A Control Loop Reduction Scheme for Wireless Process Control on Traffic Light Networks*

Junghoon Lee[1], Gyung-Leen Park[1], In-Hye Shin[1],
Choel Min Kim[2,]**, and Sang-Wook Kim[3]

[1] Dept. of Computer Science and Statistics, Jeju National University
[2] Dept. of Computer Education, Jeju National University
[3] College of Information and Communications, Hanyang University
{jhlee,glpark,ihshin76,chkim}@jejunu.ac.kr, wook@hanyang.ac.kr

Abstract. This paper designs a loop reduction scheme and measures its performance for the wireless process control application running on the grid-style traffic light network, aiming at improving the response time of the control system. Based on the WirelessHART standard, which assigns a slot to each (sender, receiver) pair according to the routing schedule, the allocation scheme puts all the pairs having no interference to a single slot, reducing the loop length. For further reduction, the classic Dijkstra's shortest path algorithm is modified such that the number of end-to-end paths starting from the horizontal links and the vertical link, respectively, is almost same. The transmission of the controller, which initiates all message delivery and is the main bottleneck point, will not be blocked. The simulation result demonstrates that the proposed scheme can significantly reduce the control loop length, and the modified path finding algorithm further achieves about 9.8 % improvement, just sacrificing the 3.2 % of transmission success ratio.

1 Introduction

Vehicular telematics networks can keep the vehicle connected to a network even while it is on the move. This network exploits matured wireless communication technologies and essentially includes static elements which provide the access point to moving vehicles. Correspondingly, each component communicates in two-level hierarchy in vehicular networks [1]. Level 1 corresponds to connection between static nodes and level 2 between a static node and moving vehicles trying to access the network. Level 1 communication creates a kind of wireless mesh networks which have been intensively researched and widely deployed into diverse field areas, exploiting commonly available protocols such as IEEE 802.11, Zigbee, and the like [2]. Based on this network, many vehicular applications can be developed and serviced.

---

* This research was supported by the MKE, Korea, under the ITRC support program supervised by the NIPA. (NIPA-2009-C1090-0902-0040)).

** Corresponding author.

D. Taniar et al. (Eds.): ICCSA 2010, Part III, LNCS 6018, pp. 1–10, 2010.

Traffic lights are found in every street, and they can desirably install a wireless communication interface as they have sufficient power provision and their locations are highly secure. On such a traffic light network, it is possible to implement a monitor-and-control application when the network includes sensors and actuators [3]. Practically, a lot of traffic-related or environmental devices such as speed detectors, pollution meters, and traffic signal controllers, can be covered by the vehicular network. Moreover, vehicles can also carry a sensor device and report the collected sensor data to a static node when connected. The message of process control applications is necessarily time critical and how to schedule messages is the main concern. It depends on the node distribution, namely, topology, and the communication protocol. First, traffic lights are placed in each intersection of Manhattan-style road networks, so the traffic light network has grid topology. Additionally, the network protocol must provide predictable network access.

The WirelessHART standard provides a robust wireless protocol for various process control applications [4]. First of all, the protocol standard exploits the slot-based access scheme to guarantee a predictable message delivery, and each slot is assigned to the appropriate (sender, receiver) pair. In addition, for the sake of overcoming the transient instability of wireless channels, a special emphasis is put on reliability by mesh networking, channel hopping, and time-synchronized messaging. This protocol has been implemented and is about to be released to the market [5]. It can accommodate a split-merge operation to mask channel errors [6] as well as an efficient routing scheme to find the path that is most likely to successfully deliver the message [7].

For the timely control action, the network must deliver the sensor and control message not only reliably but also timely, or as fast as possible. Thus, every message transmission schedule, or slot assignment, is decided in priori. Moreover, to speed up the response time of a control process, the length of the slot schedule must be minimized. In the mesh network, more than one transmission can be accomplished simultaneously as long as transmitters are far way enough not to interfere each other. If the node is equipped with an directional antenna, more transmissions can be overlapped. However, the slot assignment permitting multiple transmissions is a NP hard problem, which has no polynomial time solution. In this regard, this paper is to design and evaluate the performance of a slot allocation scheme to the the sensor and control messages for the grid-style traffic light control network.

The paper is organized as follows: After defining the problem in Section 1, Section 2 introduces the background of this paper focusing on the WirelessHART protocol and slot allocation schemes. Section 3 designs control loop reduction schemes on the target network. The simulation result is discussed in Section 4, and finally Section 5 summarizes and concludes this paper.

2 Background and Related Work

The WirelessHART standard is defined over the IEEE 802.15.4 GHz radioband physical link, allowing up to 16 frequency channels spaced by 5 MHz guard band [8]. The link layer provides deterministic slot-based access on top of the time synchronization primitives carried out continuously during the whole network operation time. According to the specification, the size of a single time slot is 10 ms, and a central controller node coordinates routing and communication schedules to meet the robustness requirement of industrial applications. For more reliable communication, CCA (Clear Channel Assessment) [9] before each transmission and channel blacklisting is exploited to avoid specific area of interference and also to minimize interference to others. In some WirelessHART implementation, the network manger traverses a communication graph or a grid by the breadth-first search and allocates slots according to this order [8].

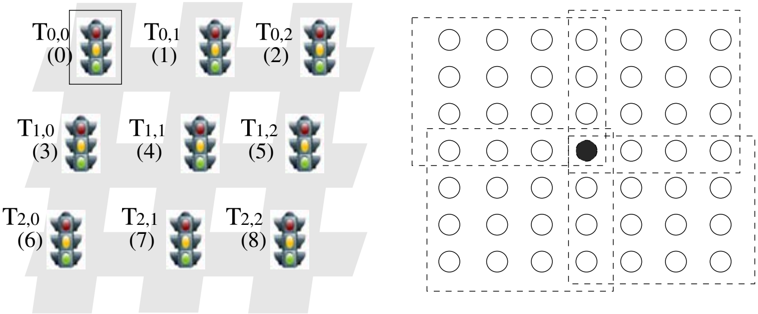
In each slot, a sender can try another channel if the CCA result of the channel on the primary schedule is not clear, even though such an operation is not yet defined in the current standard. For each destination node, a controller may reserve two alternative paths having sufficient number of common nodes. Two paths split at some nodes and meet again at other nodes. When two paths split, a node can select the path according to the CCA result in a single slot by switching to the channel associated with the secondary route. When two paths merge, the node can receive from two possible senders by a timed switch operation. Besides, the WirelessHART protocol can be reinforced by many useful performance enhancement schemes such as the virtual-link routing scheme that combines split-merge links and estimates the corresponding error rate to apply the shortest path algorithm.

Wireless mesh networks are cost-effective solutions for ubiquitous high-speed services, and its performance depends on the routing strategy. Routing has been extensively studied in wireless mesh networks, and most routing schemes are based on the shortest path algorithm [10]. The path cost is different according to the main goal of each routing scheme, including actual distance, transmission rate, and error characteristics. In most wireless process control systems, the central routing mechanism is generally taken, as not only the traffic characteristics are accurately known, but also each network access is predictable. Hence, it is possible to jointly allocate channels, route a message set, and schedule each transmission [11]. In addition, the routing procedure is executed repeatedly according to the link condition change and the schedule is distributed to each node [12].

3 Routing and Scheduling Scheme

3.1 System Model

The traffic lights form a grid network in modern cities, as a traffic light node is placed at each crossing of the Manhattan-style road network, as shown in Figure 1(a) [13]. Each node can exchange messages directly with its vertical and horizontal neighbors. Two nodes in the diagonal of a rectangle do not have a direct connection, as there may be obstacles like a tall building that blocks the wireless signal propagation. In this network, the central controller is assumed to be located at the fringe of a rectangular area, for this architecture makes the determination of the communication schedule simple and systematic. In Figure 1(a), $T_{0,0}$ is the controller node. Any grid network can be transformed into this network by partition and rotation. In the example of Figure 1(b), four 4 × 4 grids are generated and each of them can be mapped to a grid shown in Figure 1(a), regardless of the grid dimension.



(a) Traffic light network (b) Grid partition

Fig. 1. Traffic light network

A control loop consists of three phases, namely, reading state variables from sensors, deciding the control action, and sending the value of control variables to the actuators. The first and third steps involve the network and have their own communication schedules.

3.2 Loop Reduction

The allocation step basically begins with the well-known Dijkstra's shortest path algorithm which can find the optimal path from a node to the other nodes in the grid given as a cost graph. In this graph, each link can be specified with a cost value such as error rate, available network bandwidth, or pass time. As this paper focuses on the reliability of each end-to-end path, we assume that the slot error rate is given as a link cost. Each link has its own error characteristics due to different power level, obstacle distribution, and so on. The change of link error characteristics can be estimated in many ways [14], but we assume that the probing result is always correct, as the correctness of channel probing is not our concern. Figure 1(a) can be modeled as a cost graph for the 3 × 3 grid consist of 9 nodes from $T_{0,0}$ to $T_{2,2}$ and 12 links connecting each adjacent nodes. Node 0 and $T_{0,0}$ point the same node, and this paper will selectively use an appropriate notation.

Now, Dijkstra's (one-to-many) shortest path algorithm can find the best route having the lowest error rate. First, the error rate of each link can be replaced by the success probability by subtracting it from 1.0. Actually, they can be used interchangeably. Then, each iteration compares the product of success probabilities of a path extended by a expansion node, instead of the sum of link costs as in the original algorithm. In each node scan, the node having the highest success probability will be selected. The path from the controller, $T_{0,0}$, to each node is shown in Figure 2(a). The number in the parenthesis means the hop length of the path. Each column is a time slot and each slot has a (sender, receiver) pair. As each node has just one transmitter and one receiver module, neither the same sender nor the same receiver can appear twice in the same column. So, the initial allocation should be modified to avoid such a conflict.

To this end, the scheduler checks if two entries interfere. They interfere if 1) two senders are the same, or 2) two receivers are the same, or 3) the sender of one slot and the receiver of the other slot are the same. This criteria is valid just for the directional antenna which can increase spatial reuse and reduce interference by directing radio beam towards a desired direction [15]. When the node transmits using an omnidirectional antenna, which radiates or receives electromagnetic energy in all directions, the probability of interference increases, as the two transmissions will collide when the two senders are less than two hops away from each other. If two transmissions are not compatible in the same slot, the one with a shorter path will be moved one slot back, as the longer path is more critical to the whole control loop length. Such a move-back procedure is repeated until there is no change. Figure 2 (b) show the final result. In this figure, 0-3 means $T_{0,0} \rightarrow T_{1,0}$.

Slot	1	2	3	4
8 (4)	0–3	3–6	6–7	7–8
5 (3)	0–3	3–4	4–5	
7 (3)	0–3	3–6	6–7	
2 (2)	0–1	1–2		
4 (2)	0–3	3–4		
6 (2)	0–3	3–6		
1 (1)	0–1			
3 (1)	0–3			

(a) Initial setting

Slot	1	2	3	4	5	6	7	8	9	10	11
8 (4)	0–3	3–6	6–7	7–8							
5 (3)			0–3	3–4	4–5						
7 (3)					0–3	3–6	6–7				
2 (2)		0–1	1–2								
4 (2)							0–3	3–4			
6 (2)									0–3	3–6	
1 (1)				0–1							
3 (1)											0–3

(b) Final allocation

Fig. 2. Traffic light network

3.3 Routing Scheme

In the control loop scenario, traffic goes from and to $T_{0,0}$. Namely, each node sends and receives a message to and from the controller node once in the control period specified by the system requirement. Even if it is desirable to take the route which has the minimum number of hops to the destination, another detour can be advantageous in terms of delivery ratio and transmission delay. In process control applications, transmission reliability is most important. However, in addition to the end-to-end performance, it is also necessary to reduce the control message exchange time. In the example of Figure 2(b), it takes 11 slots to complete the delivery of all control messages to every node in a control round.

The control loop can be reduced by overlapping transmissions. As long as the network coordinator sticks to the classic shortest path algorithm, it can hardly expect more loop reduction. As shown in Figure 2 (a), each end-to-end route to a node starts from either $T_{0,0} \rightarrow T_{0,1}$ or $T_{0,0} \rightarrow T_{1,0}$ for the 3×3 grid. If all routes start from the same link, say, $T_{0,0} \rightarrow T_{0,1}$ the next slot cannot be overlapped as two transmission will inevitably collide. The first one is from the controller to $T_{0,1}$ (the receiver of the newly starting transmission) and the second one is from $T_{0,1}$ (the sender of the message arrived in the previous slot) to some other nodes. Table 1 describes this situation. For two destinations, namely, *Dest* 1 and *Dest* 2, suppose that *Dest* 1 takes the route $T_{0,0} \rightarrow T_{0,1} \rightarrow T_{0,2}$ while *Dest* 2 $T_{0,0} \rightarrow T_{0,1} \rightarrow T_{1,1}$, as show in Table 1.

Table 1. Example assignment

(a) Invalid assignment

	t	$t+1$	$t+2$
Dest 1	$T_{0,0} \rightarrow T_{0,1}$	$\underline{T_{0,1}} \rightarrow T_{0,2}$	
Dest 2		$T_{0,0} \rightarrow \underline{T_{0,1}}$	$T_{0,1} \rightarrow T_{1,1}$

(b) Valid assignment

	t	$t+1$	$t+2$
Dest 1	$T_{0,0} \rightarrow T_{0,1}$	$T_{0,1} \rightarrow T_{0,2}$	
Dest 2		$T_{0,0} \rightarrow T_{1,0}$	$T_{1,0} \rightarrow T_{1,1}$

As $T_{0,1}$ appears twice in slot $t+1$, one as a sender in the first row, and the other as a receiver in the second row. On the contrary, if the first link for *Dest* 2 is $T_{0,0} \rightarrow T_{1,0}$ as in Table 1(b), and two transmissions can be done simultaneously on slot $t+1$, reducing the control loop length. In case of Figure 2(a), 6 destinations takes $T_{0,0} \rightarrow T_{1,0}$ while just 2 destinations takes $T_{0,0} \rightarrow T_{0,1}$ for the first step. So, the length reduction in not so significant. To make the number of two first links as close as possible, the network can sacrifice the optimality of the shortest path algorithm.

We classify grid nodes into 3 groups. The first group, G_D, has the nodes on the diagonal from top left to bottom right. The second group, D_L, has the nodes on the lower triangle of the grid, while the third group, D_U, has the node on the upper triangle. Accordingly, each group can be specified as follows:

$$
\begin{array}{l}
G_D : \{\ T_{i,j} |\ i = j\ \} \\
G_L : \{\ T_{i,j} |\ i > j\ \} \\
G_U : \{\ T_{i,j} |\ i < j\ \}
\end{array}
$$

For nodes in G_L, we make every route start from the link $T_{0,0} \rightarrow T_{1,0}$, and for those in G_U, from the link $T_{0,0} \rightarrow T_{0,1}$. G_D nodes can take the route according to the normal Dijkstra's algorithm. For G_L nodes, after setting the slot error rate of $T_{0,0} \rightarrow T_{0,1}$ to 1.0, the scheduler runs the shortest path algorithm, and vice versa for the G_U nodes. Likewise, the number of appearances for $T_{0,0} \rightarrow T_{0,1}$ and

$T_{0,0} \rightarrow T_{1,0}$ becomes almost even. While this allocation cannot find the optimal path to each node, it can maximize the simultaneous transmissions in each slot and reduce the control loop length.

4 Performance Measurement

This section measures the performance of the proposed loop reduction scheme via simulation using SMPL which provides abundant functions and libraries for discrete event scheduling [16]. Only the downlink graph was considered for simplicity, as uplink and downlink communications are symmetric and calculated completely in the same way. For each parameter setting, 500 sets of link error rates are generated and the success ratio and the loop length ratio are averaged.

The first experiment measures the performance of our loop reduction scheme and compares with the upper and lower bounds. This experiment is conducted for the classic Dijkstra's algorithm. First, the lower bound accounts for the case that each end-to-end transmission can start one slot after another without being blocked from the largest hop-length destination. In this case, the control loop length is ideally smallest and thus the lower bound can be calculated as in Eq. (1).

$$n \times n - 1 + min(hop\ count), \tag{1}$$

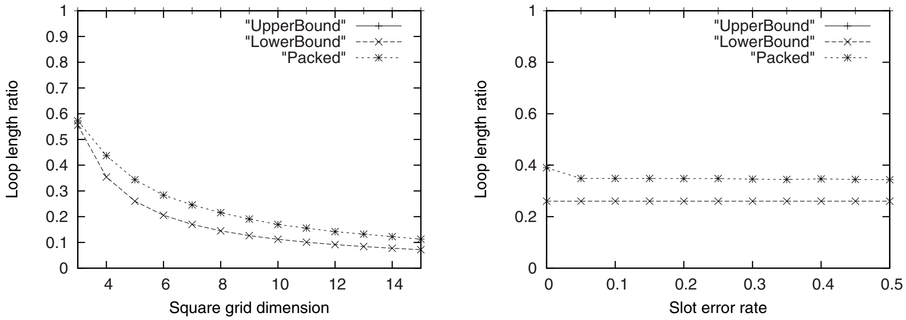
where $n \times n - 1$ is the number of noncontroller nodes in the grid, and for the last message initiated by the controller, it takes as many slots as the hop counts to the destination.

Next, the upper bound of the control loop length is estimated by assuming that there is no overlapped transmission. Hence, it corresponds to the sum of total hops to each node. For the $n \times n$ grid, the number of total transmissions for a control round is calculated as in Eq. (2).

$$\sum_{i=1}^{2(n-1)} (n - |n - i - 1|) \cdot i, \tag{2}$$

where each iteration indexed by i corresponds to the hop count from the controller. Nodes having the same hop counts need the same number of slots. The number of nodes having i hops increases by one until i reaches (n-1), and then decreases also by one until i reaches 2(n-1).

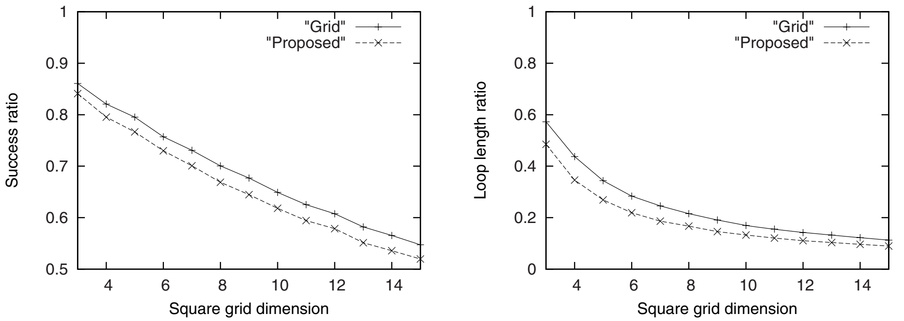
Figure 3 shows the comparison results according to the grid dimension and the slot error rate. First, in Figure 3(a), the average slot rate is 0.1, distributing exponentially, while the grid dimension ranges from 3 to 15. The curves plots the ratio of each loop length to the upper bound. Hence, the curve marked as *UpperBound* always remain 1.0. The ratio of the packet length to the upper bound is just 5.7 % larger at maximum, compared with the lower bound to upper bound ratio. When the dimension is 3, both ratios are around 0.58, as nodes are not sufficiently apart from each other and not so many transmissions can be overlapped. The gap between 2 curves reaches 5.7 % at the grid dimension of 10, but decreases afterwards. It indicates that our scheme can achieve almost ideal loop reduction without complex and time-consuming space search iterations. In addition, Figure 3(b) shows the effect of the slot error rate, which can lead to the variation in the length of each end-to-end path. Here, the grid dimension is set to 5. Even though the gap between the lower bound and our packed scheme gets smaller according to the increase of the slot error rate, its effect is insignificant.



(a) Effect of grid dimension (b) Effect of slot error rate

Fig. 3. Loop length reduction

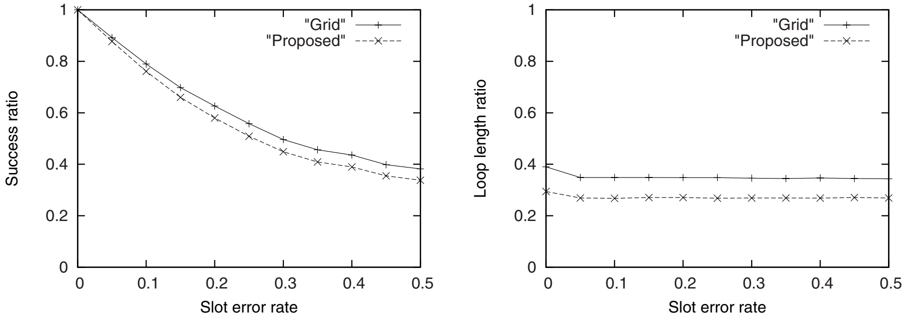
Figure 4 demonstrates the performance of the proposed routing scheme, which sacrifices a little bit the path optimality for loop reduction. Here, the average slot error rate is again set to 1.0 and the dimension is changed from 3 to 15. Figure 4(a) measures how much the success ratio is lost. For all square grid dimension range, the difference is less than 3.2 %. Figure 4(b) measures how much loop reduction we gain. The improvement reaches 9.8 % at the dimension of 3 and decreases up to 3.2 % when the dimension is 15.



(a) Success ratio decrease (b) Loop length reduction

Fig. 4. Loop reduction vs. success ratio according to grid dimension

Finally, Figure 5 shows the effect of the slot error rate in the proposed routing scheme. In this experiment, the grid dimension is also set to 5, while the slot error rate changes from 0 to 0.5. The success ratio gap increases along with the error rate, reaching 4.4 % at maximum. However, the proposed scheme can achieve consistently about 8 % loop reduction in all error ranges.



(a) Success ratio decrease (b) Loop length reduction

Fig. 5. Loop reduction vs. success ratio according to slot error rate

5 Conclusion

This paper has designed a loop reduction scheme and measured its performance for the wireless process control application running on the grid-style traffic light network, aiming at improving the response time of the control system. Based on the WirelessHART standard, which assigns a slot to each (sender, receiver) pair according to the routing schedule, the allocation scheme puts all the pairs having no interference to a single slot, reducing the loop length. For further reduction, the classic Dijkstra's shortest path algorithm is modified such that the number of end-to-end paths starting from the horizontal links and the vertical link, respectively, is almost same. The transmission of the controller, which initiates all message delivery and is the main bottleneck point, will not be blocked. The simulation result demonstrates that the proposed scheme can significantly reduce the control loop length, and the modified path finding algorithm further achieves about 9.8 % improvement, just sacrificing the 3.2 % of transmission success ratio.

As for future work, a fault-tolerant flooding scheme is expected to be very useful in wireless process control [17]. So, we will design a slot assignment scheme combining the split-merge operation to overcome node or link failure for the control message broadcast.

References

1. Bucciol, P., Li, F.Y., Fragoulis, N., Vandoni, L.: ADHOCSYS: Robust and service-oriented wireless mesh networks to bridge the digital divide. In: IEEE Globecom Workshops, pp. 1–5 (2007)
2. Gislason, D.: ZIGBEE Wireless Networking. Newnes (2008)

3. IEC/PAS 62591: Industrial communication networks - Fieldbus specifications - WirelessHART communication network and communication profile (2008)
4. Hart Communication Foundation, Why WirelessHART^{TM}? The Right Standard at the Right Time (2007), `http://www.hartcomm2.org`
5. Han, S., Song, J., Zhu, X., Mok, A.K., Chen, D., Nixon, M., Pratt, W., Gondhalekar, V.: Wi-HTest: Compliance test suite for diagnosing devices in real-time WirelessHART network. In: The 15th IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 327–336 (2009)
6. Lee, J., Shin, I., Kim, C.: Design of a reliable traffic control system on city area based on a wireless network. In: Gervasi, O., Taniar, D., Murgante, B., Laganà, A., Mun, Y., Gavrilova, M.L. (eds.) Computational Science and Its Applications – ICCSA 2009. LNCS, vol. 5592, pp. 821–830. Springer, Heidelberg (2009)
7. Lee, J., Song, H., Mok, A.K.: Design of a reliable communication system for grid-style traffic control networks. Submitted to IEEE RTAS 2010 (2010)
8. Song, S., Han, S., Mok, A.K., Chen, D., Nixon, M., Lucas, M., Pratt, W.: WirelessHART: Applying wireless technology in real-time industrial process control. In: The 14th IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 377–386 (2008)
9. Ramchandran, I., Roy, S.: Clear channel assessment in energy-constrained wideband wireless networks. IEEE Wireless Magazine, 70–78 (2007)
10. Kodialam, M., Nandagopal, T.: Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In: ACM MobiCom, pp. 73–87 (2005)
11. Wang, W., Liu, X., Krishnaswamy, D.: Robust routing and scheduling in wireless mesh networks. In: IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks, pp. 471–480 (2007)
12. Zaidi, Z., Landfeldt, B.: Monitoring assisted robust routing in wireless mesh networks. In: Mobile Network Applications, pp. 54–66 (2008)
13. Jaap, S., Bechler, M., Wolf, L.: Evaluation of routing protocols for vehicular ad hoc networks in city traffic scenarios. In: Proceedings of the 5th International Conference on Intelligent Transportation Systems Telecommunications (2005)
14. Chang, N., Liu, M.: Optimal channel probing and transmission scheduling for opportunistic spectrum access. In: Proc. ACM International Conference on Mobile Computing and Networking, pp. 27–38 (2007)
15. Dai, H., Ng, K., Wu, M.: An overview of MAC protocols with directional antennas in wireless ad hoc networks. In: Proc. International Conference on Computing in the Global Information Technology, pp. 84–91 (2006)
16. MacDougall, M.: Simulating Computer Systems: Techniques and Tools. MIT Press, Cambridge (1987)
17. Wan, P., Huang, S., Wang, L., Wan, Z., Jia, X.: Minimum latency aggregation scheduling in multihop wireless networks. In: MobiHoc, pp. 185–193 (2009)