2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Commanding Mobile Robots via Wireless Ad-Hoc Networks - A Comparison of Four Ad-Hoc Routing Protocol Implementations

Florian Zeiger and Nikolaus Kraemer and Klaus Schilling
University of Wuerzburg,
Department of Computer Science - Robotics and Telematics,
97074 Wuerzburg, Germany
{zeiger,schi}@informatik.uni-wuerzburg.de

***Abstract*— Future applications of mobile robot teams or robot teleoperation require highly dynamic network topologies. One promising approach is the use of relay nodes in wireless ad-hoc networks which require special routing protocols to provide a transparent communication network to the user. This work tests and compares four different existing ad-hoc routing protocol implementations with respect to aspects of mobile robot teleoperation. The reactive routing protocols Ad-hoc On-demand Distance Vector (AODV) and Dynamic Source Routing (DSR), the proactive Optimized Link State routing (OLSR) and B.A.T.M.A.N. are used in test scenarios to command a mobile robot via an ad-hoc network of several communication relay nodes. For all four ad-hoc routing protocols, the route reestablishing behavior is observed. In particular the packet loss and the duration of route reestablishing during test runs with real hardware in an outdoor environment are analyzed.**

I. INTRODUCTION

In the last years, the importance of the teleoperation of mobile robots and teams of mobile robots increased. Recently, more and more mobile robots are developed which are capable to operate in impassable or hazardous environments with little or no communication infrastructure. Several approaches are using wireless ad-hoc networks in many different areas of robot teleoperation. In 2007, Rooker and Birk presented multi-robot exploration with robots using wireless networks [1]. The University of Pennsylvania presented a mobile robot team connected via wireless network which performed localization and control tasks [2] in 2002. Also in the field of rescue robotics [3] or for integrating UAVs into IP based ground networks [4], the use of wireless networks is quite common nowadays. An example for the network topology of these future scenarios is given in Figure 1. The network may consist of several stationary nodes or ground stations and several mobile nodes which can be ground vehicles, aerial vehicles, or humans equipped with communication devices. All these nodes are connected by an ad-hoc wireless network which should guarantee a transparent any-to-any communication. Nevertheless, wireless communication always implies unpredictable communication delays, packet loss, or in worst case the loss of the link which makes the provision of the required quality a challenging task [5]. To avoid the loss of communication, research focused on a dynamic setup of the required telecommunication infrastructure by placing relay nodes on demand [6][7] or using mobile robots as relay nodes [8][9]. These approaches are using communication relays in wireless ad-hoc networks to setup communication networks with dynamic topologies. In these wireless networks no fixed infrastructure exists. Each mobile node not only works as host but also as router for data packets of other nodes. Dynamic topologies of wireless communication networks have advantages like providing direct and indirect any-to-any communication of each network node, redundant communication links in larger networks, no central administration, and a distribution of the traffic load in large networks. Of course, these advantages can only be used with rather complex and special routing protocols providing each node the necessary information of the network topology. The nodes itself are working as routers and must store the routing information of the complete network locally. In the field of wireless telecommunication, many ad-hoc routing protocols for wireless networks were developed [10][11][12][13]. Also some simulations for performance evaluations for larger scale telecommunication networks were done in the past [14][15][16].

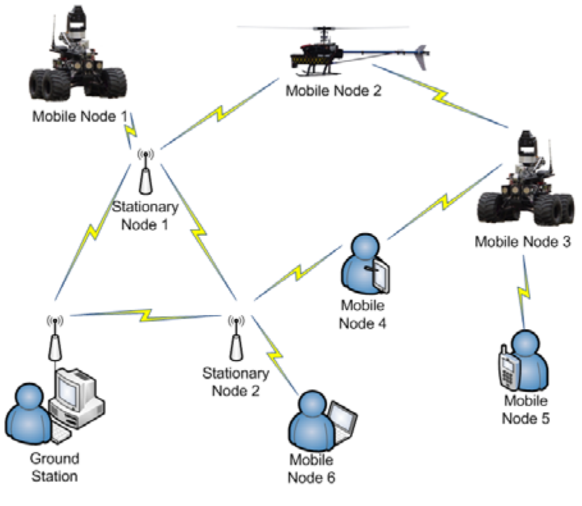

Fig. 1. Future Scenario

This work compares four existing implementations of ad-hoc routing protocols (AODV, DSR, OLSR, B.A.T.M.A.N.) with respect to the aspects of steering mobile robots via a wireless network. The teleoperation of mobile robots requires

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

a certain link quality. Hirche and Buss elaborated in [17], how human perceives the teleoperation of a 1DoF telerobotic system with time delays present. With respect to the steering task of a mobile robot, these mentioned requirements are also present in a wireless network with a highly dynamic topology. To guarantee a useful teleoperation, a certain packet loss ratio must not be exceeded. Also, limits in delay boundaries must be kept – also while reestablishing the route via new communication relay nodes – to give the user a suitable feedback of sensor data or video stream.

The publication is structured as follows. In Section II, the used command, transport, and ad-hoc routing protocols are briefly described. Section III gives a short overview of the robot architecture and the setup of the test scenarios. The following Section elaborates the results of the performed ad-hoc routing protocol comparison, and in Section V future work and conclusions are given.

II. PROTOCOLS

This section gives a short introduction to the ad-hoc routing protocols which are compared. Also, the transport and command protocols used for the mobile robot teleoperation are described.

A. Used Routing Protocols

1) Ad-hoc On-demand Distance Vector (AODV): The AODV routing protocol [12] [13] is a reactive routing protocol and searches for a route on-demand. Figure 2 shows the message exchange of the AODV protocol. In case a certain node is part of an active route, Hello messages are used to obtain the route status. These Hello messages are broadcasted periodically to all neighbors. If a neighbor does not send a Hello message within a specified time a link loss is detected and the node is deleted from the routing table. In addition, a Route Error message (RRER) is generated. To discover a route to an unknown destination, a Route Request (RREQ) message is broadcasted. Each intermediate node which is not the destination and without a route to the destination receiving a RREQ broadcasts this RREQ further. In case the RREQ is received more than once, only the first reception will result in a broadcast. To avoid uncontrolled dissemination of RREQ messages, each RREQ has a certain time to live (TTL) after which it is discarded. When the destination receives a RREQ message a Route Reply (RREP) message is generated and sent back to the source in unicast hop-by-hop fashion along the route which was determined by the RREQ message. After generating a RREP message, the RREQ message is discarded at this node. As the RREP propagates, each intermediate node creates a route to the destination. After the source receives the RREP, it records the route to the destination and begins sending data.

In case the source receives multiple RREPs, the route with the shortest hop count is chosen. The status of each route is maintained in the local routing table and timers are used to determine link failures which will result in the creation of Route Error messages (RERR). More detailed information on AODV is given in [12]. In the test scenarios of is work, AODV-UU version 0.9.5 from Uppsala University (Sweden) is used. [1]

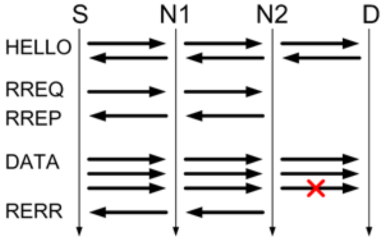

Fig. 2. AODV Messaging Example

2) Dynamic Source Routing (DSR): DSR is also a reactive ad-hoc routing protocol which works similar to AODV but without using Hello messages for route maintenance. However, it is based on source routing [18]. DSR allows the network to be completely self-organizing and self-configuring, without the need of any existing network infrastructure or administration. It does not use any periodic routing advertisement, link status sensing, or neighbor detection packets, and does not rely on these functions from any underlying protocols in the network. The core components of DSR are two main mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network. In case source node (S) wants to send data to an unknown destination host (D), S initiates the Route Discovery mechanism. S broadcasts a Route Request message which identifies the source and destination of the Route Discovery to all neighbors. A Route Request also contains a record listing the addresses of each intermediate node which was forwarding this particular copy of the Route Request. A node which receives this Route Request without being the destination looks up for a source route to the requested destination in its route cache. Without any source route present in its own route cache, the node appends its own address to the route record and broadcasts the Route Request message. In case this request message was received more than once, it is simply discarded. As soon as the Route Request message arrives at the desired destination D, a Route Reply message to S is created which contains an accumulated route record of the Route Request. After S receives this Route Reply, it caches the corresponding route in its route cache and S is ready to transmit data. Of course, there exist mechanisms to omit flooding of the network with Route Requests. A hop limit was introduced and every time a Route Request is forwarded, the hop limit is decremented by one. As soon as it reaches zero, the request is discarded. Also mechanisms for avoiding infinite recursion of Route Discoveries are implemented. A more detailed description of this protocol is given in [10] [19]. The presented work uses DSR-UU version 0.2 from Uppsala University (Sweden). [2]

3) Optimized Link State routing (OLSR): OLSR is a table-driven pro-active routing protocol for mobile ad-hoc

---

[1] http://core.it.uu.se/core/index.php/AODV-UU

[2] http://core.it.uu.se/core/index.php/DSR-UU

networks. It uses uses hop-by-hop routing – each node uses its local information to route packets. OLSR minimizes the overhead from flooding of control traffic by using only selected nodes – called Multipoint Relays (MPR) – to retransmit control messages. Each node in the network selects a set of nodes in its neighborhood, which may retransmit its messages. This set of selected neighbor nodes is called the MPR set of that node. The neighbors of node N which are not in its MPR set, receive and process broadcast messages and will never retransmit broadcast messages received from node N. The MPR set is selected such that it covers all 2-hop nodes. That means every node in the 2-hop neighborhood of N must have a link to the MPRs of N. OLSR continuously maintains routes to all destinations in the network. Therefore, it is suitable for a large set of nodes communicating with each other.

To distribute link and neighborhood information, Hello messages are exchanged periodically. These messages are also used for link sensing and for checking the connectivity. Thus, the network topology is discovered and disseminated through the network, which allows the route calculation. More details on OLSR are given in [20]. The scenario tests in the present work are performed with OLSR version 0.5.3 . [3]

4) B.A.T.M.A.N.: B.A.T.M.A.N. (Better approach to mobile ad-hoc networking) is a new approach to ad-hoc routing – unlike other algorithms that exist right now, B.A.T.M.A.N. does not calculate routes. It continuously detects and maintains the routes by receiving and broadcasting packets from other nodes. Instead of discovering the complete route to a destination node, B.A.T.M.A.N. only identifies the best single-hop neighbor and sends a message to this neighbor. These messages contain the source address, a sequence number, and a time-to-live (TTL) value that is decremented by 1 every time before the packet is broadcasted. A message with a TTL value of zero is dropped. The sequence number of these messages is of particular importance for the B.A.T.M.A.N. algorithm. As a source numbers its messages, each node knows whether a message is received the first time or repeatedly. More details on B.A.T.M.A.N are given at [4]. In the test scenarios of is work, B.A.T.M.A.N. version 0.2 is used.

B. Command and Transport Protocol

The command protocol for the robot the MERLIN Control Protocol is used. It is located at the application layer of the ISO/OSI model. During the tests, several packet types are exchanged between robot and control PC. The command packet has a payload of 13 bytes. These command packets are sent with a frequency of 10 Hz. The robot itself generates 10 different packets (cf. Table I).

For further details on the MERLIN Control Protocol refer to [21] and [22]. As transport protocol, UDP is used which will additionally add 8 bytes for the UDP header. Thus, the complete communication fully complies with the ISO/OSI reference model.

TABLE I

SENSOR PACKETS

Type	Size (bytes)	Interval (ms)
Communication Status	6	500
Link Status	8	500
Orientation	17	300
GPS 1	15	1000
GPS 2	22	1000
GPS 3	6	1000
GPS 4	17	1000
Ultrasonic	13	500
Motor Status	17	1000
Energy Status	21	1000

III. ARCHITECTURE AND TEST SCENARIO

A. Used Robot Architecture

For the performed tests, up to 6 nodes were used. One node is a PC for the operator. Up to 4 MERLIN robots (standard version) were used as stationary communication relay nodes, and one Outdoor MERLIN was used (cf. Figure 3) [21]. All MERLIN robots have a C167 microcontroller for low-level operations and sensor data processing, as well as a PC-104 for more complex and computationally more intensive tasks. The PC-104 uses a Linux operating system and all nodes are equipped with 802.11 standard WLAN equipment (Atheros chip).

Fig. 3. The Teleoperated OutdoorMERLIN Robot

For steering the mobile robot, the operator's PC is running an application which generates command packets described in II-B. These packets are sent via UDP to the mobile robot. The onboard software of the mobile robot generates a UDP packet stream of packets with variable size containing the sensor data.

B. Scenario - Increasing number of hops

In this scenario, the robot starts with a direct connection to the operator's PC. While moving along a predefined track around the obstacle with a velocity of $3 - 5km/h$ (cf. Figure 4), the direct communication between robot and operator PC will be lost and the network must establish a connection via relay node 1 (PC→N1→Robot). While the robot proceeds on its path, the communication link

---

[3] http://www.olsr.org/index.cgi?action=download

[4] https://www.open-mesh.net/batman)

must be established via more and more relay nodes. Finally, the connection will include the stationary relay nodes (PC→N1→N2→N3→Robot). While the network increases the participating relay nodes, two important factors for teleoperation must be observed. First, of course, each hop will induce a certain amount of delay. On one hand, the delay is the result of processing time required at each node. On the other hand each node can only receive or transmit (cf. IEEE 802.11 RFC). Thus, a too large number of hops (much more than used during the performed tests) will make teleoperation impossible. The much more important issue is the behavior of the reestablishment of a communication link via a new route. Here, the required time for a route reestablishing and the packet loss during this event will be measured. Also, alternating routes might occur which should be detected and characterized.

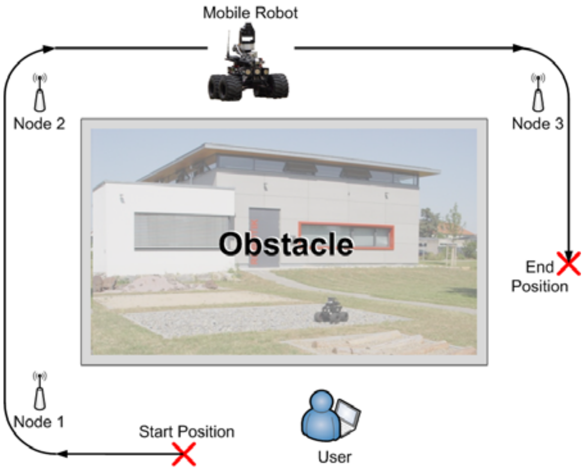

Fig. 4. Scenario 1

With respect to the teleoperation and telecommand of mobile robots, this scenario represents a worst case as only one route between control PC and mobile robot is possible. In addition, the number of communication relays for this route is increased. Thus, no redundant communication links can be used and each delay or communication loss due to the routing protocols will be observed. To assure repeatability of the test runs and to allow for comparison, the mobile robot is always moved with the same velocity while the route reestablishing procedure is initiated. In addition, the locations of the stationary nodes are identically for all tests. Thus, also the areas where the route reestablishing takes place are the same for all test runs.

IV. RESULTS

This Section starts with a single description of a test run for each routing protocol investigating the round trip times of the command packets. Finally, the packet loss during the test runs is evaluated and the time required for route reestablishment is analyzed for each protocol.

A. *B.A.T.M.A.N.*

After each node was configured with a B.A.T.M.A.N. protocol running, the robot was moved. In Figure 5, the round trip times of the command packets for this test run are displayed. At around 45 seconds test time, a direct line-of-sight connection to the controller was not possible anymore and the communication protocol was forced to include one more hop (node 1). After loss of the line-of-sight, a communication drop-out was recognized and the robot had to stop. For more than 50 seconds, the B.A.T.M.A.N. protocol was not able to find a new route between controller and robot. This happened for all tests performed with the B.A.T.M.A.N. protocol.

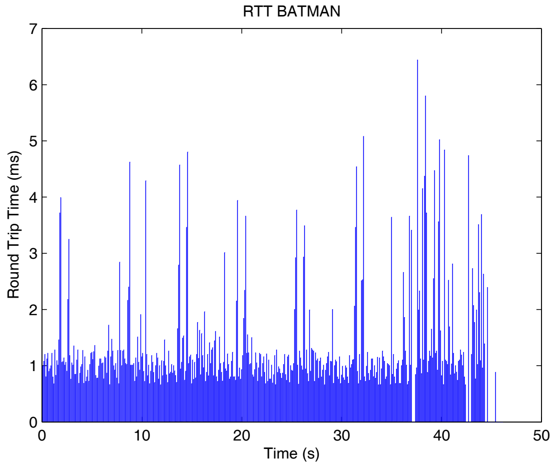

Fig. 5. Round Trip Times for B.A.T.M.A.N.

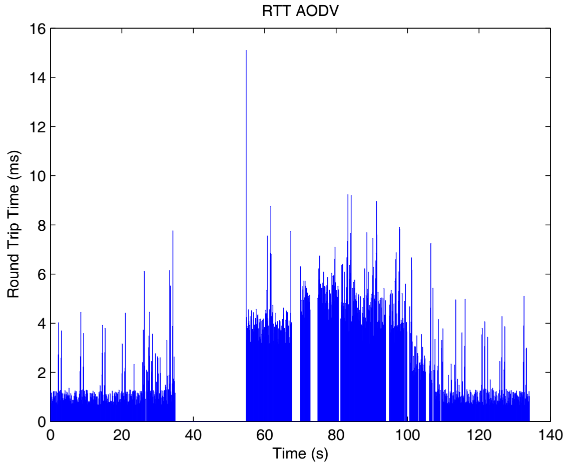

Fig. 6. Round Trip Times for AODV

B. *AODV*

In Figure 6, the round trip times of the control packets during the test of AODV are displayed. At 35 seconds, a direct communication between controller and robot was not possible anymore. Surprisingly, AODV reestablished a route via node 2 (Robot→N2→N1→PC) and 19.8 seconds later, communication continued. This behavior of selecting not the shortest route with respect to the number of nodes was observed several times during the AODV tests. At 67 seconds, the line-of-sight between the robot and node 2 was lost after moving the robot around the next corner of the obstacle. Two short communication drop-outs appeared within the next 10 seconds and a new route via node 3 was used. At 80 seconds of test time, the robot returned on the same way to the start position. On this way back the number and length of the communication drop-outs were negligible.

C. OLSR

The round trip times of the command packets during the OLSR test are given in Figure 7. After 71.5 seconds, the line-of-sight between robot and controller was lost. 10.1 seconds later, the connection via node 2 was established. At 94.5 seconds test time, the direct communication between node 1 and the robot was lost. 14.7 seconds later, a route via node 2 was active. At about 125 seconds test time, the communication was forced to include node 3. During the communication via 4 hops, several packets were lost and the variance of the round trip time is significantly higher than during the communication via links with a smaller amount of hops. On the robot's way back to the controller, node 2 was not used anymore as OLSR established a rout via node 1 after losing the line-of-sight to node 3.

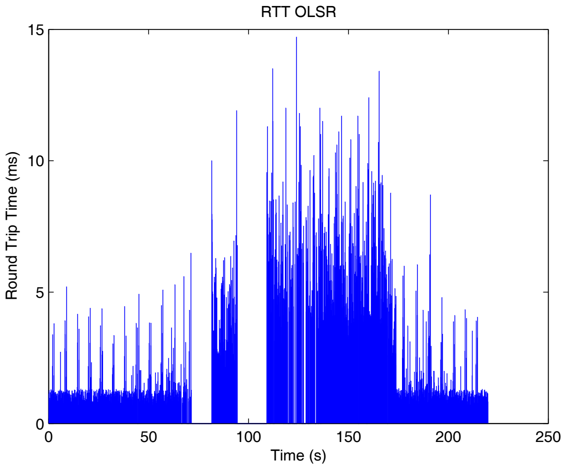

Fig. 7. Round Trip Times for OLSR

D. DSR

Figure 8 shows the command packet round trip times of the DSR test run. Between 35.2 seconds and 37.6 seconds test time, communication via node 1 is established. At 49.6 seconds, direct communication between node 1 and the robot was lost. 2.5 seconds later, node 2 is included into the route and communication is reestablished. At 62.3 seconds test time, DSR established a new route via node 3 within 2.7 seconds. On the way back, only very few packets were lost as the route reestablishing worked fast and reliable.

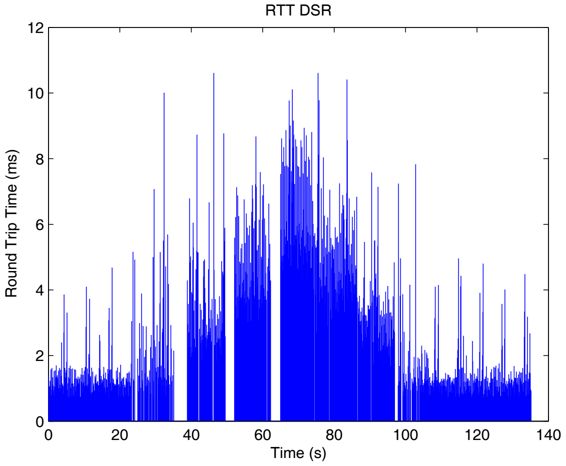

Fig. 8. Round Trip Times for DSR

TABLE II

PACKET LOSS & TIMES FOR ROUTE REESTABLISHING

Protocol	Packet loss during test run	Time for re-routing min.	max.
AODV	29.2%	$2.1s$	$> 30s$
OLSR	14.2%	$10.1s$	$> 30s$
DSR	11.2%	$2.4s$	$2.7s$

E. Packet Loss & Time for Re-routing

In the previous section about B.A.T.M.A.N., unfortunately, this protocol showed to be not useful with the default parameter setting for the tested scenario and it will not be investigated in this section. Table II shows the average packet loss during all performed test runs for each routing protocol. Also the observed minimum and the maximum of the time which is required by each protocol to reestablish a route is given. The value for the maximum time for re-routing of ($> 30s$) for AODV and OLSR means that there were test runs in which a complete communication loss appeared. The values of the time for re-routing given in Table II are only taken from the robots way from controller to node 3. On the way back, the routes were still known due to the previous path of the robot which is not appropriate for observing route discoveries. This table clearly shows that – among the four tested protocols – DSR configured with the default parameter setting performs best in the tested scenario.

V. FUTURE WORK AND CONCLUSIONS

A. Conclusions

In Section IV four different ad-hoc routing protocols are compared in a teleoperation scenario. A mobile robot was commanded in a test scenario which forced the routing protocols to increase the number of participating nodes in the communication link while the robot was moved around an obstacle. Unfortunately, it was not possible to accomplish the scenario with B.A.T.M.A.N.. During all test runs, the communication was lost after the first re-routing procedure was initiated. AODV was originally designed for highly dynamic networks. Routes are established on-demand. In some cases this re-routing took only a very short time (cf. Table II). Nevertheless, in some cases, these periods without communication were longer than 30 seconds. With respect to teleoperation, the observed periods of communication drop-outs were too long for the telecommand of a mobile robot and also the unpredictable and unstable behavior of AODV might be a problem for mobile robot teleoperation. Compared to AODV, the minimum of the required time which is required for re-routing, OLSR is much slower. Rarely, also communication drop outs were observed. With only half of the packet loss, OLSR showed a slightly better performance as AODV. Although OLSR worked more dependable than B.A.T.M.A.N. or AODV, the observed minimum time for re-routing of 10.1 seconds is quite high with respect to teleoperation and will not be appropriate for control loops via this network. DSR showed to be the most reliable and the fastest protocol which is tested. A packet loss of about 11% and a re-routing time between 2.4 and 2.7 seconds make this protocol suitable for reliable telecommand of a mobile robot. With respect to the test scenario, it was expected that DSR performs best, as only one node (the robot) is mobile and all other nodes are stationary. Here, DSR discovers the topology quite fast and only the changes due to the robot's movement result in routing messages. The used test scenario also represents a worst case in the means of route redundancy. Always, only one possible route between controller and mobile robot is possible. This could be the reason for the relatively poor performance of AODV and OLSR. Originally, these protocols were developed to handle much larger networks with a higher node mobility as it was present in the current test scenario. Nevertheless, the presented test scenario is quite typical with respect to teleoperation. In Section I, several approaches used the deployment of communication relays in case the link quality decreases below a certain threshold. This exactly results in the presented scenario where only one possible route between source and destination exists and the number of participating communication nodes is increased.

B. Future Work

As the results proved, this area gives an interesting potential for future work. The presented test was performed with standard parameter settings of the ad-hoc routing protocols. With respect to teleoperation of mobile robots, the optimization of the parameter settings of OLSR, AODV, and DSR will be investigated in future. Also, a continuous monitoring of the link level in combination with the robot's velocity might lead to a better performance. In future, also simple control loops are integrated into these wireless networks which will result in different requirements for the communication link.

REFERENCES

[1] M. N. Rooker and A. Birk, "Multi-robot exploration under the constraints of wireless networking," *Control Engineering Practice*, vol. 15, no. 4, pp. 435–445, 2007.

[2] A. Das, J. Spletzer, V. Kumar, and C. Taylor, "Ad Hoc Networks for Localization and Control," in *Proceedings of the 41st IEEE Conference on Decision and Control, (CDC 2002)*, vol. 3, 2002, pp. 2978–2983.

[3] M. N. Rooker and A. Birk, "Combining Exploration and Ad-Hoc Networking in RoboCup Rescue," in *RoboCup 2004: Robot Soccer World Cup VIII*, ser. Lecture Notes in Artificial Intelligence (LNAI), D. Nardi, M. Riedmiller, and C. Sammut, Eds. Springer, 2005, vol. 3276, pp. 236–246.

[4] F. Zeiger, C. Selbach, B. Ruderisch, and K. Schilling, "An Application Protocol to Integrate a Small Size Helicopter into an IP based Ad-Hoc Network," in *Proceedings of ROBOCOMM 2007, Athens (Greece)*, 2007.

[5] M. A. Hsieh, A. Cowley, V. Kumar, and C. Taylor., "Towards the deployment of a mobile robot network with end-to-end performance guarantees," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'06), Orlando, FL (USA)*, 2006, pp. 2085–2090.

[6] H. G. Nguyen, N. Pezeshkian, A. Gupta, and N. Farrington, "Maintaining Communication Link for a Robot Operating in a Hazardous Environment," in *Proceedings of ANS 10th International Conference on Robotics and Remote Systems for Hazardous Environments, Gainesville, Florida, March 28-31*, 2004.

[7] N. Pezeshkian, H. G. Nguyen, and A. Burmeister, "Unmanned Ground Vehicle Radio Relay Deployment System for Non-Line-Of-Sight Operations," in *Proceedings of the 13th IASTED International Conference on Robotics and Applications, Wuerzburg (Germany)*, 2007.

[8] H. G. Nguyen, N. Pezeshkian, M. Raymond, A. Gupta, and J. M. Spector, "Autonomous communication relays for tactical robots," in *The 11th International Conference on Advanced Robotics, Proceedings of ICAR 2003, Coimbra, Portugal, June 30 - July 3*, 2003.

[9] N. Pezeshkian, H. G. Nguyen, and A. Burmeister, "Unamnned Ground Vehicle Non-Line-of-Sight Operations Using Relaying Radios," in *Proceedings of the 12th IASTED International Conference on Robotics and Applications, Honolulu, Hawaii (USA)*, 2006.

[10] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networks," *Mobile Computing*, vol. 353, 1996.

[11] J. Redi and R. Welsh, "Energy-Conservation for Tactical Mobile Robots," in *Proceedings of Military Communications Conference, MILCOM*, 1999.

[12] S. Das, C. E. Perkins, and E. M. Belding-Royer, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF RFC 3561, July 2003.

[13] I. D. Chakeres and E. M. Belding-Royer, "AODV Routing Protocol Implementation Design," in *Proceedings of the International Workshop on Wireless Ad hoc Networking (WWAN), Tokyo, Japan*, March 2004, pp. pp. 698–703.

[14] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in *Mobile Computing and Networking*, 1998, pp. 85–97.

[15] S. R. Das, C. E. Perkins, E. M. Royer, and M. K. Marina, "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks," *IEEE Personal Communications Magazine special issue on Ad hoc Networking*, vol. February 2001, pp. 16–28, 2001.

[16] T. D. Dyer and R. V. Boppana, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing, Long Beach, CA, USA*, 2001, pp. 56–66.

[17] S. Hirche and M. Buss, *Advances in Telerobotics*. Springer-Verlag Berlin Heidelberg, 2007, ch. Human Perceived Transparency with Time Delay, pp. 191–209.

[18] DARPA Internet Program, "RFC 791, Internet Protocol Specification," September 1981.

[19] Y.-C. Hu, D. B. Johnson, and D. A. Maltz., "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," IETF MANETWorking Group, IETF RFC 4728, July 2004.

[20] P. J. T. Clausen, "RFC 3626, Optimized link state routing protocol (OLSR)," IETF, Network Working Group, October 2003.

[21] D. Eck, M. Stahl, and K. Schilling, "The Small Outdoor Rover MERLIN and its Assistance System for Tele-Operations," in *Proceedings of International Conference on Field and Service Robotics (FSR), Chamonix (France)*, 2007.

[22] M. Frank, S.Busch, P. Dietz, and K. Schilling, "Teleoperations of a Mobile Outdoor Robot with Adjustable Autonomy," in *Proceedings of the 2nd International Conference on Machine Intelligence, ACIDCA-ICMI, Tozeur, Tunisia*, 2005.