A New Intra-Cluster Scheduling Scheme for Real-Time Flows in Wireless Sensor Networks

Gohar Ali [1], Fernando Moreira [2], Omar Alfandi [3], Babar Shah [3,*] and Mohammed Ilyas [1]

[1]   Department of Information Systems and Technology, Sur University College, Sur 411, Oman;
      goharali@suc.edu.om (G.A.); md_ilyas@suc.edu.om (M.I.)
[2]   REMIT, Universidade Portucalense, 4200-072 Porto, Portugal & IEETA, Universidade de Aveiro,
      3810-193 Aveiro, Portugal; fmoreira@uportu.pt
[3]   College of Technological Innovation, Zayed University, Abu Dhabi 144534, UAE; omar.alfandi@zu.ac.ae
*   Correspondence: babar.shah@zu.ac.ae

Abstract: Real-time flows using time division multiple access (TDMA) scheduling in cluster-based wireless sensor networks try to schedule more flows per time frame to minimize the schedule length to meet the deadline. The problem with the previously used cluster-based scheduling algorithm is that intra-cluster scheduling does not consider that the clusters may have internal or outgoing flows. Thus, intra-cluster scheduling algorithms do not utilize their empty time-slots and thus increase schedule length. In this paper, we propose a new intra-cluster scheduling algorithm by considering that clusters may have having internal or outgoing flows. Thus, intra-cluster scheduling algorithms do not differentiate the intra-cluster time slots and utilize their empty time slots. The objective is to schedule more flows per time frame, to reduce schedule length and improve the acceptance rate of flows. Simulation results show that the acceptance rate of the proposed scheme has a higher performance than the previous scheme.

Keywords: scheduling; intra-cluster; real-time; interference; clustering

1. Introduction

WirelessHART standards for sensor-actuator networks have attracted interest for real-time communication [1]. These protocols use a centralized approach for the transmission schedule and avoid concurrent transmissions with the same channel. The approach uses a different priority scheme and achieves the real-time communication through time division multiple access (TDMA) scheduling. This standard increases the amount of delay and limits scalability. In the quality of service-aware cluster-based data sending schemes [2], each node transmits the sensing data to its cluster head, which in turn, aggregates that data and sends the results to the base station. The whole scheme consists of intra-cluster data reporting and inter-cluster control methods. For intra-cluster data, only a few member nodes were selected as reporting nodes to meet the required throughput and save energy for other rounds. At the same time, for inter-cluster controls, delay-bound data were routed using a minimum number of hops consisting of a combination of cluster heads and member nodes. The real-time query scheduling algorithm [3] assumes a pre-given routing tree for different scheduling strategies by considering the trade-offs between prioritization and throughput. The problem with TDMA scheduling for flat wireless sensor networks (WSNs) is that they are not scalable to the large networks, and have less slot reuse due to global view of the network [4] or large message exchanges. Hence, they lack in performance metrics like delay which is not efficient for real-time communication. Therefore, TDMA scheduling under cluster architecture is proposed in WSNs [5–9]. Adaptive distributed randomized [10] is a cluster-based TDMA scheme for wireless sensor networks.

The slots are assigned to the node based on the one-hop and two-hop neighbor's information. The cluster heads are allocated more slots, and hence improve channel utilization and throughput. However, for energy balance, the cluster head is alternated by other cluster members, which increases overhead due to the reassignment of slots.

The cluster-tree model [10] based on a single-channel and time division cluster schedule—proposes a receiver-oriented scheduling algorithm that offers bounded latency for data gathering applications and reduces the interference among clusters. In [11], real-time query services were scheduled where data are routed from many-to-one communication by avoiding the interference by using the interference graph. Cluster-based real-time communication scheme [12] represented interference by using graph theory, while graph coloring is used to avoid the collision. In [13], real-time packets are scheduled under the WirelessHART model. This model contains multiple channels, avoiding the interference by allowing one transmission in each channel in a time slot.

In [14], an adaptive clustering scheme was proposed for event-driven applications. In this scheme, the clusters were adapted according to the change in event sensing so that cluster-based structure proved performance in real-time WSNs. However, information accuracy was reduced by combining the packets. In [15], the real-time capacity for wireless network by assuming ideal TDMA was proposed to deliver the data by their deadlines. They provided relevant contributions to the theoretical understanding of real-time capacity of multi-hop WSNs. However, the applicability of results to a real WSN may not be applied in practice due to implementation of a medium arbitration with zero overhead.

Real-time WSN are becoming an ever more important emerging communication infrastructure for real-time applications. Examples of such applications are habitat monitoring, forest monitoring, disaster management, seismic monitoring and smart cities [16]. These types of systems are time-critical and performance-sensitive. In time-critical event detection applications, detecting events after the predefined time frame may have catastrophic consequences. Therefore, data must reach the monitoring stations before the expiration of the deadline [17]. In real-time wireless sensor networks, the media access control (MAC) layer plays an important role in determining the channel access delay. The techniques to handle the channel access in WSN are mainly divided into contention-based and TDMA-based schemes. In a contention-based MAC—due to the distributed and random back-off nature—it is difficult to provide deterministic channel access guarantees, resulting in unpredictable delays which are inappropriate for real-time system. Compared to contention-based schemes, TDMA-based MAC schemes provide a deterministic medium access delay through time slot scheduling. Therefore, many research works focus on the TDMA-based MAC approach for real-time communications in WSNs [18,19].

In WSNs, real-time communication is more difficult to achieve due to the natural property of node interference in broadcast wireless communication. The communication between pair of nodes is affected by other transmitting nodes within the interference, thus, interference-aware scheme becomes important in WSN. Especially, when it comes to guaranteeing real-time delivery, therefore contention-free scheme such as time division multiple access (TDMA) is preferred due to deterministic operation [20,21]. In adaptive DRAND [22] the scheduler assign slot to node across 2-hop neighborhood to avoid the interference while in [23] assign colors to node different from 2-hop neighbor node. The mentioned scheme assigns slots to nodes across 2-hop or 3-hop away to avoid the interference. That consider that if the time slots are not used in 2-hop or 3-hop neighborhood then that time slots can be reused. However, these schemes do not consider the concurrent transmission of nodes in the network. This is due to cumulative nature of interference, where trans- missions of other nodes that are n-hop away can interference with a transmission as they transmit simultaneously.

Real-time communication in WSNs have been addressed previously in [24–28]. Real-time query services are scheduled in [16] where data are routed from many-to-one communication by avoiding interference using the conflict graph. In [24], under the Wireless Hart model, real-time packets are scheduled by avoiding the interference using multiple channels, therefore allowing one transmission in each channel in a time slot. In [25], a conflict-free real-time scheduling algorithm for periodic real-time

flows is provided with consideration of interference in the network. Furthermore, cluster-based real-time communication scheme characterized interference by using graph theory, while graph coloring is used to avoid the collision. Similarly, in [26], three cluster-based TDMA scheduling algorithms are proposed for real-time flows. The representative approach has been described to schedule real-time data; however, some schemes have centralized approach and therefore are not scalable. Cluster-based coloring schemes for real-time flow increase the schedule length and hence are not suitable for a real-time system. The cluster-based real-time flows are scheduled in three different time slots, namely IntraSend, InterComm and IntraRecv slot to avoid interference [27]. However, this scheme uses static scheduling and does not consider the incoming and outgoing flows in the intra-cluster scheduling therefore some flows cannot utilize the available empty time slots. Several TDMA scheduling algorithms have been proposed to transmit data from source to destination using multi-hop [25,28,29]. Two centralized, node-based and level-based TDMA scheduling algorithms are proposed in [25]. In both algorithms, base station assigns slots to nodes by considering interference relation. These algorithms attempt to find TDMA schedule that minimize the number of time slots. In [28,29], conflict- free TDMA scheduling algorithms for multihop intra- and inter- cluster are provided. The algorithm schedules every node across 3-hop neighbor for the purpose to avoid the interference, hence to improve the delay, throughput and energy efficiency. Similarly, in some TDMA scheduling the knowledge of 2-hop neighborhoods is assumed to avoid the interference.

In this paper, we propose a new intra-cluster scheduling scheme to schedule the flows from source to destination node in the TDMA time slots using three scheduling algorithms, namely the IntraSend, InterComm and IntraRecv algorithms. The IntraSend scheduling algorithm schedules the data in the IntraSend time slots from the source node to the source cluster head. The InterComm scheduling algorithm schedules the data in the InterComm time slots from the source cluster head to the destination cluster head. The IntraRecv scheduling algorithm schedules the data in the IntraRecv time slots from the destination cluster head to the destination node. Similarly, for the outgoing cluster flows in IntraSend scheduling, the proposed scheme utilizes both the IntraSend and IntraRecv time slots in the IntraSend scheduling. Furthermore, if the cluster has incoming flows the proposed IntraRecv scheduling can utilize both the IntraSend and IntraRecv time slots. Thus, the proposed scheme is scheduling more flows per time frame and reduces the schedule length and as a result increases the acceptance rate of flows.

2. Proposed Scheme

The symbol r_i represents the time at which all the flows Fi are released for scheduling. The flow Fi that arrives at the source cluster head is denoted by a_i^{SH}. The notation a_i^{DH} denotes the flow Fi arriving at the destination cluster-head. The notation for the node that arrives at the destination node is represented by a_i. We divide the time slots to avoid interference. Therefore, the proposed scheme is represented by assuming IntraSend, InterComm, IntraRecv as 1, 2, 1 respectively shown in Figure 1. The minor frame in Figure 1 shows the number of time slots per frame. IntraSend means the scheduling algorithm which will schedule the flows from the source node to the source cluster head. InterComm represent the scheduling algorithm that will schedule the flows from the source cluster heads to the destination cluster heads. IntraRecv represents the scheduling that will schedule the flows from the destination cluster head to the destination node. *H1*, *H2*, *H3*, *H4*, *H5* represent cluster head *1*, *2*, *3*, *4* and *5* respectively as shown in Figure 2. Transmission $u \rightarrow v$ means that node u can send the data to node v.

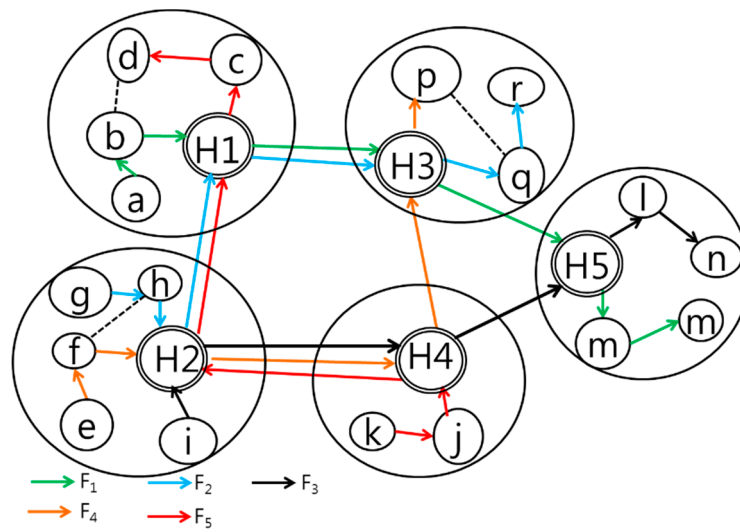| IntraSend	InterComm	IntraRecv
1	2	1

Figure 1. Minor frame.

Figure 2. Cluster-based real-time flows.

Procedure of the Proposed Scheme

The proposed scheme schedules the flows from the source to destination node by using three scheduling algorithms, namely IntraSend, InterComm and IntraRecv. In the first scheduling algorithm, the flows are scheduled from a source node to the source cluster head. Cluster heads are scheduled from source to destination in the second scheduling algorithm. Finally, in the third scheduling algorithms, the flows are scheduled from destination cluster head to destination node. As shown in Figure 2, the proposed scheme first initializes the number of flows to each cluster head. Each cluster head knows about the number of incoming and outgoing flows in their cluster. After the initialization of IntraSend scheduling, if the cluster has only incoming flows then the IntraSend scheduling can schedule the flows both in IntraSend and IntraRecv time slots. The cluster can utilize IntraSend time slots and cannot utilize IntraRecv time slots in-case if the IntraSend scheduling have both incoming and outgoing flows. For the InterComm scheduling the flows will be schedule from the source cluster head to destination cluster head. As the cluster head has high transmission power, therefore, it can utilize the InterComm time slots and cannot utilize the IntraSend and IntraRecv time slots. Finally, after the initialization of IntraRecv scheduling if the cluster has outgoing flows then IntraRecv scheduling can utilize both the IntraSend and IntraRecv time slots. Similarly, if the cluster have both incoming and outgoing flows then the IntraRecv scheduling can utilize the IntraRecv time slots and cannot utilize the IntraSend time slots.

We first schedule the flows from the source node to the destination node where each intra-cluster will have utilized only their respective time slots and cannot utilize each other empty time slots.

The cluster *H1* will schedule the transmission *a→b* and *b→H1* of *F1* in the slot#*1* and slot #*5* respectively as shown in IntraSend Scheduling in Table 1. Transmission *j→c* of *F5* cannot be scheduled in the slot #*1* due to the interference with the transmission *a→b* of high priority *F1*. Therefore, the transmission *j→c* will schedule in the next time slots of IntraSend time slot #*5*. For transmission *c→H1* of *F5* as there is no interference with higher priority flow so it will be scheduled by *H1* in the slot #*9*. Next cluster *H2* will scheduled transmission *i→h* of *F2*, *p→H2* of *F3* and *e→g* of *F4* in the slot #*1* due to no conflict with transmissions of high priority flows. Transmission *h→H2* of *F2* will be scheduled in the slot #*5* while transmission g→H2 will be cannot be schedule in the time slot#5 due to interference with transmission h→H2 of F2. Therefore, the transmission g→H2 of F4 will be scheduled in the time slot #9. Next in the InterComm scheduling, the flows from F1 to F5 are scheduled from the source node to the destination node as shown in the following Table 2.

Table 1. Results of previous scheme IntraSend scheduling of Figure 2.

			Previous IntraSend Scheduling			
Cluster	Flow#	r_i	1	5	9	a_i^{SH}
H1	F1	0	a→b	b→H1		5
	F5	0		j→c	c→H1	9
H2	F2	0	i→h	h→H2		5
	F3	0	p→H2			1
	F4	0	e→g		g→H2	9

Table 2. Results of previous scheme InterComm scheduling of Figure 2.

				Previous InterComm Scheduling							
Flow#	a_i^{SH}	2	3	6	7	10	11	14	15	18	a_i^{DH}
F1	5			H1→H3	H3→H5						7
F2	5					H2→H1	H1→H3				11
F3	1	H2→H4	H4→H5								3
F4	9					H2→H4	H4→H3				14
F5	9								H1→H2	H2→H4	18

In the IntraRecv scheduling, each cluster will schedule the transmissions from the source cluster head to the destination node. For example, cluster H5 will schedule the transmissions of F1 and F5 while transmission of F2 and F4 will be schedule by cluster H3. H4 will schedule the transmissions of F3 from the source cluster head to their respective destination node as shown the Table 3 below.

Table 3. Results of previous scheme IntraRecv scheduling of Figure 2.

			Previous IntraRecv Scheduling						
Cluster	Flow#	a_i^{DH}	4	8	12	16	20	24	a_i
H5	F1	7		H5→m	m→q				12
	F5	18					H5→n	n→t	24
H3	F2	11			H3→k	k→l			16
	F4	14				H3→d			16
H4	F3	3	H4→r	r→o					8

The proposed IntraSend scheduling system for cluster *H1* is shown in Table 4. In Figure 2, cluster *H1* has only outgoing flows, therefore, the cluster could use both *NIS* and *NIR* time slots for the IntraSend Scheduling. *F1* is scheduled first from node *a* to *H1*; thus *F5* is scheduled in *H1* cluster at time slot *#1*. Since the transmission *a→b* of *F1* and transmission *j→c* of *F5* are conflicting with each other, the transmission *a→b* of *F1* is therefore scheduled in slot#1 due to high priority. This is interesting, as there are only one incoming flow in cluster *H1*. Thus, cluster H1 can utilize the IntraRecv time slots for IntraSend scheduling and schedule the transmission *j→c* of *F1* and *h→H2* of *F5* in the same slot *#4* due to non-interference. Similarly, cluster H2 schedule transmission i→h of F2 in slot *#1*. Transmission *p→H2* of *F3* and *e→g* of *F4* are also scheduled in slot #1 due to non-interference with high priority flows. Cluster *H2* has also outgoing flows. In this, it can utilize the IntraRecv time slots. Therefore, transmission *h→H2* is scheduled in time slot *#4*. Transmission of F3 is already reached to cluster head so it cannot be schedule in slot *#4*. Finally, the transmission *g→H2* of *F4* is conflicting with transmission *h→H2* of *F2*, therefore, the transmission *h→H2* is delayed by one slot and schedule in the next time slot *#5* as shown in Table 4.

Table 4. Results of proposed scheme IntraSend scheduling of Figure 2.

			IntraSend Scheduling			
Cluster	Flow#	r_i	1	4	5	a_i^{SH}
H1	F1	0	a→b	b→H1		4
	F5	0		j→c	c→H1	5
H2	F2	0	i→h	h→H2		4
	F3	0	p→H2			1
	F4	0	e→g		g→H2	5

In InterComm scheduling, the global cluster header schedule the flows from *F1* to *F5* from source cluster head to destination head as shown in Table 5.

Table 5. Results of proposed scheme InterComm scheduling of Figure 2.

				InterComm Scheduling				
Flow#	a_i^{SH}	2	3	6	7	10	11	a_i^{DH}
F1	4			H1→H3	H3→H5			7
F2	4					H2→H1	H1→H3	11
F3	1	H2→H4	H4→H5					3
F4	5			H2→H4		H4→H3		10
F5	4				H1→H2		H2→H4	11

IntraSend and IntraRecv scheduling can utilize each other slots but cannot utilize the InterComm time slots due to the high transmission power of the cluster heads. As shown in Table 5, the global cluster head could schedule the flow *F1* from source cluster-head *H1* to neighbor the cluster head *H3*. Furthermore, the neighbor cluster-head *H3* could schedule the data to destination cluster *H5*. Similarly, for the *F2* flow, the global cluster-head cannot schedule the flow *H2→H1* in the time slot *#6* due to the interference of high priority flow *F1*. In the same way in slot #7, the flow *F2* transmission *H2→H1* is interference with transmission *H3→H5* of *F1* and hence delayed. Flow *F2* transmission *H2→H1* is scheduled in the time slot *#10* while transmission *H1→H3* is scheduled in the time slot *#11*. For flow *F3*, the IntraSend transmission arrives at the source cluster-head in time slot *#1*. Hence, the transmission *H2→H4* and *H4→H5* are scheduled by the global cluster-head in the InterComm time-slots *#2* and *3* respectively. For flow *F4*, the transmission *H2→H4* will be scheduled by the global head in the InterComm time-slot *#6*, due to no interference with high priority flow. Hence, the transmission *H4→H3* of *F4* has interference with transmission *H3→H5* of high priority flow *F1* in the InterComm time slot *#7*, therefore, the transmission *H4→H3* will be delayed by one time-slot. Transmission *H4→H3* of *F4* will scheduled in the InterComm time slot *#10* due to no interference with transmission *H2→H1* of high priority flow *F2*. For the flow *F5*, InterComm scheduling the transmission *H1→H2* is conflict with both transmission *H1→H3* of *F1* and *H2→H4* of *F4*, therefore, transmission *H1→H2* will be scheduled in the InterComm time slot *#7* as it has no interference with transmission *H3→H5* of higher Flow 1. Similarly, transmission *H2→H4* of *F5* cannot be schedule in the InterComm time slot *#10* due to interference with high priority flows transmission *H2→H1* of flow *F2* and transmission *H4→H3* of *F4*. Finally, the transmission *H2→H4* of *F2* will be scheduled in the InterComm time slot *#11*, due to non-conflict with high flow.

Finally, consider the IntraRecv scheduling of clusters as shown in Table 6. As shown in Figure 2 cluster *H3*, *H4* and *H5* have only incoming flows, so in these, IntraRecv scheduling does not differentiate IntraSend and IntraRecv time slots and can schedule IntraRecv flows both in IntraSend and IntraRecv time slots. For example, as shown in the Table 6, cluster *H5* schedule *F1* transmission *H5→m* in the IntraRecv time slots while transmission *m→q* in the IntraSend time slots, as there is only

incoming flows in cluster *H5*. Similarly, cluster *H3* schedule transmission *H3→k* in the IntraRecv time slot whereas transmission *k→l* in the IntraSend time slots. Next cluster *H4* schedule transmission *H4→r* in the IntraRecv time time-slot while transmission *r→o* in the IntraSend time slots so utilized both the IntraSend and IntraRecv time slots as shown below in Table 6.

Table 6. Results of proposed scheme IntraRecv scheduling of Figure 2.

Cluster	Flow#	a_i^{DH}	4	5	8	9	12	13	a_i
				IntraRecv Scheduling					
H5	F1	7		H5→m	m→q				9
	F5	11					H5→n	n→t	13
H3	F2	11					H3→k	k→l	13
	F4	10					H3→d		12
H4	F3	3	H4→r	r→o					5

To compare the previous and proposed system, we consider the flows arrived at their destination node. In the previous systems, *F1* arrived at destination node at time slot *#12* while in the proposed system *F1* arrived at destination node at time slot *#9*. The reason is *F1* in the proposed system utilized the empty time slots *#4* of IntraRecv time slots while *F1* in the previous systems did not utilize the empty time slots of IntraRecv scheduling. F2 arrived at time slot *#16* to the destination node in the previous scheme while in the proposed scheme due to utilization of IntraRecv time slot *#4*, *F2* reached to the destination node at slot *#13*. In the previous scheme, *F3* arrived at the destination node in slot *#8* whereas flow *F3* in the proposed scheme arrived at the destination node at time slot *#5*. Because *F3* in the proposed scheme utilized the empty IntraRecv time slot *#4*, by the IntraSend scheduling, an empty time slot of IntraSend is utilized by IntraRecv scheduling. Similarly, *F4* in the proposed scheme arrived at the destination node at time slot*#12* while in the previous scheme F4 arrived at time slot *#16* due to the non-utilization of the IntraRecv time slot. Finally, *F5* in the previous scheme arrived at the destination node at time slot *#24* while in the proposed scheme *F5* arrived at the destination node at time slot *#13* because the IntraSend scheduling utilized the empty time slots of IntraRecv. Similarly, the empty time slots of IntraSend is utilized by the IntraRecv scheduling. The flow diagram describing different states of the proposed scheme is shown in Figure 3.
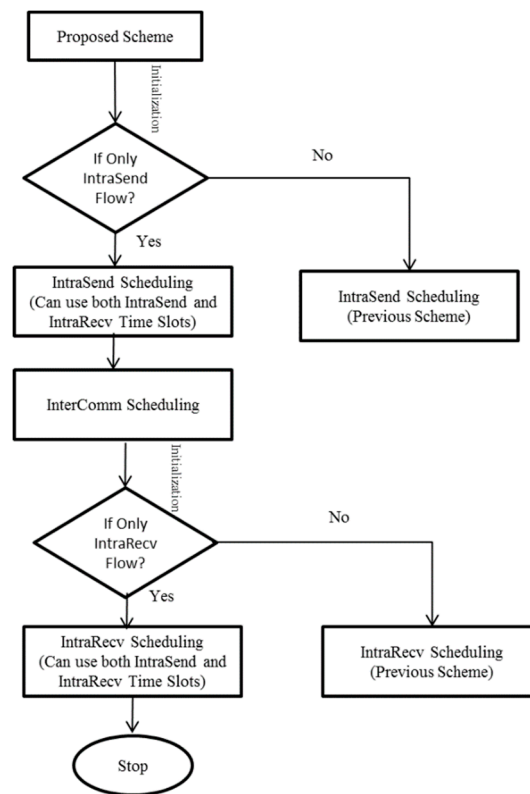
Figure 3. Flow diagram of proposed scheme.

3. Performance Evaluation

We use GENSEN [30] tool to perform the simulation of proposed scheme. The nodes are placed randomly in the area of 100 m × 100 m area.

The proposed scheme compared the acceptance rate of flows with the previous scheme in [30]. In the previous scheme the IntraSend scheduling schedule the flows in the IntraSend time slots and not utilized the empty time slots of IntraRecv time slots. Similarly, the IntraRecv scheduling not utilizing the empty time slots of IntraSend Scheduling. While in the proposed scheme the clusters that have only outgoing flows then the IntraSend scheduling can utilize the empty IntraRecv time slots. Similarly, if the cluster has only incoming flows then the IntraRecv scheduling can utilize the empty IntraSend time slots. We analyzed the performance of compared schemes based on six metrics, i.e., the impact of a deadline, the impact of flows, the impact of clusters, the impact of intra-cluster slots, the utilization of IntraSend time slots and utilization of IntraRecv time slots.

3.1. Impact of Deadline

To analyze the performance of the proposed and previous scheme with respect to the deadline. We take a short deadline from 1 to 3 while the numbers of flows and clusters are fixed to 6 and 6 respectively. In Figure 4a, the acceptance rate of the proposed and previous schemes are compared with respect to the deadline = 1. In the proposed scheme, the clusters that have either IntraSend or IntraRecv flows can make use of both IntraRecv and IntraSend time-slots. In contrast to the proposed scheme, the previous schemes provide a lower acceptance rate. Because in the previous scheme, although clusters have only outgoing or incoming flows the IntraSend and IntraRecv scheduling are not utilizing the empty IntraRecv and IntraSend time slots respectively. In Figure 4b,c, the deadline increases to 2 and 3 respectively. As the deadline increases the acceptance rate also increases in both schemes. However, the acceptance of the proposed scheme is higher than the previous scheme. As the proposed

scheme are utilizing both the IntraSend and IntraRecv time slots. While the previous schemes utilize the IntraSend time slots and did not utilize the IntraRecv empty time slots.
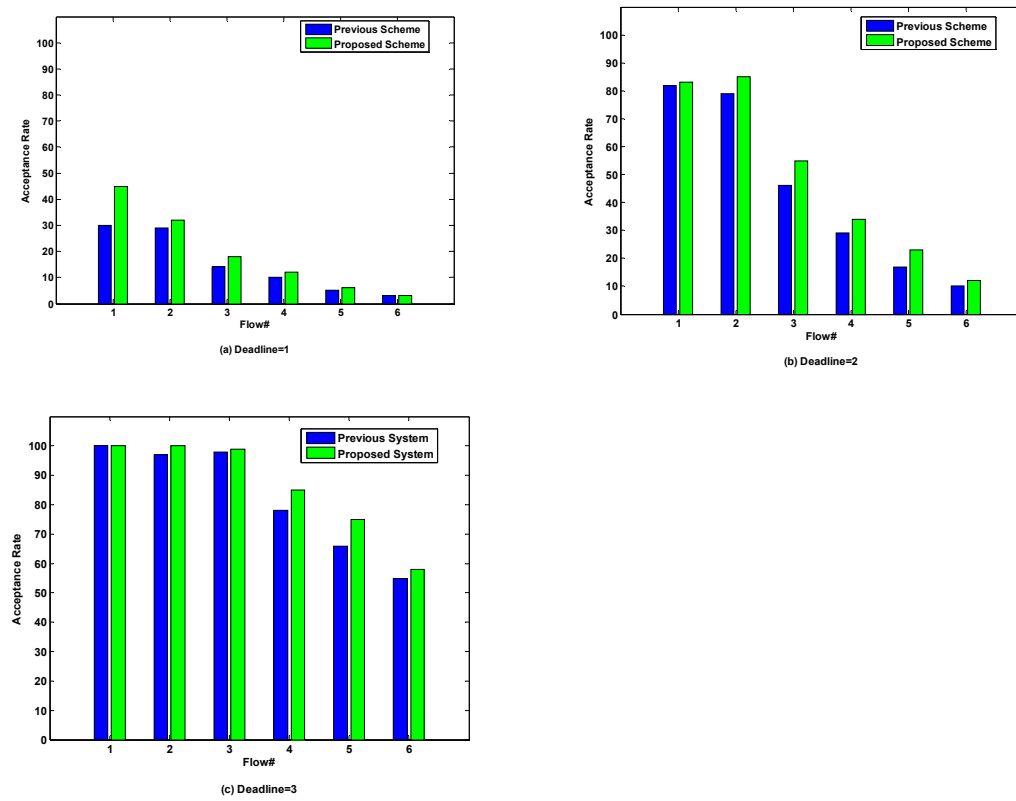

Figure 4. Simulation results for w.r.t deadline.

3.2. Impact of Flows

For the impact of flow analysis, the number of clusters is fixed to 6, while the number of flows ranges from 4 to 8. As shown in Figure 5a–e in contrast to the previous schemes—the proposed scheme shows a high acceptance rate. The acceptance rate is high in the proposed scheme because the clusters utilized slots of both IntraSend and IntraRecv flows. For instance, if there are no incoming flows in the cluster then IntraSend flows can utilize the IntraRecv slots. Similarly, if there are no outgoing flows in the cluster then the IntraRecv scheduling can utilized the IntraSend time slots. While in the previous scheme for clusters that have incoming or outgoing flows then the IntraSend and IntraRecv flow cannot use one another empty time slots.

As shown in Figure 5a–e, as the number flows increases the acceptance rate decreases. Since the number of flows is increasing then the interference is also increasing. An increase in the interference means the acceptance rate will be decreasing. However, the acceptance rate of the proposed scheme is higher than the previous scheme.

Because in the proposed scheme when the flows have interfered in the IntraSend time slots then the interference flows can be scheduled in the IntraRecv time slots. Whereas in the previous scheme if the flows are interfering in the IntraSend time slots then the flows cannot be scheduled in the IntraRecv time slots, hence increasing the scheduling. Therefore, fewer flows are scheduled in the given deadline and hence the acceptance rate is decreasing.
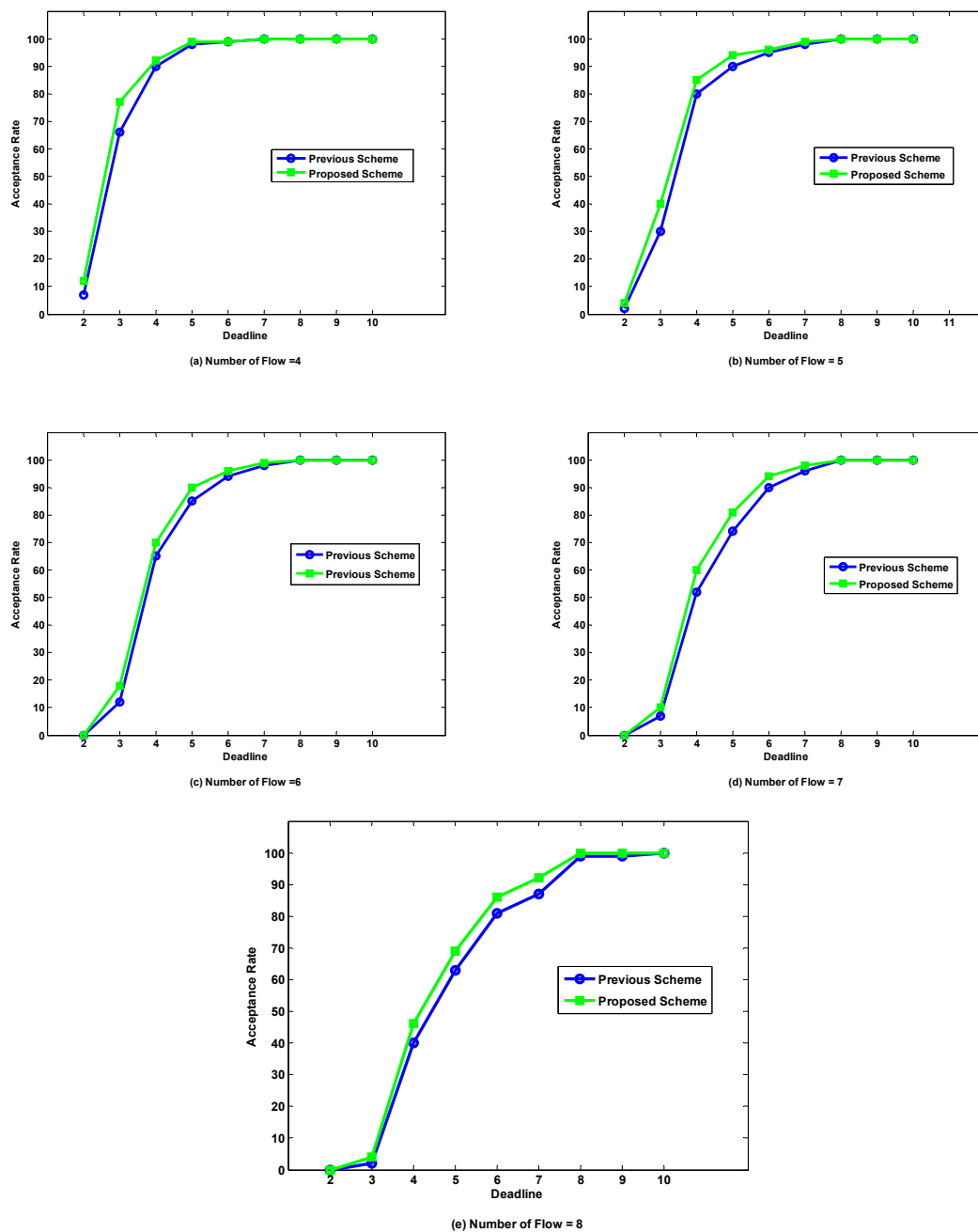
Figure 5. Simulation results w.r.t flows.

3.3. Impact of Clusters

In this section, we analyze the performance of the proposed and previous scheme with respect to a number of clusters. We fix the number of flows to 6 while the numbers of clusters are ranges from 4 to 8. As shown in Figure 6a–e, the proposed scheme shows high acceptance compare to the previous scheme in terms of a number of clusters. Because for clusters that have only outgoing or incoming flows, in that case, the intra-cluster schedule cannot use intra-cluster slots of each other.

Whereas in the proposed scheme for clusters that have either incoming or outgoing flows, then intra-cluster schedule can still utilize the intra-cluster slots if one another at that time.
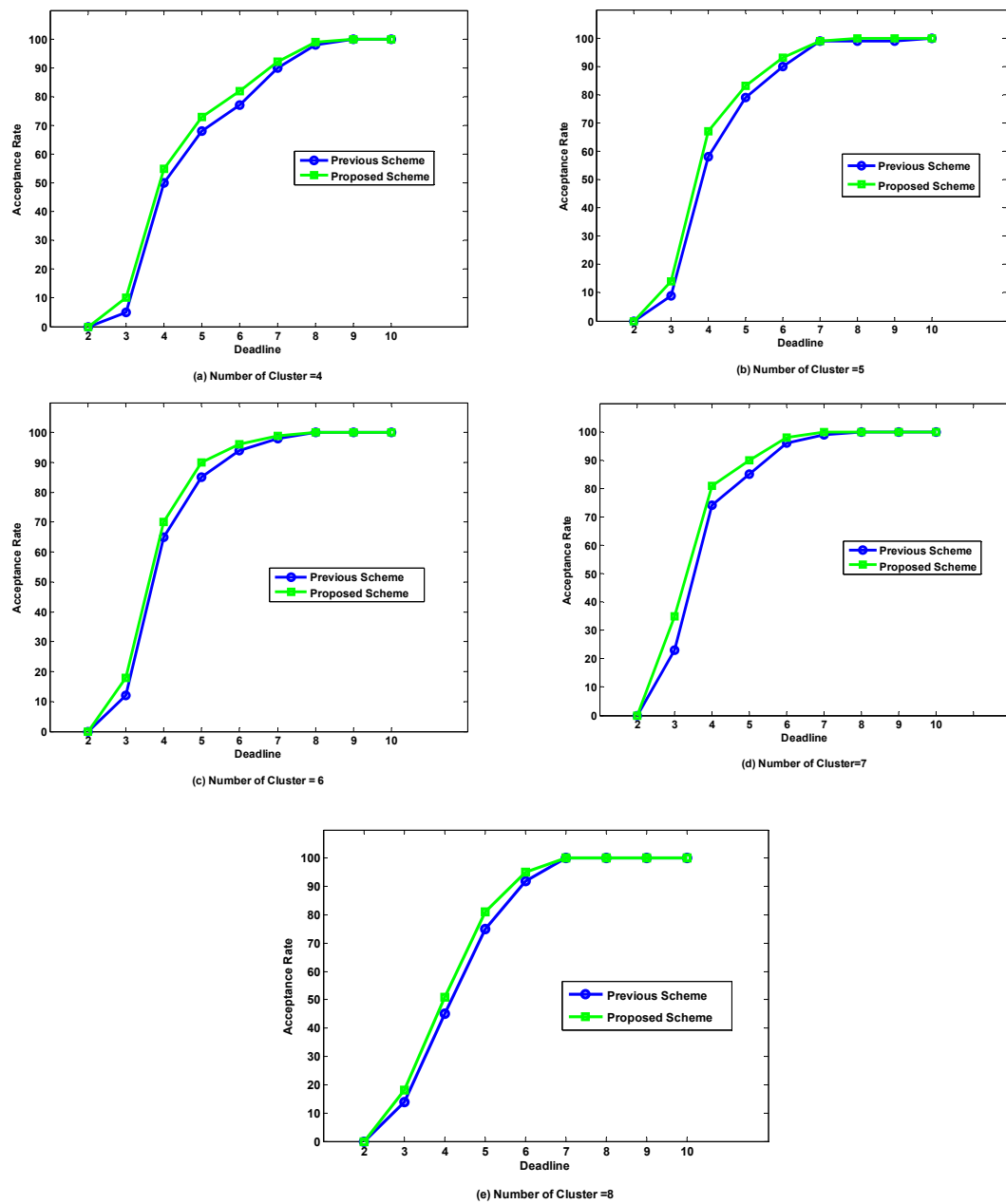
Figure 6. Simulation results for w.r.t Clusters.

3.4. Impact of Intra-Cluster Time Slots

In this section to analyze the performance of the intra-cluster time slots. The numbers of IntraSend and IntraRecv time slots were varied from 1 to 3. The numbers of clusters and flows are fixed to 6. As shown in Figure 7a–c the proposed scheme has a high acceptance rate than flows in the previous scheme. The reason is, for IntraRecv scheduling has allocated 1 slot to schedule flows which is insufficient. Therefore, in the proposed scheme if the clusters have only incoming flows then that clusters can utilize the IntraSend time slots.

(a)Sequence:1-2-1

(b) Sequence:2-2-1

(c) Sequence:3-2-1
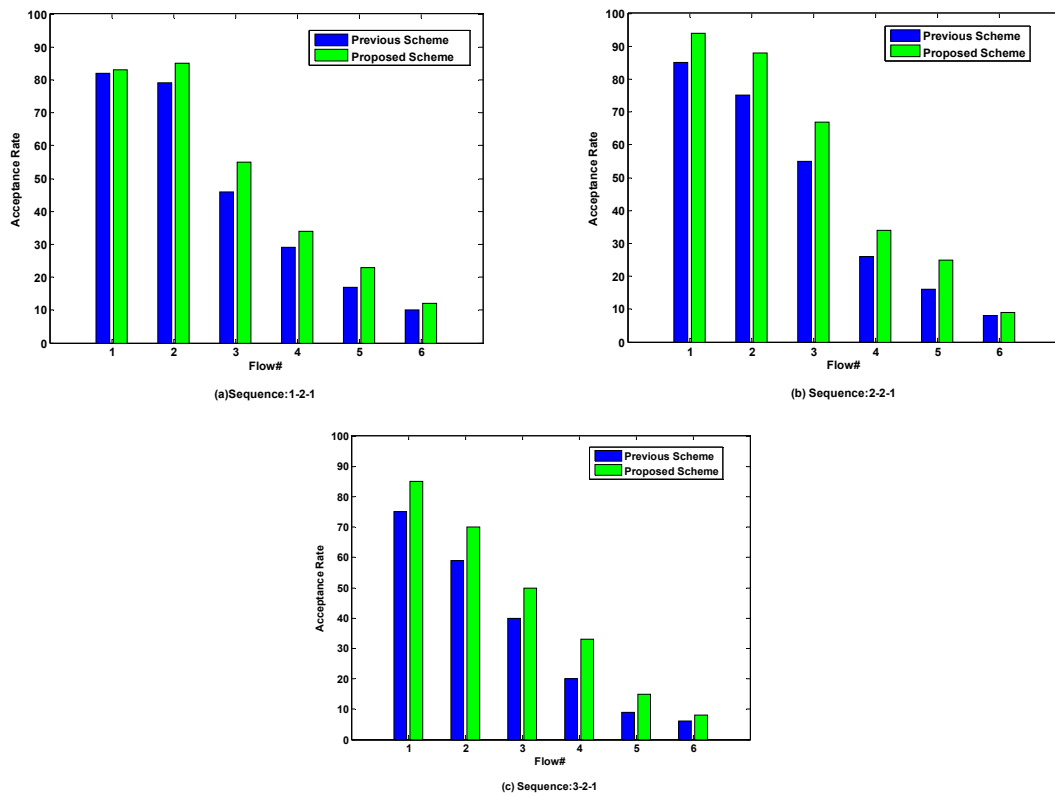
Figure 7. Simulation results for w.r.t IntraSend time slots.

In Figure 8a,b the proposed scheme has also a high acceptance rate than flows in a previous scheme. The reason is, for IntraSend scheduling has allocated 1 slot to schedule flows which is insufficient. Therefore, in the proposed scheme the clusters that have only incoming flows then that clusters can utilized the IntraSend times time slots. In contrast to the proposed scheme, the previous scheme for the clusters that have outgoing flows then the IntraSend scheduling cannot utilize the IntraRecv time and vice versa.

(a) Sequence:1-2-2

(b) Sequence:1-2-3
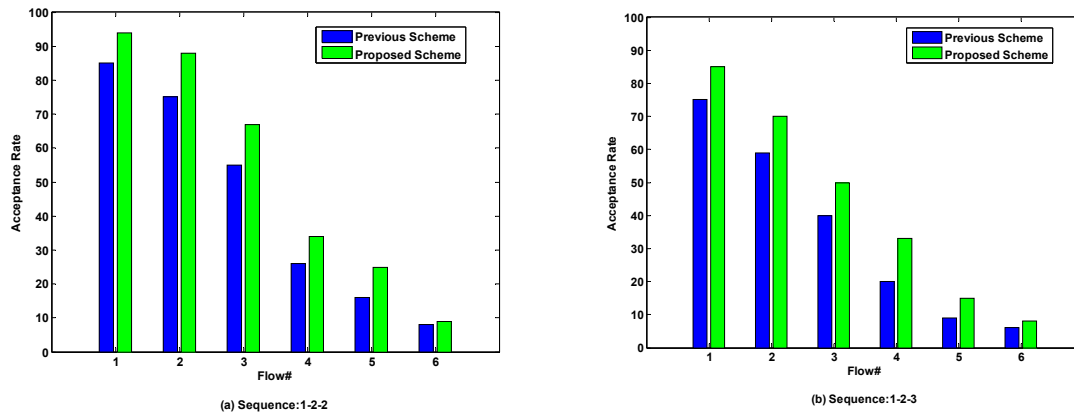
Figure 8. Simulation results for w.r.t IntraRecv time slots.

3.5. IntraSend Time Slots Utilization

To further analyzed the intra-cluster time slots. We compare the previous scheme and proposed scheme based on the number of IntraSend slots which did not utilized by the IntraRecv scheduling in the previous scheme whereas the proposed scheme IntraRecv scheduling utilized these empty IntraSend time slots.

As shown in Figure 9 the proposed scheme shows higher utilization of IntraSend time slots as compared to previous scheme. The reason is that in the IntraRecv scheduling the clusters that have only incoming flows and there are no outgoing flows then the IntraRecv scheduling in the previous scheduling will only utilized the IntraRecv time slots and will not utilized the empty time slots of IntraSend time slots. While in the proposed scheme for the clusters that have only outgoing flows then that cluster head can utilize the time slots of IntraRecv scheduling as well as the empty time slots of IntraSend time slots. Hence the proposed scheme efficiently utilized the IntraSend and IntraRecv time slots.
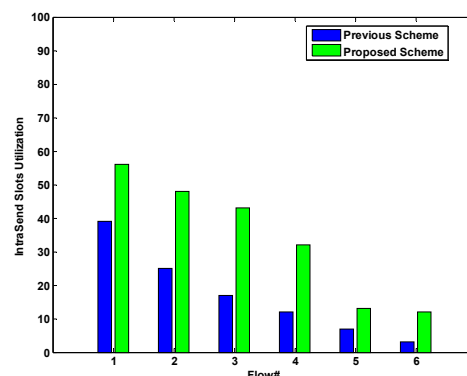
Figure 9. IntraSend slot utilization.

3.6. IntraRecv Time Slots Utilization

As shown in Figure 10, compared to previous scheme IntraSend Scheduling the proposed scheme IntraSend scheduling efficiently utilized the IntraRecv time slots. Because in the proposed scheme the clusters that have only outgoing flows the IntraSend scheduling will utilized the IntraSend time slots and not utilizing the empty time slots of IntraRecv. Whereas in the proposed scheme that clusters that have only outgoing flows the IntraSend scheduling will utilized the IntraSend time slots as well as the empty time slots of IntraRecv and hence efficiently utilized both the IntraSend and IntraRecv time slots.
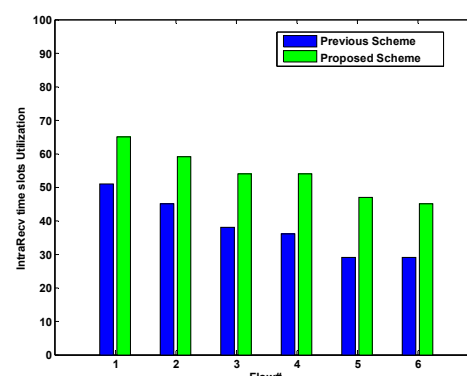

Figure 10. IntraRecv time slot utilization.

4. Conclusions

In this paper, a new intra-cluster scheduling scheme for real-time flow was proposed. In the proposed scheduling algorithm, the IntraSend scheduling utilizes both the IntraSend and IntraRecv time slots if the cluster has outgoing flows. Similarly, if a cluster has only incoming flows, the IntraRecv scheduling algorithm utilizes both the IntraRecv and IntraRecv time slots. In simulation results, we first analyzed the impact of various parameters on our proposed scheme. Then, we compared our scheme to the existing algorithm, and showed that our scheme has more flows delivered within the deadline than the existing solutions. The simulation results showed that the performance of

the proposed scheme had a higher acceptance rate than the previous scheme. Further enhancements for our technique may certainly be to enhance our scheduling scheme by delivering more real-time flows within the deadline. In this work, we focused our attention only on an intra-cluster scheduling scheme to schedule the real-time flow within the deadline, by considering interference within the cluster and among the clusters. The proposed scheme can be further enhanced by reducing the number of slots of inter-cluster real-time flow between the cluster head. Importantly, the cluster head signal is too strong to interfere with other cluster headers. Therefore, more time slots are required for the headers nodes to schedule their data between cluster-headers. Thus, to reduce the inter-cluster time slots, it would be efficient for flows in the same cluster that send their message to nodes in other similar clusters to apply some aggregation function and send them in one slot, instead of sending in multiple slots. A second improvement to the scheme could be to apply a compositional real-time scheduling into the proposed frame works. The last improvement to the scheme would be to extend the proposed model to a more realistic model; a routing protocol and distributed approach will be considered for the proposed scheme.

References

1. Jin, X.; Saifullah, A.; Lu, C.; Zeng, P. Real-Time Scheduling for Event-Triggered and Time-Triggered Flows in Industrial Wireless Sensor-Actuator Networks. In Proceedings of the IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019.
2. Soua, R.; Minet, P. Multichannel assignment protocols in wireless sensor networks: A comprehensive survey. *Pervasive Mob. Comput.* **2015**, *16*, 2–21.
3. Bhatia, A.; Hansdah, R.C. A Distributed TDMA Slot Scheduling Algorithm for Spatially Correlated Contention in WSNs. *Mob. Inf. Syst.* **2015**. [CrossRef]
4. Bakshi, M.; Jaumard, B.; Kaddour, M.; Narayanan, L. On TDMA scheduling in wireless sensor networks. In Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Vancouver, BC, Canada, 15–18 May 2016; pp. 1–6.
5. Ramanathan, S.; Lloyd, E.L. Scheduling algorithms for multi-hop radio networks. *IEEE/ACM Trans. Netw.* **1993**, *1*, 166–177. [CrossRef]
6. Ali, G.; Kim, K.H.; Kim, K.; Aldwairi, M. Interference Aware Real-Time Flows Scheduling in Cluster Based Wireless Sensor Networks. *Int. J. Eng. Technol. Innov.* **2016**, *6*, 93–102.
7. Gobinath, T.; Tamilarasi, A. RFDCAR: Robust failure node detection and dynamic congestion aware routing with network coding technique for wireless sensor network. *Peer Peer Netw. Appl.* **2019**. [CrossRef]
8. Umar, I.A.; Hanapi, Z.M.; Sali, A.; Zulkarnain, Z.A. Towards overhead mitigation in state-free geographic forwarding protocols for wireless sensor networks. *Wirel. Netw.* **2019**, *25*, 1017–1030. [CrossRef]
9. Lenka, M.R.; Swain, A.R.; Sahoo, M.N. Distributed Slot Scheduling Algorithm for Hybrid CSMA/TDMA MAC in Wireless Sensor Networks. In Proceedings of the IEEE International Conference on Networking, Architecture and Storage (NAS), Long Beach, CA, USA, 8–10 August 2016; pp. 1–4.
10. Xuelin, C.; Zuxun, S. An overview of slot assignment (SA) for TDMA. In Proceedings of the IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Ningbo, China, 19–22 September 2015; pp. 1–5.
11. Di Francesco, M.; Pinotti, C.M.; Das, S.K. Interferencefree scheduling with bounded delay in cluster-tree wireless sensor networks. In Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, New York, NY, USA, 21–25 October 2012; pp. 99–106.

12. Chatterjee, P.; Ghosh, S.C.; Das, N. Load Balanced Coverage with Graded Node Deployment in Wireless Sensor Networks. *IEEE Trans. Multi-Scale Comput. Syst.* **2017**, *3*, 100–112. [CrossRef]

13. Oladimeji, M.O.; Turkey, M.; Dudley, S. HACH: Heuristic Algorithm for Clustering Hierarchy protocol in wireless sensor networks. *Appl. Soft Comput.* **2017**, *55*, 452–461. [CrossRef]

14. Hasan, M.Z.; Al-Rizzo, H.; Al-Turjman, F. A Survey on Multipath Routing Protocols for QoS Assurances in Real-Time Wireless Multimedia Sensor Networks. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1424–1456. [CrossRef]

15. Rhee, I.; Warrier, A.; Min, J.; Xu, L. DRAND: Distributed Randomized TDMA Scheduling for Wireless Ad Hoc Networks. *IEEE Trans. Mob. Comput.* **2009**, *8*, 1384–1396. [CrossRef]

16. Li, Y.; Chen, C.S.; Song, Y.Q.; Wang, Z. Real-Time QOS Support in Wireless Sensor Networks: A survey. *IFAC Prec. Vol.* **2007**, *40*, 101–108. [CrossRef]

17. Diallo, O.; Rodrigues, J.C.; Sene, M. Real-time data management on wireless sensor networks: A survey. *J. Net. Comp App.* **2012**, *35*, 1013–1021. [CrossRef]

18. Suriyachai, P.; Brown, J.; Roedig, U. Time-critical data delivery in wireless sensor networks. In Proceedings of the 6th IEEE International Conference on Distributed Computing in Sensor Systems, Santa Barbara, CA, USA, 21–23 June 2010; pp. 216–229.

19. Zhang, H.; Soldati, P.; Johansson, M. Optimal link scheduling and channel assignment for convergecast in linear WirelessHART networks. In Proceedings of the WiOPT, Seoul, Korea, 23–27 June 2009.

20. Saifullah, A.; Xu, Y.; Lu, C.; Chen, Y. End-to-end delay analysis for fixed priority scheduling in WirelessHART networks. In Proceedings of the 17th IEEE Real-Time and Embedded Technology and Applications Symposium, Chicago, IL, USA, 11–14 April 2011; pp. 13–22.

21. Saifullah, A.; Xu, Y.; Lu, C.; Chen, Y. Real-time scheduling for WirelessHART networks. In Proceedings of the 31st IEEE Real-Time Systems Symposium, San Diego, CA, USA, 30 November–3 December 2010.

22. Cheng, P.; Zhang, F.; Chen, J.; Sun, Y.; Shen, X. A Distributed TDMA Scheduling Algorithm for Target Tracking in Ultrasonic Sensor Networks. *IEEE Tran Ind. Elect.* **2013**, *60*, 3836–3845. [CrossRef]

23. Pawar, P.M.; Nielsen, R.H.; Prasad, N.R.; Ohmori, S.; Prasad, R. M-GCF: Multicolor-Green Conict Free scheduling algorithm for WSN. In Proceedings of the 15th International Symposium on Wireless Personal Multimedia Communications, Taipei, Taiwan, 24–27 September 2012; pp. 143–147.

24. Ergen, S.C.; Varaiya, P. TDMA scheduling algorithms for wireless sensor networks. *Wirel. Netw.* **2010**, *16*, 985–997. [CrossRef]

25. Pawar, P.M.; Nielsen, R.H.; Prasad, N.R.; Ohmori, S.; Prasad, R. GCF: Green Conflict Free TDMA scheduling for wireless sensor network. In Proceedings of the IEEE International Conference on Communications (ICC), Ottawa, ON, Canada, 10–15 June 2012; pp. 5726–5730.

26. Huang, P.; Xiao, L.; Soltani, S.; Mutka, M.W.; Xi, N. The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey. *IEEE Comm. Sur. Tut.* **2013**, *15*, 101–120. [CrossRef]

27. Pawar, P.M.; Nielsen, R.H.; Prasad, N.R.; Prasad, R. H-GCF: A Hybrid Green Conflict Free scheduling algorithm for mobile Wireless Sensor Networks. In Proceedings of the 16th International Symposium on Wireless Personal Multimedia Communications, Atlantic City, NJ, USA; 2013; pp. 1–5.

28. Ali, G.; Kang, S.; Kim, K.H.; Kim, K. Towards Cluster-Based Real-Time Flow Scheduling in Interference-Aware Wireless Sensor Networks. In Proceedings of the IEEE 16th International Conference on Computational Science and Engineering, Sydney, Australia, 3–5 December 2013; pp. 523–530.

29. Pawar, P.M.; Kulkarni, N.P.; Mantri, D.S. Secure Scheduling for Cluster-based TDMA Schedule MAC in Wireless Sensor Network. In Proceedings of the IEEE Global Conference on Wireless Computing and Networking (GCWCN), Lonavala, India, 23–24 November 2018; pp. 119–123.

30. Camilo, T.; Silva, J.S.; Rodrigues, S.A.; Boavida, F. GENSEN: A topology generator for real wireless sensor networks deployment. In Proceedings of the 5th IFIP Workshop on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS 2007), Santorini, Greece, 7–8 May 2007.