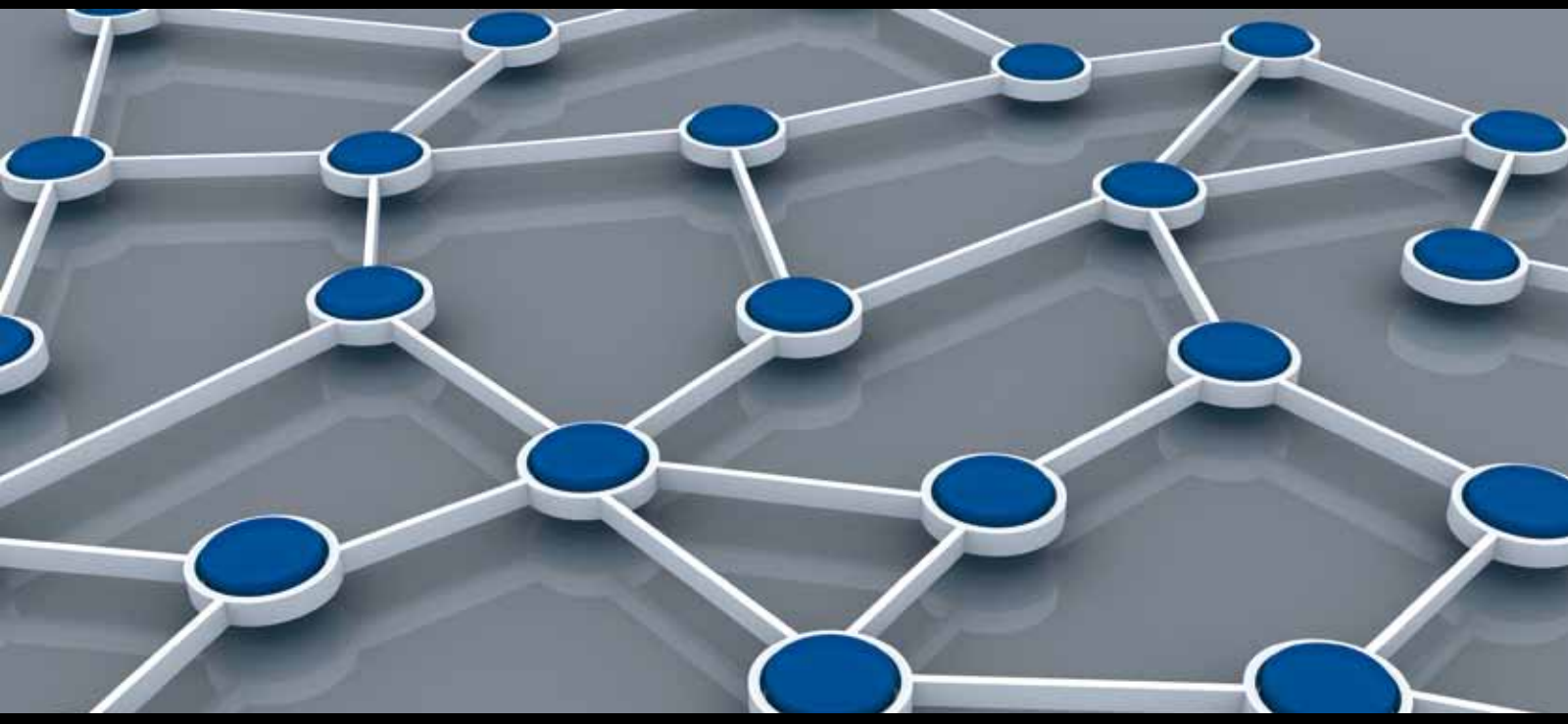
International Journal of Distributed Sensor Networks

Mobile Sensor Networks: Theory, Control, Communication, and Computation Issues

Guest Editors: V. Cagri Gungor, Kayhan Gulez, Kuniaki Kawabata, Nazif Cihan Taş, and Gurkan Tuna



Mobile Sensor Networks: Theory, Control, Communication, and Computation Issues

International Journal of Distributed Sensor Networks

Mobile Sensor Networks: Theory, Control, Communication, and Computation Issues

Guest Editors: V. Cagri Gungor, Kayhan Gulez, Kuniaki Kawabata, Nazif Cihan Taş, and Gurkan Tuna

Editorial Board

Contents

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 875702, 2 pages
http://dx.doi.org/10.1155/2013/875702

Editorial

Mobile Sensor Networks: Theory, Control, Communication, and Computation Issues

V. Cagri Gungor,[1] Kayhan Gulez,[2] Kuniaki Kawabata,[3] Nazif Cihan Taş,[4] and Gurkan Tuna[5]

[1] *Department of Computer Engineering, Abdullah Gul University (AGU), 38039 Kayseri, Turkey*
[2] *Department of Control and Automation Engineering, Yildiz Technical University, Esenler, 34220 Istanbul, Turkey*
[3] *Riken-XJTU Joint Research Unit, Saitama 351-0198, Japan*
[4] *Siemens Corporate Research, Princeton, NJ 08540, USA*
[5] *Department of Computer Programming, Trakya University, 22020 Edirne, Turkey*

Correspondence should be addressed to V. Cagri Gungor; cagri.gungor@agu.edu.tr

Received 1 September 2013; Accepted 1 September 2013

In recent years, a rapid growth has been witnessed in the use of wireless sensor technologies and mobility for different application scenarios. It is envisaged that, due to the advantages offered, the trend toward mobility will not only continue but is likely to increase over the years to come. However, such employment unveils a variety of opportunities and problems. The steady increase in the use of wireless sensor networks that are designed based on a requirement for mobility is necessarily shifting the more traditional centralized network architecture toward a distributed topology. A direct consequence of this is that many open issues related to theory, control, communication, and computation need to be addressed to ensure the fail-safe operation of mobile sensor networks (MSNs).

MSNs bridge several existing research areas, including multiagent systems, sensor networks, robotics, control theory, and machine learning. Recent years have witnessed the proliferation of application scenarios in which MSNs are used, ranging from environment monitoring to emergency search and rescue operations whereby large numbers of pervasive computing devices are connected to a wireless networking infrastructure in an ad hoc manner.

In an effort to disseminate current advances on MSNs, this special issue aims at bringing together some of the most promising state-of-the-art exemplars in the field of MSNs. The 6 papers contained in this special issue cover the bases in terms of theoretical rigor along with practical implementation. The application areas discussed involve some of the more traditional along with some of those newly emerging. Overall, however, the emphasis here has been on selecting papers for this issue that contain innovative, exciting, and insightful solutions to the problems witnessed.

In the paper "*An evolutionary game-based trust cooperative stimulation model for large scale mANETs*" by X. Wang et al., in order to realize a methodical, effective cooperative stimulation for large scale mobile ad hoc networks (MANETs) and search dynamic trust cooperative stimulation scheme in environment under a high malicious ratio, an evolutionary game-based trust cooperative stimulation model is proposed. The authors demonstrated that their model can effectively stimulate cooperation among members and meanwhile be robust under the condition where the environment is harsh under a high original malicious ratio in large scale MANETs.

In "*HMM and rule based hybrid intruder detection approach by synthesizing decisions of sensors*," a novel methodology to unify the decisions from individual sensors on a sensor field through the hidden Markov model (HMM) and rules is proposed. By the use of contextual knowledge, the success of a decision process is improved. Also, a discretization method to express the state space of sensor field is proposed in the paper.

The paper "*An efficient resource management protocol for handling small resource in wireless sensor networks*" reported that wireless sensor nodes can feasibly borrow the memory or computational resource from the gateway or the server. In this respect, a resource management protocol (RMP) that enables wireless sensor nodes to efficiently use the resources including the memory and the CPU in the gateway or the server is proposed, and the effectiveness of the RMP is validated by experiments.

The paper entitled "*A TDMA scheme for mobile sensor networks*" proposes a time division multiple access (TDMA)-based protocol for MSNs. In this paper, a mechanism is used to overcome the shortcomings of the existing TDMA-based protocols in dynamic networks where the cluster memberships may change frequently. The proposed mechanism provides significant performance improvements compared to the other existing approaches in terms of different network performance metrics.

In the paper, "*A multichannel cross-layer architecture for multimedia sensor networks*" by T. Çevik and A. Zaim, a multichannel cross-layer architecture for Quality of Service (QoS) constrained multimedia sensor networks is proposed. The proposed architecture considers both the time and energy efficiency concepts. The authors demonstrated that the proposed architecture provides higher performance than the greedy approach and the LEERA scheme.

In "*Self-stabilizing TDMA algorithms for dynamic wireless ad hoc networks*," a self-stabilizing MAC algorithm for dynamic wireless ad hoc networks that guarantees a short convergence period, and can facilitate the satisfaction of severe timing requirements, is proposed. The results of simulation studies validated that the proposed algorithm can facilitate the implementation of MAC protocols that guarantee satisfying severe timing requirements.

We are hopeful that these papers will prove to be useful sources of reference for both the researchers and the practitioners.

V. Cagri Gungor
Kayhan Gulez
Kuniaki Kawabata
Nazif Cihan Taş
Gurkan Tuna

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 639761, 17 pages
http://dx.doi.org/10.1155/2013/639761

Research Article

Self-Stabilizing TDMA Algorithms for Dynamic Wireless Ad Hoc Networks

Pierre Leone[1] and Elad M. Schiller[2]

[1] *Computer Science Department, Centre Universitaire d'Informatique, University of Geneva, Battelle bâtiment A, Route de Drize 7, Carouge, 1227 Geneva, Switzerland*
[2] *Chalmers University of Technology, Rännvägen 6B, 412 96 Göteborg, Sweden*

Correspondence should be addressed to Elad M. Schiller; elad@chalmers.se

Received 31 January 2013; Accepted 11 June 2013

Academic Editor: Kayhan Gulez

In dynamic wireless ad hoc networks (DynWANs), autonomous computing devices set up a network for the communication needs of the moment. These networks require the implementation of a medium access control (MAC) layer. We consider MAC protocols for DynWANs that need to be autonomous and robust as well as have high bandwidth utilization, high predictability degree of bandwidth allocation, and low communication delay in the presence of frequent topological changes to the communication network. Recent studies have shown that existing implementations cannot guarantee the necessary satisfaction of these timing requirements. We propose a self-stabilizing MAC algorithm for DynWANs that guarantees a short convergence period, and by that, it can facilitate the satisfaction of severe timing requirements, such as the above. Besides the contribution in the algorithmic front of research, we expect that our proposal can enable quicker adoption by practitioners and faster deployment of DynWANs that are subject changes in the network topology.

1. Introduction

Dynamic wireless ad hoc networks (DynWANs) are autonomous and self-organizing systems where computing devices require networking applications when a fixed network infrastructure is not available or not preferred to be used. In these cases, computing devices may set up a short-lived network for the communication needs of the moment, also known as an ad hoc network. Ad hoc networks are based on wireless communications that require implementation of a *medium access control* (MAC) layer. We consider MAC protocols for DynWANs that need to be autonomous and robust and have high bandwidth utilization, a high predictability degree of bandwidth allocation, and low communication delay [1] in the presence of frequent changes to the communication network topology. Existing implementations cannot guarantee the necessary satisfaction of timing requirements [2, 3]. This work proposes an algorithmic design for self-stabilizing MAC protocols that guarantees a short convergence period and, by that, can facilitate the satisfaction of severe timing requirements. The proposed algorithm possesses a greater degree of predictability, while maintaining low communication delays and high throughput.

The dynamic and difficult-to-predict nature of wireless ad hoc networks give rise to many fault tolerance issues that requires efficient solutions. DynWANs, for example, are subject to transient faults due to hardware/software temporal malfunctions or short-lived violations of the assumed settings for modeling the location of the mobile nodes. Fault tolerant systems that are *self-stabilizing* [4] can recover after the occurrence of transient faults, which can cause an arbitrary corruption of the system state (so long as the program's code is still intact), or the model of dynamic networks in which communication links and nodes may fail and recover during normal operation [5]. The proof of self-stabilization requires convergence from an arbitrary starting system state. Moreover, once the system has converged and followed its specifications, it is required to do so forever. The self-stabilization design criteria liberate the application designer from dealing with low-level complications, such as bandwidth allocation in the presence of topology changes, and provide an important level of abstraction. Consequently, the

application design can easily focus on its task and knowledge-driven aspects.

The IEEE 802.11 standard is widely used in wireless communications. Nonetheless, the research field of MAC protocols is very active and requires further investigation. In fact, the IEEE 802.11 amendment, IEEE 802.11p, for wireless access in vehicular environments (WAVE), has just being published. It was shown that the standard's existing implementations cannot guarantee channel access before a finite deadline [2, 3]. Therefore, applications with severe timing requirements cannot predictably meet their deadlines, for example, safety-critical applications for vehicular systems.

ALOHAnet and its synchronized version slotted ALOHA [6] are pioneering wireless systems that employ a strategy of "random access." Time division multiple access (TDMA) [7] is another early approach, where nodes transmit one after the other, each using its own timeslot, say, according to a defined schedule. Radio transmission analysis in ad hoc networks [8] and relocation analysis of mobile nodes [9] show that there are scenarios in which MAC algorithms that employ a scheduled access strategy have lower throughput than algorithms that follow the random access strategy. However, the scheduled approach offers greater predictability of bandwidth allocation and communication delay, which can facilitate fairness [10] and energy conservation [11].

Our design choices have basic radio technology in mind, whilst aiming at satisfying applications that have severe timing requirements. We consider TDMA frames with fixed number of fixed-length timeslots. The design choice of TDMA frames with fixed-length radio time fits well applications that have severe delay requirements. By avoiding the division of fixed-length frames into timeslots of non-equal length, as in [10, 12], we take into consideration the specifications of basic radio technology.

In the context of the previous design choices, there are two well-known approaches for dealing with contention (timeslot exhaustion): (1) employing policies for administering message priority (for meeting timing requirements while maintaining high bandwidth utilization, such as [13]) or (2) adjusting the nodes' individual transmission signal strength or carrier sense threshold [14]. The former approach is widely accepted and adopted by the IEEE 802.11p standard, whereas the latter has only been evaluated via computer simulations. The proposed algorithm facilitates the implementation of both of the previous approaches. We consider implementation details of the standard approach in Section 7.

For the sake of presentation simplicity, we start by considering a single priority MAC protocol and base the timeslot allocation on vertex coloring, before considering multipriority implementation in Section 7. The proposed algorithm allocates timeslots to a number of nearby transmitters, that is, a number that is bounded by the TDMA frame size, whereas nonallocated transmitters receive busy channel indications. The analysis considers saturated situations in which the node degree in the message collision graph is smaller than the TDMA frame size. As explained previously, this analysis assumption does not restrict the number of concurrent transmitters when implementing the proposed MAC algorithm.

1.1. Related Work. We are not the first to propose a MAC algorithm for DynWANs that follows the TDMA's scheduled approach. STDMA [15] and Viqar and Welch [16] consider global navigation satellite system -based scheduling (GNSS) [17] according to the nodes' geographical position and their trajectories. Autonomous systems cannot depend on GNSS services, because they are not always available, or preferred not to be used, due to their cost. Arbitrarily long failure of signal loss can occur in underground parking lots and road tunnels. We propose a self-stabilizing TDMA algorithm that does not require GNSS accessibility or knowledge about the node trajectories. Rather, it considers an underlying self-stabilizing local pulse synchronization, such as [18, 19], which can be used for TDMA alignment; details appear in [18].

When using collision detection at the receiving side [14, 15, 20–22], it is up to the receiving side to notify the sender about collisions via another round of collision-prone transmissions, say, by using frame information (FI) payload fields that include T entries, where T is the TDMA frame size. Thus far, the study of FI-based protocols has considered stochastic resolution of message collision via computer network simulation [15, 20, 22–25]. Simulations are also used for evaluating the heuristics of MS-ALOHA [14] for dealing with contention (timeslot exhaustion) by adjusting the nodes' individual transmission signal strength and/or carrier sense threshold. We do not consider lengthy frame information (FI) fields, which significantly increase the control information overhead, and yet we provide provable guarantee regarding the convergence time. Further analysis validation of the proposed algorithm via simulations and testbed implementation can be found in Section 8, and respectively, in [18].

The proposed algorithm does *not* consider collision detection mechanisms that are based on signal processing or hardware support, as in [26]. Rather, it employs a variation of a well-known strategy for eventually avoiding concurrent transmissions among neighbors. This strategy allows the sending side to eventually observe the existence of interfering transmissions. Before sending, the sender waits for a random duration while performing a clear channel assessment using basic radio technology (details appear in Section 3).

There are several MAC algorithms that are based on clear channel assessment. A recent example, [12], focuses on fair bandwidth allocation for single-hop-distance broadcasting while basing the interference model on discrete graphs. The authors do not consider self-stabilization. This work also considers clear channel assessment. However, we employ a strategy of random transmission delay in a way that allows the recipients to notice, in a probabilistic manner, prospective transmissions. We show that after a small number of rounds, the system is able to use the previous strategy for allocating the network bandwidth for single-hop-distance broadcasting when basing the interference model on discrete graphs. Further mitigation efforts of transmission pathologies, such as hidden terminal phenomena when unicast are considered, can be taken, for example, self-stabilizing two-hop-distance

vertex coloring [27], equalizing transmission power, and coding-based methods [28], to name a few.

An abstract MAC layer was specified for DynWANs in [29]. The authors mention algorithms that can satisfy their specifications. However, they do not consider predictability.

Local algorithms [30, 31] consider both theoretical and practical aspects of MAC algorithms ([32] and references therein) and the related problem of clock synchronization; see [33] and references therein. For example, the first partly-asynchronous self-organizing local algorithm for vertex coloring in wireless ad hoc networks is presented in [34]. However, this line currently does not consider dynamic networks and predictable bandwidth allocation.

Two examples of self-stabilizing TDMA algorithms are presented in [10, 35]. The algorithms are based on vertex-coloring and the authors consider (nondynamic) ad hoc networks. Recomputation and floating output techniques ([4], Section 2.8) are used for converting deterministic local algorithms to self-stabilization in [36]. The authors focus on problems that are related to MAC algorithms. However, deterministic MAC algorithms are known to be inefficient in their bandwidth allocation when the topology of the communication network can change frequently [9]. There are several other proposals related to self-stabilizing MAC algorithms for sensor networks, for example, [37–40]; however, none of them consider dynamic networks, and their frame control information is quite extensive.

The MAC algorithms in [9, 18, 41, 42] *have no proof* that they are self-stabilizing. The authors of [9] present a MAC algorithm that uses convergence from a random starting state (inspired by self-stabilization). In [18, 41, 42], the authors use computer network simulators for evaluating self-$\star$ MAC algorithms.

1.2. Our Contribution. This work proposes a self-stabilizing MAC algorithm that demonstrates rapid convergence without the extensive use of frame control information. Our analysis shows that the algorithm facilitates the satisfaction of severe timing requirements for DynWANs.

We start by considering transient faults and topological changes to the communication network, that is, demonstrating self-stabilization in Theorem 2. We then turn to focus on bounding the algorithm's convergence time after an arbitrary and unbounded finite sequence of transient faults and changes to the network topology. Theorem 3 shows that the expected local convergence time is brief and bounds it in (7). Theorem 7 formulates the expected global convergence time in (21). Moreover, for a given probability, the global convergence time is calculated in (22).

For discussion (Section 8), we point out the algorithm's ability to facilitate the satisfaction of severe timing requirements for DynWANs. Moreover, the analysis conclusions explain that when allowing merely a small fraction of the bandwidth to be spent on frame control information and when considering any given probability to converge within a bounded time, the proposed algorithm demonstrates a low dependency degree on the number of nodes in the network (as depicted by Figures 2 and 3).

We note that some of the proof details appear in the Appendix for the sake of presentation simplicity.

2. Preliminaries

The system consists of a set, P, of N anonymous communicating entities, which we call *nodes*. Denote every node $p_i \in P$ with a unique index, i.

2.1. Synchronization. Each node has fine-grained, real-time clock hardware. We assume that the MAC protocol is invoked periodically by synchronous *(common) pulse* that aligns the starting time of the TDMA frame. This can be based, for example, on TDMA alignment algorithms [18], GPS [44], or a distributed pulse synchronization algorithm [19]. The term *(broadcasting) timeslot* refers to the period between two consecutive common pulses, t_x and t_{x+1}, such that $t_{x+1} = (t_x \bmod T) + 1$, where T is a predefined constant named the *frame size*. Throughout the paper, we assume that $T \geq 2$. In our pseudocode, we use the event timeslot(t) that is triggered by the common pulse. We assume that the timeslots are aligned as well.

2.2. Communications and Interferences. At any instance of time, the ability of any pair of nodes to communicate is defined by the set, $N_i \subseteq P$, of *(direct) neighbors* that node $p_i \in P$ can communicate with directly. Wireless transmissions are subject to interferences (collisions). We consider the potential of the nodes to interfere with each other's communications. The interference model in this paper is based on discrete graphs.

The set $\mathcal{N}_i \supseteq N_i$ is the set of nodes that may interfere with p_i's communications when any nonempty subset of them, $I \subseteq \mathcal{N}_i : I \neq \emptyset$, transmits concurrently with p_i. We call $\mathcal{N}_i$ the *(extended) neighborhood* of node $p_i \in P$, and $d_i = |\mathcal{N}_i|$ is named the (extended) degree of node p_i. We assume that at any time, for any pair of nodes, $p_i, p_j \in P$; it holds that $p_j \in \mathcal{N}_i$ implies that $p_i \in \mathcal{N}_j$. Given a particular instance of time, we define the *(interference) graph* as $G := (P, E)$, where $E := \cup_{i \in P}\{(p_i, p_j) : p_j \in \mathcal{N}_i\}$ represents the interference relationships among nodes.

2.3. Communication Schemes. We consider (basic technology of) radio units that raise the event carrier_sense() when they detect that the received energy levels have reached a threshold in which the radio unit is expected to succeed in carrier locking; see [45]. Timeslots allow the transmission of DATA packets using the primitives of transmit() and receive() after fetching (MAC_fetch()) a new packet from the upper layer, and respectively, before delivering (MAC_deliver()) the packet to the upper layer. A *beacon* is a short packet that includes no data load, rather the timing of the event carrier_sense() is the delivered information [12]. We assume that every node $p_i \in P$ that invokes the operation transmit() causes the event carrier_sense() to be raised by its neighbors, $p_j \in \mathcal{N}_i$, within the exposure time, ε. Before the transmission of the DATA packet in timeslot t, our communication scheme

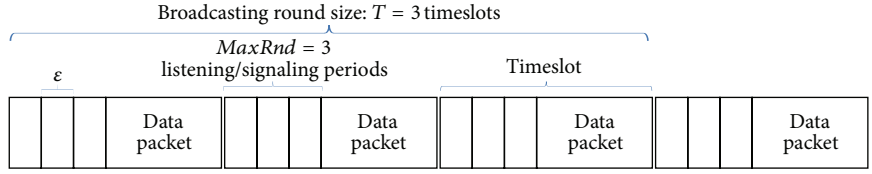


Figure 1: An example of TDMA frame, with three timeslots and three listening/signaling periods of size ε (signal exposure time). Each timeslot has a constant number, $MaxRnd = 4$, of *listening/signaling periods* in which beacons can be transmitted. The duration of each listening/signaling period is ε (signal exposure time); the period during which a beacon that is sent by node $p_i \in P$ is transmitted and raises the ca received by all neighbors $p_j \in \mathcal{N}_i$. Namely, the period between p_i's transmission and p_j's rise of the carrier_sense() event.

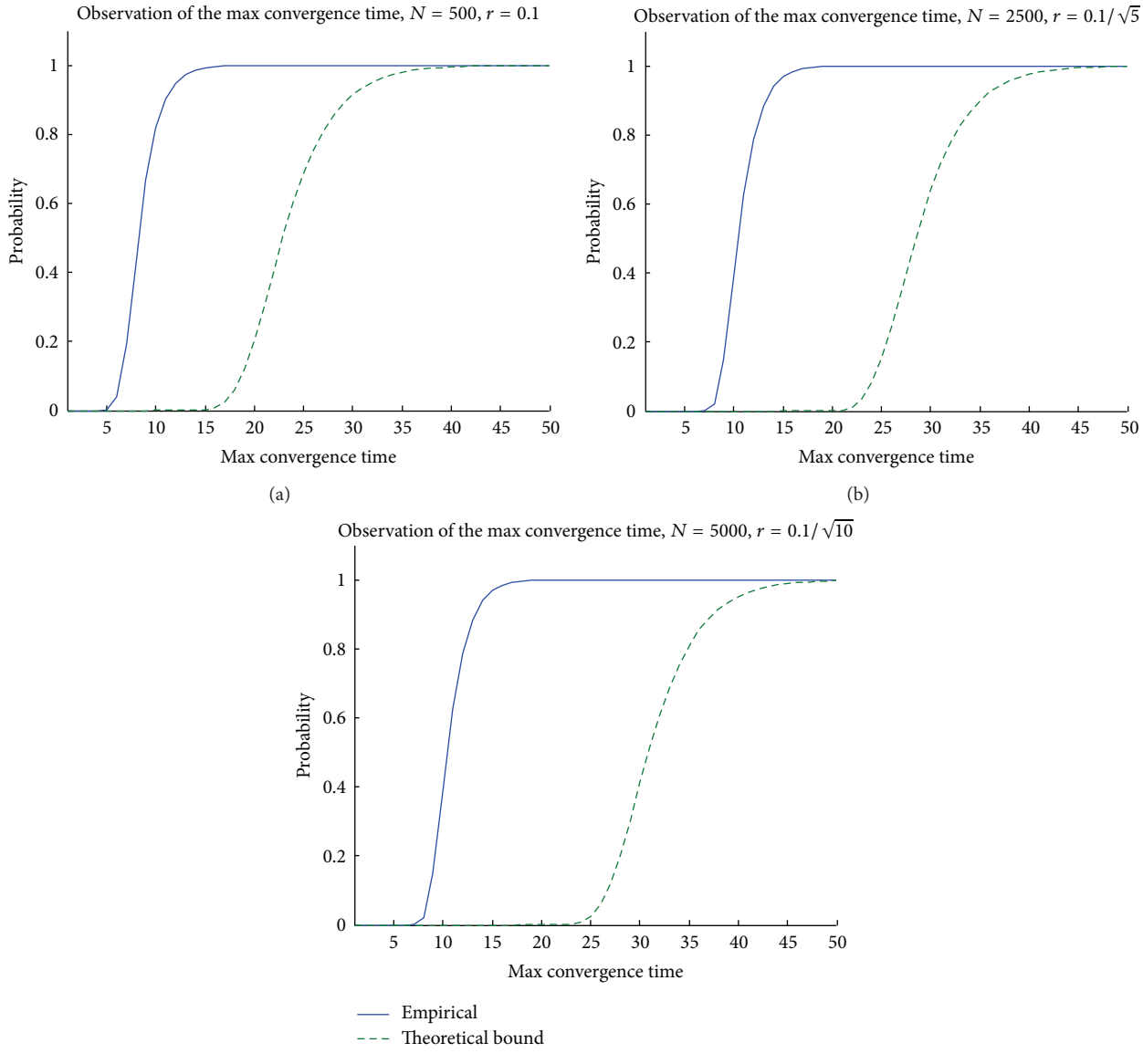


Figure 2: Numerical validation of Theorem 7's bound on the network-wise convergence time. We compare the bound, $P(t_{\max} < k) = (1 - (1-q)^k)^N$, with the numerical results, which consider random geometric graphs in which the nodes are randomly placed on the unit square. The charts considers $N \in \{500, 2500, 5000\}$ nodes (from left to right). All experiments considered 2 listening/signaling periods, interference range of $0.1/\sqrt{(N/500)}$, which result in an average extended degree of 15, $d_i/T = 1$ on average, and $q_i = 1/4$.

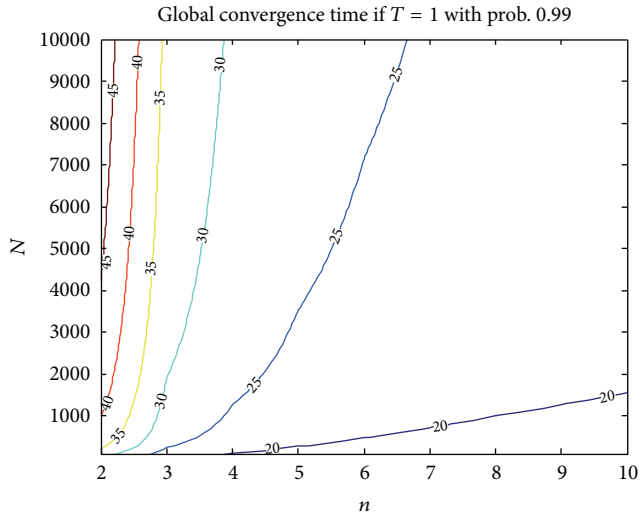


FIGURE 3: Contour plot of (22) for $s = d/T = 1$. *Contour charts* [50] present two parameter functions, for example, the convergence time function, $k(n, N)$ presented in (22). Contour lines in Figure 3 connect values of $k(n, N)$ that are the same (see the text tags along the line). When N nodes attempt to access the medium, the convergence time, $\mathcal{S}$ (cf. the contour lines), is stable in the presence of a growing number, n, of listening/signaling periods.

uses beacons for signaling the node's intention to transmit a DATA packet within t; see Figure 1.

2.4. System Settings. We consider the interleaving model [4]. Every node, $p_i \in P$, executes a program that is a sequence of *atomic steps*. The *state* st_i of a node p_i consists of the value of all the variables of the node (including messages in transit for p_i). Variables are associated with individual node states by using the subscript notation, that is, var_i is the value of variable *var* in p_i's state. The term *configuration* is used for a tuple of the form $(G, \{st_i\}_{i=1}^{N})$, where G is the (interference) graph, and $\{st_i\}_{i=1}^{N}$ are the nodes' states (including the set of all incoming communications). An *execution* (run) $R := (c(0), c(1), \ldots)$ is an unbounded sequence of system configurations $c(x)$, such that each configuration $c(x+1)$ (except the initial configuration $c(0)$) is obtained from the preceding configuration $c(x)$ by the execution of steps, $\{a_i(x)\}_{p_i \in P}$, taken by all nodes.

Let τ (task) be a specification (predicate) set and LE a set of all executions that satisfy task τ. Let us consider TDMA-based MAC protocols for which the task τ_{TDMA} requires that every node has its own broadcasting timeslot that is unique within its neighborhood. We note that τ_{TDMA}'s requirements are obviously satisfiable when the ratio between the extended degree and the frame size is less than one; that is, there is no timeslot exhaustion when for all $p_i \in P : 1 \lneq T/d_i$. Therefore, the studied task also deals with *timeslot* exhaustion by delivering busy channel indications, $\perp$, to the nodes for which there were no *timeslot* left. We define LE_{TDMA} to be the set of legal executions, R, for which for all $p_i \in P$: $(((s_i \in [0, T-1]) \wedge (p_j \in \mathcal{N}_i)) \Rightarrow s_i \neq s_j) \vee (s_i = \perp \Rightarrow$ for all $t \in [0, T-1]\ \exists p_j \in \mathcal{N}_i : s_j = t)$ holds in all of R's configurations.

We say that configuration c_{safe} is safe if there is an execution $R \in LE$, such that c_{safe} is R's starting configuration. Let R be an execution and $c \in R$ its arbitrary starting configuration. We say that R *converges* with respect to τ if within a bounded number of steps from c, the system reaches a safe configuration c_{safe}. The *closure* property requires that $R \in LE$, for any execution, R, that starts form c_{safe}. An algorithm is said to be *self-stabilizing* if it satisfies both the convergence and the closure properties.

We describe execution R as an unbounded number of concatenated finite sequences of configurations. The finite sequence, $R(x) = (c_0(x), \ldots c_{T-1}(x))$, $x > 0$, is a *broadcasting round* if (1) configuration $c_0(x)$ has a clock value, t, of 0 and immediately follows a configuration in which the clock value is $T-1$, and (2) configuration $c_{T-1}(x)$ has a clock value of $T-1$ and immediately precedes a configuration in which the clock value is 0.

3. Algorithm Description

The proposed MAC algorithm periodically performs clear channel assessments. It uses these assessments when informing each node about the nearby unused timeslots. The nodes use this information for selecting their broadcasting timeslots, assessing the success of their broadcasts and reselecting timeslots when needed.

The MAC algorithm in Algorithm 1 satisfies the τ_{TDMA} task. During the convergence period, several nodes can be assigned to the same timeslot. Namely, we may have $p_i \in P : p_j \in \mathcal{N}_i \wedge s_i = s_j$. The algorithm solves such timeslot allocation conflicts by letting the nodes p_i and p_j go through a (listening/signaling) competition before transmitting in its broadcasting timeslot. The competition rules require each node to choose one out of *MaxRnd* listening/signaling periods for its broadcasting timeslot; see Figure 1. This implies that among all the nodes that attempt to broadcast in the same timeslot, the ones that select the earliest listening/signaling period win this broadcasting timeslot and access the communication media. Before the winners access their timeslots, they signal to their neighbors that they won via beacon transmission. The signal is sent during their choice of listening/signaling periods; see Figure 1. When a node receives a beacon, it does not transmit during that timeslot, because it lost this (listening/signaling) competition. Instead, it randomly selects another broadcasting timeslot and competes for it on the next broadcasting round.

In detail, the MAC algorithm in Algorithm 1 is invoked at the start of every timeslot, t. When t is the first timeslot, the algorithm tries to allocate the broadcasting timeslot, s_i, to p_i (line 11) by randomly selecting a timeslot for which there is no indication to be used by its neighbors. Later, when the timeslot t becomes p_i's broadcasting timeslot, s_i, the node attempts to broadcast (by calling the function send() in line 13). We note that the start of timeslot t also requires the marking of t as an unused timeslot and the removal of stale information (line 12). This indication is changed when the carrier_sense(t) event is raised (line 27) due to a neighbor transmission; namely, when the detected energy levels reach

```
     Constants, variables, macros and external functions
(2)  MaxRnd (n in the proofs) : integer = bound on round number
     s : [0, T − 1] ∪ {⊥} = next timeslot to broadcast or null, ⊥
(4)  signal : boolean = trying to acquire the channel
     unused[0, T − 1] : boolean = marking unused timeslots
(6)  unused_set = {k : unused[k] = true} : unused timeslots, macro
     MAC_fetch()/MAC_deliver() : MAC layer interface
(8)  transmit/receive/carrier_sense : communication primitives

(10) Upon timeslot(t)
       if t = 0 ∧ s = ⊥ then s := select_unused(unused_set)
(12)   (unused[t], signal) := (true, false) (∗ remove stale info. ∗)
       if s ≠ ⊥ ∧ t = s then send(MAC_fetch())
(14)
     Upon receive(⟨DATA, m⟩) do MAC_deliver(⟨m⟩)
(16)
     Function send(m) (∗ send message m to p_i's neighbors ∗)
(18)   for ((signal, k) := (true, 0); k := k + 1; k ≤ MaxRnd) do
         if signal then with probability ρ(k) = 1/(MaxRnd − k) do
(20)       signal := false (∗ quit the competition ∗)
           transmit(⟨BEACON⟩) (∗ try acquiring the channel ∗)
(22)     wait until the end of competition round (∗ exposure period alignment ∗)
       if s ≠ ⊥ then transmit(⟨DATA, m⟩) (∗ send the data packet ∗)
(24)
     Upon carrier_sense(t) (∗ defer transmission during t ∗)
(26)   if s = t ∧ signal then s := ⊥ (∗ mark that the timeslot is not unique ∗)
       (signal, unused[t]) := (false, false) (∗ quit the competition ∗)
(28)
     Function select_unused(set) (∗ select an empty timeslot ∗)
(30)   if set = ∅ then return ⊥ else return uniform_select(set)
```

Algorithm 1: Self-stabilizing TDMA-based MAC algorithm, code of node p_i.

a threshold in which the radio unit is expected to succeed in carrier locking; see [45].

When a node attempts to broadcast, it uses the (listening/signaling) competition mechanism for deciding when to signal to its neighbors that it is about to transmit a DATA packet. The competition has *MaxRnd* rounds, and it stops as soon as the node transmits a beacon or a neighbor succeeds in signaling earlier (lines 18 to 23). We note that this signaling is handled by the carrier_sense(t) event (line 27). Moreover, beacons are not required to carry payloads or any other information that is normally stored in packet headers. They are rather used to invoke the carrier sense event in $\mathcal{N}_i$.

The carrier sense in timeslot t indicates to each node that it needs to defer from transmission during t (line 25). In particular, it should stop using timeslot t for broadcasting, stop competing, and mark t as a used timeslot. Lastly, arriving DATA packets are delivered to the upper layer (line 15).

4. Correctness Proof: Outline and Notation

The proof starts by considering networks that do not change their topology and for which the ratio between the extended node degree and the frame size is less than one, that is, for all $p_i \in P : 1 \lneq T/d_i$. (We deal with cases in which for all $p_i \in P : 1 \lneq T/d_i$ does not hold in Section 8). For these settings, we show that the MAC algorithm in Algorithm 1 is self-stabilizing with respect to task τ_{TDMA} (Appendices A to B), before considering the convergence time within a single neighborhood (Section 5) and the entire neighborhood (Section 6). These convergence estimations facilitate the exploration of important properties, such as predictability, and dealing with changes in the network topology of DynWANs (Section 8).

4.1. Proof Outline. The exposition of the proof outline refers to Definition 1, which delineates the different states at which a node can be in relation to its neighbors. Definition 1 groups these states into three categories of *relative states*: (1) Ready to be allocated, when the node state depicts correctly its neighbor states, (2) Obtaining a timeslot, when the node is competing for one, but there is no agreement with its neighbor states, and (3) Allocated to a timeslot, when the node is the only one to be allocated to a particular timeslot in its neighborhood. The correctness proof shows that the MAC algorithm in Algorithm 1 implements τ_{TDMA} in a self-stabilizing manner by showing that eventually all nodes are allocated with timeslots; that is, all nodes are in the relative state Allocated; see Definition 1.

Let R be an execution of the MAC algorithm in Algorithm 1 and $R(x)$ the xth complete broadcasting round of R, where $x > 0$ is an integer. We simplify the presentation by using uppercase notation for the configurations, $c_t^{\text{name}}(x)$, where $t \in [0, T-1]$ is a timeslot. This notation includes the *name* of the first event to be triggered immediately after configuration c, that is, $R(x) = (c_0^{\text{timeslot}}(x), \ldots c_{T-1}^{\text{carrier\_sense/receive}}(x))$.

Definition 1. We say that node $p_i \in P$ is Ready (*to be allocated*) to a timeslot in configuration $c_0^{\text{timeslot}}(x)$, if properties (1), (2), and (3) hold for node p_i, but Property (4) does not. We say that p_i is Obtaining timeslot s_i in configuration $c_0^{\text{timeslot}}(x)$, if properties (1) to (4) hold for node p_i, but Property (5) does not. We say that node $p_i \in P$ is in Allocated state, with respect to timeslot s_i in configuration $c_0^{\text{timeslot}}(x)$, if properties (1) to (5) hold for node p_i as follows:

$$\text{signal}_i = \text{false} \tag{1}$$

$$\left(t \in \text{unused}_i \wedge t \neq s_i\right) \longleftrightarrow \left(\forall p_k \in \mathcal{N}_i : s_k \neq t\right) \tag{2}$$

$$s_i \neq \perp \vee \text{unused\_set}_i \setminus \{s_i\} \neq \emptyset \tag{3}$$

$$s_i \neq \perp \tag{4}$$

$$\forall p_j \in \mathcal{N}_i : \left(\left(s_i \neq s_j\right) \wedge \left(\text{unused}_j\left[s_i\right] = \text{false}\right)\right). \tag{5}$$

Property (1) implies that node p_i finishes any broadcast attempts within a timeslot. Properties (2) to (3) consider the case in which p_i's internal state represents correctly the timeslot allocation in its neighborhood. In particular, property (2) means that processor p_i views timeslot t as an unused one if, and only if, it is indeed unused. Property (3) implies that when node p_i is not using any timeslot, there is an unused timeslot at its disposal. Property (4) says that node p_i is using timeslot s_i. Property (5) refers to situations in which p_i's neighbors are not using p_i's timeslot during the next broadcasting round.

Starting from an arbitrary configuration, we show that node p_i becomes Ready within two broadcasting rounds (or one complete broadcasting round); see Appendix A. Then, we consider the probability, $OnlyOne_i(x)$, that a node enters the relative state Allocated from either Ready or Obtaining; see (6) (and Appendices B and D). Namely, (6) considers the probability that node p_i is the *only one* to use its broadcasting timeslot in its neighborhood, where $\rho_k = 1/MaxRnd = 1/n$ is p_i's probability to select the kth listening/signaling period for transmitting its beacon.

Consider

$$OnlyOne_i(x) \geq \sum_{k=1}^{n} \rho_k \left(1 - \sum_{\ell=1}^{k} \rho_\ell\right)^{d_i/T}. \tag{6}$$

Theorem 2 demonstrates self-stabilization.

Theorem 2 (self-stabilization, the proof appears in Appendix C). *The MAC algorithm in Algorithm 1 is self-stabilizing with respect to the task τ_{TDMA}.*

Bounding the convergence time. We bound the time it takes the MAC algorithm in Algorithm 1 to converge by considering the relative states, Ready, Obtaining, and Allocated and describe a state machine of a Markovian process. This process is used for bounding the convergence time of a single node (Section 5) and the entire network (Section 6).

In detail, given node $p_i \in P$, its neighborhood $\mathcal{N}_i$, we define a random environment of a Markov chain; see Box 1. By looking at this random environment, we can focus our analysis on p_i's relative states while avoiding probability dependencies and considering average probabilities [46]. Suppose that p_i's environment, e, is known. Theorem 3 estimates two bounds on the expectation of probability $q_i|_e$, which is literally the probability q_i, given that the environment is e.

In order to do that, we consider a set, $\mathcal{R}$, of executions of the MAC algorithm, such that each execution $R \in \mathcal{R}$ starts in a configuration, $c \in R$, in which (I) for any node $p_j \in P$, properties (1), (2), and (3) hold, and (II) node p_i is in the relative state Ready, which implies that (III) eventually, node p_i arrives to the relative state Allocated.

With this convention, we can add a probability 1 to transit from the relative state Allocated to Ready; see the dashed line in the state machine diagram of Box 1. This allows us to estimate the expected time to reach the final relative state Allocated from relative state Ready by the expectation of the first hitting time of the irreducible Markov chain [43].

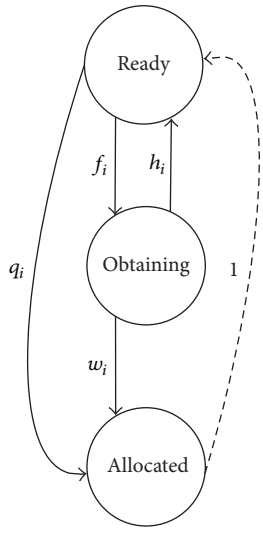
When computing the expected time for node p_i to reach state Allocated within its neighborhood, we see that it is sufficient to consider the lower bound of the probability $OnlyOne_i(x)$ to obtain an upper bound on the expected time to convergence; see Section 5. Moreover, when considering the network convergence time, that is, the expected convergence time of all nodes in the network, we see that the most dominant parameter is the mean neighborhood size. We do that by applying the arithmetic mean versus geometric mean (AM-GM) inequality and bounding the expected network convergence time; see Section 6.

4.2. Notation. Throughout the paper, we denote the states of the Markov chain by $\{X_t\}_{t\geq 0}$, $T_i^+ = \min\{t > 0 \text{ such that } X_t = i\}$ and $E_i(\cdot)$ is the expectation, given that we start in relative state i, $E_i(T_i^+) = E(T_i^+ \mid X_0 = i)$. In this paper, the states 1, 2, and 3 of the Markovian process correspond, respectively, to states Ready, Obtaining and Allocated and the time $t = 0, 1, \ldots$ corresponds to configuration $c_0^{\text{timeslot}}(x + t) \in R(x + t)$, where $R(x)$ is the first complete broadcasting round in R that starts in a configuration, $c_0^{\text{timeslot}}(x)$, in which all nodes are in the relative state Ready. For example, $E_3(T_3^+)$ is the expected time to reach the Allocated state.

Let $p_i \in P$ be a node for which $s_i \neq \perp \wedge \exists p_j \in \mathcal{N}_i : s_j = s_i$ in configuration $c_0^{\text{timeslot}}(x)$. We define $M_i(x) = \{p_j \in \mathcal{N}_i : s_i = s_j\}$ to be the set of p_i's (broadcasting timeslot) *matching* neighbors, which includes all of p_i's neighbors that, during broadcasting round $R(x)$, are attempting to broadcast in p_i's timeslot. In our proofs, we use n as the number of *listening/signaling periods*, $MaxRnd$.

5. Convergence within a Neighborhood

Theorem 3 bounds the expected time, $\mathcal{S}_i$, for a node to reach the relative state Allocated, and follows from Proposition 5 and (12). Note that $\mathcal{S}_i \leq 4$ when the number of listening/signaling periods is $n \geq 2$ and considering saturated situations in which the extended node degree $d_i < T$ is



We look at p_i's state transition with relation to its neighbors; see Definition 1. The figure on the top defines p_i's relative states as a 3-state Markov chain. The probabilities, q_i, w_i, f_i, and h_i (solid lines arrows), that node p_i change its relative state depend on its neighbor's state. For instance, q_i is the probability that p_i goes from the relative state Ready to Allocated. It is environment-dependent; that is, the states of p_i's neighbors are random as well. We added the dotted edge between the state Allocated and the state Ready in order to make the Markov chain irreducible and to allow working with the invariant probability. Namely, once node p_i arrives to Allocated, it returns to Ready with probability 1. With this convention, we can estimate the expected time to reach the final relative state Allocated from relative state Ready by the expectation of the first hitting time of the irreducible chain [43]

Box 1: Markov chain describing p_i's relative state transitions.

smaller than the TDMA frame size. Namely, the proposed algorithm convergence with a neighborhood is brief.

Theorem 3 (local convergence). *The expected time, $\mathcal{S}_i$, for node $p_i \in P$ to reach the relative state* Allocated *satisfies* (7), *where n is the number of listening/signaling periods, T is the TDMA frame size, and d_i is p_i's extended degree.*

Consider

$$\mathcal{S}_i \leq \min\left\{\left(\frac{2n}{n-1}\right)^{d_i/T}, \frac{d_i/T+1}{n}\left(\frac{n}{n-1}\right)^{d_i/T+1}\right\}. \quad (7)$$

We look into the transition probability among relative states by depicting the diagram of Box 1 as a homogeneous Markov chain. We estimate the diagram transition probabilities in a way that maximizes the expected time for reaching the diagram's final state, Allocated. It is known that the first hitting time is given by $E_i(T_i^+) = 1/\pi_i$, where $\pi = (\pi_1, \pi_2, \pi_3)$ is the invariant probability vector [43]. Let $\mathcal{S}_i$ be the expected time it takes node p_i that starts at the relative state Ready to reach Allocated. It is clear that $\mathcal{S}_i = T_3^+ - 1$, because $T_3^+ - 1$ is the return time of the relative state Allocated. In our case, the transition matrix P is given by the following:

$$P = \begin{pmatrix} 1-f_i-q_i & f_i & q_i \\ h_i & 1-h_i-w_i & w_i \\ 1 & 0 & 0 \end{pmatrix}. \quad (8)$$

The invariant probability vector π satisfying $\pi P = \pi$ is given by

$$\pi = \frac{(h_i + w_i, f_i, q_i h_i + q_i w_i + f_i w_i)}{h_i + w_i + f_i + h_i q_i + q_i w_i + f_i w_i}. \quad (9)$$

The estimation of the maximal expected time necessary to assign the node p_i to a timeslot requires us to compute bounds on the probabilities f_i, h_i, q_i and w_i that maximize as follows

$$E_3\left(T_3^+\right) = \frac{1}{\pi_3} = \frac{h_i + w_i + f_i + h_i q_i + q_i w_i + f_i w_i}{q_i h_i + q_i w_i + f_i w_i}. \quad (10)$$

The expected time for p_i to reach the relative state Allocated is bounded in

$$\mathcal{S}_i = E_3\left(T_3^+\right) - 1 = \frac{h_i + w_i + f_i}{q_i h_i + q_i w_i + f_i w_i}. \quad (11)$$

Equation (7) has a compact and meaningful bound for (11). We achieve that by studying the impact of the parameters T and n on the MAC algorithm in Algorithm 1. Lemma 4 and (11) imply

$$\mathcal{S}_i \leq \frac{h_i + w_i + f_i}{q_i h_i + q_i w_i + f_i q_i} = \frac{1}{q_i}. \tag{12}$$

Lemma 4. *Suppose that $n \geq 2$ is the number of listening/signaling periods; see line 2 of the code in Algorithm 1. Then $w_i \geq q_i$.*

Proof. Let us consider node $p_i \in P$ that is in relative state Ready. Given that p_i has v_i neighbors that compete for the same timeslot, the probability that p_i gets allocated, $q_i|_{v_i}$, is given by (13).

$$q_i|_{v_i} = \sum_{k=1}^{n-1} \rho_k (1 - \rho_1 - \cdots - \rho_k)^{v_i}. \tag{13}$$

Consider next that p_i is in relative state Obtaining, and thus, we know that p_i transmitted during the preceding broadcasting round and transited from relative state Ready to Obtaining. Moreover, p_i is using the same timeslot for the current broadcasting round. The only neighbors of p_i that are using the same timeslot are the neighbors that are also in relative state Obtaining and have chosen the same listening/signaling period as p_i during the preceding broadcasting round. Let us denote by ℓ_i, the number of such neighbors. Given ℓ_i the probability $w_i|_{\ell_i}$ that p_i is allocated to the timeslot is given by

$$w_i|_{\ell_i} = \sum_{k=1}^{n-1} \rho_k (1 - \rho_1 - \cdots - \rho_k)^{\ell_i}. \tag{14}$$

We have that ℓ_i is stochastically dominated by v_i [47], that is, $E(\ell_i) \leq E(v_i)$. Indeed, v_i is a random variable that counts the number of neighbors that choose the same timeslot as p_i, while ℓ_i counts the number of neighbors that choose the same timeslot *and* listening/signaling period as p_i. For $n \geq 2$, ℓ_i's expected value is smaller than v_i's expected value. To conclude, we remark that expressions (13) and (14) are the same decreasing function, $f_i \to \sum_{k=1}^{n-1} \rho_k (1 - \rho_1 - \cdots - \rho_k)^{f_i}$, that is evaluated at two different points, v_i and ℓ_i, respectively. Moreover, since ℓ_i is stochastically dominated by v_i, (15) holds as follows:

$$w_i = E\left(w_i|_{\ell_i}\right) \geq E\left(q_i|_{v_i}\right) = q_i. \tag{15}$$

□

Proposition 5 demonstrates (16) and leads us toward the proof of Theorem 3.

Proposition 5. *Let $\rho_i = 1/MaxRnd$. Equation* (16) *bounds from below the probability q_i; see Appendix D.*

Consider

$$q_i \geq \max\left\{\left(\frac{n-1}{2n}\right)^{d_i/T}, \frac{1}{d_i/T + 1}\left(1 - \frac{1}{n}\right)^{d_i/T+1}\right\}. \tag{16}$$

The first bound, $1/q_i \leq (2n/(n-1))^{d_i/T}$ ((7)), has a simple intuitive interpretation. Let us consider first that two nodes compete for a same timeslot. The two nodes choose independently any of the n listening/signaling periods and there are n^2 different possible outcomes. Among these outcomes n correspond to the situation where the two nodes choose the same listening/signaling period and there is no winner. We then have $n^2 - n = n(n-1)$ outcomes that lead to a winner. There is then a probability of $n(n-1)/n^2 = (n-1)/n$ that one of the nodes wins the (listening/signaling) competition. Since the game is symmetric, the probability that p_i wins is $(n-1)/(2n)$. The fact that we have T timeslots divides the number of competing nodes, d_i, and implies that there are d_i/T competing nodes for the same timeslot. If we interpret the game as a collection of d_i/T independent games, where for each game p_i wins with probability $(n-1)/(2n)$, thus, the probability q_i that p_i wins is $((n-1)/2n)^{d_i/T}$. The inverse of this expression gives the average time for the event to occur and is the bound by (7).

6. Network Convergence

We estimate the expected time for the entire network to reach a safe configuration in which all nodes are allocated with timeslots. The estimation is based on the number of nodes that are the earliest to signal in their broadcasting timeslot. These nodes are winners of the (listening/signaling) competition and are allocated to their chosen timeslots. However, counting only these nodes leads to underestimating the number of allocated nodes, which then results in an overestimation of the convergence time. Indeed, node $p_i \in P$ might have a neighbor $p_j \in \mathcal{N}_i$ that selects the earliest listening/signaling period in $\mathcal{N}_i$, but p_j does not transmit because one of its neighbors, $p_k \in \mathcal{N}_j \setminus \mathcal{N}_i$, had transmitted in an earlier listening/signaling period. Our bound considers only p_k while both p_i and p_k transmit, became p_j is inhibited by p_k's beacon.

Lemma 6 shows that the assumption that the nodes are allocated independently of each other is suitable for bounding the network convergence time, $\mathcal{S}$. Theorem 7 uses Lemma 6 for bounding the network convergence time, $\mathcal{S}$.

In Section 5, we prove a bound on the expected time, $\mathcal{S}_i$, for a *single* node to be allocated to a timeslot. We observe that the bound depends uniquely on the number of listening/signaling periods, n, as well as the ratio between the extended degree and the frame size, d_i/T. In order to obtain a bound valid for all nodes, we bound this ratio with x/T where x is as defined in Lemma 6. We note that the time needed for the allocation of timeslots to *all* the nodes depends on N, the total number of nodes.

In detail, the convergence time estimation considers the (fixed and independent) bound, q_i, for the probability that a node reaches the relative state Allocated within a broadcasting round. Then, the convergence time, t, is a random variable with geometric probability, that is, $P(t = k) = (1 - q)^{k-1} q$. Let us denotes $t_1, \ldots, t_N$ the time it takes for the nodes $p_1, \ldots, p_N$ to respectively reach the relative state Allocated.

The convergence time, $\mathcal{S}$, for all the nodes is given with $\max(\{t_1, \ldots, t_N\})$, which depends on N.

Lemma 6. *The expected number of nodes, $E(W)$, that win the (listening/signaling) competition after one broadcasting round satisfies* (17)*, where $x = 2A/N$, T is the number of timeslots, A the number of edges in the interference graph, G, and $N = |P|$ the number of nodes that attempt to access the communication media.*

Consider

$$E(W) \geq N \sum_{j=1}^{n} \rho_j \left(1 - \left(\rho_1 + \cdots + \rho_j\right)\right)^{x/T}. \tag{17}$$

Proof. The nodes that are allocated to a timeslot can previously be on relative state Ready or Obtaining. The probability of a transition from relative state Obtaining to Allocated is w_i, and a transition from relative state Ready to Allocated is q_i. As proved in Lemma 4, we always have $w_i \geq q_i$. To bound the number of nodes that get allocated during a broadcasting round, we use the lower bound on the probability q_i that a node gets allocated to a timeslot. Moreover, in the computations, we use the AM-GM bound [48], which says that if $\sum b_k = 1$ then $\prod a_k^{b_k} \leq \sum b_k a_k$ and denote by d_i the number of neighbors of node p_i. As proved in Proposition D.1, since there are T timeslots the number of neighbors of i that choose the same timeslot as i and compete for it is bounded by d_i/T. This lemma is proved by (18), where the last line of the expression holds because $\sum_i d_i = 2A$.

One has

$$\begin{aligned}
E(W) &\geq E\left(\sum_{i=1}^{N} 1_{|p_i \text{ selects the earliest signaling period}}\right) \\
&= \sum_{i=1}^{N} \left(\rho_1 (1-\rho_1)^{d_i/T} + \cdots \rho_{n-1} \left(1 - \sum_{k=1}^{n-1} \rho_k\right)^{d_i/T}\right) \\
&= \sum_{j=1}^{n} N \sum_{i=1}^{N} \frac{1}{N} \rho_j \left(1 - \sum_{k=1}^{j} \rho_k\right)^{d_i/T} \qquad (18) \\
&\geq N \sum_{j=1}^{n} \prod_{i=1}^{N} \rho_j^{1/N} \left(1 - \sum_{k=1}^{j} \rho_k\right)^{d_i/NT} \\
&= N \sum_{j=1}^{n} \rho_j \left(1 - \sum_{k=1}^{j} \rho_k\right)^{(1/TN)\sum d_i} \qquad (19) \\
&= N \sum_{j=1}^{n} \rho_j \left(1 - \sum_{k=1}^{j} \rho_k\right)^{x/T}.
\end{aligned}$$

We note that we use the AM-GM bound to reach the 4th row of (18). □

By arguments similar to the ones used in the proof of Proposition 5, we deduce that if N nodes compete, the expected number $E(W)$ of nodes that get allocated to a timeslot is lower bounded as follows:

$$E(W) \geq N \max\left\{\left(\frac{n-1}{2n}\right)^{x/T}, \frac{((n-1)/n)^{x/T+1}}{x/T+1}\right\}. \tag{20}$$

Theorem 7 bounds the system convergence time. We numerically validate Theorem 7; see Figure 2. Moreover, our experiments showed that the average convergence time of the network is below the upper bound of (21).

Theorem 7 (global convergence). *The expected number of retransmissions is smaller than $(2n/(n-1))^{d/T} - 1$, where $d = \max(\{d_i : p_i \in P\})$. Hence, one has that the expected number of broadcasting rounds, $\mathcal{S}$, that guarantee that all nodes reach the relative state* Allocated *satisfies*

$$\mathcal{S} \leq \left(\frac{2n}{n-1}\right)^{d/T}. \tag{21}$$

Moreover, given that there are N nodes in the network and $\alpha \in (0,1)$, the network convergence time is bounded by (22) *with probability $1 - \alpha$.*

Consider

$$k = 1 + \frac{\log\left(1 - \sqrt[N]{1-\alpha}\right)}{\log\left(1 - ((n-1)/2n)^{d/T}\right)}. \tag{22}$$

This means that with probability α, all nodes are allocated with timeslots in maximum k broadcasting rounds; see Figure 3.

Proof. Theorem 3 bounds the convergence time of a particular processor; see (7). Lemma 6; see (20) $E(W) \geq N((n-1)/2n)^{x/T}$, proves that this bound is still valid if we replace the term d_i/T with x/T; that is, we consider the average degree instead of the particular degree of a node. If we replace x/T by $\max\{d_i\}/T$ in expression (20) we obtain a larger bound because $x/T \leq \max\{d_i\}/T$; that is, $E(W) \geq N((n-1)/2n)^{x/T} \geq N((n-1)/2n)^{\max\{d_i\}/T}$.

The bound $E(W) \geq N((n-1)/2n)^{\max\{d_i\}/T}$ and the dis- cussion in the 1st paragraph of Section 6 show that the number of processors that are allocated during a broadcasting round is bounded by the random variable $\sum_{i=1}^{N} z_i$, where z_i are identically and independently distributed random variables that are 1 with probability $((n-1)/2n)^{\max\{d_i\}/T}$ and 0 with probability $1 - ((n-1)/2n)^{\max\{d_i\}/T}$ (the second random variable dominates the first one; see [49]). This means that we lower bound the number of processors that are allocated if we consider that they are allocated independently with probability $((n-1)/2n)^{\max\{d_i\}/T}$.

While the processors get allocated to a timeslot, the parameters d_i and T change because some timeslots are no longer available (T decreases, some nodes are allocated, d_i decreases). Actually the ratio becomes $(\max\{d_i\} - h_i)/(T - f_i)$, where $h_i \geq f_i$ because if a timeslot is allocated or sensed used by processor p_i, then T, the number of available timeslots decreases by 1 and d_i, the number of competing nodes, must decrease at least by one since there must be at least one

processor that uses the busy timeslot (there may be multiple that are in state Obtaining). Under these circumstances, we always have $\max\{d_i\}/T \geq (\max\{d_i\} - h_i)/(T - f_i)$. Thus, we can obtain a lower bound for the expected time to reach the relative state Allocated by assuming that all nodes are allocated independently with probability $x = ((n-1)/2n)^{\max\{d_i\}/T}$. We simplify the following arguments by using this definition of x.

To bound the number of broadcasting rounds, we consider the following game. The bank pays 1 unit to the nodes that get in state Allocated (get allocated to a timeslot) and receives $x/(1-x)$ units per nodes that fail to get in state Allocated. The game is fair because in each round the expected gain is $1\times x - x/(1-x)\times(1-x) = 0$. If we denote by W_i the number of processors that get in state Allocated during the ith broadcasting round and by L_i the number of processors that fail, we have that the gain is given by (23), where t denotes the total number of rounds.

One has

$$\text{gain} = \sum_{i=1}^{t}\left(\frac{x}{1-x}L_i - W_i\right). \tag{23}$$

The expected gain is 0 because the game is fair ($E(\text{gain}) = 0$) and $\sum_{i=1}^{t} W_i = N$ because eventually all the nodes get in state Allocated and the bank pays 1 unit for each such processors. If we compute the expectation on both sides of (23), we then obtain

$$N = \frac{x}{1-x}E\left(\sum_{i=1}^{t}L_i\right). \tag{24}$$

We observe that $E(\sum_{i=1}^{t} L_i)$ is the expected total number of retransmissions and $E(\sum_{i=1}^{t} L_i)/N$ is the average expected number of retransmissions whose value is $(1-x)/x$. Replacing x with its expression, we obtain that the average number of retransmission is bounded by $(2n/(n-1))^{\max\{d_i\}/T} - 1$, and, this leads to the bound (21).

To prove the second assertion, let $t_1, \ldots, t_N$ be the convergence time of nodes $1, \ldots, N$, respectively. The random variables, t_i, are bound by random variables with geometric random distribution with expectation of $(2n/(n-1))^{d/T}$, with $d = \max\{d_i : d_i \in P\}$. We require that $t_{\max} = \max\{t_1, \ldots, t_N\}$ in order to ensure that all nodes are allocated with timeslots. The fact that the random variables, t_i, are independent and identically distributed, implies (25), where t is a random geometrical random variable, that is, $\Pr(t = k') = (1-q)^{k'-1}q$ and $\Pr(t \geq k') = (1-q)^{k'-1}$.

Consider

$$\begin{aligned}\Pr\left(t_{\max} \leq k'\right) &= P\left(t_1 \leq k', \ldots, t_N \leq k'\right)\\ &= \Pr\left(t_1 \leq k'\right)\cdot\cdots\cdot P\left(t_N \leq k'\right)\\ &= P\left(t \leq k'\right)^N.\end{aligned} \tag{25}$$

Which $t_{\max} \leq k'$ satisfies (26) with probability α?

One has

$$\begin{aligned}\Pr\left(t_{\max} < k'\right) &= \Pr\left(t < k'\right)^N\\ &= \left(1 - (1-q)^{k'-1}\right)^N \geq 1 - \alpha\end{aligned} \tag{26}$$

By solving (26), we observe that (26) is satisfied for any $k' \geq k$, where k satisfies (22). This proves that, with probability $1-\alpha$, the network convergence time is bounded by (22). □

7. Implementation

Existing MAC protocols offer mechanisms for dealing with contention (timeslot exhaustion) via policies for administering message priority, such as [13]. In particular, the `IEEE 802.11p` standard considers four priorities and techniques for facilitating their policy implementation. We explain similar techniques that can facilitate the needed mechanisms.

7.1. Prioritized Listening/Signaling Periods. We partition the sequence of listening periods, $[0, MaxRnd)$, into $MaxPrt$ subsequences, $[0, MaxRnd_0), \ldots [MaxRnd_{MaxPrt-2}, MaxRnd_{MaxPrt-1})$, where $[MaxRnd_{k-1}, MaxRnd_k)$ is used only for the kth priority. For example, suppose that there are six listening/signaling periods and that nodes with the highest priority may use the first three listening/signaling periods, $[0, 2]$, and nodes with the lowest priority may use the last three, $[3, 5]$. In the case of two neighbors with different listening period parameters, say, $[0, 2]$ and $[3, 5]$, that attempt to acquire the same broadcasting timeslot, the highest priority node always attempts to broadcast before the lowest priority one.

7.2. TDMA-Based Backoff. Let us consider two backoff parameters, CW_{start} and CW_{end}, that refer to the maximal and minimal values of the contention window. Before selecting an unused timeslot, the procedure counts a random number of unused ones. Algorithm 2 presents an implementation of the select_unused() function that facilitates backoff strategies as an alternative to the implementation presented in line 29 of Algorithm 1.

The statically allocated variable *count* records the number of backoff steps that node p_i takes until it reaches the zero value. Whenever the function select_unused() is invoked with $count_i = 0$, node p_i assigns to $count_i$ a random integer from $[\text{CW}_{\text{start}}, \text{CW}_{\text{end}}]$ (cf. line 7). Whenever the value of $count_i$ is not greater than the number of unused timeslots, the returned timeslot is selected uniformly at random (cf. lines 8 to 9). Otherwise, a $\perp$-value is returned after deducting the number of unused timeslots during the previous broadcasting round (cf. lines 6 and 10).

8. Discussion

Thus far, both schedule-based and nonschedule-based MAC algorithms could not consider timing requirements within a provably short recovery period that follows (arbitrary)

```
   Additional constants and variables
(2) CW_start and CW_end : backoff parameters
    count : statically allocated variable that counts the backoff steps
(4)
    Function select_unused(set)
(6) let rtn_val = ⊥ _v_ // indicate busy channel (default return value)
    if count ≤ 0 then count ← uniform_select([CW_start, CW_end])
(8) count ← count − | set |
    if count ≤ 0 then (count, rtn_val) ← (0, uniform_select(set))
(10) return rtn_val
```

Algorithm 2: Select_unused() with TDMA-based backoff.

transient faults and network topology changes. This work proposes the first self-stabilizing TDMA algorithm for DynWANs that has a provably short convergence period. Thus, the proposed algorithm possesses a greater predictability degree, whilst maintaining low communication delays and high throughput.

In this discussion, we would like to point out the algorithm's ability to facilitate the satisfaction of severe timing requirements for DynWANs by numerically validating Theorem 7. As a case study, we show that, for the considered settings of Figure 2, the global convergence time is brief and definitive. Figure 3 shows that when allowing merely a small fraction of the bandwidth to be spent on frame control information, say, three listening/signaling periods, and when considering 99% probability to convergence within a couple of dozen TDMA frames, the proposed algorithm demonstrates a low dependency degree on the number of nodes in the network even when considering 10,000 nodes.

We have implemented the proposed algorithm, extensively validated our analysis via computer simulation, and tested it on a platform with more than two dozen nodes [18]. These results indeed validate that the proposed algorithm can indeed facilitate the implementation of MAC protocols that guarantee satisfying these severe timing requirements.

The costs associated with predictable communications, say, using cellular base stations, motivate the adoption of new networking technologies, such as MANETs and VANETs. In the context of these technologies, we expect that the proposed algorithm will contribute to the development of MAC protocols with a higher predictability degree.

Appendices

The proof of Theorem 2 uses the propositions in Appendices A and B.

A. Properties (1) to (3)

Propositions A.1, A.2 and A.3 imply that properties (1), (2), and, respectively, (3) hold within two broadcasting rounds (or one complete broadcasting round). Let R be an execution of the MAC algorithm in Algorithm 1, $x > 0$ an integer, and $c_0^{\text{timeslot}}(x)$ the first configuration in a complete broadcasting round $R(x) = (c_0^{\text{timeslot}}(x), \ldots c_{T-1}^{\text{carrier\_sense/receive}}(x))$. We note that $c_0^{\text{timeslot}}(x)$ follows an arbitrary starting configuration.

Proposition A.1 shows that, within a broadcasting round from $c_0^{\text{timeslot}}(x)$, Property (1) holds.

Proposition A.1. *In* $c_0^{\text{timeslot}}(x+1)$*, it holds that* signal_i $=$ *false.*

Proof. The value of signal_i is updated in line 18 (assigned to true) and in lines 12, 20, and 27 (assigned to false). Let us look into these assignments.

In every timeslot, the value false is assigned to signal_i (cf. line 12). Suppose that the function send() is called, and thus, true is assigned to signal_i (line 18). We propose that before returning from the function send() and after true is assigned to signal_i (line 18), node p_i must assign false to signal_i either in line 20 or 27. To see that, let us look at lines 18 and 19. Eventually either signal_i = false (because of an assignment in line 27) or $\rho(k)$ = true (line 19) holds (note the condition when $k = MaxRnd$). The latter case implies the execution of line 20. □

Proposition A.2 shows that, within a broadcasting round from $c_0^{\text{timeslot}}(x)$, Property (2) holds.

Proposition A.2. $(\exists t \in \textit{unused\_set}_i \setminus \{s_i\}) \leftrightarrow (\nexists p_k \in \mathcal{N}_i : s_k = t)$ *in* $c_0^{\text{timeslot}}(x+1)$.

Proof. Recall that $\text{unused\_set}_i = \{k : \text{unused}_i[k] = \text{true}\}$ (see line 6) and that the proposition statement does not consider the cases in which: (1) $s_i = s_k$ (because $t \neq s_i$) in $c_0^{\text{timeslot}}(x+1)$, or (2) There exists a configuration $c \in R(x)$, such that $s_k \neq \ \perp$ in c and $s_k = \perp$ in $c_0^{\text{timeslot}}(x+1)$ (because by unused_set's definition, $\perp$ is never in unused_set_i).

We note that in every broadcasting round, node $p_k \in P$ *at most once* (1) Allocates the broadcasting timeslot s_k (when $t_k = 0$; see line 11), (2) transmits a packet (when $t_k = s_k$; see line 13), and (3) deallocates the broadcasting timeslot s_k (by assigning $\perp$ to s_k when $t_k = s_k$ and the carrier_sense(t) event is raised; see line 26). Moreover, node p_i updates $\text{unused}_i[t]$ only in lines 12 (true) and 27 (false), when p_i removes stale information just before timeslot t, and respectively, when the event carrier_sense(t) is raised.

Line 12 is executed at the start of every timeslot, whereas line 27 is executed after and only when the event

Defining optimal transmission probabilities for any choices of T, n, and d_i is not possible. We choose to consider and look for optimal choices when $d_i \simeq T$ (the "hard" case) and make a case for a uniform probability $\rho_i = 1/n : i \in [1, n]$. Let us consider node $p_i \in P$ that competes, together with $k-1$ other neighbors, for the same unique timeslot. The probability that node p_i wins the (listening/signaling) competition is $\rho_1(1-\rho_1)^{k-1}$, where ρ_1 is the probability of choosing the first listening/signaling period. The value $\rho_1 = 1/k$ maximizes this probability. In the more general case where there is more than one timeslot, we consider a strategy that aims at guessing the number, k, of competing neighbors, which the optimal probability of transmission depends on. During the first listening/signaling period, the strategy considers the case in which there are $n = MaxRnd$ signaling nodes, and thus, the transmission probability is $1/MaxRnd$, where $MaxRnd \simeq T$. During the second listening/signaling period, the strategy considers the case in which there are $MaxRnd - 1$ neighbors, and thus, the transmission probability is $1/(MaxRnd - 1)$, and so on. This *sequential* selection of the listening/signaling period leads to a uniform choice of a listening/signaling neighbor. The above strategy is driven by a heuristic in which nodes signal with probability that is optimal for the case of $n \simeq T$, and thus, it depends on the number of competing neighbors.

Box 2: Transition probability, ρ_i, for listening/signaling periods (line (19) in Algorithm 1).

carrier_sense(t) is raised. The event carrier_sense(t) is raised after and only when the node $p_k \in \mathcal{N}_i$ transmits in timeslot t. In other words, *none* of p_i's neighbors, $p_k \in \mathcal{N}_i$, that transmits in timeslot $s_k = t$, can avoid causing the event carrier_sense(t) to be raised, and timeslot t to be included in unused_set$_i \setminus \{s_i\}$. □

Proposition A.3 shows that, within a broadcasting round from $c_0^{\text{timeslot}}(x)$, Property (3) holds.

Proposition A.3. $(s_i \neq \perp) \vee (unused\_set_i \setminus \{s_i\} \neq \emptyset)$ *holds in* $c_0^{\text{timeslot}}(x+1)$.

Proof. If $s_i \neq \perp$ in $c_0^{\text{timeslot}}(x+1)$, we are done. Let us suppose that $s_i = \perp$ in $c_0^{\text{timeslot}}(x+1)$ and show that unused_set$_i \setminus \{s_i\} \neq \emptyset$ in $c_0^{\text{timeslot}}(x+1)$.

Let us assume, in the way of proof by contradiction, that, unused_set$_i \setminus \{s_i\} = \emptyset$ and show that $d_i/T > 1$, that is, a contradiction with the assumption that for all $p_i \in P$: $d_i/T \leq 1$.

Recall that unused_set$_i = \{k : \text{unused}_i[k] = \text{true}\} \subseteq [0, T-1]$ (see line 6). Therefore, the assumption that $s_i = \perp$ implies that unused_set$_i$ = unused_set$_i \setminus \{s_i\} \subseteq [0, T-1]$, because by unused_set's definition, $\perp$ is never in unused_set$_i$.

By Proposition A.2, we can say that for all $t \in [0, T-1] : (\nexists t \in \text{unused\_set}_i) \leftrightarrow (\exists p_k \in \mathcal{N}_i : s_k = t)$. Since unused_set$_i \subseteq [0, T-1]$, we can write $[0, T-1] \setminus$unused_set$_i \subseteq \{s_k \in [0, T-1] : p_k \in \mathcal{N}_i\}$. By the fact that unused_set$_i = \emptyset$, we have that $T \leq |\{s_k \in [0, T-1] : p_k \in \mathcal{N}_i\}|$. Since $d_i = |\mathcal{N}_i|$ (by definition), we have that $|\{s_k \in [0, T-1] : p_k \in \mathcal{N}_i\}| \leq d_i$, which implies $T \leq d_i$: a contradiction with the assumption that $d_i/T \leq 1$. □

B. Properties (4) to (5)

Appendix A shows that, starting from an arbitrary configuration, node $p_i \in P$ enters the relative state Ready within two broadcasting rounds. This section considers the probability for p_i to enter the relative states Obtaining and Allocated.

Let $x > 0$ and R be an execution of the MAC algorithm in Algorithm 1. Suppose that $c_0^{\text{timeslot}}(x)$ is the first configuration in a complete broadcasting round $R(x)$ for which properties (1) to (3) hold in configuration $c_0^{\text{timeslot}}(x)$ with respect to node $p_i \in P$; that is, p_i is in relative state Ready, Obtaining or Allocated. Propositions B.1, B.2 and B.3 show that there is a nonzero probability that node p_i enters the relative state Allocated from either Ready or Obtaining in configuration $c_0^{\text{timeslot}}(x+1)$.

Proposition B.1 shows that p_i attempts to broadcast once in every round.

Proposition B.1. *During broadcasting round* $R(x)$, p_i *executes line 13 and calls the function* send().

Proof. If $s_i \neq \perp$ in $c_0^{\text{timeslot}}(x)$, we are done by lines 11 and 13. Let us consider the case of $s_i = \perp$ in $c_0^{\text{timeslot}}(x)$. By Property (4), unused_set$_i \neq \emptyset$, and thus, when line 11 is executed, the function select_unused() returns a non-$\perp$ element from unused_set$_i$ and $s_i \neq \perp$ when executing line 13. □

Propositions B.2 and B.3 consider the set $M_i(x+1) = \{p_k \in \mathcal{N}_i : s_k = t'\}$ and the number $m_i = |M_i(x+1)|$ of p_i's neighbors that attempt to broadcast during p_i's timeslot, t', of broadcasting round $R(x)$.

Let ρ_j be the probability for p_i to transmit in the jth listening/signaling period of timeslot t' (cf. line 19). This paper considers the concrete transmission probability $\rho_i = 1/MaxRnd$. We motivate our implementation choice of the transmission probability, ρ_i, in Box 2. Note that the *sequential* selection of the broadcasting rounds with probability $1/(MaxRnd - k + 1)$ leads to the uniform selection $\rho_k = 1/MaxRnd$.

Proposition B.2 considers p_i's chances to be the only one to transmit in its neighborhood.

Proposition B.2. *There is a nonzero probability, $OnlyOne_i(x)$ (cf. (B.1)), that only node p_i transmits in its broadcasting timeslot, t', of broadcasting round $R(x)$.*

One has

$$OnlyOne_i(x)\,|_{m_i>0} = \rho_1(1-\rho_1)^{m_i} + \rho_2(1-\rho_1-\rho_2)^{m_i} + \cdots + \rho_{n-1}\left(1-\sum_{\ell=1}^{n-1}\rho_k\right)^{m_i} \quad \text{(B.1)}$$

Proof. We show that there is a nonzero probability that only node p_i transmits in its broadcasting timeslot, t', of broadcasting round $R(x)$. Let us look at p_i and the nodes in $M_i(x)$ while they attempt to broadcast in the steps $a_i^{\text{timeslot},t'}(x)$ and $a_k^{\text{timeslot},t'}(x)_{|k\in M_i(x)}$. All of these steps include the execution of line 19; namely, each node chooses to transmit in listening/signaling period $\ell \in [0, MaxRnd]$ with probability $\rho_\ell = 1/(MaxRnd - \ell)$. Therefore, for any $MaxRnd > 0$, there is a nonzero probability, $OnlyOne_i(x)$, that, during timeslot t', node p_i transmits in the listening/signaling period $a \in MaxRnd$ and no node in $M_i(x)$ transmits in round a (or in an earlier one).

We note that the fact that p_i transmits first during timeslot t' implies that it is the only one to transmit during t'. This is because once p_i transmits a beacon in step $a_i^{\text{timeslot},t'}(x)$ (which includes the execution of line 21), node $p_j \in \mathcal{N}_i \supseteq M_i(x)$ raises the event carrier_sense(t') immediately after $a_i^{\text{timeslot},t'}(x)$. Thus, for all $p_j \in M_i(x)$ we have that immediately after step $a_i^{\text{timeslot},t'}(x)$, node p_j takes step $a_j^{\text{carrier\_sense},t'}(x)$, which includes the execution of lines 26 and 27 that assign $\perp$ to s_j and false to signal$_j$. Thus, p_j leaves the (listening/signaling) competition for timeslot t' (see line 18) and does not transmit its DATA packet (see line 23).

We now turn to calculate $OnlyOne_i(x)$. Let the variable $m_i = |M_i(x)|$ denote the number of nodes that select the same timeslot as p_i in configuration $c_0^{\text{timeslot: s}\neq\perp}$. The value of $OnlyOne_i(x)$ depends on the value of m_i, and we denote this dependence with the notation $q(i)|_{m_i}$ (conditional probability). It means the value of $OnlyOne_i(x)$ depends on the value of m_i. The value of $OnlyOne_i(x)$ for $m_i = 0$ is $OnlyOne_i(x)|_{m_i=0} = 1$. For the case of $m_i > 0$, $OnlyOne_i(x)$'s value is given by (B.1) (that appears again next), where ρ_j is the probability for transmitting in the jth listening/signaling period.

Consider

$$OnlyOne_i(x)\,|_{m_i>0} = \rho_1(1-\rho_1)^{m_i} + \rho_2(1-\rho_1-\rho_2)^{m_i} + \cdots + \rho_{n-1}\left(1-\sum_{\ell=1}^{n-1}\rho_k\right)^{m_i} \quad [\text{clone of (B.1)}] \quad \text{(B.2)}$$

We note that the jth term in (B.1) is the probability that node p_i selects the jth listening/signaling period and all its neighbors select a later listening/signaling period. □

Proposition B.3 shows that once a node is the only one in its neighborhood to transmit during its broadcasting timeslot, it enters the relative state Allocated.

Proposition B.3. *$M_i(x) = \emptyset$ (or having none of the nodes in $M_i(x)$ transmitting during timeslot t') implies that node p_i is in the relative state* Allocated *in $c_0^{\text{timeslot}}(x+1)$.*

Proof. We need to show that, in $c_0^{\text{timeslot}}(x+1)$, we have that $s_i = t' \neq \perp$ and for all $p_j \in \mathcal{N}_i$: $s_i \neq s_j$.

Showing That $s_i = t' \neq \perp$ in $c_0^{\text{timeslot}}(x+1)$. The proposition assumes that $t' \neq \perp$ in $c_0^{\text{timeslot}}(x)$. We wish to show that $s_i = t'$ in $c_0^{\text{timeslot}}(x+1)$, which implies that $s_i \neq \perp$ holds in $c_0^{\text{timeslot}}(x+1)$ and throughout $R(x+1)$.

Since the variable s_i is assigned only in lines 11 (when $t_i = 0$) and 26 (when $t_i = t'$), it is sufficient to show that line 26 is not executed by any step during timeslot t' of broadcasting round $R(x)$, that is, $a_i^{\text{carrier\_sense},t'}(x) \notin R(x)$.

Node p_i raises the event carrier_sense only during timeslots in which p_i's neighbor, p_j, transmits. By the proposition assumptions that, during timeslot t' of broadcasting round $R(x)$, none of p_i's neighbors transmits, we have $a_i^{\text{carrier\_sense},t'}(x) \notin R(x)$. Moreover, $a_i^{\text{timeslot},t'}(x+1)$ does not include an execution of line 11 that changes the value of s_i, because $s_i = t' \neq \perp$ in $c_0^{\text{timeslot}}(x+1)$.

Showing That for All $p_j \in \mathcal{N}_i : s_i \neq s_j$ in $c_0^{\text{timeslot}}(x+1)$. The proposition assumes that for all $p_j \in \mathcal{N}_i : s_i \neq s_j$ in $c_0^{\text{timeslot}}(x)$. We wish to show that the same holds in $c_0^{\text{timeslot}}(x+1)$. Since the variable s_j is assigned to a non-$\perp$ value only in line 11 when $t_i = 0$, it is sufficient to show that when line 11 is executed in step $a_j^{\text{timeslot},0}(x+1)$ the function select_unused() considers a set that does not include p_i's timeslot, s_i. This is implied by the facts that for all $p_j \in \mathcal{N}_i$: unused$_j[t']$ = false (Claim 10.1) and $s_i = t'$ (first item of (II) of this proof) in $c_0^{\text{timeslot}}(x+1)$. □

C. Theorem 2

Theorem 2 shows that all nodes are allocated eventually with timeslots (convergence) and once all nodes are allocated, they stay this way (closure).

Theorem 2 (self-stabilization). *The MAC algorithm in Algorithm 1 is a self-stabilizing algorithm with respect to the task τ_{TDMA}.*

Proof. After the previous proof of propositions, we can demonstrate this theorem.

(i) *Convergence.* We need to show that properties (1) to (5) eventually hold in configuration $c_0^{\text{timeslot}}(x+y)$ for a finite value of $y > 0$. Propositions A.1, A.2 and A.3 imply that properties (1), (2), and, respectively, (3) within two broadcasting round.

Propositions B.1, B.2 and B.3 show that there is a nonzero probability that node p_i enters the rela-

tive state Allocated from either Ready or Obtaining within one broadcasting round. Thus, by analyzing the expected time of the scheduler-luck games [4, 51], we have that y has a finite value. Further analysis of y appears in Theorems 3 and 7.

(ii) *Closure.* Suppose that $c_0^{\text{timeslot}}(x) \in R$ is a safe configuration and let $p_i \in P$ be any node. By the assumption that $c_0^{\text{timeslot}}(x)$, we have that p_i is in the relative state Allocated; that is, properties (1) to (5) hold for any node p_i. We need to show that properties (1) to (5) hold in configuration $c_0^{\text{timeslot}}(x+1)$.

Propositions A.1, A.2, and A.3 imply properties (1), (2), and respectively, (3) (within one complete broadcasting round).

Properties (4) to (5) are implied by Proposition B.3 and the fact that Properties (4) to (5) hold in $c_0^{\text{timeslot}}(x)$, that is, $M(x) = \emptyset$. □

D. Bounding $OnlyOne_i(x)$

Propositions 5 and D.2 bound $OnlyOne_i(x)$'s value, where $R(x)$ is the xth broadcasting round in execution R of the MAC algorithm in Algorithm 1. We assume that properties (1) to (5) hold in the first configuration, $c_0^{\text{timeslot}}(x)$, of $R(x)$. These bounds are obtained by computing the expectation of $q_i|_{m_i}$ with respect to m_i, where $M_i(x) = \{p_k \in \mathcal{N}_i : s_k = t'\}$ in $c_0^{\text{timeslot}}(x)$ and $m_i = |M_i(x)|$. The reason is that m_i is a random variable, that is, $q_i = E(OnlyOne_i(x)|_{m_i})$, where the expectation is computed with respect to the random variable m_i.

We note that all the terms in (B.1) are convex functions of m_i. This means that by Jensen's inequality, we obtain a lower bound of q_i in (D.1) by evaluating the expression $q_i|_{m_i}$ at m_i's expectation, $E(m_i)$ [52].

One has

$$q_i = E\left(q_i|_{m_i}\right) \geq q_i|_{E(m_i)} \tag{D.1}$$

The expression on the right side of the inequality can be again lower bounded if we estimate an upper bound for $E(m_i)$. We proceed to the computations in the proof of Proposition D.2 after demonstrating Proposition D.1 which shows that $E(m_i)$ is bounded by the ratio d_i/T, which is rather intuitive but needs to be proved.

Proposition D.1. *In configuration $c_0^{\text{timeslot}}(x)$ it holds that $E(m_i) \leq d_i/T$, where $m_i = |M_i(x)|$.*

Proof. We show that $E(m_i) = d_i/T$ by considering configuration $c_0^{\text{timeslot}}(x)$. The maximal number of p_i's neighbors that might choose the same timeslot as p_i in configuration $c_0^{\text{timeslot}}(x)$ is $\sum_{p_j \in \mathcal{N}_i} 1_{\{s_j = \perp\}}$, because any node, $p_j \in \mathcal{N}_i$, that chooses a new broadcasting timeslot immediately before $c_0^{\text{timeslot}}(x)$ must have $s_j = \perp$ in configuration $c_0^{\text{timeslot}}(x)$. We compute the expected value of m_i in (D.2) as a function of the number of empty timeslots, e_i, that p_i selects from when choosing a new broadcasting timeslot, where $e_i = |\text{unused\_set}_i|$ in configuration $c_0^{\text{timeslot}}(x)$.

Consider

$$\begin{aligned} E(m_i) &= \sum_{t \in E_i} E(m_i \mid s_i = t) \Pr(s_i = t) \\ &= \sum_{t \in E_i} \frac{1}{e_i} E(m_i \mid s_i = t) \\ &= \sum_{t \in E_i} \frac{1}{e_i} E\left(\sum_{p_j \in \mathcal{N}_i} 1_{\{p_j \text{ chooses timeslot } t\}} \mid s_i = t \right) \\ &= \sum_{t \in E_i} \frac{1}{e_i} \sum_{p_j \in \mathcal{N}_i} \frac{1}{|E_j|} 1_{\{t \in E_j\}} 1_{\{s_j = \perp\}}. \end{aligned} \tag{D.2}$$

Our assumption that $d_i \leq T - 1$ implies that $e_i > 0$. Using that $d_i = \sum_{p_j \in \mathcal{N}_i} (1_{\{s_j \neq \perp\}} + 1_{\{s_j = \perp\}})$ and, $e_i \geq T - \sum_{p_j \in \mathcal{N}_i} 1_{\{s_j \neq \perp\}}$, we obtain the following:

$$\begin{aligned} E(m_i) &\leq \sum_{t \in E_i} \frac{1}{T - d_i + \sum_{p_j \in \mathcal{N}_i} 1_{\{s_j = \perp\}}} \sum_{p_j \in \mathcal{N}_i} \frac{1_{\{t \in E_j\}} 1_{\{s_j = \perp\}}}{|E_j|} \\ &= \frac{1}{T - d_i + \sum_{p_j \in \mathcal{N}_i} 1_{\{s_j = \perp\}}} \sum_{p_j \in \mathcal{N}_i} \frac{1_{\{s_j = \perp\}}}{|E_j|} \underbrace{\sum_{t \in E_i} 1_{\{t \in E_j\}}}_{|E_i \bigcap E_j|} \\ &\leq \frac{\sum_{p_j \in \mathcal{N}_i} 1_{\{s_j = \perp\}}}{T - d_i + \sum_{p_j \in \mathcal{N}_i} 1_{\{s_j = \perp\}}} \leq \frac{d_i}{T}. \end{aligned} \tag{D.3}$$

□

Proposition D.2. *One has*

$$q_i \geq \sum_{k=1}^{n} \rho_k \left(1 - \sum_{\ell=1}^{k} \rho_k\right)^{d_i/T} \quad [\textit{clone of } (6)]. \tag{D.4}$$

Proof. Proposition D.1 shows that $E(m_i) \leq d_i/T$. The proposition is demonstrated by evaluating expression (B.1) at $E(m_i) = d_i/T$; see (D.1). □

Proposition 5 considers the concrete transmission probability $\rho_i = 1/MaxRnd$.

Proposition 5. *Let $\rho_i = 1/MaxRnd$. Equation (16) bounds from below the probability q_i.*

Proof. In this proof, we use the letter n instead of $MaxRnd$ for reason of space. We replace ρ_i with $1/n$ in (6) to obtain (D.5).

Consider

$$q_i \geq \sum_{k=1}^{n} \frac{1}{n} \left(1 - \frac{k}{n}\right)^{d_i/T}. \tag{D.5}$$

Equation (D.6) is more compact than (D.5) and it is obtained by the fact that the function $(1 - x)^s$ is convex.

Consider

$$q_i \geq \sum_{k=1}^{n} \frac{1}{n}\left(1-\frac{k}{n}\right)^{d_i/T}$$
$$= \frac{1}{2n}\sum_{k=1}^{n}\left[\left(1-\frac{k}{n}\right)^{d_i/T}+\left(1-\frac{n-k+1}{n}\right)^{d_i/T}\right]$$
$$\geq (\text{convexity})\ \frac{1}{n}\sum_{k=1}^{n}\left(1-\frac{n+1}{2n}\right)^{d_i/T} = \left(1-\frac{n+1}{2n}\right)^{d_i/T}. \tag{D.6}$$

Another way to bound (D.5) is by considering the decreasing function $y \to (1-y)^x$, as in the following:

$$q_i \geq \sum_{k=1}^{n} \frac{1}{n}\left(1-\frac{j}{n}\right)^{d_i/T}$$
$$\geq \int_{1/n}^{1} (1-y)^{d_i/T}\,dy = \frac{1}{d_i/T+1}\left(1-\frac{1}{n}\right)^{d_i/T+1}. \tag{D.7}$$

□

Acknowledgments

The authors thank Thomas Petig and Oscar Morales for helping with improving the presentation. The work of Elad M. *Schiller* was partially supported by the EC, through project FP7-STREP-288195, KARYON (kernel-based architecture for safety-critical control). The proceeding and technical versions of this work appears in [53] and, respectively, [54].

References

[1] H. Hartenstein and K. Laberteaux, *VANET: Vehicular Applications and Inter-Networking Technologies*, Wiley, 2010.

[2] K. Bilstrup, E. Uhlemann, E. G. Ström, and U. Bilstrup, "Evaluation of the IEEE 802.11p MAC method for vehicle-to-cehicle communication," in *Proceedings of the 68th Semi-Annual IEEE Vehicular Technology (VTC '08)*, pp. 1–5, September 2008.

[3] K. Bilstrup, E. Uhlemann, E. G. Ström, and U. Bilstrup, "On the ability of the 802.11p MAC method and STDMA to support real-time vehicle-to-vehicle communication," *Eurasip Journal on Wireless Communications and Networking*, vol. 2009, Article ID 902414, 2009.

[4] S. Dolev, *Self-Stabilization*, MIT Press, 2000.

[5] S. Dolev and T. Herman, "Superstabilizing protocols for dynamic distributed systems," *Chicago Journal of Theoretical Computer Science*, 1997.

[6] N. Abramson, "Development of the ALOHANET," *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 119–123, 1985.

[7] W. G. Schmidt, "Satellite time-division multiple access systems: past, present and future," *Telecommunications*, vol. 7, no. 8, pp. 21–24, 1973.

[8] M. Haenggi, "Outage, local throughput, and capacity of random wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 8, pp. 4350–4359, 2009.

[9] P. Leone, M. Papatriantafilou, and E. Michael Schiller, "Relocation analysis of stabilizing MAC algorithms for large-scale mobile ad hoc networks," in *Proceedings of the 5th International Workshop on Algorithm Aspects of Wireless Sensor Networks (ALGOSENSORS '09)*, pp. 203–217, 2009.

[10] T. Herman and S. Tixeuil, "A distributed TDMA slot assignment algorithm for wireless sensor networks," in *Proceedings of the 5th International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS '04)*, vol. 3121 of *Lecture Notes in Computer Science*, pp. 45–58, Springer, 2004.

[11] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE INFOCOM*, pp. 1567–1576, June 2002.

[12] A. Cornejo and F. Kuhn, "Deploying wireless networks with beeps," in *Proceedings of the 24th International Symposium on Distributed Computing (DISC '10)*, N. A. Lynch and A. A. Shvartsman, Eds., vol. 6343 of *Lecture Notes in Computer Science*, pp. 148–162, Springer, 2010.

[13] R. Rom and F. A. Tobagi, "Message-based priority functions in local multiaccess communication systems," *Computer Networks*, vol. 5, no. 4, pp. 273–286, 1981.

[14] R. Scopigno and H. A. Cozzetti, "Mobile slotted Aloha for Vanets," in *Proceedings of the IEEE 70th Vehicular Technology Conference Fall (VTC '09)*, pp. 1–5, September 2009.

[15] F. Yu and S. Biswas, "Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1526–1537, 2007.

[16] S. Viqar and J. L. Welch, "Deterministic collision free communication despite continuous motion," in *Proceedings of the 5th International Workshop on Algorithm Aspects of Wireless Sensor Networks (ALGOSENSORS '09)*, pp. 218–229, 2009.

[17] R. Scopigno and H. A. Cozzetti, "GNSS synchronization in Vanets," in *Proceedings of the 3rd International Conference on New Technologies, Mobility and Security (NTMS '09)*, pp. 1–5, December 2009.

[18] M. Mustafa, M. Papatriantafilou, E. Michael Schiller, A. Tohidi, and P. Tsigas, "Autonomous TDMA alignment for VANETs," in *Proceedings of the IEEE 76th Vehicular Technology Conference (VTC '12)*, 2012.

[19] A. Daliot, D. Dolev, and H. Parnas, "Self-stabilizing pulse synchronization inspired by biological pacemaker networks," in *Proceedings of the 6th International Symposium on Self-Stabilizing Systems (SSS '03)*, S.-T. Huang and T. Herman, Eds., vol. 2704 of *Lecture Notes in Computer Science*, pp. 32–48, Springer, 2003.

[20] Y. Tadokoro, K. Ito, J. Imai, N. Suzuki, and N. Itoh, "Advanced transmission cycle control scheme for autonomous decentralized TDMA protocol in safe driving support systems," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 1062–1067, June 2008.

[21] H. A. Cozzetti and R. Scopigno, "RR-Aloha+: a slotted and distributed MAC protocol for vehicular communications," in *Proceedings of the IEEE Vehicular Networking Conference (VNC '09)*, pp. 1–8, October 2009.

[22] M. Lenoble, K. Ito, Y. Tadokoro, M. Takanashi, and K. Sanda, "Header reduction to increase the throughput in decentralized TDMA-based vehicular networks," in *Proceedings of the IEEE Vehicular Networking Conference (VNC '09)*, pp. 1–4, October 2009.

[23] R. Scopigno and H. A. Cozzetti, "Evaluation of time-space efficiency in CSMA/CA and slotted vanets," in *Proceedings of the IEEE 72nd Vehicular Technology Conference Fall (VTC '10)*, pp. 1–5, September 2010.

[24] F. Abrate, A. Vesco, and R. Scopigno, "An analytical packet error rate model for WAVE receivers," in *Proceedings of the IEEE 74th*

Vehicular Technology Conference (VTC '11), pp. 1–5, September 2011.

[25] H. A. Cozzetti, R. M. Scopigno, L. Casone, and G. Barba, "Comparative analysis of IEEE 802.11p and MS-aloha in vanet scenarios," in *Proceedings of the IEEE Asia-Pacific Services Computing Conference (APSCC '09)*, pp. 64–69, December 2009.

[26] M. Demirbas and M. Hussain, "A MAC layer protocol for priority-based reliable multicast in wireless ad hoc networks," in *Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems (BROADNETS '06)*, October 2006.

[27] J. R. S. Blair and F. Manne, "An efficient self-stabilizing distance-2 coloring algorithm," in *Proceedings of the SIROCCO*, S. Kutten and J. Zerovnik, Eds., vol. 5869 of *Lecture Notes in Computer Science*, pp. 237–251, Springer, 2009.

[28] S. Gollakota and D. Katabi, "Zigzag decoding: combating hidden terminals in wireless networks," in *Proceedings of the ACM SIGCOMM Conference on Data Communication (SIGCOMM '08)*, V. Bahl, D. Wetherall, S. Savage, and I. Stoica, Eds., pp. 159–170, August 2008.

[29] F. Kuhn, N. A. Lynch, and C. C. Newport, "The abstract MAC layer," in *Proceedings of the 23rd International Symposium on Distributed Computing (DISC '09)*, I. Keidar, Ed., vol. 5805 of *Lecture Notes in Computer Science*, pp. 48–62, Springer, 2009.

[30] M. M. Halldórsson and R. Wattenhofer, "Wireless communication is in APX," in *Proceedings of the ICALP*, S. Albers, A. Marchetti-Spaccamela, Y. Matias, S. E. Nikoletseas, and W. Thomas, Eds., vol. 5555 of *Lecture Notes in Computer Science*, pp. 525–536, Springer, 2009.

[31] O. Goussevskaia, R. Wattenhofer, M. M. Halldórsson, and E. Welzl, "Capacity of arbitrary wireless networks," in *Proceedings of the INFOCOM*, pp. 1872–1880, IEEE, 2009.

[32] R. Wattenhofer, "Theory meets practice, it's about time!," in *Proceedings of the 36th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM '10)*, Špindlerův Mlýn, Czech Republic, January 2010.

[33] C. Lenzen, T. Locher, P. Sommer, and R. Wattenhofer, "Clock synchronization: open problems in theory and practice," in *Proceedings of the 36th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM '10)*, Špindlerův Mlýn, Czech Republic, January 2010.

[34] J. Schneider and R. Wattenhofer, "Coloring unstructured wireless multi-hop networks," in *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC '09)*, S. Tirthapura and L. Alvisi, Eds., pp. 210–219, August 2009.

[35] A. Jhumka and S. S. Kulkarni, "On the design of mobility-tolerant TDMA-based media access control (MAC) protocol for mobile sensor networks," in *Proceedings of the ICDCIT*, T. Janowski and H. Mohanty, Eds., vol. 4882 of *Lecture Notes in Computer Science*, pp. 42–53, Springer, 2007.

[36] C. Lenzen, J. Suomela, and R. Wattenhofer, "Local algorithms: self-stabilization on speed," in *Proceedings of the 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS '09)*, R. Guerraoui and F. Petit, Eds., vol. 5873 of *Lecture Notes in Computer Science*, pp. 17–34, Springer, 2009.

[37] A. Lagemann, J. Nolte, C. Weyer, and V. Turau, "Mission statement: applying self-stabilization to wireless sensor networks," in *Proceedings of the 8th GI/ITG KuVS Fachgespräch "Drahtlose Sensornetze" (FGSN '09)*, pp. 47–49, August 2009.

[38] M. Arumugam and S. S. Kulkarni, "Self-stabilizing deterministic TDMA for sensor networks," in *Proceedings of the 2nd International Conference on Distributed Computing and Internet Technology (ICDCIT '05)*, G. Chakraborty, Ed., vol. 3816 of *Lecture Notes in Computer Science*, pp. 69–81, Springer, 2005.

[39] M. Arumugam and S. S. Kulkarni, "Self-stabilizing deterministic time division multiple access for sensor networks," *Journal of Aerospace Computing, Information and Communication*, vol. 3, no. 8, pp. 403–419, 2006.

[40] S. S. Kulkarni and M. Arumugam, "Infuse: a TDMA based data dissemination protocol for sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2, no. 1, pp. 55–78, 2006.

[41] P. Leone, M. Papatriantafilou, E. Michael S, and G. Zhu, "Analyzing protocols for media access control in large-scale mobile ad hoc networks," in *Proceedings of the Workshop on Self-Organising Wireless Sensor and Communication Networks (Somsed '09)*, Hamburg, Germany, October 2009.

[42] P. Leone, M. Papatriantafilou, E. Michael Schiller, and G. Zhu, "Chameleon-mac: adaptive and self-? algorithms for media access control in mobile ad hoc networks," in *Proceedings of the12th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS '10)*, S. Dolev, J. Arturo Cobb, M. J. Fischer, and M. and Yung, Eds., vol. 6366 of *Lecture Notes in Computer Science*, pp. 468–488, Springer, 2010.

[43] D. Aldous and J. Fill, "Reversible markov chain and random walks on graph," 1999, http://www.stat.berkeley.edu/~aldous/RWG/book.html.

[44] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System. Theory and Practice*, Springer, 1993.

[45] K. Jamieson, B. Hull, A. Miu, and H. Balakrishnan, "Understanding the real-world performance of carrier sense," in *Proceedings of the ACM SIGCOMM Workshop on Experimental Approaches to Wireless Network Design and Analysis (E-WIND '05)*, pp. 52–57, August 2005.

[46] R. Cogburn, "The ergodic theory of Markov chains in random environments," *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, vol. 66, no. 1, pp. 109–128, 1984.

[47] S. M. Ross, *Stochastic Processes*, John Wiley & Sons, 1996.

[48] J. M. Steele, *The Cauchy-Schwartz Master Class*, Cambridge University Press,, 2004.

[49] T. Lindvall, *Lectures on the Coupling Method*, Dover Publications, 1992.

[50] A. M. Crocker, W. L. Godson, and C. M. Penner, "Frontal contour charts," *Journal of the Atmospheric Sciences*, vol. 4, no. 3, 1947.

[51] S. Dolev, A. Israeli, and S. Moran, "Analyzing expected time by scheduler-luck games," *IEEE Transactions on Software Engineering*, vol. 21, no. 5, pp. 429–439, 1995.

[52] J. L. W. V. Jensen, "Sur les fonctions convexes et les inégalités entre les valeurs moyennes," *Acta Mathematica*, vol. 30, no. 1, pp. 175–193, 1906.

[53] P. Leone and E. Michael Schiller, "Self-stabilizing tdma algorithms for dynamic wireless ad-hoc networks," in *Proceedings of the ALGOSENSORS*, A. Bar-Noy and M. M. Halldórsson, Eds., Lecture Notes in Computer Science, Springer, 2012.

[54] P. Leone and E. M. Schiller, "Self-stabilizing TDMA algorithms for dynamic wireless ad-hoc networks," abs 1210.3061, submitted in *CoRR*, 2012, http://arxiv.org/abs/1210.3061.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 503965, 16 pages
http://dx.doi.org/10.1155/2013/503965

Research Article

HMM and Rule-Based Hybrid Intruder Detection Approach by Synthesizing Decisions of Sensors

Kyungmin Kim,[1] Kwang Il Park,[1] Yewon Jeong,[1] June Seok Hong,[2] Hak-Jin Kim,[3] and Wooju Kim[1]

[1] *Department of Information and Industrial Engineering, College of Engineering, Yonsei University, 134 Shinchon-dong, Seodaemoon-Gu, Seoul 120-749, Republic of Korea*
[2] *Division of Business Administration, Kyonggi University, 94-6 Yiui-gu, Yeongtong-gu, Kyonggi 443-760, Republic of Korea*
[3] *School of Business, Yonsei University, 134 Shinchon-dong, Seodaemoon-Gu, Seoul 120-749,, Republic of Korea*

Correspondence should be addressed to Wooju Kim; wkim@yonsei.ac.kr

Received 1 February 2013; Revised 14 May 2013; Accepted 14 May 2013

Academic Editor: Gurkan Tuna

Combining individual sensor decisions can be an effective way for the enhancement of the final decision on sensor fields for intruder detection. This paper proposes a novel methodology to unify the decisions from individual sensors on a sensor field through the (hidden Markov model) HMM and rules. The HMM especially provides a stochastic decision out of the individual sensor decisions on the sensor field; then it is filtered through rule inferences reflecting the knowledge of movement patterns on the level of the sensor field, such as spatial-temporal information and factual information on the movement of objects. This use of contextual knowledge remarkably improves the final decision for the detection. Also, this paper proposes the discretization method to express the state space of sensor field, and the performance evaluation is given by simulations.

1. Introduction

Good quality intruder detection is a critical issue in many applications such as surveillance in military zones and security and protection of mission-critical facilities. The human surveillance has many limitations in the quality of detection because of physical limitations of human beings such as the hardness of a consistent concentration on surveillance and the lack of reliability due to the capriciousness in human emotion. To overcome such limitations, the automated surveillance using sensors can be introduced because sensors do not get exhausted and keep alarmed stably without interruptions by changes like those in human emotion. A weakness in using sensors, however, is the lack of intelligence in detection which drops the detection quality. This implies that how intelligence equips sensor networks is critical for the automated surveillance and detection. On the other hand, the sensitivity of individual sensors of different types is varied depending on their deployed environment, and their detection performances are wide in their values—especially in case of outdoor sensor networks [1, 2]. This makes it a challenge in practice how to deploy sensors in a sensor network so that it may fully reflect their environment states in spite of their variegated changes. For instance, the true and false alarm rates in an intruder detection sensor network change over time according to the states of the given environment, that is, sunny, snowing, raining, and so forth. This fact commands the robustness of outdoor sensor networks from the changeable environment [3].

The unification of individual sensor decisions on a sensor field, a deployed sensor network, could be a viable option to construct a robust sensor network under the uncertain environment. The decision-making step after collecting the knowledge of detections out of individual sensors may provide a chance to determine the extent of utilization of these individual decisions towards the final decision in the sensor network. This unification of small individual decisions has several advantages. First, the final decision in the sensor network mediates the individual sensor decisions, rather than leaving them isolated, that it could provide one conclusive decision. This decision synthesizing out of individual sensor

decisions may definitely help to avoid misjudgments or confusions caused by those individual sensor decisions solely, though this mediation is varied and not easy in complexity [4]. Second, during the process of unifying individual sensor decisions, it is possible to combine and exploit knowledge beyond those of the individual sensors in order to enhance the final decision.

This addition of knowledge should undoubtedly heighten the intelligence of the detection system, because it considers extensive knowledge systematically towards the final decision. In usual practical sensor networks, however, since decisions are made mainly based on the information collected from individual sensors that is mostly scalar sensing data [5, 6], most of holistic knowledge sensor networks can use tends to be ignored in general; this may lead to misunderstandings in their decision makings. For example, usual sensor networks care about signal data, such as frequency, amplitude, and intensity, rather than the whole contextual knowledge on sensor fields: the spatial information (e.g., the relative location with respect to the whole network where a sensor reacts), the temporal information (e.g., the sensing pattern over time), and the factual information (e.g., the sensor type that makes a sensor node alarmed). These kinds of information are indeed precious in that they could contribute to the final decision for intruder detection. Hence, this paper purports to propose a methodology of decision making using the contextual knowledge by unifying individual sensor decisions on a sensor field. Although the proposed methodology in this paper is designed for intruder detection on sensor fields, it could be extended to other purposes on sensor networks.

This study focuses on knowledge obtained from the three kinds of information mentioned above, called patterns on a sensor field. Table 1 shows the considerable patterns needed in the decision making of intruder detection.

Proper movement patterns of an object on a sensor field could be investigated by the HMM with the Viterbi algorithm as shown in Figure 1. In a given observation, an explicit tracking of an object is available under the consideration of a stochastic movement pattern. The movement pattern retrieved by the HMM could be used for the intruder detection on the sensor field. For example, when the expected type of the target object is "Person," the contiguity of the locations of the chased movement becomes a measurement to decide whether the moving object is "Person" or not; that is, when the retrieved trace of the moving object is improper, it could be filtered out by the consideration of the proper trace of the expected type of the target object. This is very helpful for the identification of the intruder. The trace of the detected object is also used for estimating its average speed so that the comparison of the estimated speed with the reported standard average speed of the expected type of the object can give useful information for the intruder identification. Furthermore, the movement pattern of the object can be restricted by the state and shape of the sensor field in many ways. If the corresponding movement patterns are explicitly definable by reflecting the restrictions, that knowledge could be useful for the intruder detection. For example, if the sensor field contains mine fields or cliff edges in a military zone,

Table 1: The considerable patterns for intruder detection.

The considerable patterns for intruder detection
(1) Proper Movement Patterns of an object on a sensor field. (e.g., adjacency between alarmed sensors, proper track of moving object, etc.).
(2) Detected average speed of an object on a sensor field.
(3) Specific movement patterns of an object on a sensor field.
(4) The types of sensor reacting with an object on a sensor field.

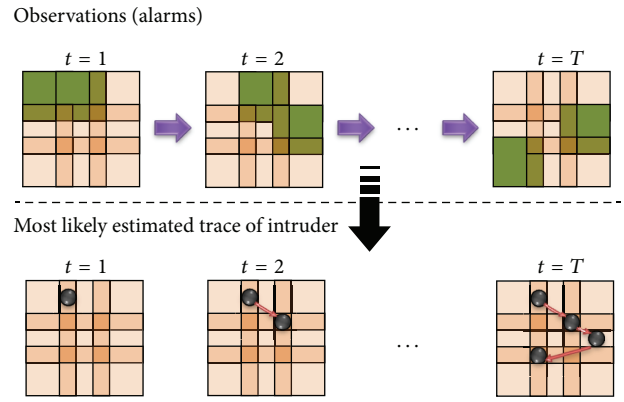


Figure 1: Proper movement patterns retrieved by HMM with Viterbi algorithm.

the movement of the target type "Person" can be hindered by the obstacles, and hence the movement of the intruder is predictable. The estimation of the trace of the moving object thus could be beneficial from such a state of the sensor field to increase the performance of the detection and the identification of the intruder.

The knowledge about the types of sensors that respond to the types of the moving object is very useful for the intruder detection. Figure 2 shows the responding sensor types for the type of each object. According to the figure, it is recognized that the moving object cannot be an animal if a magnetic sensor reports detection because magnetic sensors do not react to animals. These pieces of knowledge are contained in inference rules, by which any detection in a sensor can be inferred to a conclusion about the type of an object.

Different from sensor network systems in signal processing, the main purpose of this paper is to construct an enhanced intruder detection model as a decision model unifying individual sensor decisions in a sensor network by using the HMM and inference rules. This paper also proposes a dynamic discretization method to express the state space for a sensor field. The remainder of the paper is organized as follows. In the next section, this paper continues with the description of the state space representation for the sensor field to which the HMM is applied. The structure of the HMM for intruder detection comes next and suggests the unifying decision-making approach that this paper uses. This part mainly consists of two subparts with an example, the stochastic decision, and the rule based decision. The conclusion follows lastly.

Target	Detection factor	Target location	Responding sensor
Person	Belongings	Ground	Magnetic
	Body heat	Ground	PIR
	Sound	Ground	Acoustic
	Vibration	Ground	Pressure
	Movement	Ground	UWB
Animal	Body heat	Ground	PIR
	Sound	Ground	Acoustic
	Vibration	Ground	Pressure
	Movement	Ground	UWB
Vehicle or tracked vehicle	Sound	Ground	Acoustic
	Vibration	Ground	Pressure
	Mignetic	Ground	Magnetic
	Movement	Ground	UWB
Airplane	Sound	Midair	Acoustic
	Vibration	Midair	Pressure
Bird	Sound	Midair	Acoustic

Figure 2: The corresponding sensors for objects.

2. The State Space Representation of a Sensor Field for HMM

In order to deal with extended observation space O^+ above, one of the options this paper adopts HMM [7–9] because of its strengths in finding out what the class sequence was. For the state space representation of a sensor field using HMM, let this paper introduce HMM with the formula of object function which is defined as

$$\arg \max_{x^{t_1},\ldots,x^{t_n}} P\left(x^{t_1},\ldots,x^{t_n} \mid o^{t_1:t_n}\right), \tag{1}$$

where $n \in \mathbb{N}$; denote the given observation of $t_1,\ldots,t_n$ by $o^{t_1:t_n}$ and the state space of $t_1,\ldots,t_n$ by $x^{t_1},\ldots,x^{t_n}$. The underlying HMM model λ is the triple $\lambda(A,B,\Pi)$ where $\Pi = \{\pi\} = P(X^{t_1} = x_{t_1})$, $A = a_{ij} = P(X^k = x_j \mid X^{k-1} = x_i)$, $B = b_{ij} = P(O^k = o_j \mid X^k = x_i)$, $i,j,k \in \mathbb{N}$, $i,j,k > 0$, and $j > i$. Denote the initial state probability by Π, the transition probability by A, and the emission probability by B. With consideration of the formula (1) and λ, the state space representation should be affordable for the calculation of conditional probability (i.e., emission probability), involve state and observation parts, and be capable of describing time-series state and observation lists.

2.1. State Formulation on HMM for a Sensor Field. Consider the following state space in discrete form:

$$S = \bigcup_{i=1}^{n} R_i, \tag{2}$$

where $R_i = \{X_j \mid \exists j \in \mathbb{N}, X_j \in R_i\}$; denote by S all possible state space, by R_i the subset of S, and by X_j the element state of R_i. Note that the state space S points out all possible state space for a given sensor field which are categorized into the space in the detection of a sensor field and the space out of the detection of a sensor field; the categorized state space can be separated into several subcategories as described in Figure 3.

Then, (2) is rewritten as follows:

$$S = R_1 \cup R_2 \cup R_3 \cup R_4 \cup R_5. \tag{3}$$

Assumption 1. Let this paper make assumption for a sensor field as follows:

(1) the sensor field consist of sensor nodes, a sensor nodes is the set of sensors;

(2) there is no duplicated detection area of sensor node but a sensor does;

(3) the sensors on sensor node are aligned to the same direction;

(4) there is at least one omnisensor having maximum detection distance among sensors in a sensor node.

Based on the above assumption, this paper suggests discretization method for the state space of dynamic sensor field; obviously, the sensor field of MSN (mobile sensor network) cannot but be dynamic [10, 11]. Through Figure 3, with Assumption 1, we notice that the range of sensor detection is the criteria for the distinction of $R_1 \cup R_2$ and $R_3 \cup R_4$. More specifically, R_1 and R_2 are discriminated by the detection distance of sensor; R_3 and R_4 are distinct by geometrical features derived from triangulation and rectangulation. The discretization process follows Figure 4. Basically, the state space of "the area in the range of sensor detection ($R_1 \cup R_2$)" corresponds to the ability of sensor detection; the state space of "out of the range of sensor detection area ($R_3 \cup R_4$)" depends on the deploy locations of sensors. The internal space of a sensor field not including sensor detection area (R_3) especially, is obtained by Delaunay triangulation [12] as subtracting from triangle areas to $R_1 \cup R_2$; the border of

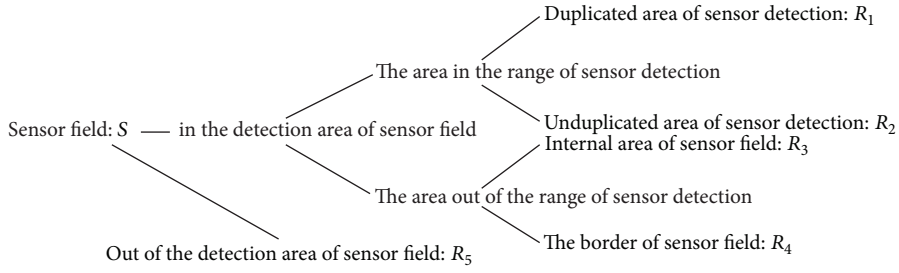


Figure 3: All the possible states on a sensor field.

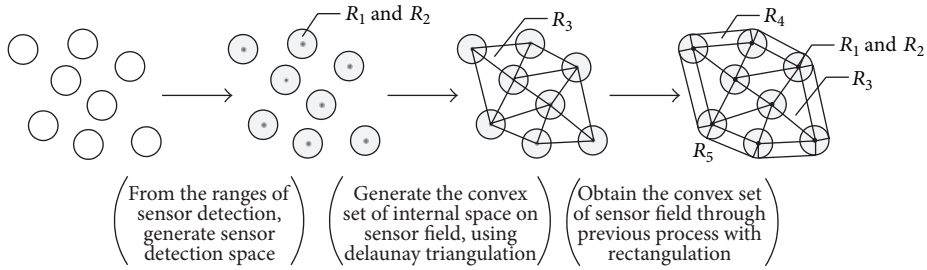


Figure 4: The discretization process for a dynamic sensor field.

a sensor field (R_4) is obtained by rectangulation from deployed location of sensors with subtracting from rectangle areas to $R_1 \cup R_2 \cup R_3$; the remaining area is R_5.

Based on Figures 3 and 4, this paper points out R as follows:

$$R_1 = \left\{X_j \mid X_j \cap S_i \neq \emptyset, \exists i \neq j\right\},$$
$$R_2 = \left\{X_j \mid X_j \cap S_i = \emptyset, \forall i \neq j\right\}, \tag{4}$$

where $X_j = S_j - \bigcup_{i \in \{l \mid S_l \subset S_j\}} S_i$ and S_j denotes the detection area of jth sensor.

As for the areas of R_1 and R_2, these are determined by the properties of sensor detection such as the radius of detection and the angle of detection. This paper introduces an example of that in the following section (Section 2.1.1 is an example for state formulation). The possible state space for undetectable area is defined as follows:

delaunayTriangulation: DT ($\{c_i \mid c_i$ is the centroid of ith sensor node$\}$) $\rightarrow \{T_1, T_2, \ldots, T_u\}$ where, u is the number of generated triangles.

convexSet : $CS = \bigcup_{i=1}^{u} T_i$ define the set of boundary edges of *CS* as *BE*.

boundaryPoint: $BP = \{p_i \mid p_i \in BE$ and p_i is the centroid of ith sensor node$\}$

boundarySegment: $BS = \{(p_i, p_j) \mid p_i, p_j \in BP$ and $\overline{p_i p_j} \subset BE\}$.

circumscribedQuadrilateral: $CQ(BS) \rightarrow \{Q_1, Q_2, \ldots, Q_v\}$, where, v is the number of generated quadrilaterals.

$$R_3 = \left\{X_j \mid X_j = T_j - \bigcup_{X_i \in (R_1 \cup R_2)} X_i\right\}, \quad \text{where } j = 1, \ldots, u,$$
$$R_4 = \left\{X_j \mid X_j = Q_j - \bigcup_{X_i \in (R_1 \cup R_2 \cup R_3)} X_i\right\}, \quad \text{where } j = 1, \ldots, v, \tag{5}$$
$$R_5 = \left\{X_0 \mid X_0 = \left(\bigcup_{X_i \in (R_1 \cup R_2 \cup R_3 \cup R_4)} X_i\right)^c\right\}.$$

Denote the set of center coordinations of sensor nodes by c_i and the set of triangles derived from $c_1, \ldots, c_m$ by $T_1, \ldots, T_u$. Note that m is the numbers of sensor nodes and u is the numbers of triangles which satisfy the objectives of Delanuary Triangulation for given $c_1, \ldots, c_m$. There are three cases of undetectable areas of sensor field which are R_3, R_4, and R_5. $R_3 \cup R_4$ points out the undetectable areas of in sensor field, while R_5 indicates the area of out of sensor field.

Note that the state space representation ensures that it presents all the possible state space. For the R_1 and R_2, those are described precisely by the given parameters—which means that the range of detection area of the sensor is given explicitly. However, it has ambiguity to define "The area out of the range of sensor detection" ($R_3 \cup R_4$) because the shape

of that area is flexible in concordance with the deployment information such as the number, the location, and the type of sensor. Hence, we need to make sure that $R_3 \cup R_4$ covers all the areas of "The area out of the range of sensor detection" in any case. For this purpose, this paper defines those areas separately by triangle and rectangular. From the Delaunay triangulation, we guarantee that the obtained area "Internal area of a sensor field" is convex set and completely covers internal area of a sensor field; after that, by adding circumscribed quadrilateral (R_4), this paper guarantees that the obtained area of $R_1 \cup R_2 \cup R_3 \cup R_4$ addresses all the space of a sensor field consequently and obtained set is convex.

2.1.1. An Example for State Formulation. Let this paper explain our discretization method in the previous section with an example.

As seen in Figure 5, there are two types of sensor node. The first one consists of pressure, acoustic, and magnetic sensor, and the other consists of pressure, PIR, and UWB sensor. Consider R_1 and R_2 as the branch of "The area in the range of sensor detection." Then detection distances of sensors in sensor node are only factors to distinct R_1 "Duplicated area of sensor detection" and R_2 ("Unduplicated area of sensor detection") so that we define R_1 as the detectable area of sensors which have less detection distance than the one that has a maximum detection distance of a sensor node. In order to allocate R_1 and R_2 on sensor field, Let us define S_j (the detection area of sensor j):

$$A_j = \left\{ \left(c_x, c_y\right) \mid \left(c_x - m_j\right)^2 + \left(c_y - n_j\right)^2 < r_j^2 \right\}, \text{ where}$$

$$c_x, m_j, c_y, n_j, r_j \in \mathbb{R}, \quad j \in \mathbb{N},$$

B_j :

$$\begin{cases} \left\{ \left(c_x, c_y\right) \mid \text{ArcTangent}\left(\dfrac{c_y - n_j}{c_x - m_j}\right) \in DetectionAngle_j \right\}, \\ \qquad \text{where } c_x - m_j \geq 0 \text{ and } c_y - n_j \geq 0 \\ \left\{ \left(c_x, c_y\right) \mid \text{ArcTangent}\left(\dfrac{c_x - m_j}{c_y - n_j}\right) + 90^\circ \in DetectionAngle_j \right\}, \\ \qquad \text{where } c_x - m_j < 0 \text{ and } c_y - n_j > 0 \\ \left\{ \left(c_x, c_y\right) \mid \text{ArcTangent}\left(\dfrac{c_y - n_j}{c_x - m_j}\right) + 180^\circ \in DetectionAngle_j \right\}, \\ \qquad \text{where } c_x - m_j > 0 \text{ and } c_y - n_j > 0 \\ \left\{ \left(c_x, c_y\right) \mid \text{ArcTangent}\left(\dfrac{c_x - m_j}{c_y - n_j}\right) + 360^\circ \in DetectionAngle_j \right\}, \\ \qquad \text{where } c_x - m_j > 0 \text{ and } c_y - n_j < 0. \end{cases} \tag{6}$$

Then, we define S_j as

$$S_j = \left\{ S_j \mid S_j = A_j \cap B_j \right\}. \tag{7}$$

Denote x-coordinate and y-coordinate on a sensor field by c_x and c_y. m_j, n_j, and r_j indicate the center coordinates and the detection distance of sensor $sensor_j$, respectively, where $sensor_j$ is one of the sensors of sensor node and $r_j < r_{\max}$ ($r_{\max}$ is the maximum detection distance on sensor node). In addition $DetectionAngle_j$ points out that $\text{minAngle}_j \leq$ the detection angle of $sensor_j \leq \text{maxAngle}_j$. Notice that there are two types of sensor node, sector type and circle type. Basically, (7) is derived from the sector type of a sensor node. However, the circle type of sensor node has $0^\circ \leq DetectionAngle_j \leq 360^\circ$ such that (7) is held in case of circle type of sensor node as well. As for the detection area of R_1 and R_2, numbers of sensors, $n-1$ duplicated areas are there. R_1 and R_2 are described as

$$R_1 = \left\{ X_j \mid X_j = S_j - S_i \right\}, \tag{8}$$

where $r_{sensor_j} < r_{sensorNode_k}$, $r_{sensor_j} > r_{sensor_i}$ and $sensor_j$, $sensor_i \in sensorNode_k$.

$$R_2 = \left\{ X_j \mid X_j = S_j - S_i \right\}, \tag{9}$$

where $r_{sensor_j} = r_{sensorNode_k}$, $r_{sensor_j} > r_{sensor_i}$ and $sensor_j$, $sensor_i \in sensorNode_k$.

Denote the detection radius of a sensor and a sensor node by r_{sensor} and $r_{sensorNode}$ repectively. Note that $sensorNode_k = \{sensor_1, \ldots, sensor_n\}$. If $sensor_j$ has the same detection radius of its sensor node, then the obtained X_j binds to R_2, otherwise R_1.

As mentioned in Section 2.1, on account of the flexibility of undetectable area in a sensor field, this paper describes that area as triangles and circumscribed quadrilateral to ensure that our approach represents all the possible state space on a sensor field in any shape. The way to achieve the goal is fairly simple in principle. From the Delaunay triangulation, we obtain guaranteed convex set for the internal space of undetectable area on a sensor field; after that as using line segments of triangles for circumscribed quadrilateral, facilely we generate optimal state space, and the meaning of optimal state space here is that all the area is evenly separated as possible as a given parameter. The acquisition of the internal space of undetectable area is achieved by

$$\begin{aligned} T_j = \Big\{ \left(c_x, c_y\right) \mid & f_1\left(p_{1_x}^{tri_k}, p_{1_y}^{tri_k}\right) f_1\left(c_x, c_y\right) \geq 0, \\ & f_2\left(p_{2_x}^{tri_k}, p_{2_y}^{tri_k}\right) f_2\left(c_x, c_y\right) \geq 0, \\ & f_3\left(p_{3_x}^{tri_k}, p_{3_y}^{tri_k}\right) f_3\left(c_x, c_y\right) \geq 0, \\ & p^{tri_k} \in TrianglesApexes \Big\}, \text{ where} \end{aligned}$$

$$p, c \in \mathbb{R}^2, \quad k \in \mathbb{N}, \tag{10}$$

$$\begin{aligned} f_1(x, y) = & \left(x - p_{2_x}^{tri_k}\right)\left(p_{2_y}^{tri_k} - p_{3_y}^{tri_k}\right) \\ & - \left(y - p_{2_y}^{tri_k}\right)\left(p_{2_x}^{tri_k} - p_{3_x}^{tri_k}\right), \end{aligned} \tag{11}$$

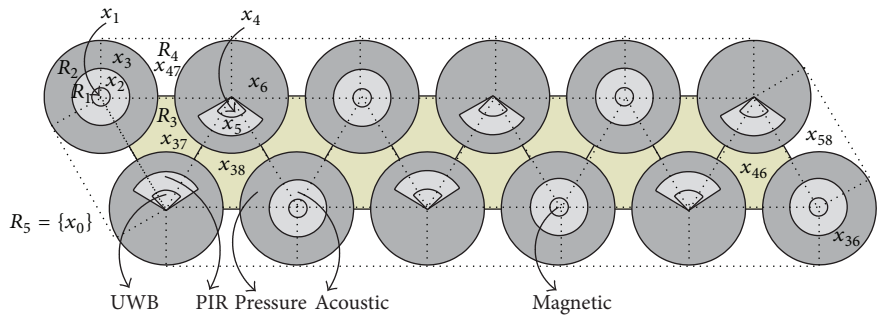


Figure 5: State space representation—an example a sensor field.

$$f_2(x,y) = \left(x - p_{3_x}^{tri_k}\right)\left(p_{3_y}^{tri_k} - p_{1_y}^{tri_k}\right) - \left(y - p_{3_y}^{tri_k}\right)\left(p_{3_x}^{tri_k} - p_{1_x}^{tri_k}\right), \tag{12}$$

$$f_3(x,y) = \left(x - p_{1_x}^{tri_k}\right)\left(p_{1_y}^{tri_k} - p_{2_y}^{tri_k}\right) - \left(y - p_{1_y}^{tri_k}\right)\left(p_{1_x}^{tri_k} - p_{2_x}^{tri_k}\right). \tag{13}$$

Denote the apexes of the triangle by p^{tri_k}. The function $f(p_x, p_y)f(c_x, c_y)$ evaluates whether or not the points p and c are located in the same region (i.e., $f(p_x, p_y)f(c_x, c_y) \geq 0$) for a given linear equation f_1, f_2, and f_3, respectively. Note that by (10)~(13) T_j indicates the interior area of the given triangle. Hence, the undetectable area inside a sensor field R_3 is described as

$$R_3 = \left\{X_j \mid X_j = T_j - \left(X_a \cup X_b\right)\right\}, \text{ where } X_a \subseteq R_1, \quad X_b \subseteq R_2. \tag{14}$$

Related to the given parameters of (18), this paper adopts Delaunay triangulation to generate undetectable areas shaped in triangles. Being the apexes of triangle, the center coordinates of sensors are applied for the triangulation. The procedure of Delaunay triangulation is as follows: (1) Define the center coordinate of sensors $P = \{p_1, \ldots, p_n\}$. (2) Define p_1 as an uppermost y-point of elements that is the maximum y-value of P. (3) Generate two arbitrary points p_{left} and p_{right} for the triangle composed of p_1, p_{left}, and p_{right} to cover all coordinates in P. (4) Index the rest of elements in P from p_2 to p_n. So the number of points used for Delaunay triangulation is $n + 2$. (5) Do $triangulation(p_i)$ in order. (5.1) If the generated triangle (e.g., the triangle of p_1, p_{left}, and p_{right}) contains p_i, then make three triangles by drawing three lines $\overline{p_i p_1}$, $\overline{p_i p_{\text{left}}}$, and $\overline{p_i p_{\text{right}}}$. (5.2) If p_i exists on $\overline{p_2 p_3}$ of the triangle by p_1, p_2, and p_3, then make two triangles by drawing line $\overline{p_i p_1}$. (5.3) If p_i is located on the line $\overline{p_2 p_3}$ that two triangles share (e.g., the triangles of p_1, p_2, p_3 and p_2, p_3, p_4), then make four triangles by drawing two lines $\overline{p_i p_1}$, $\overline{p_i p_4}$ with the legal edge condition. (6) Figure 6(a) points out the illegal edge condition in which encountering angles (a, b) is bigger than 180°. With the illegal edge condition, $\overline{p_1 p_2}$ has to be removed to generate new edge $\overline{p_3 p_4}$ for the legal edge condition as shown in Figure 6(b). (7) Remove arbitrary generated points p_{left} and p_{right}.

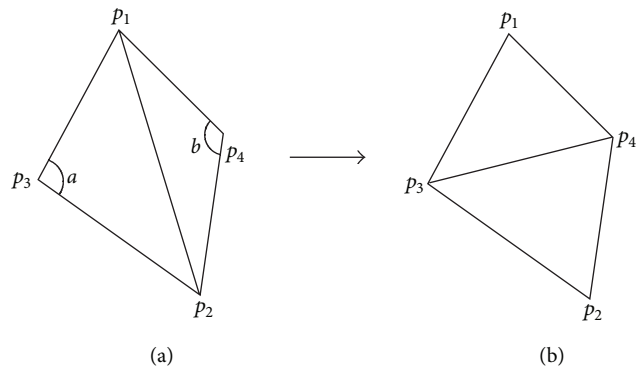


Figure 6: Edge flip condition in Delaunay triangulation.

For the border of a sensor field, consider the following equation:

$$Q_j = \left\{\left(c_x, c_y\right) \mid c_x \in RetangleArea\left(d, e, p, q\right), \; c_y \in RetangleArea\left(d, e, p, q\right)\right\}, \text{ where } d, e, p, q \in \mathbb{R}^2, \quad j \in \mathbb{N}, \tag{15}$$

$$R_4 = \left\{X_j \mid X_j = Q_j - \left(X_a \cup X_b \cup X_c\right)\right\}, \text{ where } X_a \subseteq R_1, \quad X_b \subseteq R_2, \quad X_c \subseteq R_3. \tag{16}$$

Note that the equation to calculate the area of rectangular is similar as (10) in principle by using linear equation. Hence, we skip explaining the specific function $RetangleArea(d, e, p, q)$. Denote by d, e, p, and q the coordinate of rectangular by X_a, X_b, and X_c the area involved in R_1, R_2, and R_3. Note that $\overline{pq}$ is line segment of circumscribed quadrilateral and of triangle as well; $\overline{pq}$ is acquired by triangulation which means that p, q are the center coordinates of sensor. Basically, d and e are easily derived from given p, q by Pythagorean theorem. From the reason of that p, q are the center coordinates of circle or sector, the derived rectangles are always circumscribed quadrilateral such that

in any case pythagorean theorem is valid in the calculation to obtain d, e. Let us explain the calculation for d, e with the definition of which $radius(x)$ is the function retrieving the radius of the sensor having the center coordinate x. In the 2-D coordination, Euclidian distance is

$$distance\,(p,q) = \sqrt{\sum_{i=1}^{2}(p_i - q_i)^2}, \text{ where}$$
$$p_1 = p_x, \quad p_2 = p_y. \tag{17}$$

From (17), $\overline{dq}$ is $\sqrt{distance(p,q)^2 + radius(p)^2}$, hence and $\overline{dq}$ is given; after that

$$\overline{dq} = \sqrt{(d_x - q_x)^2 + (d_y - q_y)^2},$$
$$radius\,(p) = \sqrt{(d_x - p_x)^2 + (d_y - p_y)^2}. \tag{18}$$

We have two variables (d_x, d_y) and (18) such that d is calculated. Through the same procedures of (17) and (18), the circumscribed quadrilaterals based on line segments of triangles are generated. As a consequence, from (16) we generate "the border of a sensor field" state. R_5 is simply defined as $X_0 = \{\emptyset\}$ because of "out of the area on a sensor field."

2.1.2. The Representation of Observations of the Example. According to the sensing factors, different types of sensors determine its decision through signals that react with thresholds [13]. In this point, the decisions from these sensors are regarded as the decisions reflecting the features of each type of sensors. In our point of view, this is meaningful in terms of sensor network. Hence, this paper exploits the decisions of each sensor in a sensor field as the observation of HMM.

The observation in HMM can either be discrete or continuous [14, 15]. This paper applies discrete observation which represents "Detect" (Active) and "No respond." The possible discrete observation space in our case is 2^n (n is the number of sensors in a sensor field). In order to deal with all possible observation space, this paper represents the observation as "$ob_1, \ldots, ob_{2^n}$" with the interpretation of n digits code. For example, if there are four sensors in a sensor field, we interpret the observation as "0 | 1, 0 | 1, 0 | 1, 0 | 1" (0: No respond, 1: Detect). In addition, for the calculation of conditional probability, the state representation of HMM is interpreted by n digit code in the same way as well. The emission probabilities are calculated by means of those representations, and the transition probabilities are obtained with first order Markov assumptions [9].

Figure 7 depicts an example of the representation of states and observations for Figure 5. There are 36 sensors on a sensor field. The whole possible numbers of observations are $2^{36} = 68719476136$. We define the symbol ob_0 to indicate that no alarm is reported and ob_3 for what sensor 1 and sensor 3 make an alarm. The indexes of symbol for an observation are obtained by transforming the binary to the decimal notation. In case of the representation of states for a sensor field, we adopt the suggested discretization approach at Section 2.1. According to the deploy information of sensors and its detection ranges, the symbol indicating a state is assigned with the proper range of a region. And the proper range assigned for that is interpreted by the detection ranges of sensors. For example, x_1 could be depicted by the ranges of sensor 1, sensor 1, and sensor 3. The one advantage of describing states by the detectable ranges in the manner of sensor is that it enables the calculation of emission probability through the given observations and states.

3. The Structure of HMM for Intruder Detection

The structure of HMM could be represented by $\lambda(A, B, \Pi)$. Denote initial probability by Π, transition probability by A, and emission probability by B. Define that initial probability $\Pi = \{\pi\} = P(X^{t_1} = x_{t_1})$, $A = a_{ij} = P(X^k = x_j \mid X^{k-1} = x_i)$ and $B = b_{ij} = P(O^k = o_j \mid X^k = x_i)$, where $i, j, k \in \mathbb{N}$ and $j > i$. Note that an emission probability is calculated by a given observation with the performance of sensor which is $P_{detection\,success} = P(alarm \mid Target)$, $P_{false\,detection} = P(alarm \mid NonTarget)$, $P_{detection\,failure} = P(noResponse \mid Target)$, and $P_{noResponse} = P(noResponse \mid NonTarget)$. However, in order to obtain initial and transition probability of HMM, the actual movement of an object is required. Hence, this paper applies Gaussian mobility model [15] to gain proper movements of objects. Gaussian mobility model is a well-known model to generate reliable movement of object by manipulating the parameters. The following indicates movement equations of the model.

The equation for speed and direction calculation

$$s_n = \alpha s_{n-1} + (1-\alpha)\,\overline{s} + \sqrt{(1-\alpha^2)}\sigma_s,$$
$$d_n = \alpha d_{n-1} + (1-\alpha)\,\overline{d} + \sqrt{(1-\alpha^2)}\sigma_d. \tag{19}$$

The equation for coordinate generation (2 Dimention)

$$x_n = x_{n-1} + s_{n-1}\cos d_{n-1},$$
$$y_n = y_{n-1} + s_{n-1}\sin d_{n-1}. \tag{20}$$

Denote the speed and direction at time n by s_n and d_n, respectively, the tuning parameter for adjustment of randomness where $0 \le \alpha \le 1$ by α, the mean values of speed and direction of object by $\overline{s}$ and $\overline{d}$, and the standard of deviation of speed and direction by σ_s and σ_d. Note that according to the tune of α with $\overline{s}$, $\overline{d}$, σ_s, and σ_d, the movement of a desired object is modeled.

The movement model of "Person," "Animal (Deer)," and "Tracked Vehicle" has been approximated with given parameters as follows: Person: (4.32, 0.3888, 10.0, 0.75) for $\overline{s}$ (km), σ_s (km), σ_d (°), and α; Animal: (29.00, 2.6000, 20.0, 0.50); Tracked Vehicle: (40.0, 3.6000, 10.0, 0.95). Through the generated movement by random sampling from the conducted Gaussian mobility model, the state matrix of HMM is calculated (Figure 8). In our case, we generated 200,000 samples for

The state space representation for a given sensor field

States for hidden Markov model

Symbolic representation	Binary representation (interpretation)				
Symbol	Sensor index				
	s_{36}	...	s_3	s_2	s_1
x_1	0	0 ··· 0	1	1	1
x_2		0 ··· 0	1	1	
x_3	0	0 ··· 0	1	0	0
⋮			⋮		
x_{58}	0	0 ··· 0	0	0	0

Observations for hidden Markov model

Symbolic representation	Binary representation (interpretation)				
Symbol	Sensor index				
	s_{36}	...	s_3	s_2	s_1
ob_0	0	0 ··· 0	0	0	0
ob_1	0	0 ··· 0	0	0	1
ob_2	0	0 ··· 0	0	1	0
⋮			⋮		
$ob_{68719476736}$	1	1 ··· 1	1	1	1

Figure 7: The state space representation—an example.

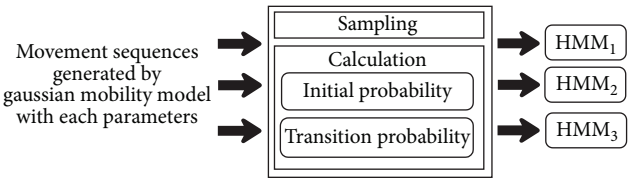


Figure 8: Initial and transition probability generation.

each movement (Person, Animal, and Tracked Vehicle) and then by calculating the frequencies of state changes, the state matrix of HMM is conducted with the following equations.

Inital probability:

$$\Pi_i = \frac{count\left(x_{0i}\right)}{count\left(\sum_{k=1}^{n} x_{0k}\right)}, \tag{21}$$

where $i, j, k \in \mathbb{N}$, and n is the total numbers of states.

Transition probability:

$$x_{ij} = \frac{count\left(x_{ij}\right)}{count\left(\sum_{k=1}^{n} x_{ik}\right)}. \tag{22}$$

Basically, the model λ requires transition matrix, initial probability, and emission matrix. The way of obtaining those factors of HMM is that for the case of transition matrix, random sampling is used as we mentioned above. Secondly, initial probability is obtained by the state change from X_0 to X_n which means the entering probability to a sensor field. It is reasonable because we define $X_0 \in R_5$ (out of the sensor field) at Section 2. Thirdly, emission matrix is dynamically calculated with the interpretation of Section 2.1.2. Assume that there are sensors having the same performance $P_{detection\ success} = 0.8$, $P_{detection\ failure} = 0.1$, $P_{false\ detection} = 0.1$, and $P_{noResponse} = 0.5$ for an example of emission probability calculation. The calculation for that is depicted in Figure 9. The representation shown in Figure 9 is derived from the sensor field having 1 sensor node comprised of three sensors. From this, obtained emission probability for $P(x_2 \mid ob_2)$ is "$0.8 \times 0.1 \times 0.1 = 0.008$." In the details, x_2 points out the state detectable by sensor 3 and sensor 2; in the mean time ob_2 indicates the observation of alarmed by sensor 3 and sensor 1. Hence, the emission probability is calculated with the false alarm rate of sensor 1, the rate of detection failure of sensor 2, and the rate of detection success of sensor 3.

4. Decision-Making Methodology Using HMM with Rules

As a first phase of our methodology for combining decisions, this paper provides the decision-making methodology based on stochastic model by adopting the suggested discretization method to HMM. The motivation of our methodology is quite naïve. Simply, there are distinct advantages between stochastic and cause-and-effect deterministic model. With an assumption of which all the events have probabilities to be happened, stochastic model is more explainable than cause-and-effect model for a given phenomenon. However, at some points, the cause-and-effect model could complement stochastic model for the enhanced decision. For example, the knowledge which has a difficulty in the representation of stochastic model can be easily extended to the rules for better decisions (experienced knowledge, statistical values, common sense, and others). Figure 10 indicates the architecture for our approach to combine stochastic (HMM) and rule-based decision within complexity of $O(N^2T)$ and $O(RACT)$, respectively. Denote evaluated costs by $O(T)$, numbers of states of HMM by N, numbers of rules by R, the number of assertions A, and the approximate number of conditions per rule by C.

As an approach for unifying decisions of sensors on a sensor field, this paper adopts HMM and rules. The first part of synthesizing decisions is achieved by HMM and then as a second part, this paper adopts rule-based decision. As shown in Figure 10, sensors in the sensor field have made sensor decisions by the input signals such as frequencies, amplitudes, decibels, and others. Then the decisions (*Detect* | *NoResponse*) become the input parameters to HMM as the observation O of a moving object. In our methodology, the unified decision by HMM is achieved by means of judging the acceptability of model $\lambda_{1:n}$ (i.e., $\mathrm{HMM}_1, \mathrm{HMM}_2, \ldots, \mathrm{HMM}_n$) by thresholds θ_1, θ_2, and θ_3 and selecting one of the judged model with

$$\arg\max_{k=1,\ldots,n} \left(\widehat{\delta}_k\right), \quad \text{where } n \in \mathbb{N},\ \widehat{\delta}_k \in \mathbb{R}. \tag{23}$$

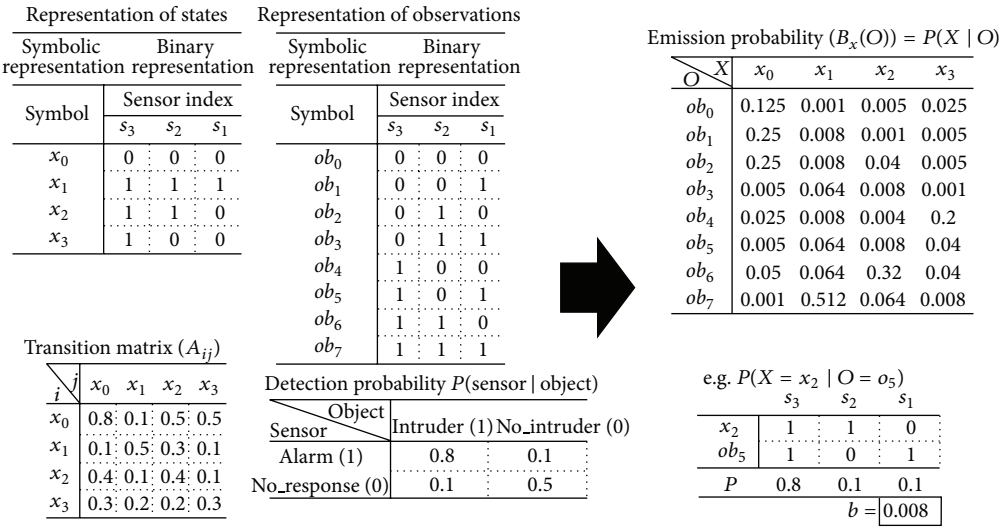


Figure 9: An example of calculation of emission probability on a sensor field.

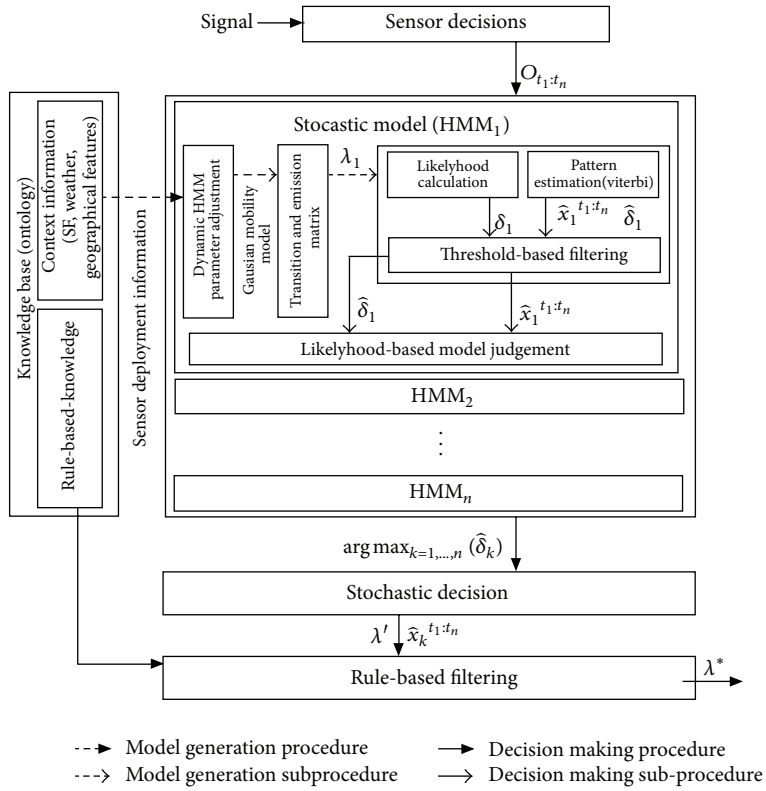


Figure 10: The architecture to combine of stochastic (HMM) and rule-based decision.

Denote by k the number of HMM model and by $\hat{\delta}_k$ the probability of the most likely state sequence for given observations on kth model (e.g., Person, Animal, or Tracked Vehicle). Note that (23) selects one model whatever it is, according to the probability of the most likely state sequence. Then the identification of moving object is performed by the selected model. Hence, it is necessary to measure the acceptability of HMM model and filter the stochastic decision, the decision which has been made by HMM. The acceptability of HMM model is judged with thresholds θ_1, θ_2, and θ_3 and then the stochastic decision is filtered by rules.

4.1. Stochastic Decision-Making Model with HMM on a Sensor Field. The procedure of stochastic decision is depicted by Figure 11. The procedure of stochastic decision making starts with calculating likelihood probability for the sequences of observations and obtaining maximum likelihood probability

for the estimated movement patterns by Viterbi algorithm. Likelihood probability δ indicates how given observations are explainable by the established model (HMM), while $\hat{\delta}$ points out how estimated patterns occur by given observations on the model. Both of them would be criteria for choosing a model for stochastic decision. However, this paper mainly focuses on the movement of an object. We adopt $\arg\max_{k=1,\ldots,n}(\hat{\delta}_k)$ as a measure of HMM decision.

The δ in Figure 11 is calculated by the observations in time $t_{1:n}$ as follows:

$$\begin{aligned} P(O \mid \lambda) &= \sum_X P(O, X \mid \lambda) \\ &= \sum_X P(O \mid X, \lambda) P(X \mid \lambda) \\ &= \sum_X \pi a_{x^{t_1} x^{t_2}} b_{x^{t_2}}\left(o_{t_2}\right) b_{x^{t_3}}\left(o_{t_3}\right) \\ &\quad \times b_{x^{t_4}}\left(o_{t_4}\right) \cdots b_{x^{t_n}}\left(o_{t_n}\right) \\ &\quad \times a_{x^{t_2} x^{t_3}} a_{x^{t_3} x^{t_4}} \cdots a_{x^{t_{n-1}} x^{t_n}} \\ &= \sum_X \pi a_{x^{t_1} x^{t_2}} \prod_{i=2}^{n} b_{x^{t_i}}\left(o_{t_i}\right) a_{x^{t_{i-1}} x^{t_i}}, \end{aligned} \tag{24}$$

where $X \in AllPossibleStatesSequences$ in time $t_1, \ldots, t_n$. Denote by O observations, by X states, and by π initial probability. Note that the function $b_{x^{t_n}}(o_{t_n})$ indicates the emission probability of observation for the state and $a_{x^{t_{n-1}} x^{t_n}}$ points out transition probability. In our case, observations are the detection decisions of sensors on a sensor field and states indicate the area of a sensor field.

This paper applies Viterbi algorithm [16] to find the most likely underlying explanation of the sequence of observation $\widehat{X}$ and the probability of estimated state sequence $\hat{\delta}$ with the following object functions:

$$\hat{x}^{1,\ldots,t} = \arg\max_{x^1,\ldots,x^t} P\left(x^1, \ldots, x^t \mid o^{1:t}\right). \tag{25}$$

Note that $\delta_t(x_j) = \max_{x^1,\ldots,x^t} P(x^{1,\ldots,t-1}, o^{1:t}, x^t = x_j) = \max_{x^{t-1}} \delta_{t-1}(x_i) a_{x_i x_j} b_{x_j}(o^t)$ so that estimated movement pattern could be obtained from $\hat{x}^t = \max_{x^t} \delta_t(x_j)$. And the probability of estimated sequence called maximum likelihood could be calculated with

$$\hat{\delta}^t = P\left(\hat{x}^{1,\ldots,t}, o^{1:t} \mid \lambda\right). \tag{26}$$

Basically, stochastic decision is established with

$$\begin{aligned} &\acute{\lambda} = \lambda_k = \arg\max_{k=1,\ldots,n}\left(\hat{\delta}_k^t\right), \text{ where} \\ &\acute{\lambda} \in \mathrm{HMM}_n. \end{aligned} \tag{27}$$

To ensure that $\acute{\lambda}$ is a proper decision because the function of arg max anyway chooses one of the HMMs (Figure 12), this paper adopts three thresholds θ_1, θ_2, and θ_3. Those thresholds are the filtering condition of, $\hat{\delta}$, and $P(x_0^{1,\ldots,t} \mid o^{1:t}, \lambda)$. Each of probabilities of those indicates the possibilities of all possible state sequences (δ), estimated state sequence ($\hat{\delta}$), and *Nothing State* ($P(x_0^{1,\ldots,t} \mid o^{1:t}, \lambda)$).

We define θ_1 by δ, θ_2 by $\hat{\delta}$, and θ_3 by $P(x_0^{1,\ldots,t} \mid o^{1:t}, \lambda)$ with the following equations:

$$\begin{aligned} \theta_1 &= \min_{i=1,\ldots,n} \delta_i = \min_{i=1,\ldots,n}\left(P_i\left(o^{1:t} \mid \lambda\right)\right), \\ \theta_2 &= \min_{i=1,\ldots,n} \hat{\delta}_i = \min_{i=1,\ldots,n}\left(P_i\left(x^{1:t} \mid o^{1:t}, \lambda\right)\right), \\ \theta_3 &= \max_{i=1,\ldots,n}\left(P_i\left(x_0 \mid o^{1:t}, \lambda\right)\right). \end{aligned} \tag{28}$$

θ_1 and θ_2 are configured by minimum δ, and $\hat{\delta}$ is obtained by model simulation with the correct movement ($n = 100$); θ_3 is configured by maximum value of $P(x_0^{1,\ldots,t} \mid o^{1:t}, \lambda)$ from the simulation. Using threshold-based filtering, stochastic decision is finalized with

$$\begin{aligned} \acute{\lambda} &= \lambda_k = \arg\max_{k=1,\ldots,n}\left(\hat{\delta}_k^t\right) \\ &= \begin{cases} \lambda_k, & f_1(\delta), f_2\left(\hat{\delta}\right) \text{and} f_3\left(P\left(x_0 \mid o^{1:t}, \lambda\right)\right) \text{ are 1,} \\ \text{Nothing,} & f_1(\delta), f_2\left(\hat{\delta}\right) \text{or} f_3\left(P\left(x_0 \mid o^{1:t}, \lambda\right)\right) \text{ are 0.} \end{cases} \end{aligned} \tag{29}$$

Note that

$$\begin{aligned} f_1(\delta) &= \begin{cases} 0, & \delta < \theta_1, \\ 1, & \delta \geq \theta_1, \end{cases} \\ f_2\left(\hat{\delta}\right) &= \begin{cases} 0, & \hat{\delta} < \theta_2, \\ 1, & \hat{\delta} \geq \theta_2, \end{cases} \\ f_3\left(P\left(x_0 \mid o^{1:t}, \lambda\right)\right) &= \begin{cases} 1, & P\left(x_0 \mid o^{1:t}, \lambda\right) \leq \theta_3, \\ 0, & P\left(x_0 \mid o^{1:t}, \lambda\right) > \theta_3. \end{cases} \end{aligned} \tag{30}$$

4.2. Rule-Based Filtering to Enhance the Stochastic Decision. The simplest idea of rule-based filtering is that a tactic and implicit knowledge could be conducted as hypothesizes for the decision making. For example, the knowledge such as "the average speed of a moving object," "stop pattern (specific pattern of an object)," and "corresponding types of sensors to an object" could be utilized as hypothesizes for intruder detection. This paper intends to represent those hypothesizes as rules for filtering HMM decision to enhance intruder detection. Hence, as shown in Figure 13, the final decision of our approach is defined as

Final Decision $\widehat{\lambda}$:
$\acute{\lambda}$ has inTheRangeOfAverageSpeed $\wedge$ $\acute{\lambda}$ has regularPath $\wedge$ $\acute{\lambda}$ is the ProperType for Sensor Responding $\rightarrow$ intuderType is $\acute{\lambda}$.

The rule for final decision $\widehat{\lambda}$ could be depicted by the rule tree (Figure 14). $\widehat{\lambda}$ considers at first the specific patterns of given objects. In our case, we conduct the rules for "Person,"

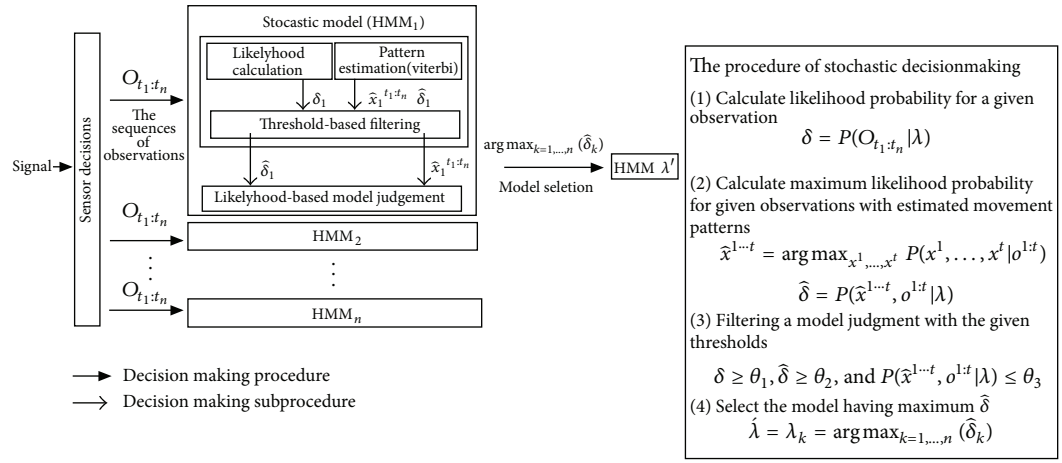


Figure 11: The procedure of stochastic decision using HMM.

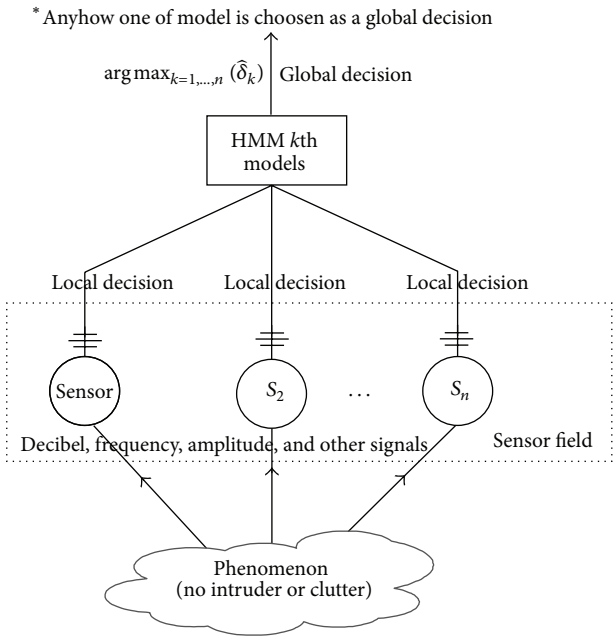


Figure 12: The measurements of filtering required.

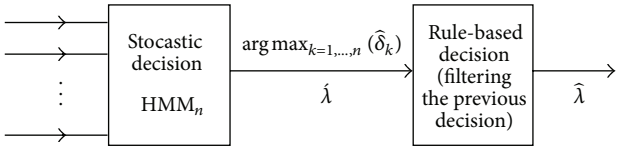


Figure 13: Decision-making using HMM with rules.

"Animal," and "Tracked Vehicle". Secondly, as filtering criteria, this paper investigates the estimated speed from retrieved pattern using Viterbi algorithm, through the comparison of an estimated speed of the object and a statistic average speed of the object. At last, the knowledge of corresponding types of sensors is employed for final decision $\widehat{\lambda}$.

The rules for $\widehat{\lambda}$ are conducted through SWRL (Semantic Web Rule Language), and the derived rules from Figure 14 are as follows:

Final Decision Rules

$EstimatedTrajectory(?x) \wedge$
$hasEstimatedSpeed(?x, ?z) \wedge$
$hasHMMEstimatedIntruderType(?x, ?y) \wedge$
$isRegularPath(?x, ?g_1) \wedge$
$isInAverageSpeed(?z, ?g_2) \wedge$
$ProperIntruderTypes(?y) \wedge$
$swrlb: notEqual(?y, Animal) \wedge\ ?g_1 \wedge\ ?g_2 \wedge ?g_3 \rightarrow$
$intruderType(?y)$

$EstimatedTrajectory(?x) \wedge$
$hasEstimatedSpeed(?x, ?z) \wedge$
$hasHMMEstimatedIntruderType(?x, ?y) \wedge$
$isRegularPath(?x, ?g_1) \wedge$
$isInAverageSpeed(?z, ?g_2) \wedge$
$ProperIntruderTypes(?y) \wedge$
$swrlb: notEqual(?y, Animal) \wedge$
$swrlb: booleanNot(?g_1, ?g_2) \wedge$
$swrlb: booleanNot(?g_2, ?g_3) \wedge$
$swrlb: booleanNot(?g_3, ?g_1) \rightarrow$
$intruderType(Nothing)$

$EstimatedTrajectory(?x) \wedge$
$hasEstimatedSpeed(?x, ?z) \wedge$
$hasHMMEstimatedIntruderType(?x, ?y) \wedge$
$isRegularPath(?x, ?g_1) \wedge$
$isInAverageSpeed(?z, ?g_2) \wedge$
$ProperIntruderTypes(?y) \wedge$
$swrlb: notEqual(?y, Animal) \wedge$
$swrlb: equal(?g_1, False) \wedge$
$swrlb: equal(?g_2, False) \wedge$

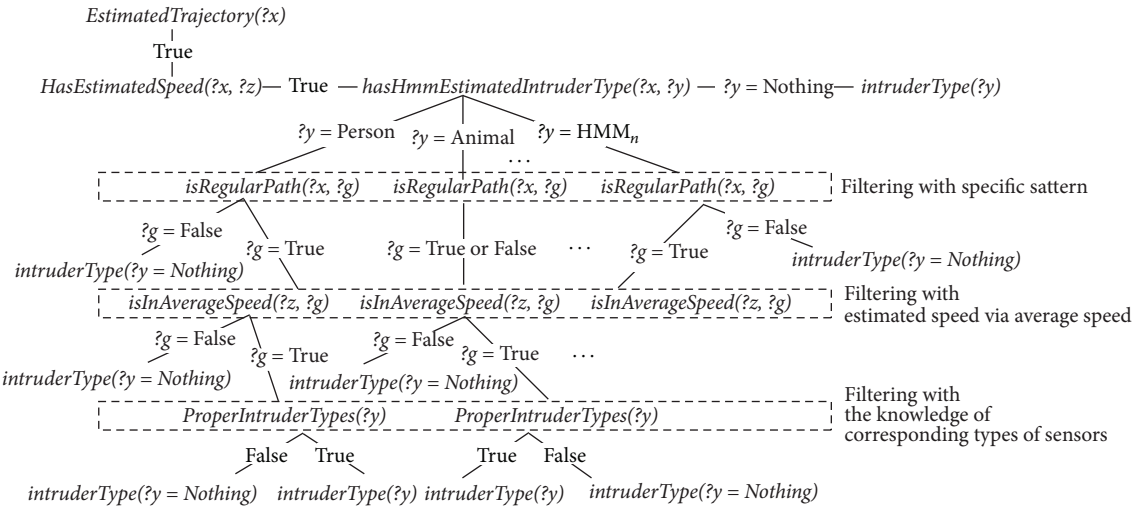


Figure 14: The rule tree for final decision.

$swrlb{:}\ equal(?g_3, False) \rightarrow intruderType(Nothing)$

$$EstimatedTrajectory(?x) \wedge hasEstimatedSpeed(?x, ?z) \wedge hasHMMEstimatedIntruderType(?x, ?y) \wedge isInAverageSpeed(?z, ?g_1) \wedge ProperIntruderTypes(?y) \wedge swrlb{:}\ notEqual(?y, Animal) \wedge ?g_1 \wedge ?g_2 \rightarrow intruderType(?y).$$

$\widehat{\lambda}$ has three criteria which are *isRegularPath*, *isInAverageSpeed*, and *ProperIntruderTypes*. *isRegularPath* represents the specific pattern of an object. For example, assume that there are mine area or cliff edge on a sensor field; in this case we suppose that the allowed area to the a sensor field are restricted for "Person" which means if nonallowed areas are traced by HMM, then we can decide explicitly that the type of intruder is not "Person" as follows:

The restrict condition of an intruder movement (=Person)

$$EstimatedTrajectory(?x) \wedge EstimatedSeq(?y) \wedge EstimatedSeq(?y_2) \wedge hasHMMEstimatedIntruderType(?x, \text{"Person"}) \wedge hasSequence(?x, ?y) \wedge hasSequenceType(?y, Inital\_Penetration\_Pathway) \wedge hasHiddenState(?y, X_3) \wedge hasSequence(?x, ?y_2) \wedge hasSequenceType(?y_2, Penetration\_Pathway) \wedge swrlb{:}booleanNot(hasHiddenState(?y_2, X_7), True) \wedge swrlb{:}booleanNot(hasHiddenState(?y_2, X_8), True) \wedge swrlb{:}\ booleanNot(hasHiddenState(?y_2, X_9), True) \rightarrow isRegularPath(?x, \text{"True"}).$$

The rule of *isInAverageSpeed* is referred to by the parameters of Gaussian mobility model for each objects. The purpose of adopting rules is filtering the misjudgment of a stochastic decision. In this point, we establish the deviation of the average speed to cover the wide range of speed, because of two reasons. The first one is that our approach applies discrete variable to represent the state space of HMM and that makes it hard to measure precise estimated speed. In the second one, basically we assume that stochastic decision has an admissible performance for detecting intruder. Hence, we intend to filter ridiculous estimated speed only by rules. In other word, we expect that the observations from earthquake, airplane, and any other unclassified objects are filtered during the sarcastic decision procedure. However, if it is not filtered by HMM, then rule-based filtering would be operated. The following rule is conducted for the case in which moving object is person with reflecting above argument:

$$EstimatedTrajectory(?x) \wedge hasHMMEstimatedIntruderType(?x, \text{"Person"}) \wedge hasEstimatedAverageSpeed(?x, ?y) \wedge swrlb{:}\ lessThanOrEqual(?y, 8.32) \wedge swrlb{:}\ greaterThanOrEqual(?y, 0.32) \rightarrow isInAverageSpeed(?y, True).$$

Knowing the type of the corresponding sensor to object is beneficial to determine the type of intruder. For example, Table 2 points out the sensors being available for the detection in given objects. From Table 2, this paper conducts *ProperIntruderTypes* rules shown in Table 3.

Table 2: The corresponding sensors to an object.

	Magnetic	PIR	Acoustic	Pressure	UWB
Person	1	1	1	1	1
Animal	0	1	1	1	1
Vehicle	1	0	1	1	1
Airplane	0	0	1	1	0
Birds	0	0	1	0	0

Table 3: The rules for *ProperIntruderType*.

	Types of sensor responding (= alarmed sensor)						Moving objects				
	Magnetic ∧	PIR ∧	Acoustic ∧	Pressure ∧	UWB	→ *ProperIntruderType* =	{person,	animal,	Vehicle,	Airplane,	birds}
Rule 1	0	0	0	0	0	→	1	1	1	1	1
Rule 2	0	0	0	0	1	→	0	0	0	1	1
Rule 3	0	0	0	1	0	→	0	0	0	0	1
Rule 4	0	0	0	1	1	→	0	0	0	1	1
Rule 5	0	0	1	0	0	→	0	0	0	0	0
Rule 6	0	0	1	0	1	→	0	0	0	1	1
Rule 7	0	0	1	1	0	→	0	0	0	0	1
Rule 8	0	0	1	1	1	→	0	0	0	1	1
⋮	⋮	⋮	⋮	⋮	⋮	→	⋮	⋮	⋮	⋮	⋮
Rule 31	1	1	1	1	0	→	0	1	1	1	1
Rule 32	1	1	1	1	1	→	0	1	1	1	1

5. Experimental Result

5.1. The Sensor Field Applied for the Simulation. This paper produces simulated results through the sensor field comprised of different types of 36 sensors having 90 percent of detecting rate and 1 percent of false alarm rate. Figure 15 indicates the utilized sensor field for the simulation. We assumed that a sensor field is deployed along the road.

We conduct three kinds of HMM model which are "Person," "Animal," and "Tracked Vehicle" with parameters of Gaussian mobility model. Among those models, this paper defines the possible types of intruders (target objects) on the sensor field as "Person" and "Tracked Vehicle." The other incoming objects are classified as nontarget objects of a sensor field (animal, airplane, and other clutters). Assume that the expected initial points of intruders are "X_3," "X_{42}," and "X_6," while non-target objects have the random initial points.

5.2. The Performance of Our Approach Using HMM with Rules for Intruder Detection. The performance of Table 4 is measured by three kinds of a sensor field. Each sensor field has (80%, 1%), (90%, 3%) and (99%, 5%) of detection rate and false alarm rate, respectively. Define the measurements for the performance test as follows:

Detection rate of sensor: $P(Sensor_{alarm} \mid Target)$,
False alarm rate of sensor: $P(Sensor_{noRepond} \mid \neg Target)$

Intruder detection rate of a sensor field
$M_{ds} = P(Decision_{intruder} \mid Intruder)$

False detection rate of a sensor field
$M_{fd} = P(Decision_{intruder} \mid \neg Intruder)$

Detection fail of a sensor field
$M_{df} = P(Decision_{nonIntruder} \mid Intruder)$

No response of a sensor field
$M_{nr} = P(Decision_{nonIntruder} \mid \neg Intruder)$.

Note that $Target$ = {Person, Animal, Tracked Vehicle}, $Intruder$ = {Person, Tracked Vehicle}, and $NonIntruder$ = $\neg Intruder$ (any other objects or clutters except "Person" and "Tracked Vehicle"). The listed results in Table 4 are obtained by 100 times simulation for each object. The movements of "Person," "Animal," and "Tracked Vehicle" are generated through Gaussian mobility model, and random noise is acquired by random observations. The average performance of our approach using HMM with rules represents 98.3% of intruder detection rate (M_{ds}) and 0% of false detection rate (M_{fd}). From the results, in overall, this paper insists that our approach successfully makes a decision for intruder detection. The simulation for intruder detection is designed to investigate the sensitivity of a sensor field. Hence, each movement of objects has been generated to a different sensor field (SF = $\{(80, 1), (90, 3), (99, 5)\}$). The obtained results point out that the performance of intruder detection is more sensitive to "False alarm rate" than "Detection rate" of sensors. In the same manner, we assume that "False detection rate" M_{fd} is one of the key measures to evaluate the performance of a sensor field.

5.3. The Effectiveness Analysis of Our Approach. Our approach is mainly designed to focus on reducing M_{fd} as using threshold-based filter and rule-based filtering. Table 5 demonstrates the effectiveness in reducing "False detection rate" of our approach. In Table 5, "Misclassification rate" is obtained from stochastic decision using HMM only, while "False detection rate" is calculated with our approach. The fed observations to each model as follows:

(i) Random noise: random observations

(ii) Airplane: simultaneously, pressure and acoustic sensors only react

(iii) Animal*: force magnetic sensor to react with animal movement; make every movements pass by an magnetic sensor at least once.

From the thresholds for stochastic decision, "Random Noise" and "Airplane" are filtered as "Nonintruder." In case of "Animal*," the sequences of observations are classified into 91% of Person and 9% of Animal. Hence, "Misclassification rate" is measured as 1 because by definition of "Animal*" the

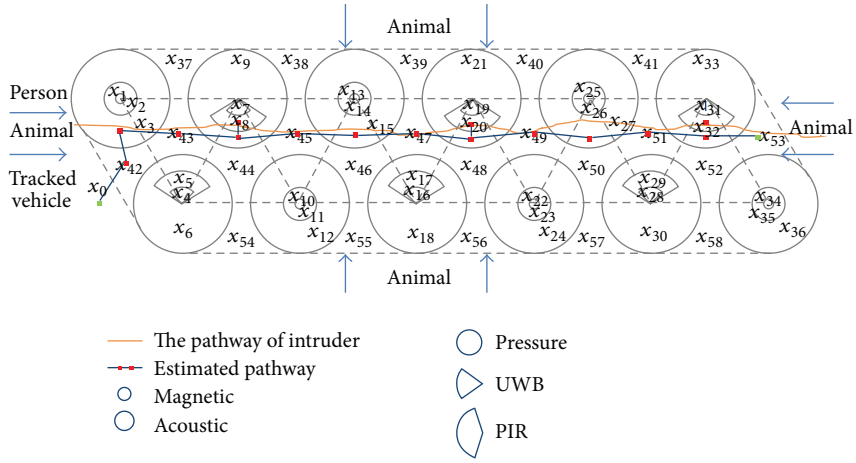


Figure 15: The sensor field simulated.

Table 4: The performance of intruder detection using HMM with rules.

Moving objects	The performance of sensors on a sensor field		The performance of our approach (%)
	Detection rate (%)	False alarm rate (%)	
Intruder			
Person	80	1	98
	90	3	97
	99	5	95
Tracked vehicle	80	1	100
	90	3	100
	99	5	100
Nonintruder			
Animal	80	1	0
	90	3	0
	99	5	0
Random noise	80	1	0
	90	3	0
	99	5	0

Intruder detection rate

False detection rate

movement of that should not be classified as Person, Animal, and Tracked Vehicle.

The interesting point in Table 5 is that nine movement patterns of "Animal*" are classified as animal. Basically, magnetic sensors do not react with animal movements. As a result, in principal, the emission probability of HMM for "Animal" ($P(X \mid O)$) is always 0, when magnetic sensors are responded. That means that magnetic sensors make alarms, whenever the movement is not an animal. For this issues, there are two exceptional cases determining an animal for given observations of "Animal*." The first is that the false alarm rate of the sensors could contribute to emission probability when calculating its probability. In this case, we have disabled the probability of detection and of false alarm as well in the establishment of HMM model for "Animal." The second one is that the rate of detection failure for "Animal*" could make the observations which magnetic sensor does not react to. This situation causes the movement of "Animal*" to be classified into an animal (misclassification) and in general, practically it makes the measurements of M_{fd} and M_{df} considerable when establishing detection model. From this point of view, keeping least rates of detection fail and false detection, not to mention detection rate, is a main objective for intruder detection model. Table 6 and Figure 16 reveal the outstanding performance of our approach.

We conduct SDR (simple decision rule) to compare with our approach. SDR_n makes a decision when the consecutive alarms are reported at least n times. In terms of detection rate, $\text{SDR}_{(1)}$ is the most effective model than any other models but it also has the worst performance on false detection rate. Basically, the numbers of consecutive alarms are in inverse proportion to detection fail and in direct proportion to detection and false detection. Hence, the optimal numbers of SDR could be obtained by the analyzing sensitivity of measurements (detection, detection fail, and false detection). According to the sensitivity of those measurements in some

Table 5: The results of false detection simulation with a sensor field (detection rate: 0.9, false alarm rate: 0.01 for sensors).

	Person	Animal	Tracked vehicle	Misclassification rate (HMM decision)	False detection rate (HMM with rules decision)
Random noise	0	0	0	0	0%
Airplane	0	0	0	0	
Animal*	91	9	0	1	

Table 6: Decision-making performance.

%	Measurements	The performance of sensors (detection rate, false detection) $SDR_{(1)}$	SDR_3	SDR_5	SDR_{10}	SDR_{20}	SDR_{30}	SDR_{40}	HMM with rules
(80, 1)	Detection	100	100	100	98	13	2	0.5	99
	Detection fail	0	0	0	1.5	84	98	99.5	1
	False detection	100	65	52.67	33.33	26.00	3.33	0	0
(90, 3)	Detection	100	100	100	100	44	14.5	2.5	98.5
	Detection fail	0	0	0	0	56	85.5	97.5	1.5
	False detection	100	66	61.67	33.33	33.33	30	12	0
(99, 5)	Detection	100	100	100	100	68	45	33	97.5
	Detection fail	0	0	0	0	32	55	67	2.5
	False detection	100	71	64.33	33.33	33.33	33.33	33.33	0

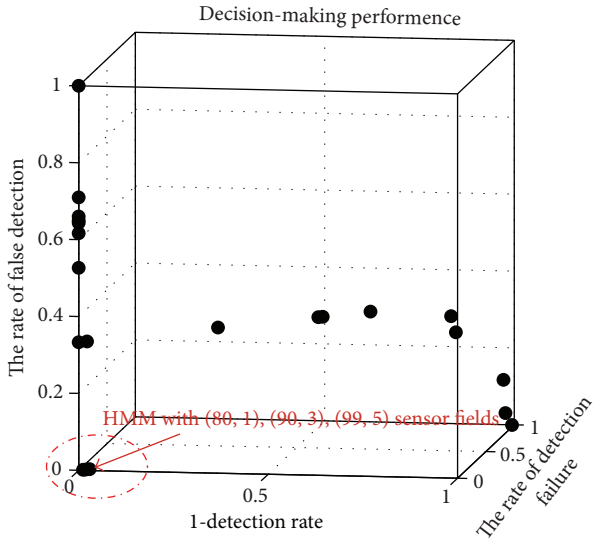


Figure 16: Decision-making performance.

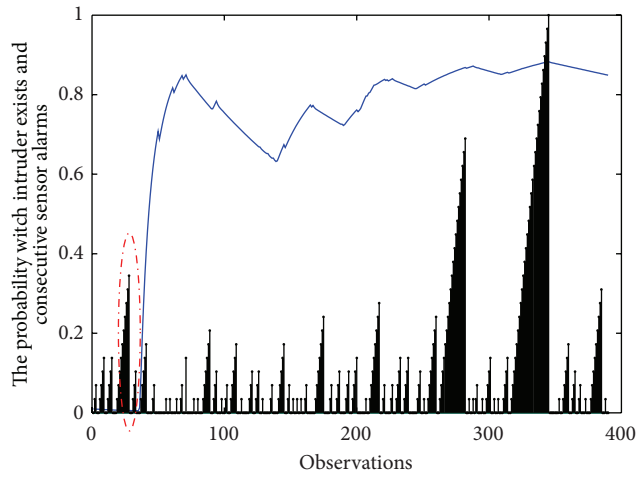


Figure 17: Consecutive sensor alarms.

specific problems, an optimal SDR sometimes could be more effective than our approach. However, in general we insist that our approach is more distinguish than SDR because of holding the admissible detection rate with low level of false detection and detection fail (Figure 16).

As the final simulation, we investigate the possibility finding out the initial point for decision-making process. We recognized that once enough observations are accumulated, then the movement decoding using Viterbi algorithm is properly operated. From this point, we assume that finding out the number of consecutive sensor alarms would be beneficial to the movement estimation of objects by delaying the initial activation time of HMM. As an example, we investigate person's movement with 100 times random samples. Figure 17 represents the tendency of conditional probability that intruder exists on a sensor field and consecutive sensor alarms at the time t. The probability that an intruder exists on a sensor field is calculated with $P_{intruder\ exist} = 1 - P(X_0 \mid O)$ and consecutive sensor alarms are measured by $\mathrm{CSA}_t = C^t / \mathrm{Max}(C^{t_1:t_n})$. Note that C^t points out the consecutive sensor alarms having the same observation at the time t. In Figure 17, we recognize if CSA_t has a first peak with predefined threshold 0.2 for person movement, and in the mean time $P_{intruder\ exist}$ closes to 1 which means some object is on a sensor field. From the result, we expect that the initial

activation time of HMM could be determined as using CSA_t. This brings to us valuable points. First, by determining the initial activation time for HMM, we could increase the performance of tracing movement. Second, with the policy of sensor activation using CSA_t, we could extend the operating time of a sensor field by delaying activation of the decision process for unnecessary observations.

6. Conclusion

In this paper, we consider a decision-making methodology for intruder detection by synthesizing the decisions on sensor network. This paper especially adopts the HMM to combine individual sensor decisions in stochastic manner and applies rules for the enhancement of the final decision. Firstly, using the HMM, this paper collects decisions of individual sensors on a sensor field and retrieves an estimated movement of a moving object. The obtained movement pattern is employed to identify the type of an object on the sensor field by taking advantage of spatial-temporal information. In this way, retrieved movement patterns on a sensor field contribute to the judgment of intruder detection beyond the simple use of signal values from individual sensors with some thresholds in their decisions. Secondly, this paper uses rules to enhance the stochastic decision obtained from the HMM. In principle, the HMM makes a decision by a given transition and emission probabilities under the assumption that all events have probabilities to occur. However, there are worth axioms and knowledge for a decision making on a sensor field that contradicts the assumption of probabilistic model (HMM). This paper adopts these knowledge and axioms as rules to enhance the decision of sensor field. As an example, this paper conducts several rules representing specific movement patterns of objects, the average speeds of the movements, and the sensor types which respond to specific objects. Since any kind of knowledge can be expressed by the rules, this proposed methodology could be easily extended for other purposes.

The contribution of this paper can be summarized as follows. First, this paper proposes the dynamic discretization method for the construction of the state space in a sensor field. Any shape of a sensor field is dynamically represented as a state space for the HMM through the proposed discretization method. Second, this paper provides a decision-making methodology for intruder detection on a sensor field by using HMM and rules, and its performance is evaluated with simulations.

Acknowledgment

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0024532).

References

[1] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem," in *Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems (DARS '02)*, 2002.

[2] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor deployment strategy for target detection," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 42–48, September 2002.

[3] C.-H. Wu, K.-C. Lee, and Y.-C. Chung, "A Delaunay Triangulation based method for wireless sensor network deployment," *Computer Communications*, vol. 30, no. 14-15, pp. 2744–2752, 2007.

[4] Q. Li, M. Zhang, and G. Xu, "A novel element detection method in audio sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 607187, 12 pages, 2013.

[5] F. Zhao, J. Liu, J. Liu, L. Guibas, and J. Reich, "Collaborative signal and information processing: an information-directed approach," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1199–1209, 2003.

[6] R. R. Tenney and N. R. Sandell Jr., "Detection with distributed sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 17, no. 4, pp. 501–510, 1981.

[7] P. Blunsom, "Hidden markov models," Lecture Notes, August 2004.

[8] S. Tugac and M. Efe, "Radar target detection using hidden Markov models," *Progress in Electromagnetics Research B*, vol. 44, pp. 241–259, 2012.

[9] J. Henderson, S. Salzberg, and K. H. Fasman, "Finding genes in DNA with a hidden Markov model," *Journal of Computational Biology*, vol. 4, no. 2, pp. 127–141, 1997.

[10] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November 2002.

[11] K. Ma, Y. Zhang, and W. Trappe, "Managing the mobility of a mobile sensor network using network dynamics," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 1, pp. 106–120, 2008.

[12] D. Lischinski, "Incremental Delaunay triangulation," in *Graphics Gems IV*, pp. 47–59, Academic Press, 1994.

[13] Y. Jian, M. Zhang, J. Tao, and X. Wang, "A novel HMM-based TTS system using both continuous HMMS and discrete HMMS," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '07)*, pp. IV709–IV712, April 2007.

[14] E. Bocchieri, "Vector quantization for the efficient computation of continuous density likelihoods," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '93)*, pp. 692–695, April 1993.

[15] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.

[16] H.-L. Lou, "Implementing the Viterbi algorithm," *IEEE Signal Processing Magazine*, vol. 12, no. 5, pp. 42–52, 1995.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 907583, 7 pages
http://dx.doi.org/10.1155/2013/907583

Research Article

A TDMA Scheme for Mobile Sensor Networks

M. Abdullah-Al-Wadud

Department of Industrial and Management Engineering, Hankuk University of Foreign Studies, 81 Wangsan, Mohyeon, Cheoin, Yongin, Gyeonggi 449-791, Republic of Korea

Correspondence should be addressed to M. Abdullah-Al-Wadud; wadud@hufs.ac.kr

Received 2 February 2013; Revised 24 April 2013; Accepted 3 June 2013

Academic Editor: Nazif Cihan Taş

We propose a TDMA-based protocol for mobile sensor networks. The proposed protocol overcomes the shortcomings of the other available TDMA-based protocols in a dynamic network where the cluster membership may change frequently. Unlike other existing TDMA-based protocols, we propose to vary the number of timeslots in the TDMA frame to allow underutilization of unused slots and successful allocation of timeslots to the newly joined sensors through minimizing collisions. The approach is fully decentralized and efficient to be used by the cluster head (leader) and the sensor nodes in a cluster. Simulations also show that the proposed mechanism provides significant performance improvements compared to the other existing approaches in terms of different network performance metrics.

1. Introduction

A sensor can be regarded as an atomic computing particle, which can capture and process different data about the surrounding geographical location where it is deployed [1]. After deployment to a target location, the sensors work as a wireless network, through which the data sensed by the sensors can be gathered for analysis.

Mobile sensor networks (MSNs) have gained much commercial as well as research interest in last few years. The critical applications, where MSNs are used, include underwater data collections, rural networks in irregular terrains [2], and other military, scientific, and industrial aspects. Since sensor networks do not need any specific infrastructure, the sensors can be deployed very much rapidly and with ease. However, if the medium access control (MAC) protocol used in MSN cannot provide the required throughput, the deployment of application may be hampered, reducing the utility of the usage of the network [2]. The conventional random access-based MAC using request/response-based contention (such as CSMA/CA) may not fulfill this requirement due to a lot of interference and collisions among the messages sent by different sensors. Moreover, fairness may also not be guaranteed as some nodes may experience a busy channel for an extended period of time.

To minimize the collisions, some time division multiple access (TDMA) [3, 4] protocols have been proposed. In TDMA protocols, one frame (alternatively known as round or cycle) of transmission time is divided into some slots specified for the sensors' transmissions. The frame repeats over time, and a sensor may send message(s) during its dedicated slot only. This ensures collision-free transmissions. The TDMA approaches usually use fixed number of timeslots in frame, which is equal to the number of nodes in the networks (or in the interference range). This works well for static networks. However, in MSNs, some sensors may not have always messages to transmit (many of them may change their statuses to the so-called "sleep" mode to conserve energy). In such cases, the timeslots dedicated to them remain unused, overlooking the opportunity to make use of these slots to improve the performance of the network. Moreover, since the sensors are mobile, the cluster membership may also change. If a sensor node moves out of its current cluster, its assigned slots will remain unused. Again, it should be allocated slot(s) in the frame being used in the new cluster where it moves in. However, use of a fixed number of timeslots in a frame may not handle such situations effectively as there may be shortage of timeslots. Moreover, contention-based approaches [2, 5, 6] are often used during making the decision scheduling the slots among sensors. This also makes it prone to a good

amount of collisions among the control messages sent by different sensors that make delays in joining to a new cluster. Focusing on some slot allocation mechanisms proposed to be used in smart antenna base stations [7, 8], Wong and Jia [6] propose a contention-based protocol to minimize collisions. Here the sizes of the contention windows and the duration (number of slots) of contention period are estimated based on the number of nodes. However, these terms may not be properly estimated if the number of nodes being served is rapidly changed, which is a very common phenomenon in MSNs.

In this paper, addressing all these issues, we propose a flexible TDMA (FTDMA) scheme, which can adapt well with the changes in an MSN. Different performance metrics also show the better performance of the FTDMA compared to the other proposed protocols.

The rest of the paper is organized as follows. Section 2 presents some other works related to ours. Section 3 describes the proposed FTDMA protocol while Section 4 presents the comparative performances of different protocols. Finally, Section 5 concludes the paper.

2. Related Works

There are two major requirements, in general, for collecting the data sensed and transmitted by the sensor nodes in an MSN: the discovery of the current network topology and the protocol to transmit the messages.

To use a TDMA scheme, cluster-based topology (or a variant) is usually used. Different methods are proposed to form the clusters as well as to select the cluster leaders [9, 10]. However, the scope of this work does not include the formation of clusters. In this paper, we adopt the approach proposed by Kothapalli et al. [9].

The IEEE 802.11 protocol based on the carrier sense multiple access with collision avoidance (CSMA/CA) is currently the most popular transmission technique. In CSMA/CA, a node transmits a packet if it finds the medium to be clear. Otherwise, it follows an exponential back-off scheme and tries to transmit later. A receiver sends acknowledgment packets (ACK) to confirm the successful receipt of the packets. Two other messages, (Request To Send RTS) and (Clear To Send CTS) are also used before transmission for negotiation. Woo and Culler [11] use carrier sensing to avoid collisions. However, fairness may not be guaranteed because some nodes may always find the channel busy with other transmissions.

In TDMA protocol, a period of time called frame is divided into a fixed number of timeslots where every timeslot is dedicated to exactly one node to send its packet(s). In traditional TDMA protocols, the number of timeslots in a frame is equal to the number of nodes in a cluster when a cluster-based topology is used. This ensures collision-free message transmissions.

Traditional TDMA protocols work well for fixed networks. However, in MSNs, the membership of the sensor nodes in clusters may change due to the mobility of nodes and/or the switching of nodes between the so-called "sleep" and "awake" modes. If some nodes move out of the current cluster and/or some node goes to the "sleep" mode (because it has nothing to transmit) for energy conservation, the slots allocated to them are still reserved but wasted. On the other hand, if some nodes move into the cluster and/or some nodes wake up from "sleep" mode, there may be shortage of slots to support their transmissions. Focusing on these issues, Herman and Tixeuil [12] propose a TDMA approach. However, it may not adapt over time. It also requires a big overhead of managing the information of extended neighborhood of a node, which makes it difficult to implement [2].

Bruhadeshwar et al. [2] propose self-stabilizing TDMA (STDMA), adopting the proposal by Kothapalli et al. [9] to form the overlay network, a TDMA approach where the slots are divided in a hierarchical manner. First, blocks of timeslots in a frame are divided among the cluster heads of the clusters. Cluster heads then assign their allocated timeslots among the member nodes. Doing so, this approach prevents the possible interferences among the transmissions of nodes in different clusters. To assign the blocks of timeslots to the cluster heads, this approach directly follows the clustering in the overlay network. However, the STDMA approach uses a fixed number of timeslots and hence may fall in short of slots when the number of member nodes increases. It may not make efficient use of unused slots too, making unwanted delay in the network. Moreover, during the contention period of STDMA, there remain high chances of collisions, and the leader needs to send a status message regarding the slot usage which adds up as an overhead.

Overhead tolerant TDMA (OLT_TDMA) [13] is another approach where a frame has a fixed number of timeslots. A newly joined node first scans the whole frame to sense and make a list of the unused slots. It then randomly picks one slot and starts transmission in the following frame. However, if more than one node selects the same slot, their messages may collide. The leader then marks the slot in the "collision map" and sends this information to all the nodes. The nodes then avoid the marked slots. Hence, it may underutilize the available timeslots. When the number of nodes becomes higher than the available slots, the OLT_TDMA allows selecting a victim node, and then both the nodes contend for the same slot. This approach may not always guarantee the collision avoidance and good performance.

D-TDMA [14, 15] is a dynamic TDMA scheduling approach addressing collisions and interferences. It also applies strategies, such as allocating contiguous slots to a node for transmission and reception, to minimize switching overhead of the nodes. However, like other TDMA protocols having a fixed number of slots in a frame, D-TDMA may also underutilize the available timeslots when network density is low. Again, when the number of requested (sender, receiver) pair becomes too high to be allocated in a frame, the D-TDMA may not always find an appropriate slot to choose without hampering other nodes' transmissions.

We have proposed a flexible TDMA (FDTMA) mechanism, which successfully overcomes the shortcomings of the other existing proposals in different membership conditions (a preliminary version of this proposal can be found in [16]). A new sensor can always be allocated a timeslot

without making any interference with other transmissions. The FTDMA allocates more slots to support more sensors when the number of sensors in a cluster grows higher. This minimizes the contentions, the number of packet losses, and the queuing delay, which ensures very good network performance. On the other hand, it decreases the number of timeslots if unused. This also minimizes unnecessary waiting time, which increases the throughput.

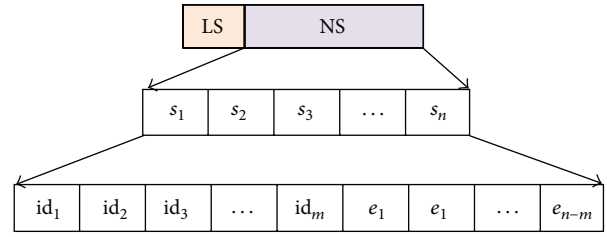


Figure 1: The structure of an FTDMA frame.

3. Proposed Method

3.1. Statement of Problem. In this work, we focus on a cluster-based protocol for propagating messages in MSNs. We can think of two issues related to such problems: discovering a so-called overlay network allowing efficient routing and the medium access control (MAC) protocol. The former one handles the problem of clustering the sensors in different clusters and selecting a leader for each of the clusters. It is much studied for the static networks. Some works are also found for the mobile nodes as presented earlier. In this work, however, we focus on the second issue—the transmission protocol.

For the discovery of the overlay network, we adopt the model proposed by Kothapalli et al. [9]. In this model, a backbone structure (the so-called overlay network) is established by efficiently making some clusters on sensors and selecting the cluster leaders. This model is based on the concept of *dominating set*. It is proved to be locally self-stabilizing and can converge quickly.

Using this overlay network, we focus on achieving a TDMA protocol that ensures inference-free transmissions of the nodes in a cluster. At the same time, a good protocol should also meet the desirable performance requirements in a network such as fairness, adaptability and throughput. No node should starve for timeslots to be assigned to it. A good overall network throughput should be maintained. In other words, the duration of control messages should be much less than the data transmissions. If some nodes leave the cluster due to mobility, energy conservation, or any other reason, the protocol should discover their unused timeslots and make use of them. On the other hand, the protocol should also welcome newly joined nodes in a cluster by assigning timeslots for their necessary transmissions. The scheme should also be self-stabilized, which can work without any central control.

Hence, the allocation of timeslots directly influences the network performance. Furthermore, proper timeslot allocation also ensures collision-free communication. So, an efficient way of scheduling the timeslots among the members of a cluster is required to maximize the network performance.

3.2. The Proposed Approach. TDMA protocols usually consider a "frame," which is a period of time split into some equal intervals called timeslots. Each timeslot is designated to a transmitting node (sensor) to transmit its data. The frame structure is repeated over time. Hence, the nodes get the chances to transmit in a round-robin fashion.

The problem of the most of the TDMA protocols is that they use fixed number of timeslots in a frame. Hence, the protocol may fall in shortage of timeslots to allocate to the newly joined nodes so that they may also perform their transmissions. Moreover, if some nodes are not transmitting, because of reasons like having no data to transmit, the timeslots allocated to them may also remain unutilized. Focusing on these shortcomings, we propose a flexible time division multiple access (FTDMA) protocol, which can make a better use of the timeslots and provide guaranteed support to the newly joined nodes' transmissions.

3.2.1. The Frame Structure. The basic structure of the frame used in the proposed FTDMA is much similar to most of the conventional TDMA protocols. An FTDMA frame has two parts: the leader segment (LS) and the node segment (NS). The leader of a cluster broadcasts the schedule information in the LS. The NS is composed of n timeslots of equal duration, and the duration of a slot is set to transmission/reception time of one (or more) data packets along with the ACKs. According to the scheduling information broadcasted by the leader during the LS, the sensor nodes in that cluster may transmit their data during their allocated timeslots in the subsequent NS.

It is noteworthy to mention here that a message may sometimes take longer time in MSN (than the static network) to reach the receiving sensor due to the increase in relative physical distance between the sender and the receiver caused by mobility of the sensors. When the transmission ranges of the nodes in a network are small, which is true for the low-power devices such as sensors, this delay is usually negligible. However, if such propagation delays become significant, the cluster leader can handle it by increasing the duration of the timeslots. In such a case, slot duration can be included in the message broadcasted by the leader during LS.

Unlike the traditional TDMA approaches, the number of timeslots, n, in the NS of FTDMA frame may vary based on the cluster's membership condition. FTDMA increases (or decreases) the value of n if some sensor nodes join (or leave) the cluster. This provides with the opportunity to successfully allocate timeslots to the newly joined nodes and reduce the possibility of idle slots, and thus FTDMA enhances the network performance.

Among the n timeslots in the NS, m slots are scheduled to m sensor nodes' transmissions, and the rest $n - m$ slots are kept empty to be used by the newly joined nodes (to whom no slot has been assigned). The scheduling is described later in detail. The structure of the proposed TDMA frame is presented in Figure 1.

3.2.2. The Frame Usage

(1) Role of Leader. In the LS of a frame, the leader of a cluster may broadcast its message containing the scheduling of the timeslots to be used by the sensors in the following NS of that frame. Simply stating, a leader allocates a slot in the subsequent NS for each *willing* (i.e., willing to transmit) sensor in its cluster. A sensor is considered *willing* if the leader has received at least one message during the previous three (this value may be preset by the network administrator depending on the mode of application) frames. If there are m *willing* sensors in the cluster, the leader puts n (where $n > m$) timeslots in the subsequent NS of the frame. The first m slots are allocated to the *willing* sensors, and the rest $n - m$ slots give the room to the newly incoming sensors (see Figure 1). The format of the message broadcasted by the leader contains the information shown in Figure 2. It contains the values n and m followed by m IDs of the sensors, which are allocated a timeslot in the subsequent NS.

Here r represents the number of rounds that a frame will repeat without any further LS from the leader. This is especially helpful when a small number of nodes transmit in a cluster. In such cases, the LS becomes an overhead if there is no change in cluster membership. The use of r thus improves the channel utilization avoiding the LS in the subsequent $r-1$ rounds of the frame. As the next LS message (i.e., allocating slots to any newly enjoined sensor in the cluster) may happen after at least r rounds of the frame, r should be set at a value that does not notably degrade the QoS for the newly joined nodes. If the leader senses a collision (among the REQUEST messages as explained later) during any of the empty slots, it may set $r = 1$ in its next broadcast in LS so that the newly joined nodes may try to send REQUEST messages in every frame.

Upon successful receipt of a REQUEST message from a sensor, a leader sends an ACK to it assuring that the corresponding sensors will be allocated a timeslot in the upcoming frame. If another sensor having the same ID as this new sensor already exists in the cluster, the leader also assigns and sends a new ID together with the ACK.

(2) Role of Member Sensors. Sensors in a cluster may be categorized as *unwilling* (having no data to send), *waiting* (a newcomer sensor who has not yet tried to get a timeslot), *willing* (a newcomer sensor who has been trying to get a timeslot but not received ACK from the leader yet), and *active* (sensor that has been assigned a timeslot or received an ACK from the leader) state sensors.

An *active* sensor scans (in listening mode) for a broadcast message from the leader during the LS of a frame. Upon getting the message, it searches its ID in it. If it finds the ID in the ith position of the broadcasted message, it transmits its data during the s_i slot of the NS. This ensures no collision with the transmission of other sensors.

A *waiting* sensor keeps scanning (because it has no idea about the frame as well as the LS in the newly joined cluster) for a broadcast message from a leader. After receiving a broadcast, it first gets the time information and synchronizes its clock with that of the leader. It then changes its status to *willing*. Note that it can send a REQUEST message in the same frame.

r	n	m	id_1	id_2	id_3	$\cdots$	id_m

Figure 2: The information included in the message broadcasted by a leader during LS of a frame.

According to the message broadcasted by the leader, r round(s) of $n - m$ empty (free) slots are going to appear. A *willing* sensor randomly (uniform random) selects a slot e_j from the $r(n - m)$ empty (free) slots and transmits a REQUEST message during the slot e_j. If it gets an ACK from the leader, it changes its status to *active* at the end of the rth round of frame. It also changes its ID if there is such instruction with the ACK.

An *unwilling* sensor may keep its radio off (some application, however, may require the radio to be on in order to receive necessary control signals) to conserve energy (a sensor at any other status performs only in one slot of NS. In the rest of the time it may also switch off its radio to preserve energy). When it has some data to transmit, it switches to the *waiting* state.

Note here that there is chance of collision of the transmissions if more than one sensor chooses the same empty slot e_j. The procedure to handle such situations is described in the next subsection.

Another point is that a sensor may also send a data packet instead of the REQUEST message. This may increase the network throughput a little, though at the cost of energy, if the transmission is unsuccessful due to collision.

3.2.3. Handling Membership Changes

(1) Incoming Nodes. To support the newly joined nodes, the leader keeps $n-m$ empty slots in the NS of a frame. In normal situation, we propose to put one empty slot on NS, that is, $n = m + 1$. This is because use of more empty slots may decrease the network throughput. When a leader successfully receives a REQUEST (or data packet) from a sensor in an empty slot without any collision, the leader allocates a slot for the sensor in the next frame (meaning that the sensor has joined the cluster successfully). At this point, it also decreases the value of n (ensuring the constraint $n > m$, of course). Otherwise, if a leader detects a collision during an empty slot, it increases the value of n (i.e., the number of empty slots) in the next frame. By increasing empty slots, the leader tries to minimize the probability of collisions in the subsequent frame. The general modification of the value of n is done using

$$n = m + \max\left(1, \mathrm{LSES} * C_f\right), \tag{1}$$

where C_f is the number the previous frames where no newly joined sensor could be allocated slots due to collisions, and (Last Successful Empty Slots LSES) represents the value $\lceil (n - m)/2 \rceil$ calculated in the last frame when the leader could successfully receive REQUEST message(s) from the newly joined sensor(s). The LSES gives an estimate of the number of empty slots that may suffice to entertain the newly joined sensors. The value $\lceil (n - m)/2 \rceil$ ensures the increase

(or decrease) of empty slots based on the network conditions, especially the rate of incoming sensors. When there is no collision, using one empty slot suffices to welcome newly joined nodes. On the other hand, the number of empty slots is increased to resolve collision(s), if any.

The proposed FTDMA thus handles the joining requests of incoming nodes much better than the other available approaches. While most of the other approaches dedicate one portion of the frame for the newly joined nodes so that they may send the REQUEST messages based on traditional contention and back-off [2, 5], the leader increases the room for the new nodes. With more empty slots, it is more likely that more sensors can be facilitated facing less number of collisions. It effectively minimizes the joining time of a sensor to a cluster in FTDMA. This leads to a very simple and easy way of management.

When the number of nodes becomes too high (very dense network) in a cluster, the performances of the different existing protocols highly degrade which is also true for the proposed FTDMA. In such a network, the allocation of a slot for every sensor in FTDMA may lead to a very long frame. This may cause delay on the channel access for the sensors. This may be critical in a network where a delay-sensitive application is deployed. In such cases, a limit on the cluster size can be imposed so that a cluster is broken into smaller clusters. In an application where packet losses are not critical, a small frame may be used where a leader schedules different subsets of the sensors in different cycles of the frame. In this work, however, we assume a moderately dense network which is not delay sensitive.

(2) Outgoing Nodes. A member sensor of a cluster may be disconnected from the leader for a number of reasons such as turning the radio off for a while (entering the *unwilling* state), dying (battery power exhausted), and moving out of the cluster. In FTDMA, a "going-to-disconnect" sensor sends a DISCONNECT message (to the leader) in its allocated slot, and the leader does not allocate any slot for it in the following frames. However, such cases may not always be predetected by the sensors nodes, and hence a node may be disconnected without sending the DISCONNECT message. To handle such situations, we propose the leader to consider a sensor node as "disconnected" if it does not receive any message from the sensor in k number of contiguous frames ($k = 3$ works well in general). The value of k can be predetermined by the network administrator according to the characteristics and availability of sensory data.

When a sensor is disconnected, the leader does not further allocate any slot for it in the following frames. It also decreases the value of m that in turn decreases n to shrink the frame size. Thus FTDMA gets rid of unused timeslots. A smaller frame also iterates quicker, which provides the *willing* sensors to have their chances to transmit more frequently. This increases the overall throughput.

4. Performance Evaluation

We have evaluated the performance of the proposed FTDMA in comparison with the IEEE 802.11, the overload tolerant TDMA (OLT-TDMA) [13], and the D-TDMA [14, 15] through simulation. In our simulation scenario, we have constructed a 400 m × 300 m rectangular field, where the sensors can move freely in any direction. A location in the field is represented as a 2D vector (x, y), where $x \in [0, 400]$ and $y \in [0, 300]$. Sensors have been deployed at different randomly selected locations on the field. The movement directions of sensors are also picked randomly from 0 to 360 degrees. After the start of simulation, no sensor goes out of the field. If any sensor approaches the boundary, it changes its direction following the law of reflection with respect to the boundary. Sensors move at a constant speed (our simulations, however, include different speed scenarios too).

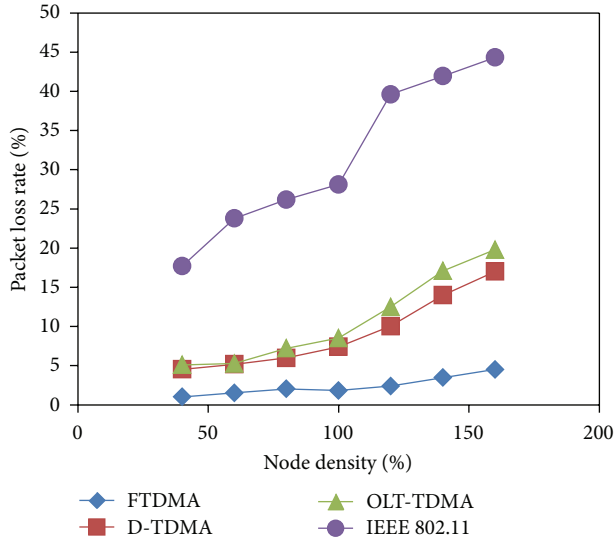


Figure 3: The packet loss ratios of the methods for different node densities.

We adopt the "Constant density spanners" approach proposed by Kothapalli et al. [9] to form the clusters and leaders and applied all the TDMA methods on the same cluster. For all the methods, the duration of the control segment (e.g., LS) of a frame is set to 14.8 ms while each timeslot assigned to a sensor is of 3.52 ms. Hence, a slot can accommodate the data transfer of 20 packets, each having the size of 44 Bytes. The number of slots in a frame is set to 10 for both OLT-TDMA and D-TDMA. A simulation has been run for 50 s. Each simulation scenario is repeated 30 times and the presented results show the average of these 30 runs.

For the first experiment, we have allowed the sensors' speeds to be selected randomly from the range $[1, 10]$. In our simulations, we have considered seven different node densities (the average number of nodes in a cluster) ranging from 40% to 160%. The approximate node densities have been achieved by increasing/decreasing the total number of sensors deployed on the field. Since D-TDMA and OLT-TDMA follow the TDMA technique with fixed number of timeslots, the node density can be considered 100% if there are approximately 10 sensors per cluster.

Figure 3 demonstrates the comparative performances of the three mechanisms in terms of packet loss ratio. It shows

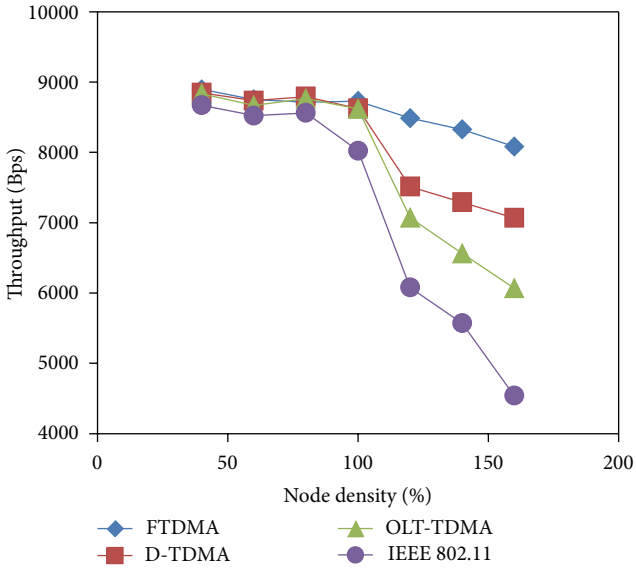


Figure 4: Throughput achieved by the methods for different node densities.

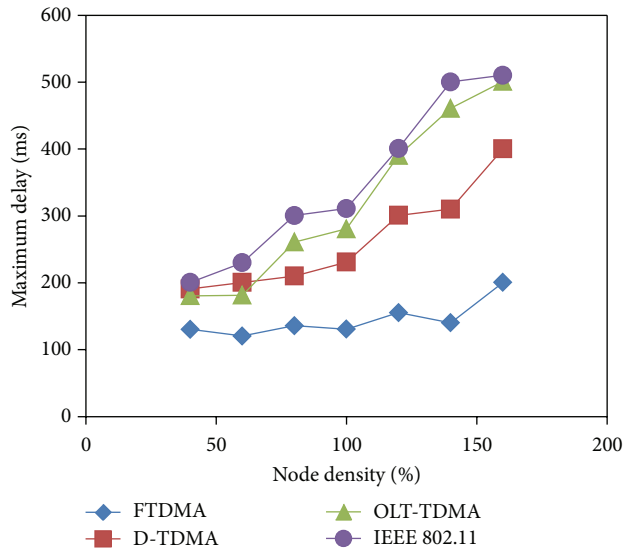


Figure 6: The maximum delays of the methods for different node densities.

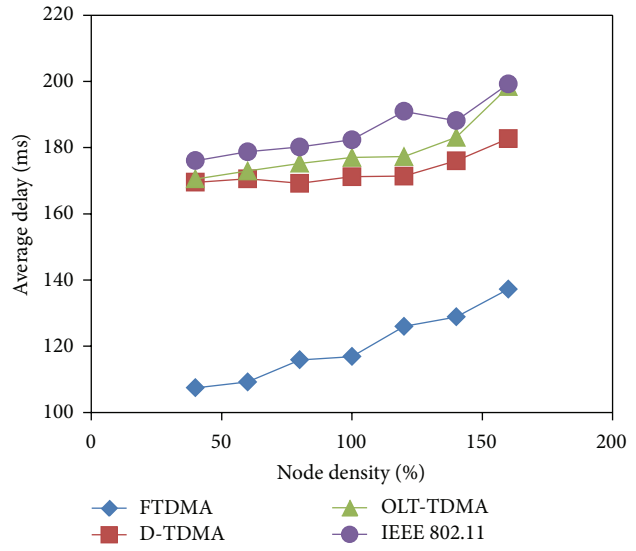


Figure 5: The average delays of the methods for different node densities.

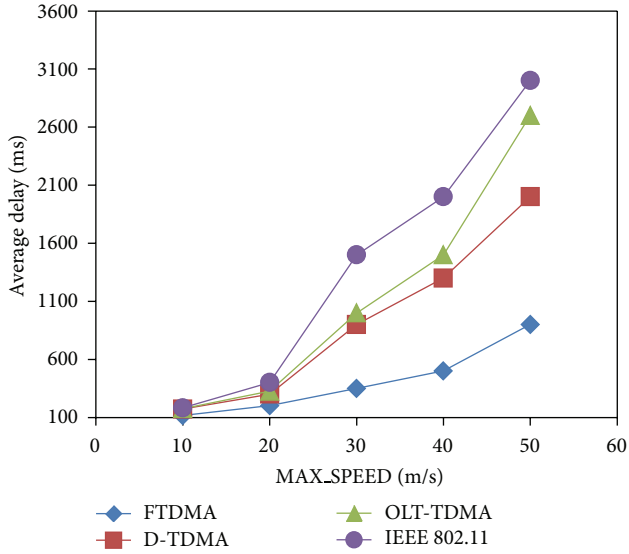


Figure 7: The average delays of the methods for different speeds of sensors.

that the packet loss rises with traffic load. However, the packet loss rate does not grow that much for the proposed FTDMA mechanism while it grows faster in the other protocols. The throughputs shown in Figure 4 also demonstrate the better performances of the FTDMA.

Figure 5 shows the average delay of all the sensors with different methods. The increase of traffic loads also increases the average queuing delay. However, with the FTDMA, the delay is significantly lower. The maximum delays shown in Figure 6 also show the similar performances.

In another experiment, we have set the node density to 100% and allow the sensors to have different speeds. In every scenario, we allow the sensors to randomly pick a speed from the range [1, MAX_SPEED], and we vary the MAX_SPEED in different scenarios. Figure 7 shows that the average delay increases with the increase of speeds of the sensors. However, the proposed FTDMA faces significantly lower delay compared to the other methods.

With the OLT-TDMA, a new node must listen to one whole frame to find an empty timeslot for it. However, with the proposed FTDMA, a node needs to listen only to the LS part of a frame, and it can choose an empty slot from the $n - m$ slots in that very frame for its transmission. If more than one sensor node selects the same slot in OLT-TDMA, a collision is detected by the leader and it marks the corresponding timeslot in the collision map in the subsequent frame.

The new joining nodes avoid this slot as there was a collision. This makes underutilization of the slots where some collisions occurred. Another delay occurs to execute the LIF algorithm for choosing a victim. And sharing a slot with the victim node requires a contention-based channel access, which may incur collisions and subsequent backoff and delay. The D-TDMA approach also faces similar shortcomings when the number of sensors per cluster increases. Furthermore, in a light-load scenario, many slots are likely to remain unused. On the other hand, all these problems are overcome by the proposed FTDMA. It minimizes the number of slots in low traffic conditions, resulting in the reduction of unnecessary slots, and maximizes the number during high node densities, ensuring less collision. All these desired points add to the better performance of the proposed FTDMA mechanism over the other existing ones.

5. Conclusion

In this paper, we focus on overcoming the shortcomings of the existing TDMA approaches and propose a flexible TDMA (FTDMA) approach, which works similar to that of the traditional TDMA approaches but with the exception that the number of timeslots in an FTDMA frame may vary depending on the cluster membership conditions. The protocol is scalable in clusters and can solely be controlled by the cluster leaders in a distributed manner. As future work, we plan to develop a general TDMA protocol that may work in different network topologies. We also have a plan to work on highly dense and delay-sensitive networks in future.

Acknowledgments

The authors would like to express their deep gratitude to the anonymous reviewers for their constructive comments that have helped to correct the flaws in the proposal. This work was supported by Hankuk University of Foreign Studies Research Fund of 2013.

References

[1] Md. Abdul Hamid, M. Abdullah-Al-Wadud, and I. Chong, "A schedule-based multi-channel MAC protocol for wireless sensor networks," *Sensors*, vol. 10, no. 10, pp. 9466–9480, 2010.

[2] B. Bruhadeshwar, K. Kothapalli, and I. R. Pulla, "A fully dynamic and self-stabilizing TDMA scheme for wireless ad-hoc networks," in *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications (AINA '10)*, pp. 511–518, April 2010.

[3] W. Crowther, R. Rettberg, D. Waldem, S. Ornstein, and F. Heart, "A system for broadcast communications: reservation ALOHA," *Proceedings of the 6th Hawaii International Conference on System Sciences*, 1973.

[4] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, "RR-ALOHA, a reliable R-ALOHA broadcast channel for ad-hoc inter-vehicle communication networks," in *Proceedings of the Med-Hoc-Net*, Baia Chia, Italy, 2002.

[5] A. Jhumka and S. Kulkarni, "On the design of mobility-tolerant TDMA-based Media Access Control (MAC) protocol for mobile sensor networks," *Proceedings of the 4th International Conference on Distributed Computing and Internet Technology*, vol. 4882, pp. 42–53, 2007.

[6] G. K. Wong and X. Jia, "An efficient scheduling scheme for hybrid TDMA and SDMA systems with smart antennas in WLANs," *Wireless Networks*, vol. 19, no. 2, pp. 259–271, 2013.

[7] F. Shad, T. D. Todd, V. Kezys, and J. Litva, "Dynamic slot allocation (DSA) in indoor SDMA/TDMA using a smart antenna basestation," *IEEE/ACM Transactions on Networking*, vol. 9, no. 1, pp. 69–81, 2001.

[8] R. Kuehner, T. D. Todd, F. Shad, and V. Kezys, "Forward-link capacity in smart antenna base stations with dynamic slot allocation," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 4, pp. 1024–1038, 2001.

[9] K. Kothapalli, C. Scheideler, M. Onus, and A. Richa, "Constant density spanners for wireless ad-hoc networks," in *Proceedings of the 70th Annual ACM Symposium on Parallelism in Algorithms and Architectures*, pp. 116–125, July 2005.

[10] M. Demirbas, A. Arora, V. Mittal, and V. Kulathumani, "A fault-local self-stabilizing clustering service for wireless ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 9, pp. 912–922, 2006.

[11] A. Woo and D. E. Culler, "A transmission control scheme for media access in sensor networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pp. 221–235, July 2001.

[12] T. Herman and S. Tixeuil, "A distributed TDMA slot assignment algorithm for wireless sensor networks," *Proceedings of the 7th International Symposium on Algorithms for Sensor Systems, Wireless Ad Hoc Networks and Autonomous Mobile Entities*, vol. 3121, pp. 45–58, 2004.

[13] R. Wang, Z. Wang, Z. Chen, and L. Zhang, "A 3G-802.11p based OLT-TDMA mechanism for cooperative safety in a dense traffic scenario," in *Proceedings of the 73rd IEEE Vehicular Technology Conference (VTC '11)*, pp. 1–5, Budapest, Hungary, May 2011.

[14] Y. Wang, P. Shi, K. Li, and Z. Chen, "D-TDMA: An approach of dynamic TDMA scheduling for target tracking in wireless sensor networks," in *Proceedings of the IEEE/ACM International Conference on Green Computing and Communications (GreenCom '10), IEEE/ACM International Conference on Cyber, Physical and Social Computing (CPSCom '10)*, pp. 546–553, December 2010.

[15] Y. Wang, P. Shi, K. Li, and Z. Chen, "An energy efficient medium access control protocol for target tracking based on dynamic convey tree collaboration in wireless sensor networks," *International Journal of Communication Systems*, vol. 25, no. 9, pp. 1139–1159, 2012.

[16] M. Abdullah-Al-Wadud, "A TDMA scheme for vehicular Ad Hoc networks," *Green University Review*, vol. 3, no. 2, pp. 53–56, 2012.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 245017, 16 pages
http://dx.doi.org/10.1155/2013/245017

Research Article

An Evolutionary Game-Based Trust Cooperative Stimulation Model for Large Scale MANETs

Xiao Wang,[1] Yinfeng Wu,[1] Yongji Ren,[1,2] Renjian Feng,[1] Ning Yu,[1] and Jiangwen Wan[1,3]

[1] *School of Instrumentation Science and Opto-Electronics Engineering, Beihang University, Beijing 100191, China*
[2] *Department of Command, Naval Aeronautical and Astronautical University, Yantai 264001, China*
[3] *Science and Technology on Communication Information Security Control Laboratory, Jiaxing 314033, China*

Correspondence should be addressed to Xiao Wang; wx04508@aspe.buaa.edu.cn

Received 3 April 2013; Accepted 13 May 2013

Academic Editor: Gurkan Tuna

In order to realize a methodical, effective cooperative stimulation for MANETs and search dynamic trust cooperative stimulation scheme in environment under a high malicious ratio, we have proposed an evolutionary game-based trust cooperative stimulation model for large scale MANETs in this paper. First, the system members' pluralistic behavior for MANETs has been covered by means of constructing the complete multirisk level strategy space. Then a trust-preferential strategy has been built through trust numerical value mapping technology, which achieves the aim that the malicious action is effectively constrained to avoid a low trust level. Furthermore, the mobility probable parameters and information propagation error matrix are introduced into game model, and the convergence condition between optimum strategy which represents payoff maximization principle and trust-preferential strategy is deduced through evolutionary analyzing finally. Both theoretical analysis and simulation experiments have demonstrated that our model can effectively stimulate cooperation among members and meanwhile be robust under the condition where the environment is harsh under a high original malicious ratio in large scale MANETs.

1. Introduction

With the development of perception theory, ubiquitous computation, and radio technology of multihop, the basic services of large scale mobile ad hoc networks (MANETs) can be autonomously deployed via local backbone nodes (BN), and accomplished by access network nodes (AN). Then the managers of each AN cooperatively upload essential information back to BN, which achieves network managements for large-scale MANETs. Hence, the cooperation among members is the premise for MANETs to provide network services.

However, realizing a methodical and effective scheme for cooperation is facing tougher challenges in MANETs. First, current networks are threatened by a wide range of attacks, such as flooding [1], spoofing [2], wormhole, and Sybil attacks [3], as well as other external attacks [4]. These threats seriously destroy cooperation in MANETs. Furthermore, even if adopting current popular secure mechanism [5, 6] to resist these attacks, due to the own inherent natures of MANETs, including limited available resources, complex deployment environment, exposed communication medium, and intermittent end-to-end links, some normal members may be unwilling to cooperate with others for saving resources to prolong their own network lifetime. The tolerable selfish behavior inevitably interrupts member cooperation of MANETs. For that reason, the exploration and researches on cooperative stimulation scheme for MANETs have been carried out all over the world. Along with current achievements, the cooperative stimulation based on game theory is the most representative. Combining with trust management, distributed system, and key protocol for MANETs, it has effectively stimulated members' cooperation by game theory for small range of wireless sensor networks and distributed networks. Obviously, the anticipation is clear for game theory as an analytical tool of MANETs: through modeling an independent strategy decision maker, users can control the whole network scene as an acentric control entity and abstract necessary hypothesis to address important problem like other mathematical models [7].

For large scale MANETs, we have found that there are some problems of current game theoretic cooperative stimulation scheme to be solved. First, it is the incompleteness of strategy and payoff space. Current scheme usually defines member's strategy space simply as "cooperative forwarding, packets dropping" and distributes each action with a payoff value. However in a large scale MANETs with infrastructure mobility, the network action chosen by members is diverse and complicated. The noncooperative actions come from malicious attacks as well as nonmalicious selfish behavior, which is not complete enough by merely describing it as packets dropping. Furthermore the cooperative action is not only the forwarding behavior. According to different network business, it shows various forms of cooperative behavior. Thus a complete strategy space reflecting realistic large scale MANETs and its rational payoff frame must be modeled. Second, it lacks a standard action selection guideline for members. In current game model, network members are modeled as rational and thus naturally selfish individual; they will make any efforts to maximize their payoffs. It is reasonable to assign a higher payoff to cooperation action in order to stimulate members taking cooperative action. In fact, in realistic network, to malicious members, they may more likely launch attacks to get a higher illegal payoff from network collapse. Since large scale MANETs are usually applied for harsh environment monitoring or military detecting, a high malicious ratio is an outstanding feature. As a matter of fact, besides payoff frame reflecting realistic network, an action selection guideline is also needed as scoring system assisting cooperative stimulation scheme based on game theory. Thirdly, the current scheme lacks evolutionary analysis for strategy space using game theory. In fact, the strategy space taken by members is not invariant as the game runs. It may be evolved according to long-term expected benefit or suffering from intrusion of instable strategy. Thus it is necessary to evaluate the evolutionary and convergence performance of each strategy space when using game theory to stimulate the member's cooperative action. Last but not least, in large scale MANETs, considering the own inherent natures of network, the scheme has to adapt dynamic property as well as propagation error when updating strategy and payoff set.

Aiming at the previous issues, in this paper, we model the transmission process as an evolutionary game and propose a trust cooperative stimulation scheme based on it; our main contributions are summarized as follows.

(1) We formulate a transmission evolutionary game defining an abstract concept of level classification in strategy space based on network risk analysis, which can cover the member's possible network actions under complicated compound attacks in large scale MANETs to enhance the completeness of strategy and payoff space.

(2) We construct a trust-preferential expected action space as strategy selection guideline for members through mapping trust management to game theoretical cooperative stimulation, which realizes effective constraint for malicious members obtaining illegal payoffs.

(3) We quantitatively analyze the stability and convergence property between our trust-preferential expected action space and payoff maximization-preferential optimum action space and then provide the sufficient and necessary numerical conditions, which can incent members to cooperate with each other.

(4) We introduce the mobility probability of members and information propagation error into the formulation of our scheme and make it approach to the realistic large scale MANETs.

2. Related Works

In the literature there are many papers proposing various methods for stimulating members' cooperation in self-organization networks which can be summarily classified into two schemes: (1) price-based schemes and (2) trust-based schemes. Price-based schemes use the tamper-proof hardware or central billing services to encourage cooperation by rewarding price credits to the cooperative nodes. For example, a cooperation stimulation scheme proposed in [8] employed a virtual currency named Nuglets as price payment for cooperative transmission, later; it was improved in [9] by using price counters. Although price-based schemes can effectively stimulate cooperation among selfish members, the requirement of tamper-proof hardware or central billing service inevitably limits their applications. What is more, the existing works are only fit for traditional multi-hop networks. The price-based schemes depend on end-to-end connections to determine how many prices each member should receive. In MANETs, since end-to-end paths are not guaranteed at all, the existing price-based schemes cannot be used. Regarding this issue, the second method to stimulate cooperation is to adopt trust-based schemes with necessary monitoring, such as CORE [10], CONFIDANT [11], and ARCS [12]. They usually rely on observing the actions of neighbor members and then use mathematical methods such as Dempster-Shafer belief theory to compute the incorporating second-hand information (reports by other nodes) to create a reputation score of members. The trust/reputation score is used for stimulating cooperation because the detected non-cooperative members will be assigned a low score as a penalty to be forced out of the network. However, in realistic dynamic environment of MANETs, for a distributed trust form, the deviating actions of a non-cooperative member are more difficult to be monitored and detected by other members since the connections with the same members are occasional.

Game theory has been widely applied to design and analyze stimulation schemes for wireless network recently. For example, in [13], a Worst Behavior Tit-for-Tat (WBTFT) incentive strategy is proposed to stimulate cooperation at the desired cooperation state, and with perfect monitoring the conditions for the proposed strategy to be subgame perfect are analyzed. In [14], a cooperation stimulation scheme are proposed based on indirect reciprocity game for the scenario where the number of interactions between any pair of players is finite. For large scale wireless networks, Xiao et al. [15]

proposed a security system that applies the indirect reciprocity principle to combat attacks in wireless networks using the evolutionarily stable strategy concept of game theory. In [16], the authors investigated whether the cooperation among members can improve energy efficiency in ad hoc wireless networks using the behavior-tracking algorithm from game theory, and then the conclusion that the cooperation can reduce power wastage at the same time maximizing the delivery rate is proved.

In addition, further researches have also been made toward mathematically analyzing cooperative stimulation for self-organized wireless network (e.g., MANETs, wireless sensor networks, delay tolerant networks) by using game theory [17–23]. Zhao et al. [17] proposed a wage-based incentive mechanism for encouraging rational individuals to provide truthful feedbacks. The feedback reporting process in a reputation system was modeled as a reporting game. They also proposed a set of incentive compatibility constraint rules including participation constraints and self-selection constraints. Ze and Haiying [18] analyzed the underlying cooperation of the reputation systems, price-based systems, and a defenseless system through game theory. Based on the results, they proposed an integrated system with a higher performance in terms of the effectiveness of cooperation and selfish node detection. Li et al. [19] showed how game theory can be a tool to analyze the behaviors of every player in role-based trust framework. Considering two types of users, cooperative and malicious, they analyze the strategy sets and payoffs of trust domains and each type of users. Charles et al. [20] investigated when for each node it was cost-effective to freely participate in the security mechanism or protect its privacy according to its own belief in others. The game theoretic framework was used to model trust, and evolutionary game theory was used to capture the dynamic evolution of trust behavior in the network. Also, the studies of cooperative stimulation conditions under correlated equilibrium of coalitional game theoretic approach in ad hoc networks have also been issued in [21–23].

In most existing studies the modeled game theoretic cooperative stimulation shows a promising incentive effect in the network with a small-range, static topology paradigm. Designing a scheme using game theory towards large scale MANETs is the purpose of this paper. The major difference between this paper and current studies is as followss: (1) we formulate a transmission evolutionary trust game constructing the complete strategy and payoff space to cover the member's possible network actions under complicated compound attacks in large scale MANETs; (2) we separate trust-preferential strategy from payoff-maximization frame, which can be used as strategy selection guideline by means of numeric mapping technology. It can effectively resist malicious members obtaining illegal payoffs from attacking network; (3) we quantitatively analyze the stability and convergence property of the proposed game model in detail, and then provide the sufficient and necessary numerical conditions which can incent members to cooperate with each other; furthermore, (4) we introduce the mobility probability parameters and information propagation error into the formulation of our scheme and make it approach to the realistic large scale MANETs.

3. Scheme Model

3.1. Information Transmission Scenario. We design the scheme model used in homogenous mobile ad hoc networks consisting of N homogenous randomly mobile nodes. For the convenience of discussing, we make the following assumptions: (1) the underlying channel model adopts disk model in order to abstract asymmetrical information away from the complicated properties of RF. (2) As for information transmitter, the probability that the arbitrary other nodes move away from its communication range or the newly nodes accesses into its communication range is ω.

In our model a typical information transmission scenario is composed of one transmitter, one intended receiver, and several information relay nodes. The transmitter generates the information and sends it to the intended receiver with the help of relay nodes. The node within the communication range of the transmitter can be selected to the relay nodes if it has the optimal link state described by medium congestion level, robustness of route protocol, mobile state prediction, and the health degree of itself. In a similar way, the relay node selects the next relay node and the link route according to the same principle until the generated information successfully gets to the intended receiver. For simplicity of mathematical expression, each node in our model becomes the relay node with the probability μ. In this paper, we use the symbol Φ to indicate whether a node is selected to the relay node. More specifically, $\Phi = 1$ indicates that the node becomes the relay node on the information transmission path while $\Phi = 0$ indicates that this node is only in charge of monitoring the behavior of other nodes and computing their trust value. Then these trust values will be exchanged by neighbor nodes via the cryptographic secure channel.

3.2. Trust Management. In MANETs trust management can effectively resist various internal attacks conducted by compromised internal members. In this paper, to stimulate the cooperation behavior among nodes we design a game model in order to enhance the information transmission throughput which needs a scoring system to evaluate such behaviors. Hence we adopt the trust management to design the scoring system.

More specifically, one node monitors and records various communication factors (i.e., transmission rate, forwarding rate, etc.). Then by means of robust mathematical calculation method (i.e., Bayesian interference, DS evidence theory, fuzzy logic classification, etc.), the quantifiable trust value of each supervised members can be obtained by trust manager via the cryptographic secure channel, which can be regarded as the members' credible extent.

3.3. Game Model. In this paper, we model the aforementioned information transmission process as a dynamic Bayesian game among all nodes in MANETs. During this game all players who participate in this game make efforts to maximize their own payoffs. That is to say, all nodes in our

game model are deemed as rational players related to game theory.

More specifically, in MANETs, there are three kinds of players which amount to $n + 2$ members engaged in our game: a transmitter, an intended receiver, and n participants within the transmitter's communication range of the game. At time t, participant p_n selects one action according to the rational principle from our designed complete strategy space to play the game, denoted as $a_{p_n}[t]$. Based on the analysis of [24], current malicious nodes in large scale MANETs have gradually changed conventional pure attack modes into purposive strategic attack modes, such as selective forwarding attack, frame flooding, spoof, selfish packet dropping attack, and black hole and Sybil attack. In this paper, unlike present research works which simply build the behavior space composed of cooperative and uncooperative actions, we consider a comprehensive situation of attacks in MANETs. In order to stimulate cooperation among nodes in MANETs when nodes are at risk of aforementioned purposive strategic attacks, we classify and abstract current network attacks into multiple levels and then put them in the behavior space of the game model. The strategy space of our game model is shown in Table 1, where $\{A_1, A_2, A_3, \ldots, A_L\}$ denotes the attack set classified and abstracted by the risk level. Note that in large scale MANETs the specific attack form corresponding to certain risk level is changing when the network operation goal is different. For example, with regard to the information monitoring network, enhance the energy utility which prolongs the network life time is the most important thing to be considered. Thus the frame flooding attack or the relevant combination of attacks which deteriorate the energy performance should be identified as the high-level risk attacks; another, as for the network that emphasizes the data transmission rate and throughput such as media ad hoc network, black hole and Sybil attack or the relevant combination of attacks which deteriorate the QoS performance should be identified as the high-level risk attacks. Besides, the elements in behavior set $\{V, C\}$ denote the action taken by the participant who violates and complies with the cooperation rule respectively. More specifically, to the relay node, $\{V, C\}$ denotes {selfish, forward}. On the contrary, to the monitoring node, $\{V, C\}$ denotes {forward, monitor}.

The payoff frame is an important factor to model as well as analyze the behavior of players. In our game model after taking one certain strategy from behavior space to participate in the game, each participant obtains a real-time payoff $R_{a_{p_n}[t]}[\Phi]$ with relay indicator Φ, where $a_{p_n}[t] \in \{A_1, A_2, \ldots, A_L, V, C\}$. Particularly, as for the information transmitter, at time t every other player that takes one action $a_{p_n}[t]$ will produce one instant payoff to it, denoted as $R_{a_{p_n}[t]}[T]$. We use $[T]$ to stand for the payoff that belongs to the transmitter. Generally speaking, the payoff is composed of two parts; one is the gain from action, and the other is the cost when taking this action. The value obtained by subtracting the cost from the gain means the payoff of taking this action. In this paper, positive payoff means that the participant earns profit from action, while negative payoff means that the participant loses some resources such as energy, throughput. More specifically

Table 1: Strategy space.

Action indicator	Meaning
A_1	Level-1 attack
A_2	Level-2 attack
A_3	Level-3 attack
⋮	⋮
A_L	Level-L attack (the highest risk level attack)
V	Violation of cooperation rule
C	Cooperation

in our game model, with regard to the participant who takes the cooperation behavior C, both information forwarding ($\Phi = 1$) and channel monitoring ($\Phi = 0$) inevitably consume its own resources; hence $R_C[\Phi = 1] < R_C[\Phi = 0] \leq 0$. On the other hand, the information transmitter would earn a profit after taking action C; thus $R_C[T] > 0$. Next, with regard to the participants who take action that attacks the network or violates the cooperation rule (denoted as the malicious behavior set $M = \{a_{p_n}[t] \mid a_{p_n}[t] \in (A_1, A_2, \ldots, A_L, V)\}$), they can earn profits from these actions, so $R_M[\Phi] \geq 0$. In this situation the information transmitter's benefit is threatened which leads to a negative instant payoff, $R_M[T] \leq 0$. In addition, according to a wide range of attacks in MANETs classified by multiple risk levels in our model, the instant payoff satisfies the following conditions for both the transmitter and the game members:

$$\begin{aligned} &R_{A_L}[\Phi] \geq R_{A_{L-1}}[\Phi] \geq \cdots \geq R_{A_2}[\Phi] \\ &\qquad \geq R_{A_1}[\Phi] \geq 0 \geq R_V[\Phi] \geq R_C[\Phi], \\ &R_C[T] \geq 0 \geq R_V[T] \geq R_{A_1}[T] \\ &\qquad \geq R_{A_2}[T] \geq \cdots \geq R_{A_{L-1}}[T] \geq R_{A_L}[T]. \end{aligned} \tag{1}$$

Note that in our modeled transmission game for MANETs, the strict transmission constraint condition is used. More specifically, the information successfully reaches the intended receiver only if all the participants on transmission path comply with the cooperation rule. Based on this condition, at time t the total instant payoff for transmitter is the minimum of all the obtained instant payoffs from other nodes, denoted as $P^T_{a_{p_i}[t]}[t] = \min_{i=1}^{n} R_{a_{p_i}[t]}[T]$. Similar to the game participant p_n, its total instant payoff can be expressed as $P^{\Phi}_{p_n, a_{p_n}[t]}[t] = R_{a_{p_n}[t]}[\Phi]$.

For convenience of understanding the game model, Table 2 lists the symbols as well as their meanings used in this paper.

4. Trust Cooperative Stimulation Scheme

In this paper we design a cooperative stimulation scheme for large scale MANETs combining game theory and trust management. On one hand, the equilibrium and stability condition of the aforementioned game model is deduced to grasp and predict the result through figuring out each node's optimal strategy. What is more, based on solution of game

Table 2: Symbols and notations.

Symbol	Meaning
N	The number of homogenous mobile members
ω	The probability that the member newly enters or moves away from communication range of transmitter
Φ	Information relay indicator
μ	The probability that member is selected as information relay node
$a_{p_n}[t]$	The action taken by game participant p_n at time t
L	Risk classification number of attacks
$R_{a_{p_n}[t]}[\Phi]$	The instant payoff obtained by participant p_n with relay indicator Φ after taking action $a_{p_n}[t]$ at time t
$P^T_{a_{p_n}[t]}[t]$	The total instant payoff obtained by transmitter after taking action $a_{p_n}[t]$ at time t
$P^{\Phi}_{p_n,a_{p_n}[t]}[t]$	The total instant payoff obtained by participant p_n with relay indicator Φ after taking action $a_{p_n}[t]$ at time t
T_{vector}	Trust vector
$\left[N_{i,j}\right]_{(L+2)\times(L+2)}$	Action decision matrix (used as action selection guideline)
$a^*_{(L+2)\times(L+2)}$	Expected action space
$\Gamma = \left(\Gamma_{A_1}, \Gamma_{A_2}, \ldots, \Gamma_V, \Gamma_C\right)$	Time factor vector
$I^{i,j}_{T_{\text{vector}}}$	The instant trust vector
Θ	The trust propagation matrix
χ_i (or ε)	The systematic probability of correctly recognizing trust level i
δ	The probability that the member becomes the information transmitter
$\widetilde{A} = \left[\widetilde{a}_{i,j}\right]_{(L+2)\times(L+2)}$	Optimum action space
$P_{i,j}$	The anticipated maximum payoff obtained by participant with trust level i towards the member with trust level j
$\text{TT}^{a_{p_n}=k}_{i,j}$	Trust transfer vector
$P^{a_{i,j}}_{i,j}$	The anticipated payoff of participant taking action $a_{i,j}$

model, the mathematical relationship between payoff and statistical parameters can be used to guide (stimulate) members to choose cooperation with each other in order to resist selfish behavior or even high-level risk attacks. On the other hand, by means of trust management mechanism, a uniform frame concerning trust distribution, trust update and behavior selection is constructed by the whole members in the network.

In this section, we mainly introduce the behavior selection frame based on trust management (i.e., each member in MANETs takes action according to its trust value) and propose a trust cooperative stimulation scheme. Trust value records the member's quantified credibility (the higher the member's trust value is, the reliable the member is), and its calculation, distribution, and updating must be accomplished via the cryptographic secure channel. Without permission, members cannot clear or temper trust value optionally. All these paradigms of trust scheme indicate that it is fit for large scale MANETs since the cooperation interactive between two nodes only related to their recorded current trust value. For example, without prior knowledge about whether you cooperated with me before, I can decide to take cooperative action with you only if you have an acceptable trust value for me. It differs from current research about cooperative stimulation based on first-hand in MANETs.

Table 3: Action-trust based mapping rule.

Trust value	Action space	Numeric indicator of trust level
$\frac{L-1}{L+2} \le T_{\text{value}} < \frac{L}{L+2}$	A_1	L
$\frac{L-2}{L+2} \le T_{\text{value}} < \frac{L-1}{L+2}$	A_2	$L-1$
$\frac{L-3}{L+2} \le T_{\text{value}} < \frac{L-2}{L+2}$	A_3	$L-2$
$\vdots$	$\vdots$	$\vdots$
$\frac{1}{L+2} \le T_{\text{value}} < \frac{2}{L+2}$	A_L	1
$\frac{L}{L+2} \le T_{\text{value}} < \frac{L+1}{L+2}$	V	$L+1$
$\frac{L+1}{L+2} \le T_{\text{value}} < 1$	C	$L+2$

The symbol in action space (i.e., A_3, C, etc.) corresponding to numerical indicator in the third column can be used to indicate trust level of the member. More specifically, the greater the member's numerical indicator, the higher the trust value it has, the more reliable it belongs to, hence the higher-level cooperative action can be serviced by other members in MANETs.

The trust cooperative stimulation scheme contains trust evaluation which is based on our previous works [5], action-trust-based mapping method, game action decision principle, and trust updating frame. In this frame, the higher the member's trust value is, the more likely this member is stimulated to take cooperative action (i.e., C), which results in spreading the cooperation behavior to the whole network. On the contrary, if the member takes a high-level risk attack behavior for obtaining a short-term positive payoff, its trust value will be rapidly declined which causes cooperative service rejection in terms of scheme rule.

Recall that the behavior space for each node participating in the transmission game is $\{A_1, A_2, \ldots, A_L, V, C\}$ amounting to $L + 2$ elements. To combine trust level with game action, we classify the trust value (ranging from 0 to 1 usually) into $L + 2$ trust levels and map each level into the behavior space $\{A_1, A_2, \ldots, A_L, V, C\}$ linearly as Table 3, which we call action-trust-based mapping method. Consequently, the higher the member's trust level is, the more likely it can get a higher level cooperative action from other members. Note that by means of this mapping method, we can use element in action space to indicate node's trust value (also trust level).

According to original mapping, each node is assigned an original trust level $j \in \{A_1, A_2, \ldots, A_L, V, C\}$ and a trust

vector $T_{\text{vector}} = (T_{v_{A_1}}, T_{v_{A_2}}, \ldots, T_{v_V}, T_{v_C})^T$, where T_{v_j} denotes the probability that the node's trust level is T_{v_j} ($0 \le T_{v_j} \le 1$, $\sum_{j=1}^{L+2} T_{v_j} = 1$).

After trust mapping, game action decision principle and trust updating frame are the two important parts affecting the performance of the trust cooperative stimulation. More specifically, at single time moment the game action decision principle is designed according to interaction between transmitter with trust level i and participant with trust level j shown as matrix $[N_{i,j}]_{(L+2)\times(L+2)}$ in the following, where element $N_{i,j}$ denotes the assigned trust level of the participant who takes action i ($i \in \{A_1, A_2, \ldots, A_L, V, C\}$) towards the transmitter with trust level j and Φ ($\Phi = 0, 1$) is the relay indicator of the participant:

$$N(\Phi)_{(L+2)\times(L+2)} = \Phi \begin{bmatrix} A_1(L) & A_1(L) & \cdots & A_1(L) & A_1(L) & A_1(L) \\ A_2(L-1) & A_2(L-1) & \cdots & A_2(L-1) & A_2(L-1) & A_2(L-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_L(1) & A_L(1) & \cdots & A_L(1) & A_L(1) & A_L(1) \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & V(L+1) \\ A_1(L) & A_2(L-1) & \cdots & A_L(1) & V(L+1) & C(L+2) \end{bmatrix} + (1-\Phi) \begin{bmatrix} A_1(L) & A_1(L) & \cdots & A_1(L) & A_1(L) & A_1(L) \\ A_2(L-1) & A_2(L-1) & \cdots & A_2(L-1) & A_2(L-1) & A_2(L-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_L(1) & A_L(1) & \cdots & A_L(1) & A_L(1) & A_L(1) \\ V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & V(L+1) \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \end{bmatrix}. \tag{2}$$

From this matrix, the game action decision principle can be explained that the node could take cooperative actions with its neighbor to obtain a higher trust level striving to restrain the attack actions. Generally speaking, at one time moment, taking action C can obtain a highest instant trust level (i.e., C) while taking action C can inevitably obtain a lower instant trust level (i.e., type A).

If the node takes actions for maintaining own high trust level in the game, we can intuitively get the expected action denoted as matrix $a^*_{(L+2)\times(L+2)}$:

$$a^*_{(L+2)\times(L+2)} = \Phi \begin{bmatrix} V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & C(L+2) \\ V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & C(L+2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & C(L+2) \\ V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & C(L+2) \\ V(L+1) & V(L+1) & \cdots & V(L+1) & V(L+1) & C(L+2) \end{bmatrix} + (1-\Phi) \begin{bmatrix} C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \\ C(L+2) & C(L+2) & \cdots & C(L+2) & C(L+2) & C(L+2) \end{bmatrix}, \tag{3}$$

where $a^*_{i,j}$ denotes the participant who takes action i ($i \in \{A_1, A_2, \ldots, A_L, V, C\}$) towards the transmitter with trust level j and the Φ ($\Phi = 0, 1$) is the relay indicator of the participant.

From matrix $a^*_{(L+2)\times(L+2)}$, for the view of maintaining a higher trust level in this game, this expected strategy space can effectively encourage participants to take cooperative actions.

Recall that each newly node participating in the game would be assigned a trust vector. Suppose that a new node has a good intention to cooperate with each other; it could be assigned a trust vector $T_{\text{vector}} = (0, 0, \ldots, 0, 1)^T$. At time $t+1$ the action i ($i \in \{A_1, A_2, \ldots, A_L, V, C\}$) taken by participant is recorded by monitoring node. Then according to game action decision principle $[N_{i,j}]_{(L+2)\times(L+2)}$, relay indicator Φ, and the trust level j of information transmitter, the participant would

obtain an instant trust level denoted as $I_T^{i,j} = N_{i,j}(\Phi)$. The trust updating process is triggered as shown in Figure 1.

From Figure 1, the participant's trust vector at time $t+1$ is expressed as

$$T_{\text{vector}}(t+1) = \Theta\left(\Gamma_{N_{i,j}(\Phi)} T_{\text{vector}}(t) + \left(1 - \Gamma_{N_{i,j}(\Phi)}\right) I^{i,j}_{T_{\text{vector}}}\right). \tag{4}$$

It is composed of three parts: the first part is instant trust vector

$$I^{i,j}_{T_{\text{vector}}} = \underbrace{(0,\ldots,[1],\ldots,0)^T}_{\text{The corresponding vector position of } I_T(t+1)} \tag{5}$$

(it is extended by instant trust level $I_T^{i,j}$ at time $t+1$, that is, filling the vector's position corresponding to $I_T^{i,j} = N_{i,j}(\Phi)$ with numerical value 1, and other position with numerical value 0). The second part is time factor of the action taken at time $t+1$. In our game, considering the coupling degree between instant and the accumulated trust vector we define the time factor vector $\Gamma = (\Gamma_{A_1}, \Gamma_{A_2}, \ldots, \Gamma_V, \Gamma_C)$ depicting the coupling degree, and thus element $\Gamma_{N_{i,j}(\Phi)}$ is the active factor contributing to the updating process. Note that the greater the Γ is, the more likely the trust vector is to depend on the previous value and the fewer effects of instant value will be received. The third part is trust propagation matrix Θ whose role is to conquer behavior monitoring error and trust vector error by channel error via trust propagation. Specifically, Θ can be denoted as

$$\Theta_{(L+2)\times(L+2)} = \begin{bmatrix} \chi_L & \frac{1-\chi_L}{L+1} & \cdots & \cdots & \frac{1-\chi_L}{L+1} & \frac{1-\chi_L}{L+1} \\ \frac{1-\chi_{L-1}}{L+1} & \chi_{L-1} & \frac{1-\chi_{L-1}}{L+1} & \cdots & \cdots & \frac{1-\chi_{L-1}}{L+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1-\chi_1}{L+1} & \cdots & \frac{1-\chi_1}{L+1} & \chi_1 & \frac{1-\chi_1}{L+1} & \frac{1-\chi_1}{L+1} \\ \frac{1-\chi_{L+1}}{L+1} & \cdots & \cdots & \frac{1-\chi_{L+1}}{L+1} & \chi_{L+1} & \frac{1-\chi_{L+1}}{L+1} \\ \frac{1-\chi_{L+2}}{L+1} & \cdots & \cdots & \cdots & \frac{1-\chi_{L+2}}{L+1} & \chi_{L+2} \end{bmatrix}, \tag{6}$$

where χ_i denotes the systematic probability of correctly recognizing trust level i.

In this section, we propose a trust cooperative stimulation scheme based on trust-game mapping idea. An expected action matrix $a^*_{(L+2)\times(L+2)}$ (trust-preferential strategy) and the trust updating frame are deduced to guide members tending to cooperation. For the view of maintaining a higher trust level in this game, this scheme can effectively encourage participants to take cooperative actions.

5. Game Theoretic Analysis

In the large scale MANETs, members who take the expected action $a^*_{(L+2)\times(L+2)}$ to select cooperative behavior can maintain a higher trust level (trust-preferential strategy) with themselves. Hence even if the network topology changes dramatically, nodes can also continue to obtain high-level cooperative network services in the new area by means of their high trust level. However in game theory, optimum actions refer to the strategy that receives a highest payoff for all the players. In this section, we mainly study whether the nodes can take the expected trust-preferential strategy also obtain a higher payoff after long-time running of the game? In addition, we adopt the evolutionarily game idea to analyze under which numerical condition can the expected trust-preferential strategy evolve to the payoff-preferential strategy (optimum strategy), that is, evolutionarily stable strategy (ESS).

5.1. Evolutionary Game Theory. Evolutionary game theory provides a new angle of view to research the network cooperative stimulation scheme. It well overcomes the difficulties about rational hypothesis and multiple equilibriums in classical game theory. What is more, it can obtain more accurate results than traditional theory by using evolutionary game theory to research network security and can realistically analyze and explain cooperative motivation. To the best of our knowledge, introducing evolutionary game theory to study the mechanism of cooperative stimulation is a method innovation in MANETs.

In evolutionary game model, if most of the members take ESS, other parts of members' alternative strategies cannot invade the ESS. First we use the expected trust-preferential action as the original strategy in the game, and then the strategy starts to evolve in terms of the payoff maximum criteria which can deduce the optimum strategy of the game. More specifically, at the original time moment 1, game participant p_n takes the action according to the game action

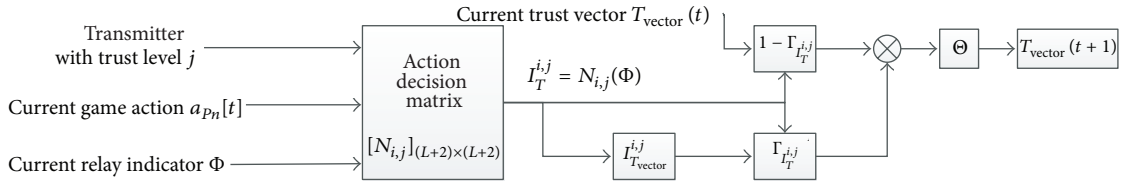


Figure 1: Trust updating process.

decision matrix $[N_{i,j}]_{(L+2)\times(L+2)}$ and the expected action matrix $a^*_{(L+2)\times(L+2)}$, and the evolutionary process is triggered. At time $t+1$, the probability of taking action $a_{p_n}[t+1] = i \in \{A_1, A_2, \ldots, A_L, V, C\}$ for node p_n is denoted as

$$\Pr\left\{a_{p_n}[t+1] = i\right\} = \frac{\Pr\left\{a_{p_n}[t] = i\right\} P_{p_n,i}[t]}{\sum_{j=1}^{L+2} \Pr\left\{a_{p_n}[t] = j\right\} P_{p_n,j}[t]}, \quad (7)$$

where $P_{p_n,i}[t]$ denotes the instant payoff obtained by participant p_n who takes the action i at time t. From (7) we can solve the ESS (i.e., optimum strategy space) as well as the corresponding stable trust vector $\widetilde{T}_{\text{vector}}$ of this evolutionary game when taking the expected trust-preferential strategy as original strategy in MANETs.

5.2. Optimum Action Space. We first define matrix $\widetilde{A} = [\widetilde{a}_{i,j}]_{(L+2)\times(L+2)}$ as optimum action space, where the element $\widetilde{a}_{i,j} \in \{A_1, A_2, \ldots, A_L, V, C\}$ denotes the instant optimum action taken by participant with trust level i towards the member with trust level j. Note that taking this instant optimum action the participant should have obtained an anticipated maximum payoff. We use symbol $P_{i,j} = \max_{1\le a_{i,j}\le(L+2)} P_{i,j}^{a_{i,j}} = P_{i,j}^{\widetilde{a}_{i,j}}$ to denote this anticipated maximum payoff. Consequently, the optimum action should satisfy the following expression:

$$\widetilde{a}_{i,j} = \arg \max_{1\le a_{i,j}\le(L+2)} P_{i,j}^{a_{i,j}}. \quad (8)$$

In our game model, considering the trust updating frame, the trust level of each participant may be transferred at different time moment. Thus we must evaluate the evolutionary optimum action space under the influence of the trust updating process. Recall that in our game the probability of participant selected to the information relay node is μ. Suppose that each behavior in set $\{A_1, A_2, \ldots, A_L, V, C\}$ has the same time factor Γ. We define trust transfer vector $\mathrm{TT}_{i,j}^{a_{p_n}=k}$ denoting the transfer probability vector after the participant with trust level i takes action k towards the participant with trust level j. Based on trust updating process shown in Figure 1, we can calculate the $\mathrm{TT}_{i,j}^{a_{p_n}=k}$ by the following expression:

$$\begin{aligned}\mathrm{TT}_{i,j}^{a_{p_n}=k} &= \left[t_{i,j}^{a_{p_n}=k}[L], t_{i,j}^{a_{p_n}=k}[L-1], \ldots, t_{i,j}^{a_{p_n}=k}[1], \right.\\ &\qquad \left. t_{i,j}^{a_{p_n}=k}[L+1], t_{i,j}^{a_{p_n}=k}[L+2]\right]^T\\ &= \Theta\left(\Gamma I_{T_{\text{vector}}}^{i,j} + (1-\Gamma)\right.\\ &\qquad \left.\times\left(\mu I_{T_{\text{vector}}}^{N_{k,j}(\Phi=1),j} + (1-\mu) I_{T_{\text{vector}}}^{N_{k,j}(\Phi=0),j}\right)\right),\end{aligned} \quad (9)$$

where vector element $t_{i,j}^{a_{p_n}=k}[l]$ denotes the probability that the trust level of the participant taking action k towards the participant with trust level j has transferred from i to l. Formula (9) takes behavior time factor and relay factor into account, which is embodied by the application of Γ and μ, respectively.

Second, we solve the game payoff $P_{i,j}^{a_{i,j}}$ obtained by participant p_n with trust level i towards the participant taking optimum action $\widetilde{a}_{j,i}$ with trust level j. If p_n takes action $a_{i,j}$, its instant payoff at time t can be denoted as $\mu P_{p_n,a_{i,j}}^{\Phi=1}[t] + (1-\mu)P_{p_n,a_{i,j}}^{\Phi=0}[t]$. In addition, consider the dynamics of MANETs, suppose that the probability of the node staying in the local area or moving to the new area is ω, and the stable trust vector $\widetilde{T}_{\text{vector}}$ does not change as time goes on; we can calculate the instant payoff obtained by nonrelay participant p_n as

$$\mu P_{p_n,a_{i,j}}^{\Phi=1}[t] + (1-\mu) P_{p_n,a_{i,j}}^{\Phi=0}[t] + \omega \sum_k \sum_l t_{i,j}^{a_{i,j}} \widetilde{T}_{v_l} P_{k,l}. \quad (10)$$

On the other hand, if p_n faces the information transmitter, similarly, its instant payoff can be calculated by

$$(1-\mu) P_{\widetilde{a}_{j,i}(\Phi=0)}^{T}[t] + \mu P_{\widetilde{a}_{j,i}(\Phi=1)}^{T}[t] + \omega \sum_l \widetilde{T}_{v_l} P_{k,l}. \quad (11)$$

To sum up, we define the probability that the node becomes the information transmitter in the game is δ consequently the game payoff $P_{i,j}^{a_{i,j}}$ can be expressed as

$$\begin{aligned}P_{i,j}^{a_{i,j}} &= (1-\delta)\left(\mu P_{p_n,a_{i,j}}^{\Phi=1}[t] + (1-\mu) P_{p_n,a_{i,j}}^{\Phi=0}[t]\right.\\ &\qquad \left. + \omega \sum_k \sum_l t_{i,j}^{a_{i,j}} \widetilde{T}_{v_l} P_{k,l}\right)\\ &\quad + \delta\left((1-\mu) P_{\widetilde{a}_{j,i}(\Phi=0)}^{T}[t] + \mu P_{\widetilde{a}_{j,i}(\Phi=1)}^{T}[t]\right.\\ &\qquad \left. + \omega \sum_l \widetilde{T}_{v_l} P_{k,l}\right).\end{aligned} \quad (12)$$

Thirdly, we give the expression of stable trust vector $\widetilde{T}_{\text{vector}}$ when the game evolves to the ESS. According to trust updating frame, when taking probability μ and relay indicator Φ of the participant p_n into account, the stable trust vector $\widetilde{T}_{\text{vector}}$ can be expressed by

$$\widetilde{T}_{\text{vector}} = \Theta\left(\Gamma\widetilde{T}_{\text{vector}} + (1-\Gamma)\left(\mu\begin{bmatrix}\sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=1)=L}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=1)=L-1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \vdots\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=1)=1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=1)=L+1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=1)=L+2}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\end{bmatrix} + (1-\mu)\begin{bmatrix}\sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=0)=L}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=0)=L-1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \vdots\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=0)=1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=0)=L+1}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\\ \sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi=0)=L+2}\widetilde{T}_{v_i}\widetilde{T}_{v_l}\end{bmatrix}\right)\right), \tag{13}$$

where $\sum_{i=1}^{L+2}\sum_{l,N_{\widetilde{a}_{j,l},l}(\Phi)=k}\widetilde{T}_{v_i}\widetilde{T}_{v_l}$ denotes the probability that the trust level of participant (with relay indicator Φ) who takes optimum action $\widetilde{a}_{i,j}$ transfers to level k. Based on previous analysis, the optimum action space has been modeled into a Markov decision process.

Combining (9), (12), and (13), the proposed optimum action space $\widetilde{a}_{i,j}$ and its corresponding stable trust vector $\widetilde{T}_{\text{vector}}$ of the evolutionary game can be solved by iterative numerical method.

5.3. Relationship between Optimum and Expected Action and Its Convergence Condition. So far by means of evolutionary game theory we have deduced the ESS of the payoff-preferential strategy (optimum strategy) when taking the expected trust-preferential strategy as the original dominant action. In this section, we continue to study the convergence condition of this game, which is depicted as the paradigm that the ESS of the game converges to the original strategy. Meanwhile, we deduce and give the numerical relationship between optimum and expected action and its convergence condition.

If our evolutionary game converges, the original dominant strategy $a^*_{i,j}$ will evolve to be the optimum strategy $\widetilde{a}_{i,j}$; that is, $\widetilde{a}_{i,j} = a^*_{i,j}$. According to (3), we have

$$\begin{aligned}\left[\widetilde{a}_{i,j}(\Phi)\right]_{(L+2)\times(L+2)} &= \begin{bmatrix}\widetilde{a}_L(\Phi)\\ \widetilde{a}_{L-1}(\Phi)\\ \vdots\\ \widetilde{a}_1(\Phi)\\ \widetilde{a}_{L+1}(\Phi)\\ \widetilde{a}_{L+2}(\Phi)\end{bmatrix} = \begin{bmatrix}\widetilde{a}_{i,L}(\Phi)\\ \widetilde{a}_{i,L-1}(\Phi)\\ \vdots\\ \widetilde{a}_{i,1}(\Phi)\\ \widetilde{a}_{i,L+1}(\Phi)\\ \widetilde{a}_{i,L+2}(\Phi)\end{bmatrix}\\ &= \Phi\begin{bmatrix}V(L+1)\\ V(L+1)\\ \vdots\\ V(L+1)\\ V(L+1)\\ C(L+2)\end{bmatrix} + (1-\Phi)\begin{bmatrix}C(L+2)\\ C(L+2)\\ \vdots\\ C(L+2)\\ C(L+2)\\ C(L+2)\end{bmatrix}.\end{aligned} \tag{14}$$

For the convenience of deducing, suppose that all the probability of correctly recognizing trust level in trust propagation matrix Θ is the same, and $\chi_i = \varepsilon$; we also have

$$\Theta_{(L+2)\times(L+2)} = \begin{bmatrix}\varepsilon & \frac{1-\varepsilon}{L+1} & \frac{1-\varepsilon}{L+1} & \cdots & \frac{1-\varepsilon}{L+1}\\ \frac{1-\varepsilon}{L+1} & \varepsilon & \frac{1-\varepsilon}{L+1} & \cdots & \frac{1-\varepsilon}{L+1}\\ \frac{1-\varepsilon}{L+1} & \frac{1-\varepsilon}{L+1} & \varepsilon & \cdots & \frac{1-\varepsilon}{L+1}\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ \frac{1-\varepsilon}{L+1} & \frac{1-\varepsilon}{L+1} & \frac{1-\varepsilon}{L+1} & \cdots & \varepsilon\end{bmatrix}. \tag{15}$$

Proposition 1. *Given the trust level* $1 \le j \le (L+2)$ *and* $1 \le m \le (L+1)$*, the following expression is obtained:*

$$P_{(L+2),j} - P_{m,j} = \frac{\delta\mu\left(P^T_{L+2} - P^T_{L+1}\right)(L+1)}{(L+1)(1-\omega\delta) - \omega(1-\delta)\Gamma(L\varepsilon-1)}. \tag{16}$$

Proof. According to (9), (14), and (15), the difference of two trust transfer vectors from optimum action strategy is

$$\begin{aligned}\mathrm{TT}^{\widetilde{a}_{m,j}}_{m,j} - \mathrm{TT}^{\widetilde{a}_{(L+2),j}}_{(L+2),j} &= \mathrm{TT}^{\widetilde{a}_j}_{m,j} - \mathrm{TT}^{\widetilde{a}_j}_{(L+2),j}\\ &= \Theta\Gamma\left(I^m_{T_{\text{vector}}} - I^{(L+2)}_{T_{\text{vector}}}\right)\\ &= \Gamma\frac{(L\varepsilon-1)}{(L+1)}\left(I^m_{T_{\text{vector}}} - I^{(L+2)}_{T_{\text{vector}}}\right).\end{aligned} \tag{17}$$

Note that in the remainder of this paper we use $V[x]$ to denote the xth element of vector $\mathbf{V}$. Combining (12), (14), and (17), we can obtain

$$\begin{aligned}
&P_{m,j} - P_{(L+2),j} P_{m,j}^{\widetilde{a}_{(L+2),j}} - P_{(L+2),j}^{\widetilde{a}_{(L+2),j}} \\
&\quad = P_{m,j} - P_{(L+2),j} \\
&\quad = (1-\delta)\,\omega \sum_k \sum_l \left(\mathrm{TT}_{m,j}^{\widetilde{a}_j}[k] - \mathrm{TT}_{(L+2),j}^{\widetilde{a}_j}[k] \right) \widetilde{T}_{v_l} P_{k,l} \\
&\qquad + \delta \Bigg((1-\mu) \left(P_{\widetilde{a}_m(\Phi=0)}^T - P_{\widetilde{a}_{(L+2)}(\Phi=0)}^T \right) \\
&\qquad\quad + \mu \left(P_{\widetilde{a}_m(\Phi=1)}^T - P_{\widetilde{a}_{(L+2)}(\Phi=1)}^T \right) \\
&\qquad\quad + \omega \sum_l \widetilde{T}_{v_l} \left(P_{m,l} - P_{(L+2),l} \right) \Bigg) \\
&\quad = (1-\delta)\,\omega \sum_k \Gamma \frac{(L\varepsilon - 1)}{(L+1)} \left\{ \left(I_{T_{\mathrm{vector}}}^m - I_{T_{\mathrm{vector}}}^{(L+2)} \right)[k] \right\} \widetilde{T}_{v_l} P_{k,l} \\
&\qquad + \delta \left(\mu \left(P_{L+1}^T - P_{L+2}^T \right) + \omega \sum_l \widetilde{T}_{v_l} \left(P_{m,l} - P_{(L+2),l} \right) \right) \\
&\quad = \delta\mu \left(P_{L+1}^T - P_{L+2}^T \right) + \omega \left(\delta + (1-\delta)\,\Gamma \frac{(L\varepsilon - 1)}{(L+1)} \right) \\
&\qquad \times \sum_{l=1}^{L+2} \widetilde{T}_{v_l} \left(P_{m,l} - P_{(L+2),l} \right).
\end{aligned} \tag{18}$$

Because $(P_{m,l} - P_{(L+2),l})$ is independent with value i, the above (18) can be rewritten as

$$\begin{aligned}
P_{m,j} - P_{(L+2),j} &= \delta\mu \left(P_{L+1}^T - P_{L+2}^T \right) \\
&\quad + \omega \left(\delta + (1-\delta)\,\Gamma \frac{(L\varepsilon - 1)}{(L+1)} \right) \\
&\quad \times \left(P_{m,j} - P_{(L+2),j} \right).
\end{aligned} \tag{19}$$

Thus we can get the conclusion of Proposition 1. □

According to Proposition 1, we can infer that the participant with trust level m (less than the highest level $L+2$) would obtain a lower bound of the payoff after taking the expected trust-preferential strategy:

$$P_{(L+2),j} - \frac{\delta\mu \left(P_{L+2}^T - P_{L+1}^T \right)(L+1)}{(L+1)(1-\omega\delta) - \omega(1-\delta)\,\Gamma(L\varepsilon - 1)}. \tag{20}$$

The mechanism of cooperative stimulation by using the expected trust-preferential strategy is depicted as follows: no matter how low the participant's trust level is or which attribute (transmitter, receiver, relay node, and monitor) the participant belongs to, the expected trust-preferential strategy can stimulate it to cooperate with other members to obtain needed payoff to be serviced by the network.

Theorem 2. *The sufficient condition that the proposed evolutionary game model can converge is*

$$\begin{aligned}
&P_1^\Phi - P_{L+2}^\Phi \\
&\quad < \frac{\omega(1-\Gamma)(L\varepsilon - 1)\,\delta\mu \left(P_{L+2}^T - P_{L+1}^T \right)}{(L+1)(1-\omega\delta) - \omega(1-\delta)\,\Gamma(L\varepsilon - 1)}, \\
&\qquad\qquad \Phi = 0, 1, \quad \varepsilon > \frac{1}{L+2}.
\end{aligned} \tag{21}$$

Proof. First, consider the situation that the participant with trust level i faces the participant with the highest trust level $(L+2)$ and relay indicator $\Phi = 1$. To inquire the expected behavior matrix $a_{i,j}^*$, it should take the expected action shown as $\widetilde{a}_{L+2}(\Phi = 1) = L+2$. So for arbitrary action indicator $m < L+2$, we have $P_{i,L+2}^{a_{i,L+2}(1)=(L+2)} - P_{i,L+2}^{a_{i,L+2}(1)=m} > 0$. Again after combining (12), we can deduce the following equation:

$$\begin{aligned}
&P_{i,L+2}^{a_{i,L+2}(1)=(L+2)} - P_{i,L+2}^{a_{i,L+2}(1)=m} \\
&\quad = (1-\delta) \Bigg(\mu \left(P_{L+2}^{\Phi=1} - P_m^{\Phi=1} \right) \\
&\qquad + \omega \sum_k \sum_l \Big(t_{i,L+2}^{a_{i,L+2}(1)=(L+2)}[k] \\
&\qquad\quad - t_{i,L+2}^{a_{i,L+2}(1)=m}[k] \Big) \widetilde{T}_{v_l} P_{k,l} \Bigg) > 0.
\end{aligned} \tag{22}$$

Combining (2), (9), and (22), we have

$$\begin{aligned}
&\mu \left(P_{L+2}^{\Phi=1} - P_m^{\Phi=1} \right) \\
&\quad > \omega \sum_k \sum_l \Big(t_{i,L+2}^{a_{i,L+2}(1)=m}[k] \\
&\qquad - t_{i,L+2}^{a_{i,L+2}(1)=(L+2)}[k] \Big) \widetilde{T}_{v_l} P_{k,l} \\
&\quad = \omega \sum_k \sum_l \Theta(1-\Gamma) \Big\{ \mu \Big(I_{T_{\mathrm{vector}}}^{N_{m,(L+2)}(\Phi=1)} \\
&\qquad - I_{T_{\mathrm{vector}}}^{N_{(L+2),(L+2)}(\Phi=1)} \Big)[k] \Big\} \widetilde{T}_{v_l} P_{k,l} \\
&\quad = \omega(1-\Gamma)\,\mu \left(\frac{L\varepsilon - 1}{L+1} \right) \sum_l \widetilde{T}_{v_l} \left(P_{m,l} - P_{(L+2),l} \right).
\end{aligned} \tag{23}$$

Taking (17) into the above (23), we can obtain

$$\begin{aligned}
\left(P_m^{\Phi=1} - P_{L+2}^{\Phi=1} \right) &< \omega(1-\Gamma) \left(\frac{L\varepsilon - 1}{L+1} \right) \left(P_{(L+2),l} - P_{m,l} \right) \\
&= \frac{\omega(1-\Gamma)(L\varepsilon - 1)\,\delta\mu \left(P_{L+2}^T - P_{L+1}^T \right)}{(L+1)(1-\omega\delta) - \omega(1-\delta)\,\Gamma(L\varepsilon - 1)}.
\end{aligned} \tag{24}$$

Thus,

$$P_1^{\Phi=1} - P_{L+2}^{\Phi=1} < \frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^T - P_{L+1}^T\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}. \tag{25}$$

Because of $P_{L+2}^T \geq P_{L+1}^T$ and $P_{L+2}^{\Phi=1} \leq P_1^{\Phi=1}$, the above formula can also indicate that $\varepsilon > 1/(L+2)$.

Second consider the situation that the participant with trust level i faces the participant with the highest trust level $(L+2)$ and relay indicator $\Phi = 0$. Similar to the above case with $\Phi = 1$, according to (14), (12) can be rewritten as

$$\begin{aligned} &P_{i,L+2}^{a_{i,L+2}(0)=(L+2)} - P_{i,L+2}^{a_{i,L+2}(0)=m} \\ &\quad = (1-\delta)\Big(\left(1-\mu\right)\left(P_{L+2}^{\Phi=0} - P_m^{\Phi=0}\right) \\ &\qquad + \omega\sum_k\sum_l\Big(t_{i,L+2}^{a_{i,L+2}(0)=(L+2)}[k] \\ &\qquad\quad - t_{i,L+2}^{a_{i,L+2}(0)=m}[k]\Big)\widetilde{T}_{v_l}P_{k,l}\Big) > 0. \end{aligned} \tag{26}$$

Combining (2), (9), and (26), we have

$$\begin{aligned} &\left(1-\mu\right)\left(P_{L+2}^{\Phi=0} - P_m^{\Phi=0}\right) \\ &\quad > \omega\sum_k\sum_l\left(t_{i,L+2}^{a_{i,L+2}(0)=m}[k] - t_{i,L+2}^{a_{i,L+2}(0)=(L+2)}[k]\right)\widetilde{T}_{v_l}P_{k,l} \\ &\quad = \omega\sum_k\sum_l(1-\Gamma) \\ &\qquad \times\Big\{\left(1-\mu\right)\Theta\Big(I_{T_{\text{vector}}}^{N_{m,(L+2)}(\Phi=0)} \\ &\qquad\quad - I_{T_{\text{vector}}}^{N_{(L+2),(L+2)}(\Phi=0)}\Big)[k]\Big\}\widetilde{T}_{v_l}P_{k,l} \\ &\quad = \omega(1-\Gamma)\left(1-\mu\right)\left(\frac{L\varepsilon-1}{L+1}\right)\sum_l\widetilde{T}_{v_l}\left(P_{m,l} - P_{(L+2),l}\right). \end{aligned} \tag{27}$$

Because of $1 \leq m \leq L+1$, taking (17) into the above formula, we can obtain

$$\begin{aligned} \left(P_1^{\Phi=0} - P_{L+2}^{\Phi=0}\right) &< \omega(1-\Gamma)\left(\frac{L\varepsilon-1}{L+1}\right)\left(P_{(L+2),l} - P_{1,l}\right) \\ &= \frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^T - P_{L+1}^T\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}. \end{aligned} \tag{28}$$

To sum up, we can get the conclusion of Theorem 2 when the participant with trust level i faces the participant with the highest trust level $(L+2)$.

Next we have to consider the situation that the participant with trust level i faces the participant with the trust level $j < (L+2)$ and relay indicator $\Phi = 1$. In this situation, it should take the expected action shown as $\widetilde{a}_j(\Phi = 1) = L+1$. (1) For action indicator $m < L+1$, we have $P_{i,j}^{a_{i,j}(1)=(L+1)} - P_{i,j}^{a_{i,j}(1)=m} > 0$. According to (12), we can deduce that

$$\begin{aligned} &P_{i,j}^{a_{i,j}(1)=(L+1)} - P_{i,j}^{a_{i,j}(1)=m} \\ &\quad = (1-\delta)\Big(\mu\left(P_{L+1}^{\Phi=1} - P_m^{\Phi=1}\right) \\ &\qquad + \omega\sum_k\sum_l\Big(t_{i,j}^{a_{i,j}(1)=(L+1)}[k] \\ &\qquad\quad - t_{i,j}^{a_{i,j}(1)=m}[k]\Big)\widetilde{T}_{v_l}P_{k,l}\Big) > 0. \end{aligned} \tag{29}$$

Simplifying it, we have

$$\begin{aligned} &\mu\left(P_{L+1}^{\Phi=1} - P_m^{\Phi=1}\right) \\ &\quad > \omega\sum_k\sum_l\left(t_{i,j}^{a_{i,j}(1)=m}[k] - t_{i,j}^{a_{i,j}(1)=(L+1)}[k]\right)\widetilde{T}_{v_l}P_{k,l} \\ &\quad = \omega\sum_k\sum_l\Theta(1-\Gamma)\left\{\mu\left(I_{T_{\text{vector}}}^{N_{m,j}(\Phi=1)} - I_{T_{\text{vector}}}^{N_{(L+1),j}(\Phi=1)}\right)[k]\right\}\widetilde{T}_{v_l}P_{k,l} \\ &\quad = \omega(1-\Gamma)\mu\left(\frac{L\varepsilon-1}{L+1}\right)\sum_l\widetilde{T}_{v_l}\left(P_{m,l} - P_{(L+2),l}\right). \end{aligned} \tag{30}$$

Hence,

$$\begin{aligned} \left(P_m^{\Phi=1} - P_{L+1}^{\Phi=1}\right) &< \omega(1-\Gamma)\left(\frac{L\varepsilon-1}{L+1}\right)\left(P_{(L+2),l} - P_{m,l}\right) \\ &= \frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^T - P_{L+1}^T\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}. \end{aligned} \tag{31}$$

Because of $1 \leq m \leq L$, taking (17) into the above, we can obtain:

$$\begin{aligned} &P_1^{\Phi=1} - P_{L+1}^{\Phi=1} \\ &\quad < \frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^T - P_{L+1}^T\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}. \end{aligned} \tag{32}$$

In the same situation, (2) for action indicator $m = L+2$, the expected action participant should take is still shown as $\widetilde{a}_j(\Phi = 1) = L+1$. So we have $P_{i,j}^{a_{i,j}(1)=(L+1)} - P_{i,j}^{a_{i,j}(1)=(L+2)} > 0$ and the following expression:

$$\begin{aligned} &P_{i,j}^{a_{i,j}(1)=(L+1)} - P_{i,j}^{a_{i,j}(1)=(L+2)} \\ &\quad = (1-\delta)\Big(\mu\left(P_{L+1}^{\Phi=1} - P_{L+2}^{\Phi=1}\right) \\ &\qquad + \omega\sum_k\sum_l\Big(t_{i,j}^{a_{i,j}(1)=(L+1)}[k] \\ &\qquad\quad - t_{i,j}^{a_{i,j}(1)=(L+2)}[k]\Big)\widetilde{T}_{v_l}P_{k,l}\Big) > 0. \end{aligned} \tag{33}$$

Simplifying above formula, we have

$$\begin{aligned}
&\mu\left(P_{L+1}^{\Phi=1}-P_{L+2}^{\Phi=1}\right)\\
&>\omega\sum_{k}\sum_{l}\left(t_{i,j}^{a_{i,j}(1)=(L+2)}[k]-t_{i,j}^{a_{i,j}(1)=(L+1)}[k]\right)\widetilde{T}_{v_l}P_{k,l}\\
&=\omega\sum_{k}\sum_{l}\Theta(1-\Gamma)\left\{\mu\left(I_{T_{\text{vector}}}^{N_{(L+2),j}(\Phi=1)}-I_{T_{\text{vector}}}^{N_{(L+1),j}(\Phi=1)}\right)[k]\right\}\widetilde{T}_{v_l}P_{k,l}\\
&=\omega(1-\Gamma)\mu\left(\frac{L\varepsilon-1}{L+1}\right)\sum_{l}\widetilde{T}_{v_l}\left(P_{j,l}-P_{(L+2),l}\right).
\end{aligned}\tag{34}$$

Hence,

$$\begin{aligned}
\left(P_{L+2}^{\Phi=1}-P_{L+1}^{\Phi=1}\right)&<\omega(1-\Gamma)\left(\frac{L\varepsilon-1}{L+1}\right)\left(P_{(L+2),l}-P_{j,l}\right)\\
&=\frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^{T}-P_{L+1}^{T}\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}.
\end{aligned}\tag{35}$$

Last, consider the situation that the participant with trust level i faces the participant with the trust level $j<(L+2)$ and relay indicator $\Phi=0$. Similar to the above proof procedure, we can obtain the following two expressions:

$$\begin{aligned}
P_{1}^{\Phi=0}-P_{L+1}^{\Phi=0}&<\frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^{T}-P_{L+1}^{T}\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)},\\
P_{L+2}^{\Phi=0}-P_{L+1}^{\Phi=0}&<\frac{\omega(1-\Gamma)(L\varepsilon-1)\delta\mu\left(P_{L+2}^{T}-P_{L+1}^{T}\right)}{(L+1)(1-\omega\delta)-\omega(1-\delta)\Gamma(L\varepsilon-1)}.
\end{aligned}\tag{36}$$

To sum up, we considered all the situations and deduced (25), (28)–(36), which can support and verify the conclusion of Theorem 2. If the parameters of our proposed stimulation scheme are set to satisfy Theorem 2, the transmission game can get into ESS and the optimum strategy space converges to the expected strategy space which can also make members obtain maximum payoff. □

6. Simulation

6.1. Simulation Setup. In this part, we conduct extensive simulations to evaluate the network performance of our proposed stimulation model. All simulations are conducted in randomly generated MANETs. 5000 members are randomly deployed in a 10000 m × 10000 m region. The Medium-Access Control (MAC) layer protocol implements the IEEE 802.11 DCF with a four-way handshaking mechanism. The default link bandwidth is 2 Mb/s. DSR is adopted as route protocol. The maximum transmission range is 100 m. In our simulated MANET, each node is moving according to the random waypoint model: a node randomly chooses a destination within the circle and moves forward to the destination at a velocity uniformly chosen in 0.5 m/s, 2.5 m/s. When arriving at the destination, the node will choose a new location and a new speed to move on.

Table 4 lists the default settings of stimulation scheme.

Table 4: Default parameter settings.

Parameter	Default value setting
N	5000
ω	0.4
μ	0.5
δ	0.25
L	3
$\Gamma=(\Gamma_{A_1},\Gamma_{A_2},\ldots,\Gamma_V,\Gamma_C)$	(0.5, 0.5, 0.5, 0.5, 0.5)
χ_i (or ε)	0.75

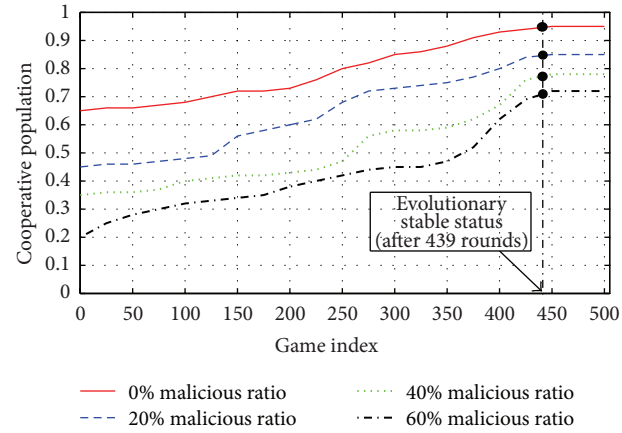


Figure 2: The overall effect of cooperative performance under various original malicious ratios.

To evaluate the network transmission performance of our proposed cooperative stimulation scheme in large scale MANETs, a proportion of malicious members who give priority to take attack action from strategy space on the basis of payoff maximization will be mixed up with normal members at the initial time, that is, original malicious ratio (i.e., we mainly set the ratio at 0%, 20%, 40%, and 60% in simulation). More specifically, take wireless medium network for instance in simulation; a 3-level attack set is provided for malicious members to make decisions; A_1 means frame flooding attack, A_2 means black hole attack, and highest risk of A_3 means packets dropping attack. Then the following indexes are measured for evaluating network performance.

(1) *Cooperative Population*: it is defined as the ratio between the total number of members taking cooperative action and that of all members in MANETs.

(2) *Average Payoffs*: it is defined as the mean value payoffs obtained by all members in MANETs after each round of the game.

(3) *Transmission Success Ratio (TSR)*: it is defined as the ratio between the total number of successfully forwarded packets and that of packets scheduled to be sent.

(4) *Normalized Network Throughput (NNT)*: it is defined as the ratio between the number of valid packets (or bits) which make them through the network per time unit and that of total network packets, which can be depicted as the data activity of MANETs.

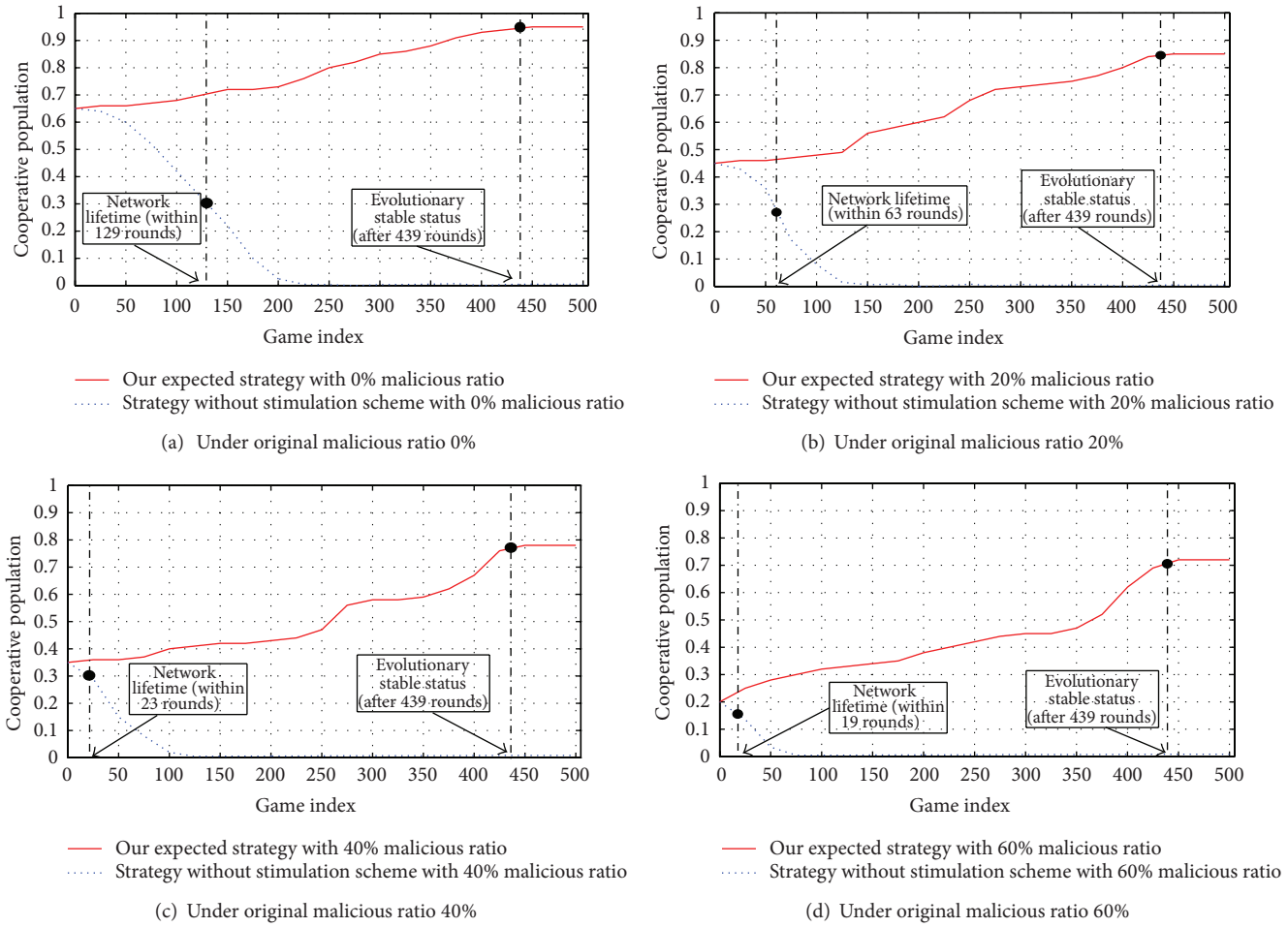


Figure 3: Comparison result of cooperative performance using our expected strategy and others without stimulation scheme under different original malicious ratios.

In our evolutionary game, each simulation is evolved 500 times to estimate these indexes. What is more, the evolutionary stable status and game convergence performance are also measured to verify our trust cooperative stimulation scheme.

6.2. Simulation Results. Figure 2 compares the overall effect of evolutionary cooperative population under various kinds of malicious ratio using our trust cooperative stimulation scheme. From Figure 2 we can see that as the cooperation game goes on, the network cooperative population, which takes strategy C, is all increased significantly during 500 game rounds under original malicious ratio at 0%, 20%, 40%, and 60%. This is because nodes could take cooperation strategy to obtain a higher trust level in the frame of cooperative stimulation scheme in order to strive for continuous network services. In addition, the simulation parameter setting meets the condition of Theorem 2; namely, the game model exists evolutionary stable status and convergence point. According to the simulation results, after the evolutionary game is played 439, rounds the evolutionary stable status (ESS) comes and the cooperative population cannot fluctuate wildly. More specifically, when original malicious ratio of network is at 0%, 20%, 40%, and 60%, respectively, the cooperative population is increased from 62.3%, 45.8%, 36%, and 19.6% to 94.3%, 83.9%, 78.6%, and 72.4% at ESS point. Even if there is small proportion of network members engaged in malicious attacking after ESS, the stable status of cooperative population is not invaded by malicious strategy. These simulation results prove that our proposed scheme can stimulate cooperation behavior among network members under a high malicious ratio as well as promote ratio of population participating in cooperative transmission so as to maintain normal services of MANETs.

Figure 3 shows the comparison of the cooperative population using our proposed stimulation scheme and the method not using stimulation scheme under original malicious ratio at 0%, 20%, 40%, and 60%, which are shown Figures 3(a), 3(b), 3(c), and 3(d), respectively. Note that in MANETs, especially in large scale MANETs, if there are more than 70% of network members refusing to cooperate with others, the network services will be impeded seriously. Thus in the simulation, if the cooperative population is less than 30% and this tendency continues 100 game rounds, the network transmission service is suspended. Without loss of generality

the round number of the game which corresponds to the point of 30% of cooperative population is defined as network lifetime. From Figure 3, we can see that the network lifetime is effectively prolonged by improving cooperative population far above 30% using stimulation scheme compared with the other method. More specifically, under original malicious ratio at 0%, 20%, 40%, and 60% the network lifetime is 129 rounds, 63 rounds, 23 rounds, and 19 rounds, respectively, by using the method without stimulation scheme. While using our scheme, until the end of 500 rounds of the simulation, the network services are still maintained by large crowd of cooperative population (94.3%, 83.9%, 78.6%, and 72.4% when it comes to ESS). It can be inferred that in large scale MANETs (member number exceeds 5000) as well as high malicious ratio (>50%), our scheme still has a better performance.

Recall that in our proposed cooperative stimulation scheme, the time factor of the action space plays a role in coupling current and accumulated trust level of the members in MANETs; hence it contributes to the improvement in cooperative performance of the network. In order to verify and evaluate the impact of the time factor on cooperative population and convergence rate of the proposed evolutionary game, a series of simulations have been conducted. Figure 4 shows the result of the two kinds of settings of the time factor; one is time factor for each action at 0.5; that is, the updated trust level of the members is equal-weighted by the current value of trust level and that of accumulated value. The other one is time factor optimally by hierarchically weighting different action element; that is, the higher the risk level corresponding to action element, the smaller its time factor is set. By using hierarchical setting of time factor, the updated trust value by taking the higher risk action relies less on accumulated value. On the contrary, because of the larger value of time factor corresponding to beneficial action, the updated trust value relies more on accumulated value. Thus once the member takes action with a higher risk level, its trust level will be reduced immediately as punishment, and while taking actions that do not threaten the network, its reduction rate of trust level slows down with the increase of coupling degree. From Figure 4, we can see that, adopting the equal-weighted setting of time factor ([0.5, 0.5, 0.5, 0.5, 0.5]), the game gets into the ESS at the point of 439 game rounds and the cooperative population at this time remains 78.6% under original malicious rate at 40%. While in the same circumstances, not only the convergence rate, but also the cooperative population is superior to the previous result by using optimal hierarchal setting of time factor ([0.3, 0.2, 0.1, 0.4, 0.5]) whose value is 361 rounds and 87.3%, respectively. To sum up, the hierarchical setting method can be regarded as user interface which adjusts the risk level of various actions in our scheme.

In previous systematic simulations, all parameters are set to satisfy Theorem 2. When the transmission game gets into ESS, the optimum strategy space converges to the expected strategy space which can also make members obtain maximum payoff. In the following simulation, we need to evaluate another important index which mainly drives members to take which actions, that is, average payoff, and verify the effect

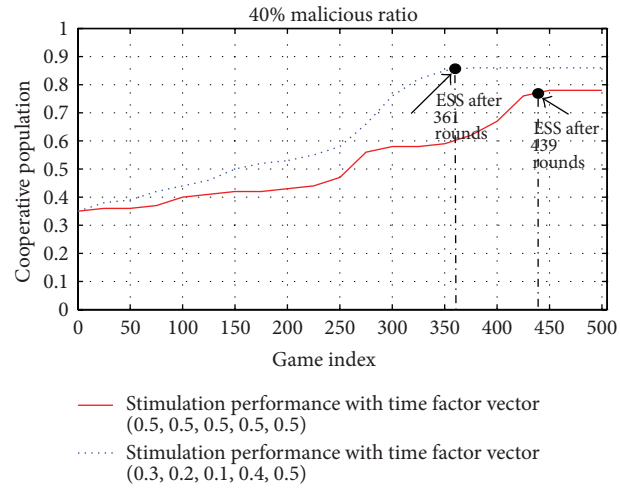
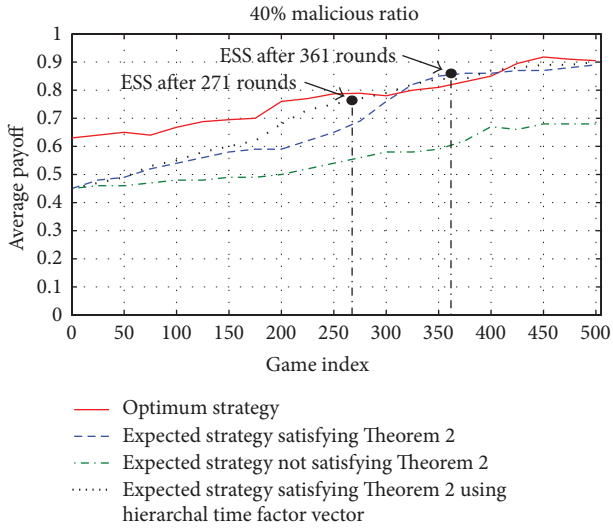


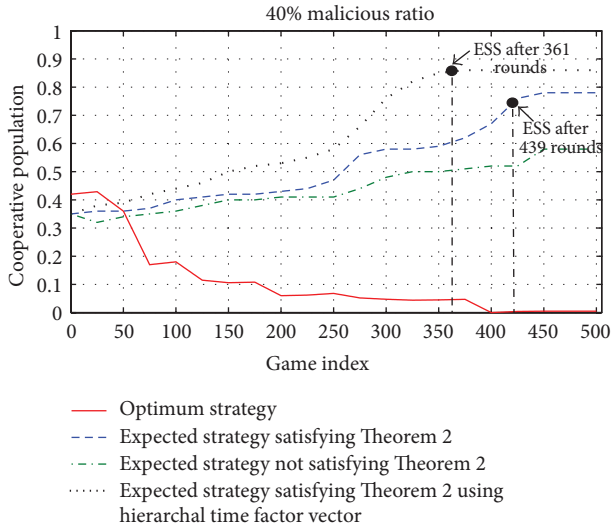
Figure 4: Impact of 2 kinds of time factor setting on cooperative performance using scheme under original malicious ratio 40%.

of Theorem 2 in this paper. As can be shown in Figure 5, we compared 4 action spaces in our game, which are optimum action space (payoff-preference), expected action space satisfying Theorem 2 (convergent trust-preference), expected action space not satisfying Theorem 2 (nonconvergent trust level-preference), and expected action space not satisfying Theorem 2 with a hierarchical time factor (optimal attack classification). From the simulation result, under original malicious ratio at 50%, the optimum action space has the highest average payoffs during each round of the game (the average value of payoffs obtained by 5000 members is greater than 0.6 and grows top to 0.92). By contrast, the average payoffs of the other 3 action spaces are lower than optimum action space (about 0.4–0.8). On the other hand when members take optimum strategy, the cooperative population of the network does not increase in spite of the maximizing members' payoffs. This is because that the strategy driven by obtaining maximized payoff principle is always attack, that is, violating to cooperate with each other. As a consequence, as shown in Figure 5(b), taking optimum strategy would reduce the cooperative population (as curve 1). According to our inference in this paper, expected strategy (trust-preference) can effectively stimulate members to cooperate with others, but it cannot bring members a satisfying payoff (as curve 4). To solve this problem, if parameters are set to satisfy Theorem 2, it can not only stimulate members to cooperate with others, but also increase average payoff of the whole network (as curves 2, 3). Moreover the strategy which is set to include hierarchal time factor performs better than that without hierarchal time factor (see curve 2), which well verifies the simulation result above.

To extend our theoretical game model to the application of realistic MANETs, there are 2 important indexes referring to network transmission service, transmission success rate (TSR) and normalized network throughput (NNT), which must be measured. So finally we conduct afterwards simulation to evaluate TSR and NNT using the proposed



(a) Average payoff of 4 representative action spaces under original malicious ratio 40%



(b) Cooperation performance of 4 representative action spaces under original malicious ratio 40%

FIGURE 5: Verification the impact of convergence condition depicted as Theorem 2 on cooperation performance and average payoff using 4 representative action spaces under original malicious ratio 40%.

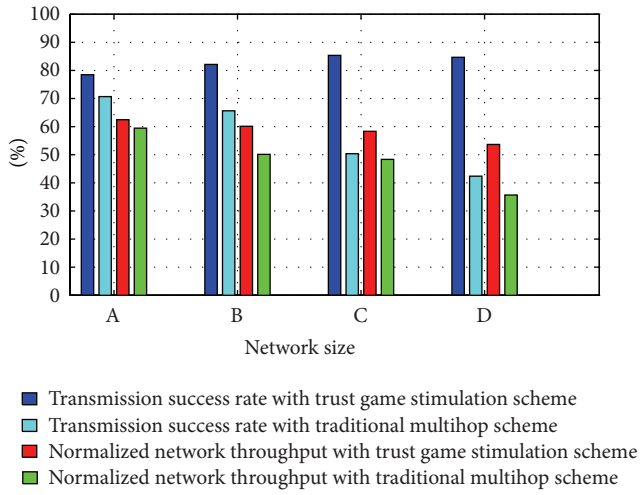


FIGURE 6: Comparison result of TSR and NNT using our stimulation scheme and traditional multihop scheme.

cooperative stimulation scheme comparing to that using traditional multihop transmission scheme in large scale MANETs. The bar chart of Figure 6 shows the simulation result, where A, B, C, and D denote the member number of the network 2500, 5000, 7500, and 10000, respectively. From Figure 6, we can see that due to the increase of the cooperative population by using stimulation scheme, the TSR has been increased from 79% to 84% as the network member number ranges from 2500 to 10000. On the contrary, under original malicious ratio at 40% by using traditional multi-hop scheme the TSR drops dramatically from 71% to 42% which results in lacking of cooperation among network members. Then to the index NNT which reflects the active degree of network information, as a matter of fact, a higher NNT means larger accommodation of data stream of MANETs. From Figure 6, the NNT has been effectively maintained from 62% only down to 54% with the growth of the network scale. But in the same situation, by using traditional multi-hop scheme the NNT has been reduced dramatically from 59% to 35%. Therefore, our proposed cooperative stimulation scheme can effectively serve date transmission in large scale MANETs with a higher malicious ratio.

7. Conclusion

In this paper, we have investigated an evolutionary game theoretic trust cooperative stimulation scheme for large scale MANETs to incent members to take cooperative actions with each other so as to maintain cooperative performance. By means of constructing the complete multirisk level strategy and payoff space and building trust-preferential strategy, the malicious action can be effectively constrained to a low trust level. Then through evolutionary analysis of game model, the convergence condition between optimum strategy which represents payoff maximization principle and trust-preferential strategy is deduced. Furthermore, the mobility probability parameters and information propagation error are also introduced into our scheme, which makes it approach to the realistic large scale MANETs. Both theoretical analysis and simulation experiments have demonstrated that although a gap may exist between the game model and reality, the game-theoretic approach can still provide thoughtful insights and helpful guidelines when stimulating members to cooperate with each other from multirisk level of purposive strategic attack in large scale MANETs. The proposed scheme can effectively stimulate cooperation among members and meanwhile be robust under the condition where the environment is harsh under a high original malicious ratio in large scale MANETs.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grants nos. 61001138 and 61201317.

References

[1] H. Kim, R. B. Chitti, and J. Song, "Novel defense mechanism against data flooding attacks in wireless ad hoc networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 579–582, 2010.

[2] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "A survey of routing attacks in mobile ad hoc networks," *IEEE Wireless Communications*, vol. 14, no. 5, pp. 85–91, 2007.

[3] M. S. Fallah and M. Mouzarani, "A game-based sybil-resistant strategy for reputation systems in self-organizing MANETs," *Computer Journal*, vol. 54, no. 4, pp. 537–548, 2011.

[4] A. Alireza, T. Helen, and V. Athanasios, "A survey of security challenges in cognitive radio networks: solutions and future research directions," *Proceedings of the IEEE*, vol. 100, no. 12, pp. 3172–3186, 2012.

[5] R. Feng, X. Xu, X. Zhou, and J. Wan, "A trust evaluation algorithm for wireless sensor networks based on node behaviors and D-S evidence theory," *Sensors*, vol. 11, no. 2, pp. 1345–1360, 2011.

[6] A. Rehan, K. Turgay, and G. V. Raju, "EMLTrust: an enhanced machine learning based reputation system for MANETs," *Ad Hoc Networks*, vol. 10, no. 3, pp. 435–457, 2012.

[7] H. Y. Shi, W. L. Wang, N. M. Kwok, and S. Y. Chen, "Game theory for wireless sensor networks: a survey," *Sensors*, vol. 12, no. 7, pp. 9055–9097, 2012.

[8] L. Blazevic, L. Buttyan, S. Čapkun, S. Giordano, J. P. Hubaux, and J. Y. Le Boudec, "Self-organization in mobile ad hoc networks: the approach of terminodes," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 166–173, 2001.

[9] L. Buttyán and J. P. Hubaux, "Stimulating cooperation in self-organizing mobile Ad Hoc networks," *Mobile Networks and Applications*, vol. 8, no. 5, pp. 579–592, 2003.

[10] P. Michiardi and R. Molva, "CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proceedings of the 6th Joint Working Conference on Communications and Multimedia Security (IFIP TC6/TC '11)*, pp. 107–121, 2002.

[11] S. Buchegger and J. Y. Le Boudec, "Performance analysis of the CONFIDANT protocol (Cooperation of nodes: fairness in dynamic ad-hoc networks)," in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '02)*, pp. 226–236, June 2002.

[12] W. Yu and K. J. Ray Liu, "Attack-resistant cooperation stimulation in autonomous Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 12, pp. 2260–2271, 2005.

[13] B. Niu, H. Jiang, and H. V. Zhao, "A cooperative multicast strategy in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 6, pp. 3136–3143, 2010.

[14] Y. Chen and K. J. R. Liu, "Indirect reciprocity game modelling for cooperation stimulation in cognitive networks," *IEEE Transactions on Communications*, vol. 59, no. 1, pp. 159–168, 2011.

[15] L. Xiao, Y. Chen, W. S. Lin, and K. J. R. Liu, "Indirect reciprocity security game for large-scale wireless networks," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1368–1380, 2012.

[16] D. A. Maurizio, O. Francesco, and R. S. Pietro, "Can cooperation improve energy efficiency in ad hoc wireless networks?" *Computer Communications*, vol. 35, no. 14, pp. 1707–1714, 2012.

[17] H. Zhao, X. Yang, and X. Li, "An incentive mechanism to reinforce truthful reports in reputation systems," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 951–961, 2012.

[18] L. Ze and S. Haiying, "Game-theoretic analysis of cooperation incentive strategies in mobile Ad Hoc networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 8, pp. 1287–1303, 2012.

[19] L. Li, W. Cai, L. Liang, and L. Fan, "Design and analysis of Bayesian game in role based trust management," in *Proceedings of the IEEE International Conference on Information Theory and Information Security (ICITIS '10)*, pp. 394–398, December 2010.

[20] A. K. Charles, P. Niki, and M. Kia, "Game theoretic modeling and evolution of trust in autonomous multi-hop networks," in *Proceedings of the IEEE International Conference on Communications (ICC '11)*, Kyoto, Japan, June 2011.

[21] D. Wu, Y. Cai, and J. Wang, "Cooperation policy selection for energy-constrained Ad Hoc networks using correlated equilibrium," *IEEE Communications Letters*, vol. 16, no. 3, pp. 349–351, 2012.

[22] T. Chen, L. Zhu, F. Wu, and S. Zhong, "Stimulating cooperation in vehicular ad hoc networks: a coalitional game theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 2, pp. 566–579, 2011.

[23] W. Yu and K. J. R. Liu, "Secure cooperation in autonomous mobile ad-hoc networks under noise and imperfect monitoring: a game-theoretic approach," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 2, pp. 317–330, 2008.

[24] F. Li, Y. Yang, and J. Wu, "Attack and flee: game-theory-based analysis on interactions among nodes in MANETs," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 40, no. 3, pp. 612–622, 2010.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 324632, 9 pages
http://dx.doi.org/10.1155/2013/324632

Research Article

An Efficient Resource Management Protocol for Handling Small Resource in Wireless Sensor Networks

Wan-Hee Cho, Jiho Kim, and Ohyoung Song

School of Electrical and Electronics Engineering, Chung-Ang University, Seoul 156-756, Republic of Korea

Correspondence should be addressed to Ohyoung Song; song@cau.ac.kr

Received 1 February 2013; Accepted 24 April 2013

Academic Editor: Kayhan Gulez

Wireless sensor nodes with single chips may have insufficient resources for certain applications. We propose a resource management protocol for applications with constrained resources to improve effectiveness by borrowing resources from a resource management server.

1. Introduction

Wireless sensor nodes are often used in harsh environments such as sewers, bridges, and the outer walls of buildings. Most are operated by small batteries, which mean that the power consumption and size of devices should be reduced. These requirements can be met through the use of an integrated circuit that reduces the usage of external components. The highly integrated single-chip approach is a preferred solution for semiconductor manufacturers, since the total system cost is a key factor for industrial and home wireless applications [1]. Cost considerations are driving implementations towards single-chip solutions with a minimum number of external components [2]. The single-chip approach is also advantageous for reducing device size through the use of well-designed integrated circuits to reduce the number of external components. The single-chip approach is a mainstream method used for developing wireless sensor network-compliant devices. However, the limitations of the device size and the power consumption still lead to various constraints, such as a slower microprocessor unit (MPU) and smaller memory size. The most recent single-chip designs, especially for wireless sensor nodes, provide about 4~16 Kbytes of SRAM and an 8-bit microprocessor capable of 16 million instructions per second (MIPS) [3, 4]. The available memory becomes much smaller after using up the memory for a ZigBee profile stack. In applications that perform complex and very repetitive processing, the usage of the microprocessor might become excessive. A general solution may be to use external memory, an additional microprocessor, or a controller, but this usually causes the cost and device size to increase.

We have shown that wireless sensor nodes can feasibly borrow the memory or computational resource from the gateway or the server in our preliminary study [5]. In this paper, we propose a resource management protocol (RMP) that enables wireless sensor nodes to efficiently use the resources including the memory and the CPU in the gateway or the server. Also, the effectiveness of the RMP is validated by several experiments through our implementation of the RMP.

The rest of this paper is organized as follows. Section 2 presents related works to efficient resource management in wireless sensor networks. Section 3 identifies the resource constraint issues and draws the requirements to relax them. In Section 4, we briefly introduce ZigBee-layered architecture. In Section 5, we propose an efficient resource management protocol. We evaluate the performance of our resource management protocol by experiments and analysis in Section 6 and make a conclusion in the last section.

2. Related Works

Wireless sensor nodes have been used for various applications in surveillance, environment and habitat monitoring, structural monitoring, healthcare, and disaster management [6]. Developers of wireless sensor nodes face technical challenges

that include dense ad-hoc deployment, dynamic topology, spatial distribution, and constrains in bandwidth, memory, computational resources, and energy [7]. Low-power sensor nodes are known for their limited resources. For instance, motes are equipped with kilobytes of RAM which may be easily insufficient for storing or processing images [8]. Typically, in visual sensor networks (VSNs) which consist of tiny visual sensor nodes called camera nodes, the camera nodes should be equipped with memories of larger capacity in order to store the data [9]. Using more powerful microcontrollers equipped with sufficient RAM for data processing would be a straightforward solution for large data processing [8]. But they usually cause the cost to increase. Traditionally, wireless sensor nodes collect data and transmit data to centrally resourceful gateway for processing because they are generally supposed to be resource constrained [8]. Memory overhead is one of the main technical concerns for sensor network security such as any replication detection protocol [10]. Existing solutions for multicast in wireless sensor networks are limited because they either support multicast only from a single source node (usually the root node) or they limit the multicast group size to constrain memory usage [11]. It becomes especially difficult to implement them when composing a large-scale wireless sensor network or controlling a peripheral device, such as an LCD, due to insufficient resources. This issue is described in Sections 3 and 6 in greater detail.

Toward efficient resource management in wireless sensor networks, several studies have been made on new protocols that are efficient in resource management and carefully managed by operating system level [12] and solutions to the scalability of resources using an open source cloud model which provides the storage and computation resources necessary to address the scalability [13]. Extension of the cloud computing paradigm to the sharing of sensor resources in wireless sensor networks results in a much promising technology called Sensor Clouds [14]. Since the resource and capability of physical sensor devices are limited, Sensor-Cloud infrastructure can be behalf of the sensor management such as availability and performance of physical sensors [15]. Various physical sensors with different owners can join Sensor-Cloud infrastructure. The templates for virtual sensors and virtual sensor groups are prepared for sharing physical sensors. Users can control their virtual sensors directly or via their Web browsers [15]. Mass data processing is required for these sensor networks. Several studies about employing virtual memory to increase the available memory have been made [16]. Virtual memory named FaTVM for data-intensive applications in wireless sensor networks makes it possible for sensor nodes to carry out complex computation with heavy memory footprint without using energy-hungry MPUs with large RAM [8]. FaTVM uses NAND flashes as secondary storage and focuses on reducing virtual memory overhead [8]. In our resource management protocol, wireless sensor nodes can borrow the resources from a server that has access to more sufficient resources in the network in such a way that is general and extensible enough to resolve the resource constraint issues in several kinds of wireless sensor networks.

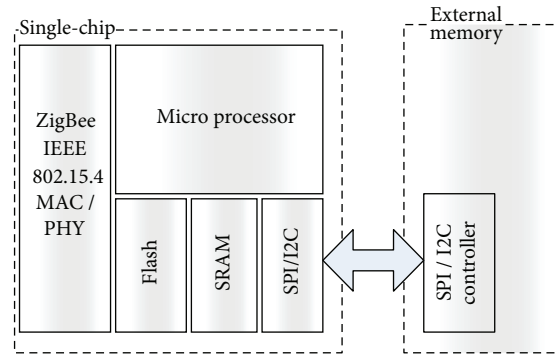


Figure 1: An Example of using an External Memory.

3. The Resource Constraint Issues

ZigBee-based wireless sensor nodes require low-power, low-cost wireless networking in residential and industrial environments [17], so that the battery in the sensor end node lasts for a long time without being plugged in. A busy state using complex computational resources consumes much more power than an idle state. Even memory resource constraints require additional memory devices, which increase the total system cost. To achieve low-power, low-cost wireless networking, the amount of memory allocated and the MPU usage in a device must be reduced. However, for large-scale sensor networks, with limited resources at every node, and deployment in environments with high access cost, the task of managing and operating these systems is extremely challenging [18].

Once a ZigBee end device (ZED) has joined a personal area network (PAN), a ZigBee coordinator (ZC) needs to update its neighbor table and store the minimum 12 bytes of information for the joined device [19]. If we assume that a maximum of 240 allowable devices have joined the network, the ZigBee Coordinator needs about 5 kilobytes of memory. In addition, ZigBee End Devices may require more memory to support the ad hoc on-demand distance vector (AODV) routing feature. Moreover, additional computational resources are required to access the memory. ZigBee end devices and the ZigBee coordinator may have various peripherals, depending on the application. For instance, in Smart Grids [20], a smart energy device that actively informs customers via in-home displays (IHD) of when or how energy is being used, has become necessary [21]. The device, which has an LCD screen, needs a great deal of screen buffer memory and computational resources to print text or to draw a picture on the screen.

To solve the resource constraint issues, a general solution is redesigning the hardware device. Though the software can be optimized, this cannot be a complete solution. Figure 1 shows an example of using external memory through a general-purpose interface, like SPI or I2C. The memory addition is simple, but it may require an API for interfacing external memory if the microprocessor or the single-chip does not provide a dedicated external memory interface. These APIs also use a substantial amount of resources.

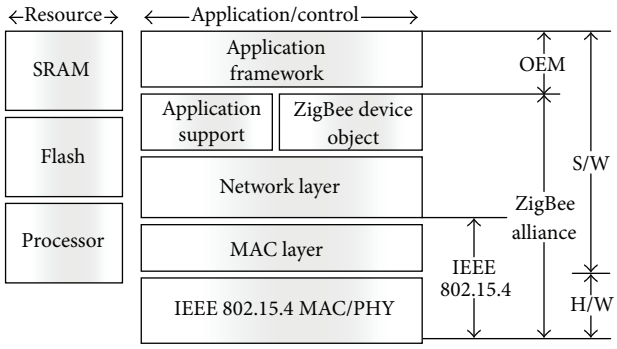


Figure 2: ZigBee-layered architecture.

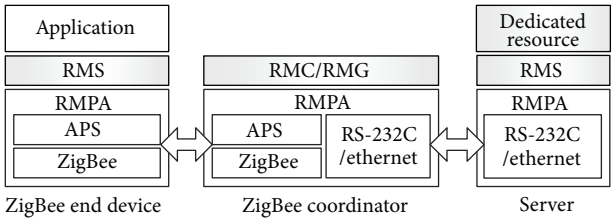


Figure 3: The resource management protocol-layered architecture.

In the same manner, we can use an additional processor or dedicated peripheral controller if more computational resources are required. However, this causes the device size to increase, and the total system cost becomes high.

4. ZigBee-Layered Architecture

ZigBee technology consists of application, network, medium access control (MAC), and physical layer. The MAC/PHY layer of ZigBee is specified in IEEE 802.15.4. ZigBee standardizes network and application layer. In other words, the specification scope of ZigBee alliance is the rest upper layer of IEEE 802.15.4 PHY/MAC. Figure 2 illustrates the ZigBee-layered architecture.

As shown in Figure 2, the application layer consists of an application support layer (APS), an application framework (APF), and a ZigBee device object (ZDO). The ZigBee alliance standardizes APS and ZDO. APF is handed over to a vendor. The network layer provides functionalities that enable an end device to communicate with other end devices. It also manages network security and routing.

4.1. Data Service. A network layer data entity (NLDE) sends and receives a data frame or controls a network header and communicates with APS using an NLDE service access point (NLDE-SAP). It also communicates with the MAC layer using a MAC common part sub-layer (MCPS-SAP) data interface.

4.2. Management Service. For the purpose of management, a network layer management entity (NLME) communicates with MAC using MLME-SAP. An application communicates with the network layer using ZDO.

Table 1: Header information block.

Layer	Header information
MAC	Length (1), MAC header (9), MAC CRC (2)
NETWORK	NWK Header (8), NWK security (14), NWK MIC (4)
APS	APS header (5~8), APS security (5), APS MIC (4)

4.3. Service Access Point (SAP). The data between two layers is transferred by SAP. Each SAP transfers an appropriate data structure, which is specified with an entity in the PAN information base (PIB).

4.4. Application Support Layer (APS). The maximum payload length in 802.15.4 is 128 bytes, including a length byte. Each layer uses an information header block, as shown in Table 1. As a result, the maximum payload length in the APS layer is 73 bytes. Moreover, it may be shortened further if the packet includes a routing header.

The ZigBee APS packet includes an application profile ID that describes the format of the message, a cluster ID for this message, a source endpoint, a destination endpoint, a bitmask of options, a group ID for this message if it is multicast mode, and a sequence number.

5. Efficient Resource Management

In a client-server system, a client's application may not be executed due to insufficient resources, even though the system has one or more resource-rich servers. The idea of efficient resource management is based on the imbalance of resources. The resource management protocol provides a way of using server-side resources.

Figure 3 illustrates the connection between the resource management client (RMC) and the resource management server (RMS). The resource management client requests a needed resource to execute a function for an application. The resource management server allocates a requested resource, performs a requested operation, and provides the outcome of the request to the resource management client. The ZigBee coordinator also acts as a resource management gateway (RMG) that relays packets between the resource management client and the resource management server. Each resource management client can request a resource from the resource management server through the resource management gateway. The resource management protocol adaptation (RMPA) layer abstracts the APS and ZigBee-based wireless sensor network interface.

5.1. RMP Packet. The format of packets that are transmitted from the resource management protocol is shown in Figure 4. The source address, SRC, is a short address of the server or the device that generates the RMP packet. The packet body includes an RMP command and an optional data field, as shown in Figure 4. The basic RMP command set

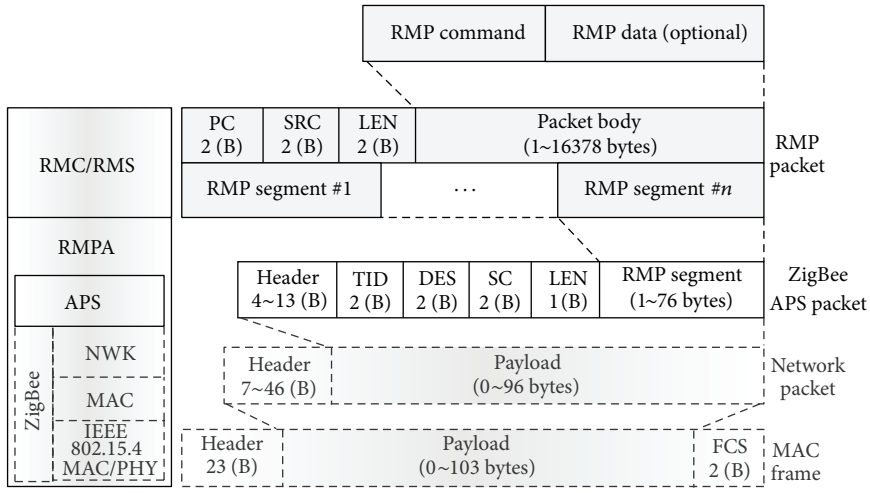


Figure 4: RMP packet structure in RMP layered architecture.

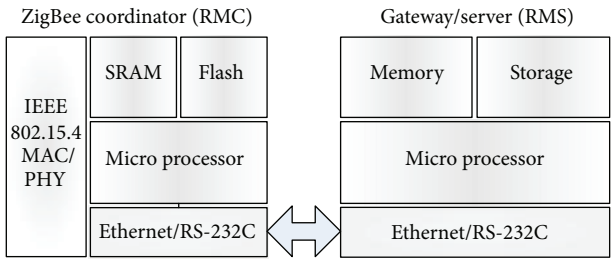


Figure 5: Data frame architecture.

Table 2: Packet types for solving issue on the large-scale ZigBee network.

Field	Description
STORE	Store data into the specified location of the RM server
LOAD	Load data from the specified location of the RM server
REMOVE	Remove data from the specified location of the RM server
FIND	Find a location of which is matched with specified condition from the RM server and load data

is shown in Table 2. It is also used for the experiment in this paper. The RMP command set can be modified or extended to conform to the requirements of other applications.

5.2. ZigBee APS Packet. The RMP packet is divided into the RMP packet segments shown in Figure 4. The APS header consists of the profile ID, the cluster ID, the source endpoint, and the destination endpoint. The transaction ID, TID, is a transient value that is specified by the initiator of the current RMP transaction when the original RMP packet is generated. The RMP packet segments that are generated by the RMP packet have the same TID. The destination address, DES, is a short address of the device that is the destination of the RMP segment. The sequence control, SC, has the number of the current segment and the number of the total segments.

5.3. Packet Concatenation. The maximum payload of a ZigBee bearer is 76 bytes, excluding the overhead of 12 bytes in the MAC layer, 26 bytes in network layer, and 13 bytes in APS layer, as shown in Table 1. Packet concatenation is needed at the destination, because a packet that exceeds the maximum payload size of 76 bytes in the resource management protocol is fragmented into several segments, which are sent to the destination. An RMP packet can be assembled by concatenating one or more raw data from the RMP packet segments that arrive at the destination. The RMP packet segments that have the same transaction ID are concatenated by the order of the sequence number.

5.4. RMP Packet Relay. The ZigBee coordinator between the resource management client and the resource management server functions not only as a resource management client but also as a resource management gateway, as shown in Figure 3. If the destination address of an APS packet that comes from the resource management server is different from the short address of the ZigBee Coordinator, the ZigBee Coordinator forwards an APS packet to the corresponding ZigBee end device without packet concatenation.

If the destination address of an APS packet that comes from the resource management client is undefined or identical to the short address of the resource management server, the ZigBee Coordinator forwards an APS packet to the resource management server. The ZigBee Coordinator receives an APS packet that includes the RMP payload and sends the APS packet to its destination address, as shown in Figure 4.

5.5. RMP Adaptation Layer. The RMP packet that comes from the resource management client or the resource management server forwards appropriate communications, which can be wireless (IEEE 802.15.4 RF) or wired (Ethernet, RS-232C) depending on the packet direction after regeneration of the packet header. In wireless communication, packet fragmentation can be applied to a packet that exceeds the

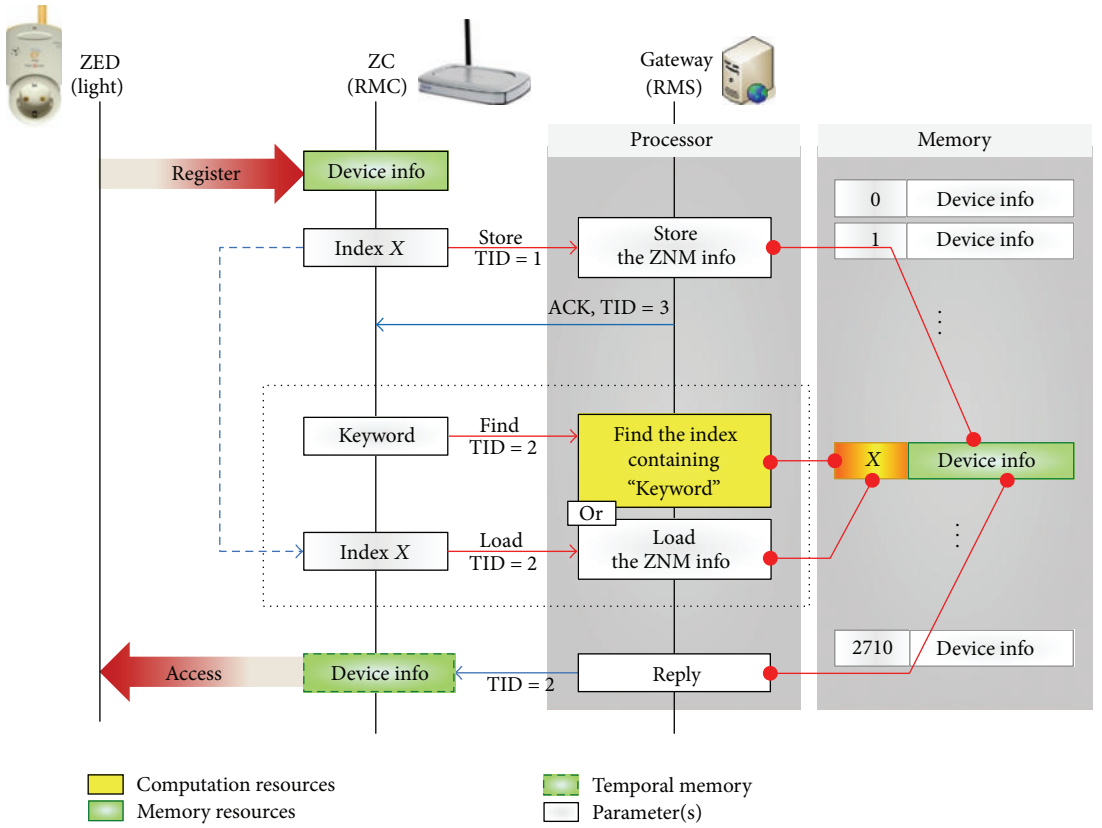
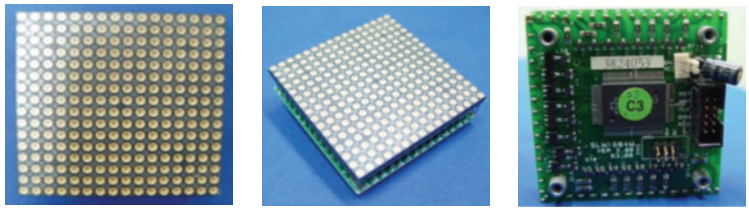


Figure 6: Sequence flow of the RMP on the large-scale ZigBee network.



(a) (b) (c)

Figure 7: 16 × 16 dot matrix.

maximum payload. The RMP adaptation layer concatenates the packet segments to the entire RMP packet prior to forwarding into the RMP layer. When transmission error occurs in the transaction, concatenation will fail and discard the packet segments if the packet segments arrive out of order. If the destination address of the RMP packet segment that arrived in the ZigBee Coordinator differs from the address of the ZigBee Coordinator, then the ZigBee Coordinator relays the RMP packet segment to the destination through the APS.

5.6. Packet Acknowledgment. Whenever the RMP packet is transmitted successfully, the resource management client or the resource management server has to reply with an acknowledgement within 2 seconds. If the retransmit time expires, then the sender starts the retransmit timeout procedures and retransmits the RMP packet using the same transaction ID.

5.7. RMP Types. Various packet types can be defined for different kinds of applications. Packet types can define various operations between the resource management server and the client, including memory allocation, memory access, data encoding, data decoding, and arithmetic operation. The RMP command and data shown in Figure 4 should be defined appropriately for an application.

6. Implementation & Experiments

We show actual implementations and experiments to validate the feasibility and effectiveness of the resource management

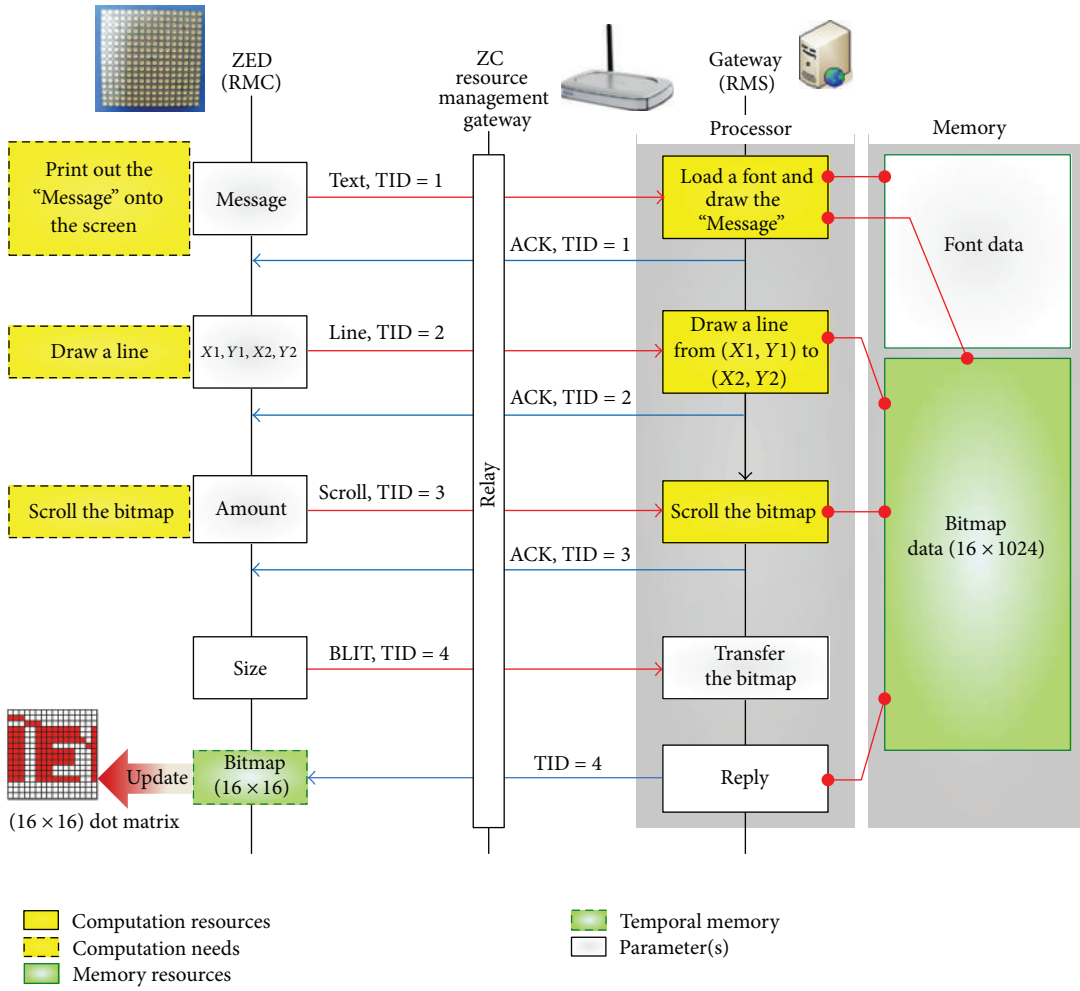


FIGURE 8: Sequence flow of the RMP for controlling a dot matrix.

protocol. Some packet types mainly used in wireless sensor networks with resource constraints are defined in Table 2.

6.1. The Resource Management Protocol on the Large-Scale ZigBee Network. In Figure 5, the ZigBee Coordinator acting as a resource management client typically connects with the resource management server via ethernet or an RS-232C cable. This means we can ignore the data transmission latency between the resource management client and the resource management server.

Figure 6 shows the sequence flow of the resource management protocol in a large-scale ZigBee network. The resource management client with insufficient memory sends an RMP packet with the packet type "STORE" and an index that represents the memory address in the server. The resource management server stores RMP data into its own memory with the specified index. When a resource management client needs to use the data, it sends a request packet with the packet type "LOAD" and the index. The resource management server then replies with the data associated with the index from its own memory. A packet with the packet type "FIND" enables the resource management client to look up the memory without using its own computational resources.

Now, we consider a building with 10,000 lights, each of which is connected and controlled by a ZigBee End Device in the ZigBee mesh network. They are controlled by the gateway, which is connected to the ZigBee Coordinator using RS-232C. During the peak time, an average of 1,000 control commands occur. The response time of the service T_s is 50 ms, which was obtained through experimentation. Referring to Little's Law,

$$\rho = \lambda T_s = 0.05 * \frac{1,000}{60} = 0.83 = 83\%, \quad (1)$$

where ρ is the server utilization and λ is the service requests per second

$$T_r = \frac{T_s}{1 - \rho} = 0.29 \text{ sec}, \quad (2)$$

where T_r is the response time for a service.

Because this shows that the server utilization ρ is less than 1 (100%) and the response time of 290 ms is acceptable

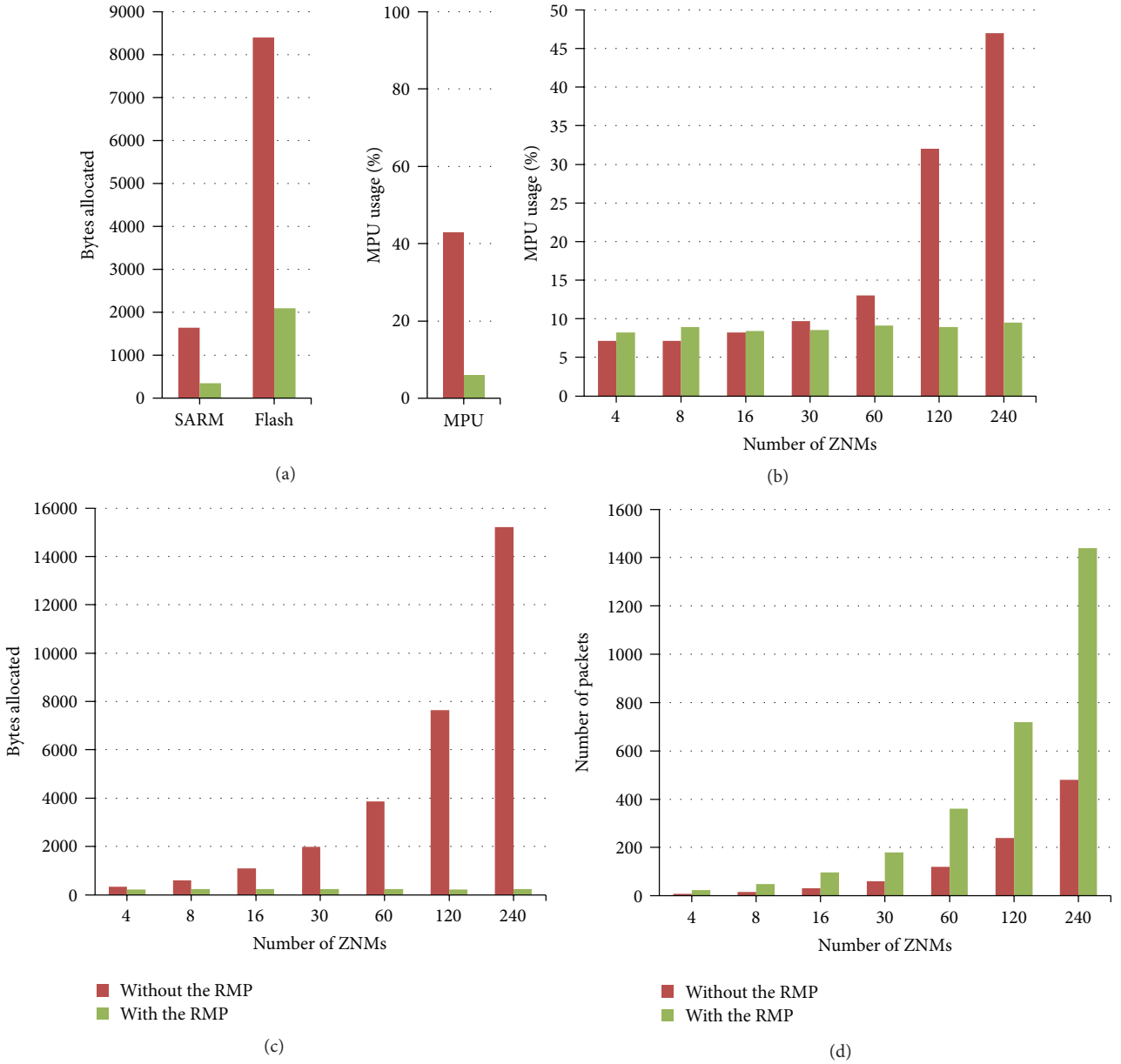


Figure 9: Performance of the RMP.

for a light control service, the assumption is reasonable and practical.

6.2. Controlling a Dot Matrix with the Resource Management Protocol. In order to validate the feasibility and the effectiveness of the resource management protocol for the resource constraints found in controlling peripheral devices, we set the ZigBee End Device as an in-home display using a 16 × 16 dot-matrix which has a 32-byte display buffer. Figure 7 shows an actual 16 × 16 dot matrix display used in the In-Home Display implementation. We define the RMP packet types as shown in Table 3.

A long text message with a maximum of 128 characters in several lines is then scrolled on the dot-matrix. For this application, the resource management server creates a 16 × 1024 (16 KB) drawing buffer for the 128-character-long

Table 3: Packet types for solving issues of controlling a dot matrix as a peripheral device.

Field	Description
FILL	Fill the specified region of bitmap data with a certain data
BLIT	Transfer the bitmap data to the RM client
TEXT	Print out a string to the specified location of bitmap space
SCROLL	Scroll the text or image
LINE	Draw a line onto the bitmap space of the RM server

message with an 8 × 16 font. Figure 8 shows the sequence diagram of the resource management protocol for controlling a dot-matrix. The resource management client sends a request packet which has the packet type "TEXT" and a message string to print out onto the dot-matrix. Then, the resource management server loads font data from its own memory and draws the message string onto the drawing buffer. The resource management client sends "LINE" request packets to the resource management server to draw lines onto the drawing buffer. Every 100 ms, the screen buffer is scrolled with the "SCROLL" request. When the server receives a "BLIT" request from the resource management client, its front partial 16 × 16 bitmap is transferred to the resource management client. The 16 × 16 bitmap is copied to the dot-matrix to update the display.

6.3. Experiments. For an experiment, we assembled one ZigBee Coordinator and four ZigBee End Devices to simulate 240 ZigBee End Devices. When each ZigBee End Device joins the network, a maximum of 60 ZigBee End Devices are assumed to join. Consequently, the ZigBee Coordinator stores the information 60 times with different indices. The lights are then randomly toggled by the gateway using a poisson distribution with a parameter of 100 controls per minute. We build two kinds of ZigBee End Device firmware. One uses the resource management protocol, but the other does not. They are written to the ZigBee End Devices evenly. The allocated SRAM size and the MPU usage, which are associated with controlling a dot-matrix and light, are measured from the kernel process scheduler of each ZigBee End Device. The varying numbers of the ZigBee End Devices are also measured from the ZigBee Coordinator. The needed flash memory size is evaluated with the firmware size. Figure 9 shows the result of the experiment.

As shown in Figure 9(a), by using the resource management protocol, the required memory size and MPU usage of the ZigBee End Devices that work as ZigBee Node modules (ZNMs) are largely reduced. Figures 9(b) and 9(c) show the effectiveness of the RMP for reducing the needed memory and MPU usage in the ZigBee Coordinator Module (ZCM). In Figure 9(d), when using the resource management protocol, the network traffic increases proportionally to the memory size that ZNMs need, because the ZCM converts the needed memory into RMP packets.

7. Conclusion

We have proposed a solution that enables efficient resource management of wireless sensor nodes using a resource management protocol. We have shown that it is feasible and effective for overcoming the resource constraints found in ZigBee applications through our implementations and experiments. The resource constraint issues were solved using the resource management protocol, without increasing the total system cost.

Disclosure

A preliminary version of this paper appears in Proceedings of the IEEE International Conference on Consumer Electronics, ICCE, 2011. This is the full version.

Acknowledgments

This research was supported by a Grant (SS100020) from the Seoul R&BD program funded by the Seoul Development Institute of the Korean government and supported by the Ministry of Knowledge Economy (10041725), Republic of Korea.

References

[1] W. C. Craig, "Wireless Control That Simply Works," ZigBee Alliance, 2004.

[2] W. Kluge, F. Poegel, H. Roller et al., "A fully integrated 2.4-GHz IEEE 802.15.4-compliant transceiver for ZigBee applications," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 12, pp. 2767–2774, 2006.

[3] Ember Corporation, "EM250 Single-Chip ZigBee/802.15.4 Solution," 120-0082-000R, May 2009.

[4] Atmel Corporation, "8bit AVR Microcontroller with Low Power 2.4 GHz Transceiver for ZigBee and IEEE 802.15.4," 8266AS-MCU Wireless, December 2009.

[5] W. H. Cho, J. Kim, and O. Song, "An efficient resource management protocol for handling small resource ZigBee devices," in *Proceedings of the IEEE International Conference on Consumer Electronics (ICCE '11)*, pp. 765–766, January 2011.

[6] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 54–61, 2004.

[7] R. V. Kulkarni and G. K. Venayagamoorthy, "Particle swarm optimization in wireless-sensor networks: a brief survey," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 41, no. 2, pp. 262–267, 2011.

[8] N. Lin, Y. Dong, and D. Lu, "Providing virtual memory support for sensor networks with mass data processing," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 324641, 20 pages, 2013.

[9] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Advances in Multimedia*, vol. 2009, Article ID 640386, 21 pages, 2009.

[10] M. Zhang, V. Khanapure, S. Chen, and X. Xiao, "Memory efficient protocols for detecting node replication attacks in wireless sensor networks," in *Proceedings of the 17th IEEE International Conference on Network Protocols (ICNP '09)*, pp. 284–293, October 2009.

[11] A. Marchiori and Q. Han, "PIM-WSN: efficient multicast for IPv6 wireless sensor networks," in *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM '11)*, 2011.

[12] P. Dutta and A. Dunkels, "Operating systems and network protocols for wireless sensor networks," *Philosophical Transactions A*, vol. 370, no. 1958, pp. 68–84, 2012.

[13] B. Sinha, *Wireless Sensor Network Architecture Addressing the Sclalability Issue Using Cloudsense*, SRM University, 2012.

[14] S. K. Dash, J. P. Sahoo, S. Mohapatra, and S. P. Pati, "Sensor-cloud assimilation of wireless sensor network and the cloud,"

in *Advances in Computer Science and Information Technology. Networks and Communications*, vol. 84 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 455–464, Springer, 2012.

[15] M. Yuriyama and T. Kushida, "Sensor-cloud infrastructure physical sensor management with virtualized sensors on cloud computing," in *Proceedings of the 13th International Conference on Network-Based Information Systems (NBiS '10)*, pp. 1–8, September 2010.

[16] N. Lin, Y. Dong, and D. Lu, "Fast transparent virtual memory for complex data processing in sensor networks," in *Proceedings of the International Conference on Sensor Networks (SENSORNETS '12)*, pp. 24–34, 2012.

[17] E. Callaway, P. Gorday, L. Hester et al., "Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 70–77, 2002.

[18] C. C. Han, R. Kumar, R. Shea, and M. Srivastava, "Sensor network software update management: a survey," *International Journal of Network Management*, vol. 15, no. 4, pp. 283–294, 2005.

[19] ZigBee Alliance, "ZigBee Specification," Document 053474r17, January 2008.

[20] B. Heile, "Smart grids for green communications," *IEEE Wireless Communications*, vol. 17, no. 3, pp. 4–6, 2010.

[21] ZigBee Alliance, "ZigBee Smart Energy Profile Specification," ZigBee Document 075356r14, May 2008.

Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 457045, 11 pages
http://dx.doi.org/10.1155/2013/457045

Research Article

A Multichannel Cross-Layer Architecture for Multimedia Sensor Networks

Taner Çevik[1] and Abdül Halim Zaim[2]

[1] *Department of Computer Engineering, Fatih University, Istanbul, Turkey*
[2] *Department of Computer Engineering, Istanbul Commerce University, Istanbul, Turkey*

Correspondence should be addressed to Taner Çevik; tcevik@fatih.edu.tr

Received 22 January 2013; Accepted 12 March 2013

Academic Editor: V. Cagri Gungor

Depending upon the technological developments, the same fast evolution has occurred in the structures of sensor networks, their composing devices which are sensor nodes, and their application areas. Those tiny, energy-constrained, mostly non-real-time data transmitting sensor nodes have evolved to more energy-containing, camera-adapted, real-time multimedia-data-transmitting devices. Developments in the usage areas and the capabilities have revealed some other problems such as time limited data transmission. In this paper, we propose a multichannel cross-layer architecture for Quality of Service (QoS) constrained multimedia sensor networks. The proposed architecture considers both the time and energy efficiency concepts. Energy efficiency is succeeded by ensuring the fair load distribution among the nodes during a real-time multimedia packet stream transmission. Besides ensuring the fair load distribution, on-time packet transmission is also assured by constructing the paths with a hard reservation technique depending on the predetermined QoS constraints. Simulations show that the proposed architecture provides higher performance than the Greedy approach and the LEERA scheme.

1. Introduction

Technological advances have provided great facilities and opportunities for human life. Many tasks have been carried out by computerized systems recently. Those computer-based tasks include surveillance control, fire preventing, health monitoring, and agricultural watering systems. There have been many developments that took place on issues such as hardware technology, signal processing, and communication protocols. Hence, it has become cumbersome to carry out those jobs by human beings.

Wireless Sensor Networks (WSNs) are one of those computerized systems that have started to take mission of man-power in the tasks, which are especially dangerous, time consuming, and expensive to perform manually. WSNs consist of a data collection mechanism called sink, the sensor nodes composing the backbone, and the communication protocols that define the way of data exchange between the devices. Sensor nodes are so small and low cost devices, such that to employ hundreds or thousands of them in a task is not costly. Those devices embody mainly three units: a sensing mechanism for gathering data from physical environment, a processor, and a battery unit with a radio subunit for data communication [1]. In addition to low cost, those networks also have the ability of selforganizing, which makes them the samples of ad hoc networks [2, 3].

In order to use thousands or millions of these sensor nodes, they must be very small to be produced with minimal costs. However, their small sizes cause some disadvantages, such as limited energy resources and coverage areas. Therefore, the energy limitation for WSNs must be considered while designing their protocols [4].

Recently, as a result of rapid technological evolutions, it is possible to equip the sensor nodes with tiny cameras and microphones to gather multimedia data from the environment. This new network type is called the Multimedia

Wireless Sensor Network (MWSN) [5, 6]. After these new technologies, new problems and requirements emerge, such as on time transmission, low loss rate, and small jitter. Obviously, like all other communication technologies, a constant QoS value must be supplied by the network and its communication protocols during the transmission of multimedia data. Most of the traditional WSNs gather and transmit the physical data, which is delay tolerant and does not require a specific service quality [6].

In this paper, we propose a multichannel, cross-layer structure, in which packet forwarding is made according to the residual energy levels and geographical coordinates of the nodes. These nodes are positioned along the paths, which ensure the QoS parameters defined at the beginning of the data transmission. In order to provide load distribution, next hop selection is done by considering the residual energy levels of the nodes in the coverage area. Besides, to increase channel utilization, overall channel frequency is divided into n data channels and a control channel, which is only used for request message and non-real-time data transmission. Multiple paths with different QoS constraints can be constructed by using multiple channels. Hence, different node sets can be employed in lifetime maximization. Additionally, higher channel utilization is succeeded by using multiple channels. Before the start of real-time data transmission, hard bandwdith reservation is employed as done in the ATM networks. Thereby, the required QoS is assured.

The rest of this paper is organized as follows. Section 2 gives brief information about the structure, features, and requirements of MWSNs. In Section 3, related studies about MWSNs are discussed. Section 4 defines the architecture we construct. Section 5 gives the simulation results, and finally Section 6 concludes the paper.

2. Multimedia Wireless Sensor Networks

MWSNs differ from traditional WSNs by their hardware and the data they transmit to the data collection center. Conventional sensor networks gather scalar physical data such as temperature, humidity, and pressure from environment and transmit this delay-tolerant data to the data collection center according to the protocols specific for WSNs [6]. Those protocols mainly consider energy conservation, thereby giving lifetime maximization. Many solutions at different levels of the communication protocol stack have been proposed. One of them is the duty-cycling method, on which several researches focus and develop additional features. In this method, nodes are arranged on a schedule, periodically sleep, and then wake up to employ in the data communication. Another method contributing to the lifetime maximization occurs in the next hop selection at the routing layer. Here, the purpose is to find an exact path towards the sink, which will contribute to the network lifetime maximization. Actually, the computation of the best load balance is important while finding a path towards the sink. This means that if all packets are sent along a specific path or several nodes located in the same region send their data along the same path, the nodes of this path lose their energies quickly. So, the transmission of the packets along different possible paths maintains the network lifetime [7, 8].

As described before, researches about WSNs mostly focus on energy conservation challenge. However, MWSNs have more challenges to be dealt with. Those challenges stem from the characteristics of data they gather from the environment. WSNs can measure one-dimensional scalar data. However, MWSNs can process two-dimensional data, which is called image. This increase in the dimension induces some other challenges to be considered. Firstly, all data gathered by cameras should be selected to reduce the amount of transmitted data. Different intelligent image processing algorithms have to be applied depending upon the application type [9]. The more complicated image processing there is, the more energy consumption occurs. Another challenge is that large amount of data requires a larger bandwidth. Consequently, multimedia data requires real-time and reliable transmission. Therefore, MWSNs require a specific QoS value, which is not a crucial issue for WSNs. Lastly, in order to prevent a collision caused by continuously sending of the packets inside the network, the obtained data must be stored inside the nodes for a while. But this situation causes another challenge because the sensor nodes are small sized devices and contain limited storage areas [10].

Challenges described eariler are general issues that researchers work on. Communication scientists generally deal with the delay and reliability aspects that define the QoS requirements of the application. As far as we know, researches about MWSNs mainly focus on satisfying those QoS parameters. However, in this paper, we propose a cross-layer approach, in which both the QoS and the energy issues are considered together.

3. Related Work

As described in the previous section, not all of the methods or protocols, which have been applied in traditional WSNs, are suitable in MWSN applications. Though it is a new area, several studies have been done to satisfy the requirements of MWSNs. In this section, we mention the studies dealing with the QoS aspect of MWSNs.

Magri et al. [11] investigated the delay and energy consumption aspects of the duty-cycling approach for MWSNs. They analyzed the power consumptions of different tasks that comprise the whole cycle. Besides, different duty-cycling configurations were tested and the energy consumption of each configuration was presented. By defining the energy consumption model for each individual task, it can be identified with the complete cycle too. Thereby, the authors tried to make estimation about the lifetime of an MWSN.

Isik et al. proposed two distinct routing methods. Those methods were contributed to prevent load balancing and thus a possible congestion. Reducing the congestion probability causes a reliable data transmission. The first method they proposed is the Load Balanced Reliable Forwarding (LBRF). LBRF considers the occupancy rate of the buffers of all neighbor nodes. The node with the smallest buffer occupancy rate is chosen as the next hop. The drawback of this method

is that if all of the neighbors' buffers have the same buffer occupancy ratei, next hop that is closest to the sink is chosen among the candidate ones. This idea does not include anything about residual energy levels of the candidate nodes. The second idea they presented is Directional Load Balanced Spreading (DLBS). DLBS is a combination of the LBRF and Directional Geographical Routing (DGR) methods [12]. In DGR method, a video stream is divided into multiple streams and transmitted over multiple disjoint paths towards the sink. However, while constructing the paths, none of the QoS parameters are considered. Traditionally, the authors considered the delay parameter while making their simulations. However, in this method, number of hops of the paths is just estimated according to an angle towards the sink. This idea may not always give the convenient result. Besides, the intelligence of the sensor nodes during the determination of the number of substreams was mentioned, but the details were left out. Here, when the estimated number of hops of a path is not correct and the packet arrives to the sink with an unreasonable delay, other packets of the same flow sent over this path will be able to be directed to other paths. DBLS employs the spatial disjoint multipath construction of the DGR method. Besides, load balance is provided by LBRF while transmitting the packets over those disjoint paths. However, end-to-end delay, which is vital for multimedia data communication, was not carefully considered in these studies.

Yaghmaee and Adjeroh [13], proposed a differentiated service model. With this model, packets are classified into two main groups, real-time and non-real-time. Non-real-time packets are also classified into multiple sub groups depending on the resource requirements of the packets. According to this service model, packets belonging to different service classes are stored in different queues. For the packets in real-time queue priority queuing and for the non-real-time packet queues weighted round robin methods are employed. This study only presented a model that is employed inside the node however, much more care should be taken for the events occur outside the node such as data communication.

González-Valenzuela et al. [14] presented a multichannel scheme for wireless sensor networks which is not a new idea. On the contrary, it has been applied in other types of ad hoc networks for a long time. They did not consider anything about QoS. In their method, two disjoint paths with different frequency conditions are constructed in order to prevent collisions which is a popular, already applied idea. Besides, lots of the recent studies have been done about how to assign those multiple frequencies more efficiently to the nodes on those multiple paths.

MMSPEED [15] was basely constructed on the idea of SPEED [16]. Satisfying the QoS is tried to be achieved in two domains: reliability and delay. Routing decisions are given locally by employing geographical routing method. Besides, depending on the QoS requirements of different packets, different paths are tried to be constructed. At the end, only a single path is found in SPEED protocol. However, in MMSPEED, multiple SPEED layers are virtually built and each of them is used for the packets with different QoS requirements. The delay parameter defined at the source node is revised dynamically at the intermediate nodes in order to satisfy the average original value predefined at source. In order to achieve a reliable packet delivery, multiple paths are used for the same packet delivery. The number of these paths changes according to the packet loss rate. When the loss rate and reliability demand increases, the number of paths that a packet is multicasted also increases. However, as the authors mentioned before, the drawback of this method is that it is an application specific scheme convenient for networks, which have lifetime of hours or at most a day, in that the scheme does not consider anything about energy consumption. The only challenge considered here is providing a certain QoS.

Mao et al. presented MRTP [17] that facilitates multiflow real-time data communication with the help of its companion protocol MRTCP. MRTP employs at the application layer. It mainly deals with supporting the applications to partition data into flows and to transmit these subdata over the paths associated with the subflows. MRTCP helps its companion by accomplishing tasks such as QoS feedback, session, and flow control. MRTP utilizes the paths maintained by its underlying multipath routing protocol.

Another QoS considering method for sensor networks was proposed by Gelenbe et al. [18–20]. Their routing method gives priority levels to the nodes in their coverage area according to their distances to the sink. High priorities are assigned to the neighbors located closer to the sink and vice versa. GPSR [21] is utilized when the packet generation rate decreases under a threshold value. Otherwise, the packets are classified and higher level priorities are assigned to the packets with higher QoS values. Those packets having higher priorities are forwarded to the next nodes. Remaining packets, which have low levels, are sent through the low level nodes randomly or fairly.

Saxena et al. [22] proposed an MAC protocol that considers both the energy conservation and QoS. Nodes adaptively adjust their contention window sizes according to the QoS that the packet transmission requires. Besides, duty-cycle mechanism of the nodes is also adaptively rearranged due to the QoS parameters. During delay-tolerated nonreal packet transmissions, nodes can be put into sleep state for longer times and thereby can save more energy.

A multipath power efficient transmission method was proposed by Politis et al. [23], for video transmission over sensor networks. In that study, two scheduling algorithms are proposed. First algorithm is the baseline scheduling algorithm that calculates multiple paths towards the sink that can fulfill the bandwidth requirements of the transmission. Due to the possibility of total aggregated bandwidth of the specified paths not being able to fulfill the QoS, a packet elimination mechanism, which weeds out some of the packets depending on their importance levels, is utilized. The second method, power aware packet scheduling, can estimate the residual energy levels of the nodes in the network and adaptively adjusts packet elimination according to the bandwidth requirements of the transmission and the residual energy levels of the nodes in the network. Thereby, network lifetime is tried to be maximized.

AGEM routing protocol [24] was proposed as a developed version of GPSR protocols to support multimedia data

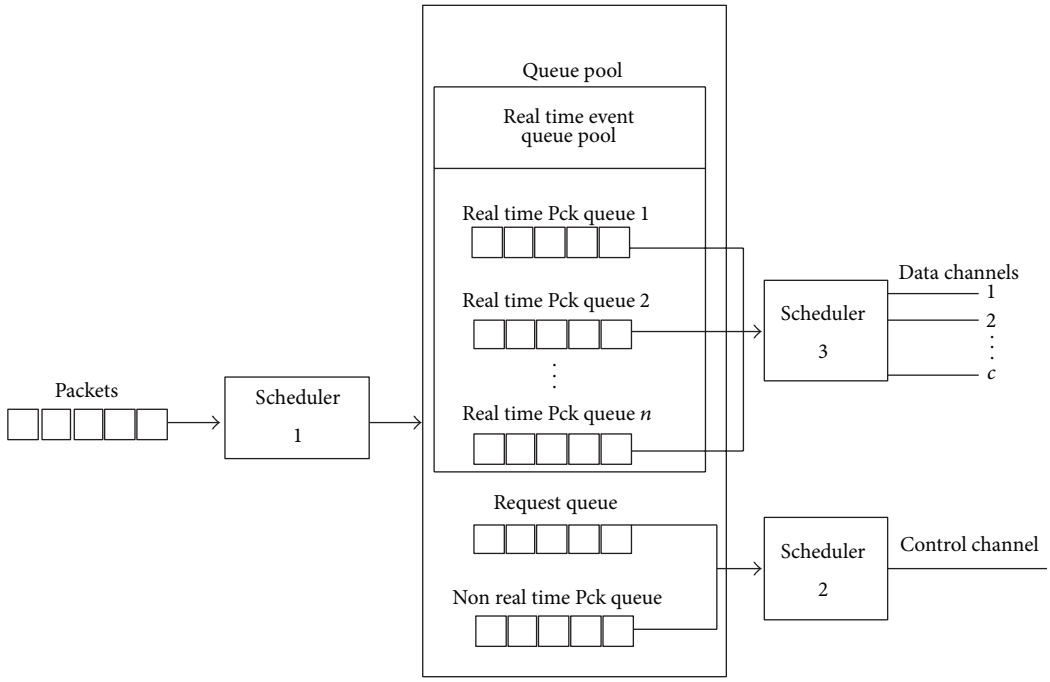


Figure 1: Schedulers utilized in the system.

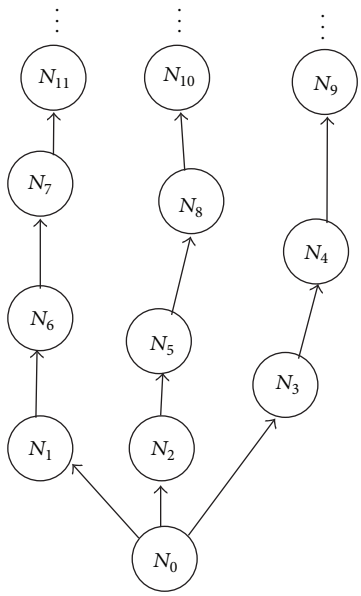


Figure 2: A sample scenario.

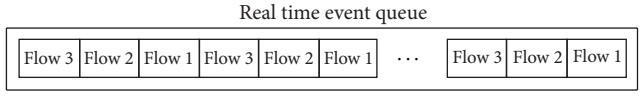


Figure 3: Flow segmentation.

transmission requiring certain QoS values. Enhancement is accomplished by adding a load-balancing feature to GPSR method. In GPSR method, packets emerge from a single node

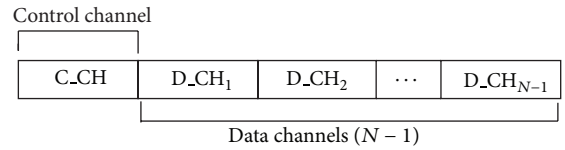


Figure 4: Partition of the total available bandwidth into N non-overlapping channels.

and always follow the same path towards the sink. In contrast, in the AGEM protocol, nodes choose next hop on the path to the sink according to a policy. The policy comprises four major criteria:

(i) residual energy levels of the nodes;

(ii) number of nodes being visited before the existing node;

(iii) distances between neighbor nodes and the existing node;

(iv) statistics about the packets belonging to the same flow.

The most popular approach utilized by the researchers is transmitting a video stream over multiple paths. In another sample of this approach [25], a heuristic method aims to find those multiple paths, which satisfy the QoS required by the source node towards the sink. According to their simulation results, possibility of satisfying the required QoS is reasonably higher than the shortest-path or the shortest-feasible methods. The second contribution of the study is a video segmentation and a scheduling algorithm. By utilizing this algorithm, source node segments the original stream,

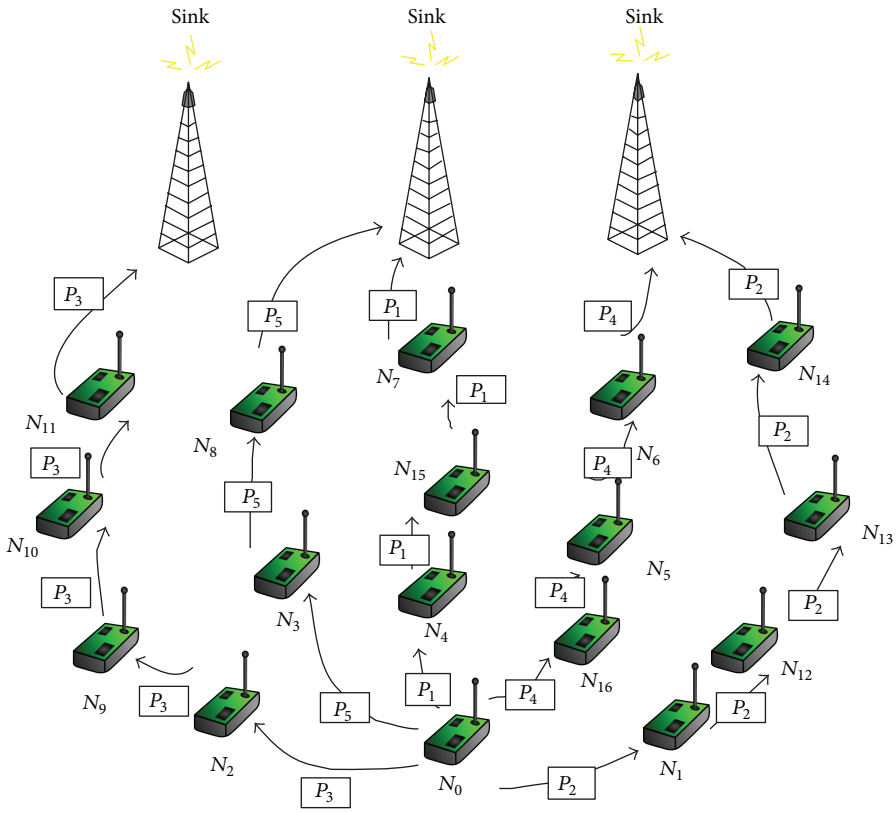


Figure 5: Next hop selection by LEERA-MS.

and the packets are sent through the paths defined by the heuristic mentioned earlier. After that, the sink can combine the arriving packets at the earliest time.

4. The Proposed Architecture

The architecture we proposed comprises five major components, each of which is utilized in order to prolong network lifetime and satisfy the required QoS. These essential components are briefly identified in the following subsections.

4.1. Scheduler. Recent sensor nodes are capable of gathering different types of information, such as scalar and multimedia data. Hence, different types of data require different types of QoS. Each node, whether it is a data generator or an intermediate one, contains schedulers that classify the emerging or arriving packets into different queues and pick them out from the queues according to their priorities. There are three types of packets processed in the queues. Route request message packets are employed for constructing paths and reserving resources before sending real time packets. They have the highest priority in the system, so as to construct the path and start the real-time packet transmission immediately. The second type is the real-time packet emerged during an unusual event such as surveillance applications. The priority level of these packets stays in the middle of the hierarchy. The lowest priority leveled packet is the non-real-time packet. Non-real-time packets emerge periodically and contain delay-tolerant data. Therefore, they can suffer from delays encountered in the queues. The structure of the scheduler subsystem is given in Figure 1.

As it is depicted in Figure 1, there are three schedulers employed in the architecture. Scheduler 1 classifies the arriving packets and places them into the appropriate queues. During data transmission, real-time packets are transmitted by the data channels. Non-real-time packets and the route-request messages are conveyed over the control channel. Therefore, Scheduler 3 is concerned only with the real-time packet queues. Packets are pulled from the real-time queues in a round-robin manner. Scheduler 2 pops only from the request queue until no request message remains. After that, it comes the turn for the non-real-time packets.

4.2. Adaptive Subflow Generation. In Multimedia Sensor Networks, such as utilized for surveillance, a continuous packet stream emerges after an unusual event occurs. If this stream is transmitted through a single path, the nodes on this path deplete the energy. However, in order to provide the load balance, if the required QoS is supplied, then the original stream is segmented into a number of flows. This flow number is defined according to the number of paths constructed during the bandwidth reservation. Each packet

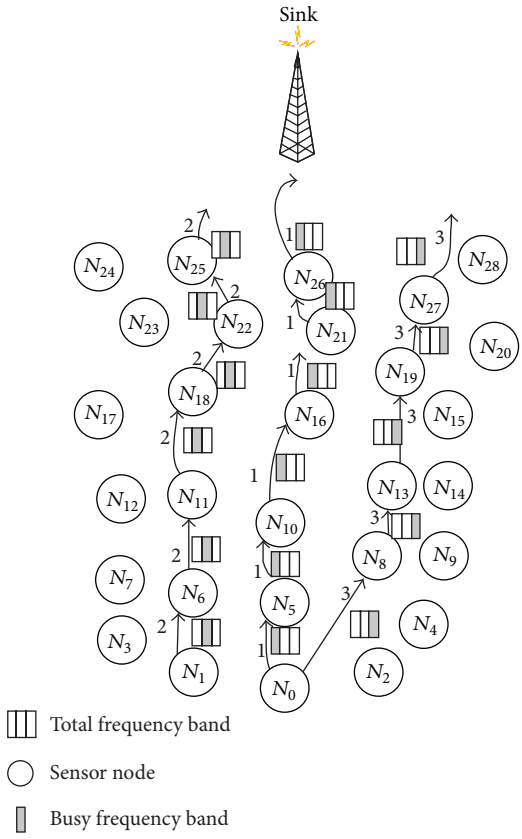


Figure 6: Scenario 1: Real-packet transmission, number of nonoverlapping channels = 5, number of hops traveled (QoS parameter) = 6, and number of packets = 3.

in the queue is labeled with a flow number in the round-robin manner and sent over the path reserved for the related flow.

If there are three paths that satisfy the required QoS, as in Figure 2, then the multimedia data flow is segmented into three subflows as shown in Figure 3.

4.3. Multichannel Structure. Single channel structures do not provide the same fair channel utilization among the nodes as do the multiple channel architectures. Since the wireless medium is a broadcast environment, during a packet transmission between a node pair, all the other nodes deployed inside the coverage area of this pair cannot send/receive any packet which is called CSMA/CA. Contention-based protocols like CSMA/CA are not convenient for multimedia applications carried on static networks. Since the packets of the stream should arrive to the destination in a definite time, a standard resource usage opportunity should be provided for these real-time packets. That is to say, the ability of using the common resources in parallel increases the network throughput and reduces the delay [26–29]. In the scheme we proposed, total bandwidth is divided into N nonoverlapping channels. One of the channels is dedicated to carrying control messages and the non-real-time data. Remaining $N - 1$ channels are utilized for real-time data transmission. Hence, each channel is assigned a bandwidth of W/N as represented clearly in Figure 4. Besides, all nodes are assumed to be equipped with N half-duplex transceivers, each assigned to a single channel statically.

4.4. Resource Reservation and Route Discovery. Some of the multichannel MAC protocols, especially the earlier ones, do not concern with the QoS and look through the data transmission as a single task. Thus, they make channel allocation and handshaking per packet. However, for data streams such as multimedia, instead of channel allocation and deallocation per packet, reservation of the channels until the end of the stream would be more efficient. Thus, the nodes suffer less overhead caused by the handshaking mechanism which occurs at the beginning of each transmission [30].

After the segmentation of the stream into flows, a route request message is created for each flow and put into the request queue. Hence, the number of request messages depends on the number of the flows created. After that, each request is pulled and sent over the control channel to the next hop defined by the routing algorithm.

In the scheme we proposed, resource reservation is made during the path construction. Paths towards the sink are discovered by using an ad hoc on-demand distance vector (AODV) [31] based route discovery algorithm. In contrast to AODV method, requests are not sent to all neighbors as flooding. Next hops are defined according to our load-balanced routing algorithm. *Number of hops traveled* is considered as the QoS parameter. Resource reservation is made per flow. In other words, once a path-channel pair is defined and the resources are reserved for a flow, each packet belonging to that flow follows the same path along the defined channel. The node getting the request looks at the number-of-hops-traveled field and checks whether the constraint is exceeded or not. If the requested value is exceeded, then a *NackForRouteRequest* message is sent back to the previous hop. This *NackForRouteRequest* message follows back the path up to the source node, and all the nodes on the path release their resources reserved for that request. Conversely, if the QoS parameter is not exceeded, the value in the number-of- hops-traveled field is increased, and the related resources are reserved. Then, the request message is forwarded to the next hop defined by the routing algorithm.

4.5. Load Balanced Routing with a Certain QoS. As mentioned in the previous section, path discovery is made by an AODV algorithm. However, the major difference is that request message is not sent to all of the neighbors. The receiver of the request message is identified according to a developed version of the load balanced routing algorithm (LEERA-MS) we proposed before [1]. We have proposed LEERA-MS for traditional sensor networks, which concern with non-real-time scalar data. Since the data being transmitted is best-effort, the major idea while sending a packet becomes to provide load balance. Hence, the first criteria while choosing the next hop is the residual energy level of the neighbors. If all of them include the same amount of energy, their distance to the corresponding sink of the sending node

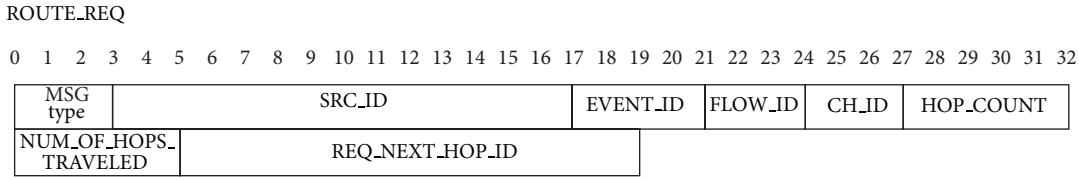


Figure 7: Route request message.

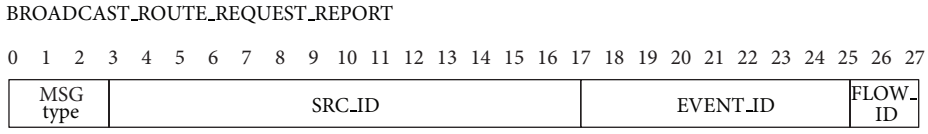


Figure 8: Route request report.

is considered. For the first packet, the shortest path is chosen as the next hop, which starts with N_4 in Figure 5. For the following packets, the furthest one to the corresponding sink of the sending node is chosen as the next hop. By this way, the paths located at the edges of the network are employed. When the data transmissions occur, paths located in the center of the topology are used for transmission. Thus, both a possible occurrence of a collision and congestion in the network are prevented, and packet transmission load is distributed over as many nodes as possible.

As clearly seen in Figure 5, paths

N_4-N_{15}-N_7 for P_1,
N_1-N_{12}-N_{13}-N_{14} for P_2,
N_2-N_9-N_{10}-N_{11} for P_3,
N_{16}-N_5-N_6 for P_4,
N_3-N_8 for P_5 are used subsequently.

In the developed version (QS_LEERA-MS), paths are defined depending on the type of packet being sent. Non-real-time packets are sent according to LEERA-MS method. For real-time transmissions, as we described eariler, firstly a path must be constructed and resources should be reserved. In contrast to LEERA-MS, paths that are located in the center of the topology are chosen first. By the time the QoS constraints are fulfilled, the paths at the edges are constructed. The idea in LEERA-MS is out-to-in, and in contrary, it is in-to-out in QS-LEERA-MS.

The operational description of QS-LEERA-MS is shown later.

As seen in Figures 6 and 3, packets emerge from N_0 with the QoS parameter as number of hops traveled = 6. Total bandwidth is divided into 3 non-overlapping channels. The format of a request message is shown in Figure 7.

SRC_ID denotes the packet generator node. It is considered in the system that a single node can generate more than one stream. Thus, for each stream, a distinct EVENT_ID is assigned. FLOW_ID represents the flow for which the resources are being reserved. HOP_COUNT identifies the number of hops traveled up to the current node. The QoS constraint is defined as the number of nodes traveled and is denoted by the field NUM_OFHOPS_TRAVELED. The receiver of the request message is identified by the routing algorithm, and the identification number of the next hop is put into the field REQ_NEXT_HOP_ID.

Request messages are sent over the control channel. According to QS-LEERA-MS, since all candidate next hops have the same amount of residual energy, the first request is sent through the shortest path, which begins with node N_5. When N_5 gets the request, it firstly checks whether the value inside the field number-of hops traveled exceeds the defined value or not. If it does, N_5 replies back to N_0 with a *NackForRouteRequest* message. Otherwise, N_5 increases the value of number of hops traveled by one. If again all neighbors of N_5 have the same amount of residual energy levels, the request is forwarded to N_{10}. The operation continues in this wise until the request arrives to Sink3. Sink3 sends a broadcast replication message as soon as it gets the request packet. The structure of this broadcasted report is given in Figure 8.

Obviously, this message announces to all nodes in the network that the path belonging to the substream denoted with the triple (SRC_ID, EVENT_ID, FLOW_ID) is constructed successively. Each node employed along the path creates a record in its routing table that holds the information about the constructed route. The structure of each record is given in Figure 9.

As the source node gets the request report, it puts the first real-time packet (the structure given in Figure 10) on the way. Since the packet is the first in the substream, the field SEQ_NO_IN_FLOW equals 1. The real-time packet 1 is sent to N_5 over channel 1. The second path construction attempt starts with creating another request message. Data channels are being checked, and the first empty one is chosen as the candidate data channel. In the scenario presented in Figure 6, the channel with ID = 2 is chosen. This time, the request message is sent to N_1. The operations for the second request repeat in the same way as in the message transmission of the first request. Lastly, the third request is generated and sent to N_8 by reserving channel 3 for the transmission of real-time

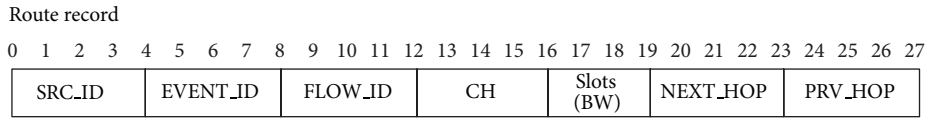


Figure 9: Route record.

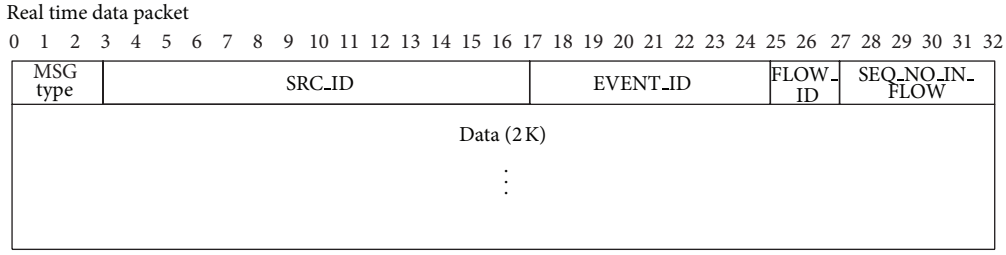


Figure 10: Real-time data packet.

data packets belonging to flow 3. After all path construction attempts are performed, all data stream is divided into n number of flows. Here, n denotes the number of distinct paths constructed successfully.

In another scenario, as similar to the one illustrated in Figures 6 and 5, packets are generated at N_0. Though there are 5 non-overlapping channels and it is possible to construct a path beginning with N_1 and N_2, requests sent via those nodes result in an NACK message sent by nodes N_{23} and N_{20}. The reason for the NACK is that the value of QoS parameter is exceeded. The predefined QoS parameter value is 6. When the request messages follow the paths starting with N_1 and N_2 and arrive at the nodes N_{23} and N_{20}, the value in the field NUM_OF_HOPS_TRAVELED is already exceeded. Thus, a NACK message with the format given in Figure 11 is sent back by the nodes N_{23} and N_{20} to their former nodes, N_{17} and N_{15}. These NACK messages are backwarded until they arrive to the originator node N_0. Each node on the way getting this NACK message releases its resources reserved for this attempt.

Figure 12 gives another scenario in which number of hops traveled (QoS parameter) is more delay tolerated. Thus, in addition to the paths constructed in the first scenario, two more routes are utilized in this way. By employing two additional paths, bandwidth utilization also increases. Besides, transmission load is distributed over the nodes more fairly than the previous scenario.

5. Evaluation and Performance Analysis

We evaluated the performance of our proposed method by comparing it with the methods of Greedy and our earlier proposal LEERA-MS. As the authors exposed before [32, 33], due to the fact that much more energy is consumed during the data communication compared with the data processing, sensor nodes consume much more energy during data transmission. As far as we know, the authors have only discussed their contributions by concerning the QoS, such as delay or reliability. However, the major issue to be considered for sensor networks is the energy scarcity problem. Therefore, we did not only concern the QoS in our method. We also considered the lifetime maximization by implementing a load distributed routing algorithm. We did the calculations of energy consumptions during the data communication both in the sending and receiving stages according to the following formulas:

$$E_{\text{Total}}(N) = E_{\text{snd}}(N) + E_{\text{rcv}}(N),$$

$$E_{Tx}(l,d) = E_{Tx\text{-elec}}(l) + E_{Tx\text{-amp}}(l,d),$$

$$E_{Tx}(l,d) = \begin{cases} (l * E_{\text{elec}}) + \left(l * \varepsilon_{fs} * d^2\right), & d < d_o, \\ (l * E_{\text{elec}}) + \left(l * \varepsilon_{mp} * d\right), & d \geq d_o, \end{cases} \quad (1)$$

$$E_{\text{rcv}}(N) = l * E_{\text{elec}}.$$

We prepared our simulations in JAVA. It is assumed that there are n non-overlapping channels and each node has two half-duplex radios. The radio with low speed and low power is statically assigned to the control channel. The other radio, which is with high speed and power, dynamically switches to the convenient channel after the approval of the route request. Besides, the sensor nodes have the ability of sensing both scalar and multimedia data. The parameters applied in simulation are represented in Table 1.

5.1. Performance Analysis of Delay and Energy Consumption versus Data Rate. Figure 13 presents the comparison of our new model with the Greedy and our earlier method LEERA-MS in terms of delay and energy consumption. Data rate affects only the network lifetime. Other parameters, such as bandwidth utilization rate, end-to-end delay, or load balance, are not affected by the data rate.

As clarified in Figure 13, the amount of energy wasted by the most energy spender node in the network is the smallest in QS-LEERA-MS. That yields the network lifetime maximization. The network using Greedy approach as the routing method has the shortest network lifetime, because of that the path employed for the first packet's transmission is also used for the remaining packets of the same stream. However, in LEERA-MS and QS-LEERA-MS, packets use

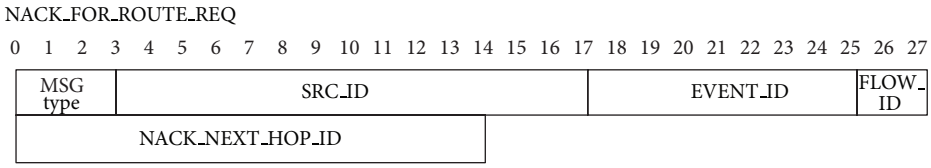


Figure 11: NACK for route request.

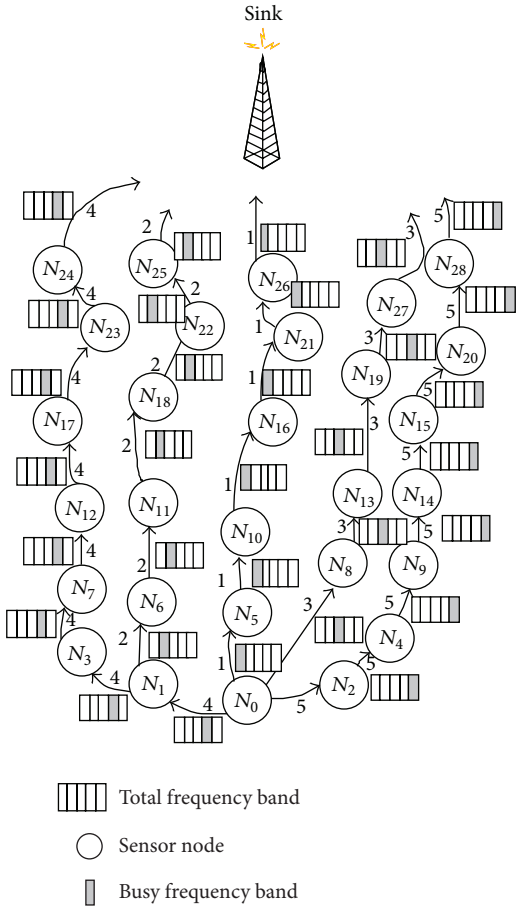


Figure 12: Scenario 2: real-packet transmission, number of non-overlapping channels = 5, number of hops traveled (QoS parameter) = 7, and number of packets = 5.

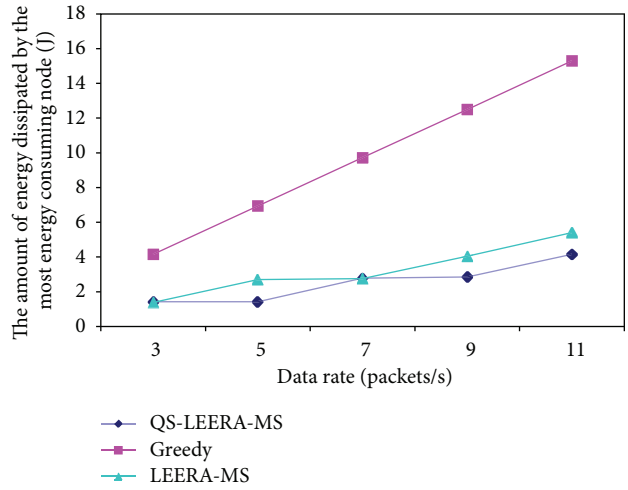


Figure 13: Comparison of energy consumptions.

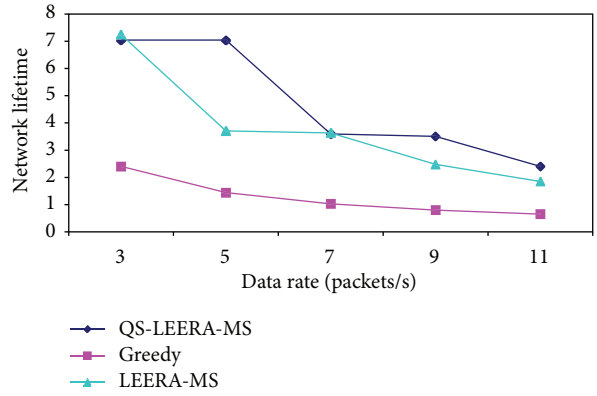


Figure 14: Comparison of network lifetimes.

different paths in order to arrive to the sink. Network lifetime comparison is depicted in Figure 14.

As it is mentioned before, increase in the number of non-overlapping channels contributes to the fair distribution of the load balance. Besides, bandwidth utilization increases due to enabling more than one node using the common broadcast medium at the same time. By utilizing multiple channels, it becomes possible to divide the original stream into multiple subflows and transmit these flows over distinct paths by considering the QoS levels. As illustrated in Figure 15, with the increase in the number of channels operated, fair distribution in the load balance is more likely to be provided. This is because much more nodes will share the mission of conveying the packets towards the sink.

However, employing multiple longer paths towards the sink causes an increase in the end-to-end delay. Nevertheless, during the path construction and resource reservation stage, QoS is already considered. Paths violating the QoS are not permitted.

Another factor affecting the network lifetime is the requested QoS. As mentioned previously, as the QoS constraint (maximum number of hops that a packet can pass over towards the sink denoted by number of hops traveled in Figure 16) gets loose, more alternative paths emerge. Thus, more nodes take in charge during packet relaying process that

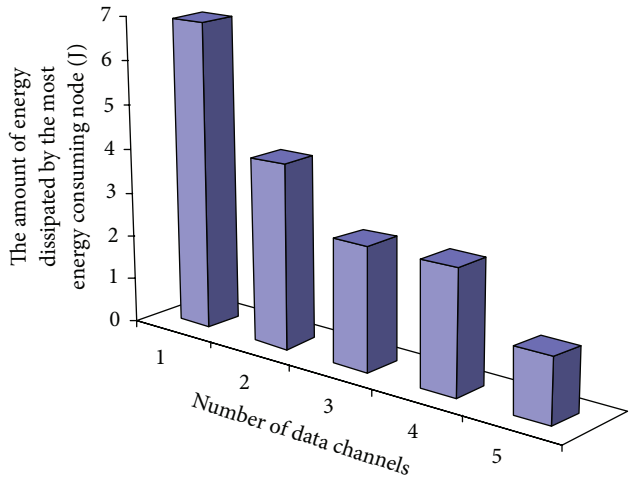


Figure 15: Effect of the number of non-overlapping data channels on the energy consumption.

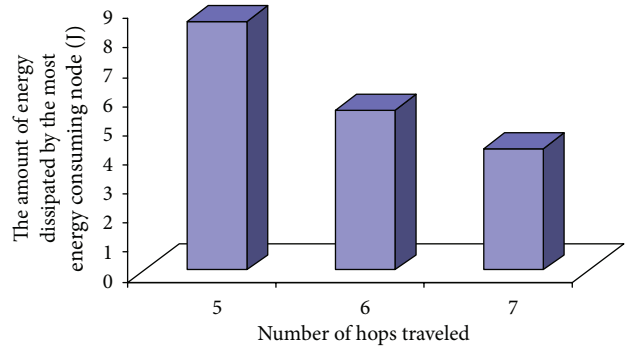


Figure 16: Effect of the requested QoS to the network lifetime.

provides load balancing. As shown in Figure 16, loosening in the QoS constraint prolongs the network lifetime.

Besides, loosening in the service quality also increases the throughput because that providing by utilizing a certain number of channels, and multiple paths can be used for relaying packets to the sink. Therefore, packets do not have to travel along the same path and wait in the queues of the nodes located on this path. Consequently, as shown in Figure 17, utilizing multiple paths by loosening the service quality causes an enhancement in the throughput.

6. Concluding Remarks

Energy scarcity is the major problem of WSNs. Hence, traditional methods and protocols used for conventional ad hoc networks are not convenient for WSNs. These traditional approaches do not mostly concern with the energy issue. However, while designing architectures or protocols to employ in WSNs, energy scarcity problem should also be considered in addition to concerning the traditional issues faced with in ad hoc networks.

With the evolution of plain sensor networks into MWSNs, additional challenges have emerged to be considered. Those sensing-capability-enhanced nodes have to transmit their

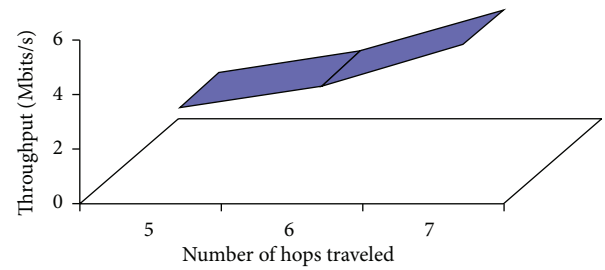


Figure 17: Effect of the requested QoS on the throughput.

Table 1: Simulation parameters.

Control CH radio transmission rate	30 kbps
Data CH radio transmission rate	250 kbps
E_{elec}	50 nj/bit
Ef_s	10 pJ/bit/m^2
Em_p	0.0013 pj/bit/m^4
Real-time data rate	~80 Kbits/s
d_0 (threshold distance value)	30 m
Request message size	51 bits
Video frame size	16424 bits
Non-real-time packet size	350 bits
Route report size	27 bits
Nack-for-route-request size	41 bits

captured multimedia data to the sink in some reasonable delays. Thus, while designing a protocol for MWSNs, QoS should also be considered as well as the energy scarcity issue.

In this paper, we proposed a multichannel cross-layer architecture with a novel load balanced routing method. The main feature of our scheme is that multiple path construction is made possible by employing multichannel structure. A single multimedia stream including multiple video frames is segmented into multiple flows according to the number of paths constructed with respect to not exceeding the QoS constraint defined in the request messages. The key point is that while constructing these multiple paths, a packet must travel maximum number of hops. We utilized this parameter in our simulations as the QoS criteria. Because, an increase in the number of hops traveled towards the sink causes additional service delays at the additional nodes. These additional service waiting delays cause an increase in the end-to-end delay. During path construction, if the QoS constraint is not exceeded, the resources on the path are reserved for that particular flow.

We compared the performance of our scheme (QS-LEERA-MS) with the Greedy method and our earlier method LEERA-MS. As the simulation results clarified, the network in our method significantly prolonged its lifetime when compared to the networks applying Greedy or LEERA-MS. Simulation results also stated that the packets transmitted over distinct paths prevented possible congestions in a single channel—single path architecture. Thereby, throughput of the system, which is a significant factor for real-time data transmission, is also increased.

References

[1] T. Cevik, A. H. Zaim, and D. Yıltaş, "Localized power aware routing with an energy efficient pipelined wakeup schedule for wireless sensor networks," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 20, pp. 964–978, 2012.

[2] T. Cevik and A. H. Zaim, "EETBR: energy efficient token-based routing for wireless sensor networks," *Turkish Journal of Electrical Engineering & Computer Sciences*. In press.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–105, 2002.

[4] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.

[5] M. A. Yiğitel, O. D. İncel, and C. Ersoy, "QoS-aware MAC protocols for wireless sensor networks: a survey," *Computer Networks*, vol. 55, no. 8, pp. 1982–2004, 2011.

[6] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[7] S. Işık, M. Y. Dönmez, and C. Ersoy, "Cross layer load balanced forwarding schemes for video sensor networks," *Ad Hoc Networks*, vol. 9, pp. 265–284, 2011.

[8] C. K. Toh, A. N. Le, and Y. Z. Cho, "Load balanced routing protocols for ad hoc mobile wireless networks," *IEEE Communications Magazine*, vol. 47, no. 8, pp. 78–84, 2009.

[9] S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "MeshEye: a hybrid-resolution smart camera mote for applications in distributed intelligent surveillance," in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 360–369, April 2007.

[10] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Advances in Multimedia*, vol. 2009, Article ID 640386, 21 pages, 2009.

[11] C. B. Magri, R. Manduchi, and K. Obraczka, "Energy consumption tradeoffs in visual sensor networks," in *Proceedings of the 24th Brazilian Symposium on Computer Networks (SBRC '06)*, 2006.

[12] M. Chen, V. C. M. Leung, S. Mao, and Y. Yuan, "Directional geographical routing for real-time video communications in wireless sensor networks," *Computer Communications*, vol. 30, no. 17, pp. 3368–3383, 2007.

[13] M. H. Yaghmaee and D. Adjeroh, "A model for differentiated service support in wireless multimedia sensor networks," in *Proceedings of the 17th International Conference on Computer Communications and Networks (ICCCN '08)*, pp. 881–886, August 2008.

[14] S. González-Valenzuela, H. Cao, and V. C. M. Leung, "A multi-channel approach for video forwarding in wireless sensor networks," in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC '10)*, January 2010.

[15] E. Felemban, C. G. Lee, and E. Ekici, "MMSPEED: multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738–754, 2006.

[16] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: a stateless protocol for real-time communication in sensor networks," in *Proceedings of the 23th IEEE International Conference on Distributed Computing Systems*, pp. 46–55, May 2003.

[17] S. Mao, D. Bushmitch, S. Narayanan, and S. S. Panwar, "MRTP: a multiflow real-time transport protocol for ad hoc networks," *IEEE Transactions on Multimedia*, vol. 8, no. 2, pp. 356–369, 2006.

[18] E. Gelenbe and E. C.-H. Ngai, "Adaptive QoS routing for significant events in wireless sensor networks," in *Proceedings of the 5th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '08)*, pp. 410–415, October 2008.

[19] L. Hey and E. Gelenbe, "Adaptive packet prioritisation for wireless sensor networks," in *Proceedings of the Next Generation Internet Networks (NGI '09)*, July 2009.

[20] E. Gelenbe and E. C.-H. Ngai, "Adaptive random re-routing for differentiated QoS in sensor networks," *The Computer Journal*, vol. 53, no. 7, pp. 1052–1061, 2010.

[21] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, August 2000.

[22] N. Saxena, A. Roy, and J. Shin, "Dynamic duty cycle and adaptive contention window based QoS-MAC protocol for wireless multimedia sensor networks," *Computer Networks*, vol. 52, no. 13, pp. 2532–2542, 2008.

[23] I. Politis, M. Tsagkaropoulos, T. Dagiuklas, and S. Kotsopoulos, "Power efficient video multipath transmission over wireless multimedia sensor networks," *Mobile Networks and Applications*, vol. 13, no. 3-4, pp. 274–284, 2008.

[24] S. Medjiah, T. Ahmed, and F. Krief, "AGEM: adaptive greedy-compass energy-aware multipath routing protocol for WMSNs," in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC '10)*, January 2010.

[25] J. Chen, S.-H. G. Chan, and V. O. K. Li, "Multipath routing for video delivery over bandwith-limited networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, pp. 1920–1932, 2004.

[26] J. Mo, H.-S.W. So, and J. Walrand, "Comparison of mutichannel MAC protocols," *IEEE Transactions on Mobile Computing*, vol. 7, pp. 50–65, 2008.

[27] S.-L. Wu, Y.-C. Tseng, C.-Y. Lin, and J.-P. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *The Computer Journal*, vol. 45, no. 1, pp. 101–110, 2002.

[28] A. Nasipuri, J. Zhuang, and S. R. Das, "A multichannel CSMA MAC protocol for multihop wireless networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, pp. 1402–1406, 1999.

[29] M. A. Marsan and D. Roffinella, "Multichannel local area networks protocols," *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 5, pp. 885–897, 1983.

[30] P.-J. Wu and C.-N. Lee, "Connection-oriented multi-channel MAC protocol for ad-hoc networks," *Computer Communications*, vol. 32, no. 1, pp. 169–178, 2009.

[31] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 90–100, February 1999.

[32] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 40–50, 2002.

[33] G. Pottie and W. Kaiser, "Wireless integrated network sensors," *Communication of ACM*, vol. 43, no. 5, pp. 51–58, 2000.