

Delay Sensitive Wireless Protocols for Telerobotics Applications*



Cihan Ulaş¹, H. Seçkin Efendioğlu¹, Onur Toker¹, Halûk Gümüşkaya²

¹Fatih University
Dept. of Electrical and Electronics Engineering
34500 Buyukcekmece
Istanbul. Turkey

²Halic University
Dept. of Computer Engineering
Sıracevizler Cad. No: 29
Bomonti-Şişli
Istanbul. Turkey

{culas, onur, hseckin}@fatih.edu.tr, halukgumuskaya@halic.edu.tr

ABSTRACT: In telerobotics applications, communication delays play a central role in overall system performance, and their minimization is a fundamental problem in this area. In this paper, we consider a master arm on the operator side, and a slave arms (Robot arm) operating in a wireless environment with sensors located on or around this slave arm. Sensor data needs to be transmitted to the operator side to provide feedback, and operator commands must be transmitted to the slave arm to operate it. In a setup like this, minimization of transmission delays is of extreme importance for smooth interactive operation. **In the first part of the paper,** we evaluate of two alternatives for wireless protocol development and analysis for our Mitsubishi robot RV-2AJ. The first one is the Universal Software Radio Peripheral (USRP) [11] which has an FPGA for baseband processing and other possible low level protocol details. USRP has analog daughterboards for different frequency bands, including 900MHz, and 2.4GHz bands. USRP allows custom protocol development in the physical, and MAC layers, as well as the layers above these two. The second alternative system considered is a powerful microcontroller with an 802.11b controller card [12, 10]. In this system, lots of physical layer details are fixed inside the 802.11b controller card, there is no operating system on the microcontroller, but one can generate custom 802.11 packets, and build custom protocols above the MAC layer for delay sensitive telerobotics applications. **In the second part of the paper,** we present the details of the proposed custom protocol and its implementation on the selected microcontroller board with an 802.11b controller. The proposed custom protocol is compared with standard data communication protocols, and it is demonstrated that the proposed protocol has smaller communication delays. Finally, in the presence of traffic congestion, or/and low received signal strength, it has been also shown that the proposed protocol has significant advantages compared to standard data communication protocols.

Keywords: Telerobotics, Wireless Communication Protocols

Received: 11 May 2010, Revised 14 June 2010, Accepted 19 May 2010

© 2009 D-Line. All rights reserved

1. Introduction

Communication delays and their minimization are central problems in telerobotics. Performance of a telerobotics system heavily depends on how fast sensor and operator commands can be transmitted from one point to the other, and delays cause significant problems for interactive telerobotic operation. In this paper, we study the design and implementation of high performance telerobotics systems by using custom wireless protocols with no operating system overhead. We first summarize our design alternatives, how certain design decisions are made, then present implementation details of the proposed approach together with delay measurement results. It has been experimentally verified that the proposed custom wireless protocol has smaller communication delays compared to standard data communication protocols, and the advantages become more significant in the presence of traffic congestion or/and low received signal strength.

Teleoperation and teleoperated system design received substantial research attention as it can be applied to many areas such as telesurgery [13], experiments or operations in dangerous environments [14, 15], rescue and reconnaissance [16], etc. Communication protocols and the associated communication delays are the key factors which determine the performance of teleoperated systems. In the literature, teleoperation with both wired and wireless communication protocols are studied in detail.

In telerobotics, wireless communication gained significant popularity due to its convenience in real applications. There are several wireless communication standards that can be implemented on PC based systems or using special hardware. In the literature, several researchers preferred to work with a single board computer (SBC) running an operating system and equipped with a IEEE 802.11b/g wireless network interface cards (NIC). Such a setup provides a user friendly and practical development environment. For example, in [18], a Wi-Fi based controlled robot arm is designed on PCM-3375 SBC, which has embedded Windows operating system and a IEEE 802.11b NIC, and TCP/IP protocols was used for wireless communication. A similar study which is based on a SBC and IEEE 802.11g with Real Time Streaming Protocol (RTSP) can be found in [19]. In [20], a mobile robot which is controlled by a PC104 SBC with embedded Windows is presented. There are several other PC based telerobotics systems with Windows or Linux, and using standard data communication protocols such as TCP/IP and UDP [21-29]. On the other hand, other wireless communication standards such as IEEE 802.15.4 (Zigbee) [30-33], 802.15.1 (Bluetooth) [34-36], and WAP [37] have also been used in certain telerobotics systems. A major drawback in all these is the operating system overhead, and its negative effect on communication delays.

As a completely different research direction, use of RF modems has also been reported in the literature. For example in [38], a high performance wireless serial RF modem which has 10 km communication range and less than 50 ms communication delay is used for remote operation of an excavator system. In a recent study [39], new possibilities for wireless communication of mobile robotic and industrial units are discussed, an ultra-low power 2.4GHz RF communication chip, called nRF24L01+, was used for wireless communication.

Different communication standards and protocols have their own advantages and disadvantages. For example, if we increase the carrier frequency and use a larger and less cluttered portion of the radio spectrum, we can achieve much better performance and lower communication delays. For fair comparison, we will fix the carrier frequency to 2.4GHz. Furthermore, certain point to point communication standards may seem to have very low communication delays, but the analysis may be based on the assumption that the used portion of the spectrum is uncluttered and this may not be valid for certain telerobotics applications. In this work, we do assume that the available spectrum is shared too, and we consider the effects of traffic on communication delays and discuss the advantages of the proposed approach. A custom built higher frequency communication device utilizing a larger bandwidth is likely to outperform the devices reported in the literature working on lower frequencies. However such custom built devices will be more expensive, and difficult to find as off the shelf equipment. We would like to stress that, another important goal of this project is to achieve high performance wireless communication by using off the shelf components only.

In data communications, the focus is on the average data transfer rate, which is also known as the throughput. However, in telerobotics the focus is on communication delays. In control systems, it is a well known fact that sensor/measurement and actuator delays are likely to cause the closed loop system to become unstable. In [40] and [41], effects of communication delays on stability and performance of a teleoperated system is discussed in detail. The use ad-hoc networking in teleoperation [23] is known to improve response times and reduce communication delays (See also [42], and [43] for related work).

TCP/IP and UDP protocols are not designed for real-time telerobotics systems. They are optimized for throughput, however a teleoperation system requires extremely low communication delays and the data rates are much smaller. Nevertheless, TCP/IP and UDP protocols are used in [18], [22], [25], and [29]. On the other hand, certain custom protocols are also reported in the literature. A RT-WMP protocol based on a token-passing approach for real time communication is implemented in Linux platform [27]. Another custom transport protocol, TFRC, for teleoperation systems is developed in [44].

This paper is organized as follows. In section 2, we consider different hardware platforms and design alternatives for implementation of a high performance telerobotics system. In Section 3, we discuss the proposed custom wireless protocol which was built on top of the MAC layer. In Section 4, we illustrate the pictures of the final telerobotic system, present communication delay measurements, and compare these with the delays associated with standard data communication protocols. Furthermore effects of traffic and received signal strength are also discussed and advantages of the proposed custom wireless protocol are illustrated as well. Finally, some concluding remarks are made in Section 5.

2. Design Alternatives

In this section, we will review different hardware alternatives for wireless protocol development.

2.1 Universal Software Radio Peripheral

Universal Software Radio Peripheral is an open source design for software defined radio applications. USRP is designed by Mark Ettus, and is commercially available from Ettus Research LLC. USRP system has a motherboard with an Altera Cyclone EP1C12Q240C8 FPGA with 12,060 Logic Elements, four 64 MS/s 12-bit analog to digital converters, and four 128 MS/s 14-bit digital to analog converters. USRP also supports various analog daughterboards for modulation and demodulation, including 900MHz TX/RX, and 2.4GHz TX/RX boards. Total two I-Q pairs are supported as output, and two I-Q pairs are supported as input. In Fig. 1, a USRP system is shown with one TX and one RX board. USRP also has a USB port which can be used for interfacing with a PC.

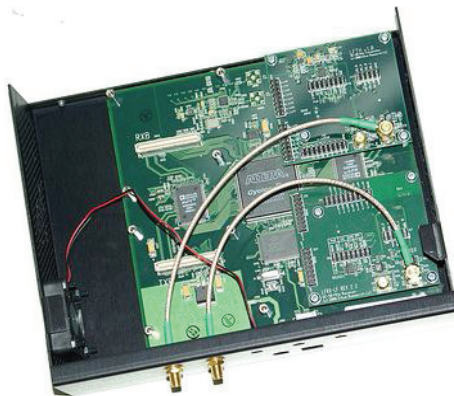


Figure 1. Universal Software Radio Peripheral (USRP) with one TX and one RX daughterboards. There is room for one more TX and one more RX daughterboards

USRP is an extremely flexible platform, and can be used for protocol development in the physical and MAC layers, as well as the layers above these two. At this point, we would like to cite the ADROIT project which has successfully implemented an 802.11 receiver on USRP [13]. In principle by re-engineering the whole 802.11b physical and MAC layers for telerobotics applications, and then building custom protocols on top of these, is likely to result much better performance. However, re-engineering the 802.11b physical and MAC layers for telerobotics applications is by itself a separate major project. USRP seems to be a universal tool for wireless protocol development. However, because of the lack of freely available IP cores for complete 802.11b physical and MAC layers, and the difficulty of developing and testing all of these for the USRP framework, we decided to use a more down to earth approach. Namely, to use an 802.11b controller card and concentrate on layers above the MAC layer. By re-engineering these layers for telerobotics applications, it is still possible to achieve better performance and higher degrees interactive operation.

2.2 ARM boards with 802.11b controller

We have evaluated two different microcontroller / microprocessor boards with 802.11b controllers. The first one was the ConnectCore™ Wi-9C board shown in Fig. 2, which is available from Digi International.



Figure 2. ConnectCore™ Wi-9C board

The ConnectCore™ Wi-9C board has an ARM9 core module which is based on the NET+ARM NS9360 processor, and an 802.11b wireless network interface. Digi International has several other ARM boards with both wireless and/or wired network interfaces. ARM processors are commonly used in several system designs because of their powerful RISC architecture, low power consumption, and abundance of easily accessible information on several ARM software and hardware related problems. However, the main reason why we have not selected this option was the lack of application notes and / or textbooks which explain in a *step by step fashion* how to use and program these ARM+802.11b cards. Because of the high risk of dealing with lots of unknowns about software and hardware related problems, and the amount of time that will be spent on searching for appropriate technical information, these ARM+802.11b cards do not seem to be the best platform choice for developing custom wireless protocols.

2.3 AirDrop-P System: A Microchip PIC18LF6722 board with 802.11b controller

Airdrop-P system has a Microchip PIC18LF6722 microcontroller, and an 802.11b controller. As shown in Fig. 3, it is a very simple, compact, and low cost system, which is available from EDTP Electronics, Inc. for under \$100. In principle, its architecture is similar to the ConnectCore™ Wi-9C board which has an ARM9 core module, and an 802.11b wireless network interface, but is significantly different from USRP. Airdrop-P system has an open source design, and there is an excellent tutorial book which covers the hardware and software of the system in detail [10]. This means that lots of technical problems that one might encounter during initial system development and testing, will be totally eliminated. Because of these, Airdrop-P system seems to be the best embedded system choice for our delay sensitive custom wireless protocol development project. Airdrop-P system software basically consists of two parts. The first part is the low level communication with the 802.11b controller card. The wireless network controller card of the Airdrop-P system is based on the Prism chipset, and has Linux drivers. This first part of the Airdrop-P system software is developed by studying the Linux drivers, and reverse engineering this code.



Figure 3. Airdrop-P system is a Microchip PIC18F6722 microcontroller based board with an 802.11b controller card

The second part of the Airdrop-P system software is based on a minimal TCP/IP stack. This part is mainly based on cooperative multitasking, and does not require a multitasking operating system environment. It is the software designer's responsibility to make sure that all tasks are performed and all timing requirements are met. This will make software development more difficult, but because of the elimination of the operating system overhead, system response times may be further optimized.

A similar approach is going to be pursued for development of custom delay sensitive wireless protocols on the Airdrop-P systems. These protocols will be lightweight, optimized for telerobotics applications only, and implemented by using the cooperative multitasking approach.

3. Proposed Wireless Protocol

In this section, we will discuss four different experiments performed on the Airdrop-P system. These are infrastructure, and ad-hoc mode tests, transmission and reception of custom 802.11 packets, and an 802.11b based wireless mouse experiment.

3.1 Infrastructure mode test

The Airdrop-P system software is by default configured in infrastructure mode. There should be an access point, and a PC to perform this test. Airdrop-P system and the PC communicate with each other over the access point. To perform the infrastructure mode test, we have changed the BSSID in the Airdrop-P system software. After completing the basic "ping" test, we have performed UDP echo tests. This requires a special program called EDTP Internet Test Panel running on the PC [10].

3.2 Ad-hoc mode

To be able to operate the Airdrop-P system in ad-hoc mode, system software needs to be modified. Basically, we have first gone through certain technical documents, and then modified the first part of the Airdrop-P system software accordingly. In

the ad-hoc mode, the Airdrop-P system can communicate with a PC without using an access point. Both connection to an ad-hoc network, and UDP echo tests are performed successfully.

3.3 Transmission and reception of custom 802.11 packets

In this experiment, we use two Airdrop-P systems operating in ad-hoc mode. A custom 802.11 packet format shown in Fig. 4 is introduced, transmission/reception experiments are performed, and basic round trip time measurements are done.

802.11 Header	802.2 LLC	802.2 SNAP	<i>PacketType</i>	<i>FrameID</i>	<i>Data</i>	CRC
30 bytes	3 bytes	5 bytes	1 byte	1 byte	5 or more bytes	4 bytes

Figure 4. Custom packet format

In the 802.2 SNAP header, custom packet type is set to 0x1001, which is different from ARP and IP packet types of 0x0800 and 0x0806. Airdrop-P system software has the so called “802.3 mode”: In this case, the wireless network interface card takes care of the 802.11 addressing for us, and the software development task will be simplified. In the above packet format, gray fields correspond to layers which will not be re-engineered, whereas the white fields correspond to layers which will be optimally designed for delay sensitive wireless communication protocols. *Packet SubType*, *Frame ID* and *Data* fields can be selected to be of different length as well. It is also possible to combine *Packet SubType*, and *Frame ID* fields into a single compound field.

3.4 An 802.11b based wireless mouse

Our final experiment was an 802.11b based wireless mouse. A PS/2 mouse is interfaced to the first Airdrop-P system via a custom designed board, and the second Airdrop-P system is connected to a PC via the serial port. A Visual Basic application is used to read received serial data, and move the mouse accordingly. It has been observed that the 802.11b based mouse works, but there is of course room for improvement. Ad-hoc mode was selected in this experiment, and a custom packet format was used. However, as of writing this paper, no optimization was done to improve the performance.

4. Performance of the Wireless Telerobot

In Figures 5a and 5b, the complete wireless telerobotic system is illustrated. The slave arm is the Mitsubishi RV-2AJ robot with an AirDrop-P card. They communicate via their standard UART ports. The master arm has a PS/2 mouse with a separate microcontroller programmed according to the PS/2 mouse protocol. The master arm also has an AirDrop-P card, and this separate microcontroller is used to simplify the interfacing with the AirDrop-P.

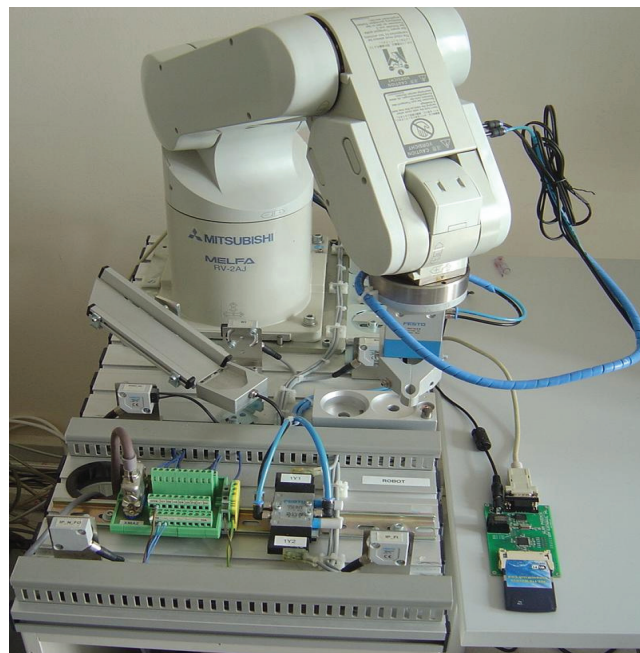


Figure 5a. Wireless telerobotic system - Slave arm

The following experiments are performed to illustrate the advantages of the proposed custom wireless protocol.

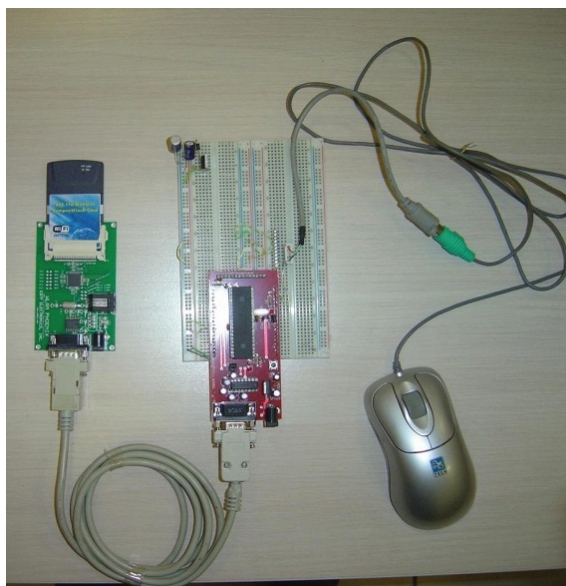


Figure 5b. Wireless telerobotic system - Master arm

4.1 Round trip time measurements

We consider packets carrying 10, 70 and 150 bytes, and measure round trip times in an echo style test setup. It is clear that the proposed custom wireless protocol outperforms UDP, i.e. has much smaller communication delays.

Data Length (bytes)	Custom RTT in ms	Time	UDP RTT in ms
10	5.32	13:00	6.06
70	7.35	13:00	10.05
150	10.09	13:00	15.50

Table 1. Round-trip times for UDP and proposed custom protocol

4.2 Effects of high data traffic for custom protocol

In the second set of experiments, we can generate high data traffic by initiating a large file transfer from a PC to another one using the same wireless medium. In this way, we test the performance of the proposed custom wireless protocol under low and high traffic levels. Regardless of the traffic levels, custom wireless protocol resulted smaller communication delays. Differences are more significant when traffic level is low and number of data bytes is small.

Data Length (bytes)	Custom RTT in ms	TRAFFIC	UDP RTT in ms
10	3.25	LOW	6.06
10	5.23	HIGH	10.05
150	13.10	LOW	15.50
150	14.60	HIGH	17.50

Table 2. Round-trip times versus Traffic levels

4.3 Effects of received signal strength

When the received signal strength is small, we also observed larger communication delays. Again differences were more significant when the number of data bytes were smaller.

5. Conclusion

In this paper, we have summarized different embedded system alternatives for delay sensitive telerobotics oriented wireless network protocols. We have also discussed some preliminary tests performed on the Airdrop-P system, including custom 802.11 packet transmission and reception in the ad-hoc mode. Then a complete telerobotics system by using the proposed custom wireless protocol is illustrated, performance tests and communication delay measurements are done. Advantages of the proposed custom wireless protocol are experimentally proven.

Acknowledgements

This work is supported by TUBITAK (Scientific and Technological Research Council of Turkey) under the Research Grant EEEAG-108E091.

References

- [1] Al-Mouhamed, M., Toker, O., Iqbal, A (2006). A Multi-Threaded Distributed Telerobotic Framework. *IEEE Transactions on Mechatronics*, p. 558-566.
- [2] Al-Mouhamed, M., Toker, O., Al-Harthy, A (2005). A 3D Vision-Based Man-Machine Interface For Hand-Controlled Telerobot. *IEEE Transactions on Industrial Electronics*, p. 1-15.
- [3] Fan, R., Cheded, L., Toker, O (2005). Internet Based SCADA: A new approach using JAVA and XML. *IEE Computing and Control Engineering*.
- [4] Fan, R., Cheded, L., Toker, O (2005). Java plus XML: A powerful new combination for SCADA systems, *IEE Computing and Control Engineering*.
- [5] Fan, R., Cheded, L., Toker, O (2005). Designing a SCADA system powered by Java and XML. *IEE Computing and Control Engineering*.
- [6] Al-Mouhamed, M., Toker, O., Iqbal, A., Islam S. (2005). Evaluation of Real-Time Delays for Networked Telerobotics. *IEEE International Conference on Industrial Informatics*, p. 351-356.
- [7] Al-Mouhamed, M., Toker, O., Iqbal, A., Nazeeruddin, A (2004). A Distributed Framework for Relaying Stereo Vision for Telerobotics, *IEEE/ACS International Conference on Pervasive Services.*, p. 221-225.
- [8] Al-Mouhamed, M., Toker, O., Iqbal, A (2003). Performance Evaluation of a Multi-Threaded Distributed Telerobotic Framework. *Proceedings of the 10th IEEE Inter. Conf. On Electronics, Circuits and Systems*, p. 1284-1287.
- [9] Al-Mouhamed, M., Toker, O., Al-Harthy, A (2003). A 3D Vision-Based Man-Machine Interface For Hand-Controlled Telerobot. *IFAC International Conference on Intelligent Control Systems and Signal Processing*, p. 586-591. Faro, Portugal.
- [10] Eady, F (2005)., Implementing 802.11 with Microcontrollers: Wireless Networking for Embedded Systems Designers. *Newnes*.
- [11] Universal Software Radio Peripheral, <http://www.ettus.com/>
- [12] 802.11b Wireless Network Controller, <http://www.edtp.com/>
- [13] Troxel, G. D. et al (2006). Adaptive Dynamic Radio Open-source Intelligent Team (ADROIT): Cognitively-controlled Collaboration among SDR Nodes. *1st IEEE Workshop on Networking Technologies for Software Defined Radio Networks*, p. 8 - 17. Reston, VA, USA.
- [14] Mitchell, J.H. L., et. al (2009). Teleoperation in Surgical Robotics – Network Latency Effects on Surgical Performance. *Engineering in Medicine and Biology Society*, p. 6860 – 6863. Minneapolis, MN, USA.
- [15] Sung-Min, M., et. al (2009). Development of Tele-operation Control Station for Intelligent Excavator. *Technologies for Practical Robot Applications*, p. 123 – 128. Woburn, MA.
- [16] Poong, W. J., Seul J (2003). Teleoperated Control of Mobile Robot Using Exoskeleton Type Motion Capturing Device Through Wireless Communication. *Advanced Intelligent Mechatronics*, (2) 1107 – 1112.
- [17] Li Y.Y., et. al (2006). Teleoperation of Robots via then Mobile Communication Networks. *IEEE International Conference on Robotics and Biomimetics*, p. 670 – 675. Shatin.
- [18] Gupta, G. S., Mukhopadhyay, S.C, Finnie M (2009). WiFi-based Control of a Robotic Arm with Remote Vision. *International Instrumentation and Measurement Technology Conference*, Singapore.
- [19] Leelasantitham, A., Chaiprapa, P (2009). A Study of Performances on an Automatic IEEE 802.11g Wireless-Standard Robot. *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, p. 1556-1560. Bangkok, Thailand.

- [20] Zheng Fang, GuoFeng Tong, Yimin Hu, Xinhe Xu (2006). An Embedded Platform for Intelligent Mobile Robot. *Proceedings of the 6th World Congress on Intelligent Control and Automation*, p. 9104-9108. Dalian, China.
- [21] Riquelme, C. P., Garcia R., Malumbres, M.P (2005). Analyzing the Behavior of a Real-Time Telerobotic System on IEEE 802.11b Wireless Networks. *IEEE International Conference on Industrial Technology*, (2) p.963-968.
- [22] Babet, F. et. al (2008). Closed Loop Control for a Tele-Echography System with Communication Delays. *Electrotechnical Conference*, p. 356-361. Ajaccio.
- [23] Zeiger, F., Kraemer N., Schilling K (2008)., Commanding Mobile Robots via Wireless Ad-Hoc Networks A Comparison of Four Ad-Hoc Routing Protocol Implementations. *IEEE International Conference on Robotics and Automation*, p.590-595. Pasadena, USA.
- [24] Zheng, Y., et. al (2007)., Development of an Internet Soccer Tele-operation System. *IEEE International Conference on Automation and Logistics*, p. 3035-3038. Jinan, China, Aug. 2007.
- [25] Ju, D., Zhong R., Takahashi M (2008)., Development of Remote Control and Monitor System for Autonomous Mobile Robot Based on Virtual Cell Phone. *Innovative Computing, Information and Control*. Kumamoto.
- [26] Wei, W., Pan, Y., Furuta, K (2005). Internet-based Tele-control System for Wheeled Mobile Robot, *IEEE International Conference on Mechatronics & Automation*, p. 1151-1156. Niagara Falls, Canada.
- [27] Tardioli, D., Villarroel, J.L (2008). Real Time Communications over 802.11 RT-WMP, *Mobile Ad-hoc and Sensor Systems*. p. 1-11. Pisa.
- [28] Kubota, N., Wakisaka, S., Yorita, A (2009). Tele-operation of Robot Partners Through IPOD Thouch. *Computational Intelligence and Intelligent Informatics*, p. 75 – 80. Luxor.
- [29] Lundy, G.W., Bekas, A.J (2002)., Wireless Communications for a Multiple Robot System. *MILCOM 97 Proceedings*, p. 1293 - 1297, Monterey, CA.
- [30] Mata, W. G. et. al (2007)., A Wireless Networked Embedded System with a New Real Time Kernel Partikle. *Congress of Electronics, Robotics and Automotive Mechanics*, p. 21-26, Morelos.
- [31] Johansson, M (2008). Designing and Implementing an Intelligent ZigBee and WLAN Enabled Robot Car. *ICI 4th International Conference*, p. 1-6. Tashkent.
- [32] Bossche, A., Val, T., Campo, E (2006). Proposition of a Full Deterministic Medium Access Method for Wireless Network in a Robotic Application. *Vehicular Technology Conference*, p. 279 – 283, Melbourne, Vic.
- [33] Young, K.D., et al (2008)., Real Time Embedded Control System Development for Wireless Mobile Platforms. *IEEE International Symposium on Industrial Electronic*, p. 2022 – 2027, Cambridge.
- [34] Huang, Y.J., et. al (2009). Design and Implementation of a Multi-Functional Mobile Robot. *ICCAS-SICE*, p. 3414 – 3417, Fukuoka.
- [35] Rissanen, H., et. al (200). Designing and Implementing an Intelligent Bluetooth-Enabled Robot Car. *Wireless and Optical Communications Networks*. p. 1 – 6, Cairo.
- [36] Mester, G (2009). Wireless Sensor-based Control of Mobile Robots Motion. *Intelligent Systems and Informatics*, p. 81-84, Subotica.
- [37] Sgouros, N.M., Gerogiannakis, S (2002). Integrating WAP-based Wireless Devices in Robot Teleoperation Environments. *IEEE International Conference on Robotics and Automation*, p.1191 – 1196, Washington, DC.
- [38] Moon, S., et al (2009). Development of Tele-operation Control Station for Intelligent Excavator. *IEEE International Conference on Technologies for Practical Robot Applications*, p. 123 – 128, Woburn, MA.
- [39] Srovnal, V., Machacek, Z (2009). Wireless Communication for Mobile Robotics and Industrial Embedded Devices. *International Conference on Networks*, p. 253 – 258. Gosier, Guadeloupe.
- [40] Zhang, Lei, et. al (2010). A Networked Teleoperation System for Mobile Robot with Wireless Serial Communication. *IEEE International Conference on Robotics and Biomimetics*, p. 2227 – 2231. Guilin, China.
- [41] Pohjola, M., Nethi, S., Jantti, R (2008). Wireless Control of Mobile Robot Squad with Link Failure. *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops*, p. 648 - 656, Berlin.
- [42] Zhigang, W., MengChu, Z., Ansari, N (2003). Ad-hoc Robot Wireless Communication. *IEEE International Conference on Systems, Man and Cybernetics*, p. 4045 – 4050.
- [43] Zeiger, F., et. al (2008). Challenges in Realizing Ad-Hoc Networks based on Wireless LAN with Mobile Robots. *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops* p. 632 – 639. Berlin.
- [44] Zhou Y., et. al (2007). TFRC-PROBE: A Transport Protocol for Teleoperation Systems of Mobile Robots. *IEEE International Conference on Information Acquisition*, p. 1492 – 1496. Weihai, China.