. Yue, Y.-G. and P. He, *A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions.* Information Fusion, 2018. **44**: p. 188-204.

14. Mahapatro, A. and P.M. Khilar, *Fault Diagnosis in Wireless Sensor Networks: A Survey.* IEEE Communications Surveys & Tutorials, 2013. **15**(4): p. 2000-2026.

15. Preparata, F.P., G. Metze, and R.T. Chien, *On the Connection Assignment Problem of Diagnosable Systems.* IEEE Transactions on Electronic Computers, 1967. **EC-16**(6): p. 848-854.

16. Malek, M. *A comparison connection assignment for diagnosis of multiprocessor systems*. in *Proceedings of the 7th annual symposium on Computer Architecture*. 1980. ACM.

17. Elias P. Duarte, J., R.P. Ziwich, and L.C.P. Albini, *A Survey of Comparison-Based System-Level Diagnosis.* ACM Computing Surveys (CSUR), 2011. **43**(3): p. 1-56.

18. Chwa, K.-Y. and S.L. Hakimi, *Schemes for fault-tolerant computing: A comparison of modularly redundant and t-diagnosable systems.* Information and Control, 1981. **49**(3): p. 212-238.

19. Somani, A.K., *System level diagnosis: A review.* Technique Report, Dependable Computer Laboratory, Iowa State University, 1997.

20. Chessa, S. and P. Santi. *Comparison-based system-level fault diagnosis in ad hoc networks*. in *Proceedings 20th IEEE Symposium on Reliable Distributed Systems.* 2001. New Orleans, LA, USA: IEEE.

21. Elhadef, M., A. Boukerche, and H. Elkadiki. *Diagnosing mobile ad-hoc networks: two distributed comparison-based self-diagnosis protocols*. in *Proceedings of the 4th ACM international workshop on Mobility management and wireless access*. 2006. ACM.

22. Jarrah, H., et al. *A Time-Free Comparison-Based System-Level Fault Diagnostic Model for Highly Dynamic Networks*. in *Proceedings of the 11th International Conference on Queueing Theory and Network Applications*. 2016. ACM.

23. Greve, F., L. Arantes, and P. Sens. *What model and what conditions to implement unreliable failure detectors in dynamic networks?* in *Proceedings of the 3rd International Workshop on Theoretical Aspects of Dynamic Distributed Systems*. 2011. ACM.

24. Greve, F., et al., *Asynchronous Implementation of Failure Detectors with partial connectivity and unknown participants.* 2011.

25. Greve, F., et al. *A Time-Free Byzantine Failure Detector for Dynamic Networks*. in *Proceedings of the 2012 Ninth European Dependable Computing Conference*. 2012. IEEE Computer Society.

26. Mostefaoui, A., et al. *From static distributed systems to dynamic systems*. in *24th IEEE Symposium on Reliable Distributed Systems (SRDS 2005)*. 2005. Orlando, Florida, USA: IEEE.

27. Aguilera, M.K., *A pleasant stroll through the land of infinitely many creatures.* ACM Sigact News, 2004. **35**(2): p. 36-59.

28. Santoro, N., *Design and analysis of distributed algorithms*. Vol. 56. 2006: John Wiley & Sons.

29. Elhadef, M., A. Boukerche, and H. Elkadiki, *A distributed fault identification protocol for wireless and mobile ad hoc networks.* Journal of Parallel and Distributed Computing, 2008. **68**(3): p. 321-335.

30. Elhadef, M. and S. Grira. *Comparison-based system level fault diagnosis using game theory*. in *2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*. 2016. IEEE.

31. Hsu, G.-H. and J.J. Tan, *A local diagnosability measure for multiprocessor systems.* IEEE Transactions on Parallel and Distributed Systems, 2007. **18**(5).

32. Lin, C.-K., et al., *Local diagnosis algorithms for multiprocessor systems under the comparison diagnosis model.* IEEE Transactions on Reliability, 2013. **62**(4): p. 800-810.

33. Koo, C.-Y. *Broadcast in radio networks tolerating byzantine adversarial behavior*. in *Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*. 2004. ACM.

34. Bhandari, V. and N.H. Vaidya, *Reliable broadcast in radio networks with locally bounded failures.* IEEE Transactions on Parallel and Distributed Systems, 2010. **21**(6): p. 801-811.

35. Sengupta, A. and A.T. Dahbura, *On self-diagnosable multiprocessor systems: diagnosis by the comparison approach.* IEEE Transactions on Computers, 1992. **41**(11): p. 1386-1396.

36. Fidge, C.J., *Timestamps in message-passing systems that preserve the partial ordering.* 1987.

37. Arantes, L., et al., *Eventual leader election in evolving mobile networks*, in *Principles of Distributed Systems*. 2013, Springer. p. 23-37.

38. Varga, A., *A practical introduction to the OMNeT++ simulation framework*, in *Recent Advances in Network Simulation*. 2019, Springer. p. 3-51.

39. Sarkar, N.I. and J.A. Gutiérrez, *Revisiting the issue of the credibility of simulation studies in telecommunication networks: highlighting the results of a comprehensive survey of IEEE publications.* IEEE Communications Magazine, 2014. **52**(5): p. 218-224.

40. Zarrad, A. and I. Alsmadi, *Evaluating network test scenarios for network simulators systems.* International Journal of Distributed Sensor Networks, 2017. **13**(10): p. 1550147717738216.