Halmstad University Post-Print

Protocol for Wireless Real-Time Systems

Henrik Bengtsson, Elisabeth Uhlemann and Per-Arne Wiberg

N.B.: When citing this work, cite the original article.

©1999 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Bengtsson H, Uhlemann E, Wiberg P. Protocol for wireless real-time systems. In: Proceedings of the 11th Euromicro Conference on Real-Time Systems, 1999. Los Alamitos, CA: IEEE Computer Society; 1999. p. 168-174.

DOI: http://dx.doi.org/10.1109/EMRTS.1999.777463
Copyright: IEEE

Post-Print available at: Halmstad University DiVA
http://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-2168

Protocol for Wireless Real-Time Systems

Henrik Bengtsson
Sjoland & Thyselius
Datakonsulter i Goteborg AB
henrik.bengtsson@st.se

Elisabeth Uhlemann
Centre for Computer Systems
Architecture, Halmstad University
elisabeth.uhlemann@cca.hh.se

Per-Arne Wiberg
Centre for Computer Systems
Architecture, Halmstad University
per-arne.wiberg@cca.hh.se

Abstract

A protocol and a communication mechanism intended for time and safety critical applications using a radio channel for information transport are considered jointly. The protocol is based on a scheme of retransmissions done on demand within a given time window. Each retransmission is coded with a varying number of redundant symbols. The set of blocks used for retransmission is controlled by two quality-of-service parameters: deadline for the transmission and the probability that the correct decoded message will reach the recipient before this deadline. Analysis of a protocol model indicates that it is possible to transmit time critical information in a mobile wireless system with very low error probabilities in an industrial environment.

1. Introduction

Wireless transmission of information is used in an increasing number of applications, of which cellular telephony is the most common. Transmission of data is on the way to the market with approaches like PCS [1]. Multimedia applications have been demonstrated in very controlled environments [2]. Wireless communication is, however, anticipated to be found in an increasing number of applications in the future. A driving force for this development is the large cabling cost. Mobility also introduces new kinds of services. In this paper we are primarily concerned with problems and solutions related to the use of wireless communication in an industrial setting.

A large number of applications in industry have real-time demands on the transmission links. In some cases the demands are also safety critical. We mean by real-time that an information package has to be received by the recipient before a certain *deadline*. If not, the consequences will be severe. How severe depends on the application and on how safety-critical the application is. These concepts are quite well founded within the real-time research community [3].

A crucial issue that has to be thoroughly addressed before a wireless system will be accepted for this kind of applications is the reliability aspects. In our work we have taken a probabilistic approach by proposing two new *quality-of-service* parameters (QoS): deadline for delivery and probability for correct delivery before this deadline. An application or a higher protocol layer can, in this scheme, negotiate these parameters with the network.

This commutation scheme is described in Section 2. Section 3 discusses access methods for industrial wireless communication. In Section 4 a protocol is proposed for transforming the QoS requests into message streams. The protocol is analysed with respect to error probability in Section 5. Results from calculations with parameters of typical industrial environment are also presented in Section 5. Section 6 concerns related work, Section 7 aspects of future work, and Section 8 contains our conclusions.

2. Quality of service in mobile real-time systems

QoS negotiations are a powerful way of handling varying demands on the network. QoS parameters used today are, for example: bandwidth, latency and error rate [4]. All these parameters reflect demands on average behavior of the network, which is typical for office applications. For industrial systems the demands are often of worst case nature. This is not taken into account by the protocols for wireless communication used today, e.g. IEEE 802.11 [5]. Therefore we suggest a probabilistic approach that focuses on the worst case behavior.

When an industrial application is designed and analysed, it is often the case that failure probabilities are calculated or set up as requirements on the system. From these failure probabilities a communication failure probability can then be derived.

A communication failure in a real-time system can be divided into information error and temporal error. By information error we mean that the communication system

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

corrupts the information in such a way that the receiver is unable to interpret it or interprets it in an erroneous way. Temporal errors occur when the communication system fails to deliver the information in time. This implies that the application sets up a deadline for the communication system before which it has to deliver. Taking this background into account we therefore propose two new QoS parameters:

- *Deadline for delivery (DL).* This states the latest time at which the receiver (application task) must receive the information being sent.

- *Probability for correct delivery before the deadline (PDL)*. This property controls how reliable the transfer must be. It does not say anything about delivery of correct data after the deadline. This is a request from the application on the communication system.

These two parameters are used to control the network behaviour. It is worth noting that these QoS parameters are useful for non-real-time applications as well. Such an application is multimedia. If a package of image data is sent the deadline is relatively tight but *PDL* can be moderate, because an incorrect delivery will appear as image noise. A too late package on the other hand will disturb the viewing more.

For a control application, it is important that correct information reaches its destination before a quite tight deadline with a fairly low error probability. This will naturally mean that more bandwidth will be required for the transmission.

3. Access method

A protocol is necessary to transform the data into a data stream meeting the QoS demands. In order to develop such a protocol we need to state the context in which the protocol shall work. A proper medium access method (MAC) and a radio frequency band must be selected.

We address the access method first. There are three options: TDMA, FDMA and CDMA [6].

TDMA divides the time axis into several time-slots. It is the most common access method used today, in the wide spread cellular telephone standard GSM. The bandwidth of this standard is sufficient for speech transmission. When it comes to time critical transport, the allocation and scheduling of time-slots becomes critical. There has been some research in this area for wirebound systems [7] [8] [9], which shows that the time-slot allocation becomes quite complicated and it will be even more complicated in a wireless system.

FDMA divides the frequency band into several channels. However, in a multipath environment the signal-to-noise ratio varies within a single time-frequency element dependent on the transfer function. The variance will thus be reduced if the signal is distributed over several time-frequency elements.

CDMA uses a number of orthogonal codes (PN-codes) for multiple access, giving each channel full use of both the frequency band and the time axis. This results in two properties that are attractive for a real-time network: *i*) CDMA rejects some of the multipath fading present in a radio channel. *ii*) Several channels can be established and used simultaneously, provided that the PN-codes are well administrated (e.g. one PN-code per user in the area). No extra administration for access is required. What it means, however, is increased interfering noise experienced by the adjacent users. In order to guarantee bandwidth, and thus QoS, there must be a bound on the number of simultaneous users within a certain area. The interference from other channels is dependent on the selected PN-codes. This is described in detail in [10].

The two advantages with CDMA mentioned above make it a suitable MAC method for mobile real-time systems. There are also disadvantages with CDMA. The two most severe drawbacks from industrial utilisation point of view, are the near-far problem and the bandwidth limitation.

The near-far problem comes from the situation when an adjacent transmitter that outputs high power to reach its recipient jams a closer recipient on another channel. A suggested solution to the problem is a more careful power control of the transmitters [11].

There is another possible solution to this problem, especially feasible for industrial systems. The cellular systems are organised around a basestation with mobile units. This topology is natural for access to a wirebound transport network. In the industrial system the major part of the traffic is found within the ”cell”. Multihop networks [12] are therefore attractive network organisations. The near-far problem can then be attacked by routing trough the neighbouring node rather than drowning it with noise. This is, however, outside the scope of this paper.

The bandwidth limitation originates from the up-scaling of frequency when the signal is spread in the frequency domain. This means that hardware components are required to have very high throughput and speed. Despite this, we think that the advantages outweigh the drawbacks of CDMA for real-time traffic.

In a recently developed communication system called Bluetooth [13] a combination of CDMA and TDMA is used. This protocol is likely to become popular in the future and we present a comparison between Bluetooth and our proposed protocol in Section 6.

Another basic parameter is which radio frequency (RF) band to utilise. The RF spectrum is rapidly getting more and more crowded, forcing us to use higher frequencies. Some frequency bands are kept unlicensed, meaning they are free to use with some restrictions. This standardisation effort is described in more detail in [5].

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

We consider the 2.4 GHz band to be the lowest frequency with reasonable throughput among these new unlicensed frequencies. There has been some development of components for this frequency band by different vendors like Harris, Motorola, and Philips, indicating that this band will be widely used. For detailed information about the products, consult the web pages of the above companies. We base our analysis below on the 2.4 GHz band and performance parameters of the PRISM chip set from Harris.

4. Protocol for real-time transmission over a wireless channel

The application makes a request to send a message with the two QoS parameters, *DL* and *PDL*. The deadline dependent coding (DDC) protocol will transform the request into a data stream with the desired probability of success.

The DDC protocol is a retransmission protocol schematically shown in Figure 1. As the name of the protocol implies, the coding of the information is made dependent on the deadline. In our analysis we use a Reed-Solomon (RS) block code.

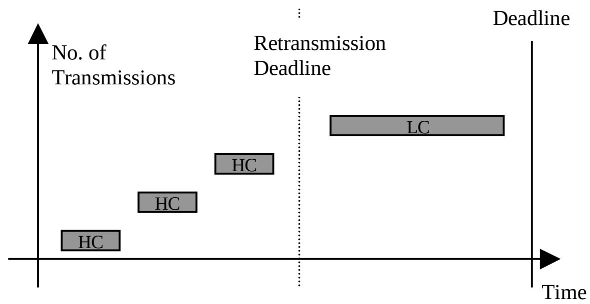

Figure 1: The DDC protocol. HC stands for high rate coded whereas LC stands for low rate coded.

RS-codes have some good properties that make them suitable for this problem. First, they have maximum distance separation, MDS, of the code words, which gives a small probability of decoder error. The latter is shown to be a very important property in the DDC protocol. Secondly, the RS-codes have very good error correcting performance, which is also advantageous in conjunction with the DDC. A comprehensive description of RS-codes is given in [14].

First a high rate coded message (i.e. few redundant bits) is sent. The receiver tries to decode the message. If the decoder fails, the sending node will retransmit the message. This is repeated in case of yet another decoder failure. When the receiver has three erroneous copies of the message, it performs a bitwise majority voting on the three copies. This results in a new message, which it tries to decode. If it still fails to decode the message, the retransmissions can continue or a low rate coded message can be sent. Which of these scenarios that are chosen depend on the QoS requested.

5. Analysis of the deadline dependent coding protocol

Since the users of the communication system request a specific probability for the transmission, a mapping from this request to the retransmission scheme must be done.

When mapping the QoS parameters onto the DDC protocol, a diversity of techniques are available, since many of the protocol parameters can be varied. The goal is to find a mapping technique that fulfils the QoS demands, with a minimum of interference to the other channels.

In the following discussion, the deadline is kept constant and the probability for correct delivery is varied. It should be noticed that for a short time window, i.e. when the deadline is close in time, more redundancy has to be introduced in order to maintain the same probability of delivery.

One approach is to keep the block length of both the high rate and the low rate coded message constant and only increase the number of retransmissions when the probability of correct delivery (*PDL*) increases. This technique is easily implemented, as the block lengths are known in advance. This method is hereafter called the dynamic retransmission scheme (DRT).

Another solution is to keep the number of retransmissions constant and instead increase the block length and thus the number of redundant bits for each retransmission. This method is hereafter called dynamic block lengths (DBL). The receiver must, however, know the block lengths in advance in order to decode the message. To solve this problem, the first message should always be of a certain block length and contain information about the following block lengths.

The DRT scheme, which provides a simple and straightforward solution, is chosen for the mapping of the QoS parameters.

In order to calculate the probabilities involved, we have developed a model of the communication channel. The communication scenario with six interfering nodes is shown in Figure 2. An important entity in our model is the Energy per bit to Noise ratio at the receiver, E_b/N_0. This entity is used for evaluating the performance of the Reed-Solomon codes.

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

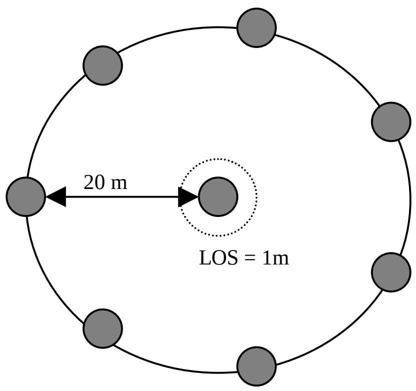

Figure 2: Communication scenario with six interfering nodes.

A transmitter is sending a message to a remote receiver at a distance, *d*, of 20 meters. We assume a line of sight, LOS, of 1 meter. The received carrier is a function of transmitted power and total path loss. The latter is due to multipath fading. The LOS propagation loss $L_f(d_0)$ [15] is:

$$L_f(d_0)[dB] = 10\log G_T + 10\log G_R - 20\log f - 20\log d_0 + \\ + 20\log\left(\frac{c}{4}\right)$$

where G_T and G_R are the transmitter and the receiver gain respectively, *f* is the radio frequency and *c* is the speed of light. The absolute mean path loss in dB can now be obtained as [15]:

$$L_f(d)[dB] = L_f(d_0) - 10{*}n{*}\log(d/d_0)$$

The path loss exponent *n* represents how much the path loss increases with distance. The path loss exponent is typically in the range of $2 \le n \le 4$ for indoor environments and $3 \le n \le 5$ for outdoor communication. We have chosen n=3.9. The calculations are made with the PRISM chipset wireless LAN parameters. The PRISM operates on the 2.4 GHz frequency band at 1 Mbps. The antenna power is 63 mW. From this we derive E_b/N_0 to be 8.12 dB.

It should be noted that in the following probability calculations neither the probability of loosing the acknowledge signal nor the probability that the receiver is unable to phase lock on the preamble are taken into account.

The performance of RS-codes is measured in terms of the probability of decoder failure and decoder error. The decoder will select the code word closest, in Hamming distance, to the received code word. If no code word is located within a predefined bounded Hamming distance a decoder failure is declared, here denoted *P(F)*:

$$P(F) = 1 - \left[\sum_{j=0}^{[(d_{\min}-1)/2]} \binom{n}{j} (1-s)^j s^{n-j}\right] - P(E)$$

where $d_{\min}$ is the minimum Hamming distance between two code words, *n* is the block length and *s* is the probability of the correct symbol being received.

If the received code word contains several errors, it may cause the decoder to select a valid code word, different from the one transmitted. This is called a decoder error, here-denoted *P(E)* [14].

$$P(E) = \sum_{j=d_{\min}}^{n} A_j \sum_{k=0}^{[(d_{\min}-1)/2]} P_k^j$$

where P_k^j is the probability that a received word is exactly at Hamming distance *k* from a weight-*j* code word [14].

$$P_k^j = \sum_{r=0}^{k} \binom{j}{k-r} \binom{n-j}{r} p^{j-k+r} (1-p)^{k-r} s^{n-j-r} (1-s)^r$$

where *p* is the probability that one particular incorrect code word symbol is received. The number of weight-*j* code words in a MDS code is [14]:

$$A_j = \binom{n}{j} (q-1) \sum_{i=0}^{j-d_{\min}} (-1)^i \binom{j-1}{i} q^{j-i-d_{\min}}$$

where *q* is the Galois field over which the RS code symbols are defined.

The probability of decoder error is relatively small, compared to the probability of decoder failure, but it becomes an important factor in a retransmission scheme. A decoder error does not result in a retransmission; hence there is no way of correcting the error. In fact, every time a retransmission is made, it gives the decoder a new chance of making a mistake as seen in Figure 3. The probability of decoder error will consequently increase with the number of retransmissions as:

$$P_{tot}(E) = \sum_{k=1}^{N} P_k(E) + P_{MV}(E) + P_{LC}(E)$$

where *N* is the number of transmissions of the high rate coded message and P_{MV} the improved probabilities after

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

the majority voting procedure. P_{LC} represent a low rate coded transmission.

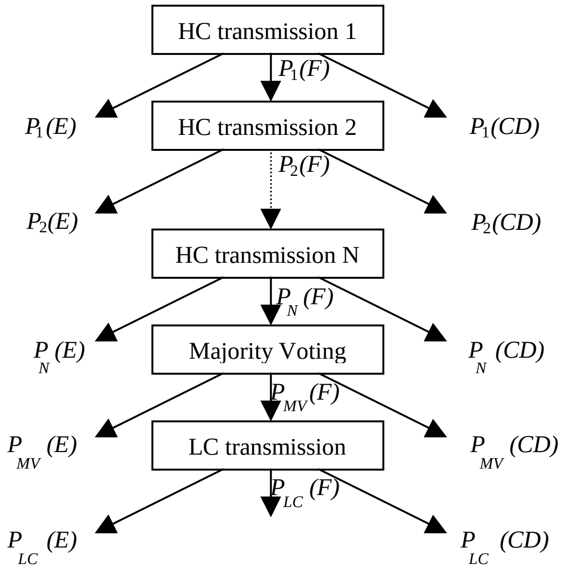

Figure 3: The probabilities in the retransmission scheme. HC stands for High rate Coded, LC for Low rate Coded whereas CD stands for Correct Decoding.

A decoder failure, on the other hand, will result in a retransmission and thus a new chance of correct decoding. Hence, the probability of decoder failure will decrease with the number of retransmissions as:

$$P_{tot}(F) = P_1(F)^N \cdot P_{MV}(F) \cdot P_{LC}(F)$$

The bit error rate is improved after a majority voting procedure [15] as:

$$P_{eMV} = \sum_{k=(N+1)/2}^{N} \binom{N}{k} P_e^k (1-P_e)^{N-k}$$

where N is the number of transmissions made and P_e is the original bit error rate.

The encoding and decoding time is highly dependent on hardware implementation and is omitted in our calculations. The PRISM chipset needs a preamble and header for each package of 272 bits. This is included in the calculations.

We define the message error rate (*MER*) as the rate at which the protocol fails to deliver before the specified deadline. We will treat two levels of *PDL* starting with the lower level.

For a low *PDL* an RS(15,11) is used for the high rate coded message and RS(63,11) for the low rate coded message. It should be noted that the selection of Reed-Solomon codes should be done with care, as it influences the system performance. The codes selected in these examples have been chosen to illustrate how the performance is affected. Three retransmissions of the high rate coded message, a Majority Voting procedure and a low rate coded transmission are done. This yields a probability of message error of $8.60 \cdot 10^{-6}$ and $1.55 \cdot 10^{-5}$ if an additional interfering node is added at a distance of 1 meter. This is approximately equal to the probability of decoder error after one transmission of the high rate coded message, $P_1(E)$. The total message error rate is thus highly dependent of the probability of decoder error.

The Majority Voting procedure improves the system performance by reducing the probability of decoder failure.

The time to transmit three high rate coded messages, one low rate coded message and to compute a Majority Voting procedure is 2.057 ms. The acknowledge signal is coded with RS(15,11).

The RS(15,11) is capable of correcting two errors. If a code with higher error correcting capability is used, the probability of message error can be substantially reduced. This is the case for a high *PDL* level.

Here we use an RS(63,32) for the high rate coded message and RS(127,32) for the low rate coded message. The message error rate is then reduced to $5.22 \cdot 10^{-29}$ for six interfering nodes and $4.71 \cdot 10^{-28}$ with an additional node at a distance of 1 meter. These values are also equal to the probability of decoder error when one transmission of the high rate coded message has been made, i.e. $P_1(E)$.

In this case the time to transmit three high rate coded messages, one low rate coded message and to compute a Majority Voting procedure is 2.409 ms. The acknowledge signal is coded with RS(63,32) in this example.

The above analysis indicates that the total message error rate is highly dependent of the probability of decoder error when only one transmission has been made.

6. Related work

Not much work has been done on real-time protocols for wireless connections. There has, on the other hand, been done quite a lot on time-critical communication over wirebound networks. Protocols in this context can be divided into the following categories: (*i*) protocols based on static allocation of time slots [16] [17], (*ii*) protocols based on token passing [18], and (*iii*) protocols based on unpredictable carriers like Ethernet made suitable for real-time traffic by means of a window technique [19].

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

A recently developed protocol for wireless communication is Bluetooth [13]. This protocol is supported by a large group of industries, and is thus likely to be accepted as a standard. It uses some of the mechanisms mentioned above. We will in this section make a comparison of this protocol to ours.

Bluetooth uses an access method based on frequency hopping CDMA. The network is divided into clusters called piconets. Each such cluster has a unique hopping sequence. The access within a piconet is controlled by a master using TDMA when communicating with the slaves in the cluster.

The slaves are accessed in a way similar to token passing. In each interval, when a slave holds the token, the master can send one message and receive one message from the slave. Each message can occupy one to five time-slots. In case of a retransmission this is done the next time the slave holds the token, thus a varying delay is introduced called token round-trip delay.

In our protocol the sender answers a request for retransmission without regarding any delay caused by a token round-trip. Bluetooth has automatic repeat request (ARQ). The retransmission is, however, delayed until the master is in turn to communicate with that particular slave. This causes problems when the deadline is smaller than the token round-trip time, which is often the case.

Bluetooth supports two types of channels: asynchronous for data mentioned above, and synchronous for voice. The synchronous channel sends voice data periodically. No ARQ is used, and the information is coded with a given number of redundant symbols.

The synchronous channel could be used for transmitting real-time data because of its predictable characteristics. The problem is, however, that variations in the channel properties are not considered. The coding does not give the very low bit error rates, necessary for critical real-time systems.

Although Bluetooth is a very interesting protocol it lacks some properties to make it suitable for real-time systems with high demands on information integrity.

Another commonly known protocol for wireless traffic is based on the standard IEEE 802.11 [5]. This communication scheme can be compared to Ethernet and has properties similar to it. The only possibility to use these protocols for real-time traffic, as we see it, is to use some of the mechanisms for wirebound networks, mentioned earlier in this section.

7. Future work

In Section 5 it was shown that the *MER* of the DDC protocol is highly dependent of the probability of decoder error *P(E)*. The probability of decoder failure *P(F)*, although higher than *P(E)*, has less impact on the system performance as it is reduced when a message is retransmitted or a Majority Voting procedure is performed. *P(E)*, on the other hand, is increased for every retransmission. Consequently, if *P(E)* could be decreased, the *MER* in the system would improve. Below, we point out two approaches that have the potential of reducing *P(E)* without increasing the network activity. They have, however, not been investigated further.

Aulin [20] has analysed a method using some of the redundant bits in the code word for error detection only, and not for error correction. In this way more errors can be detected. Obviously, the drawback is that fewer errors can be corrected, thus increasing *P(F)* and the probability of a retransmission. This method could, however, be used to decrease *P(E)*. Another advantage is that if errors are detected it is possible to use only two messages to reconstruct the correct code word, as there is a low probability for errors to occur in the same bit in the two different messages. The use of this approach has the potential of balancing *P(F)* and *P(E)* to meet QoS with better performance.

An attempt to introduce soft decision decoding will also be investigated further in the near future.

8. Conclusions

In this paper a wireless deadline dependent coding protocol (DDC) has been analysed. The DDC protocol is controlled by two Quality of Service, QoS, parameters and aims at keeping the network activity as low as possible. Different techniques to map the two QoS parameters, deadline for delivery (*DL*) and the probability of correct delivery before this deadline (*PDL*), to the DDC protocol have been discussed.

We have shown that, using available technology, it is possible to transmit time critical information in a mobile wireless system with very low error probabilities, thus fulfilling the industrial safety demands and still get enough throughput for many applications.

The dynamic retransmission scheme (DRT) that has been analysed indicates that the system performance is limited by the probability of decoder error *P(E)* after one transmission. Due to its cumulative characteristic, *P(E)* increases with the number of retransmissions, while the probability of decoder failure *P(F)* decreases. The Majority Voting procedure is shown to have significant impact on the system performance, without contributing to the overall network activity.

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.

9. References

[1] T.Y.C Woo, T. F. La Porta and K. K. Sabnoni, Pigeon: A wireless two-way messaging system, *Journal of Selected Areas in Communication*, vol. 15, no. 8, pp. 1391, October 1997.

[2] A Sampath and J. M. Holtzman, Access control of data in integrated voice/data CDMA systems: Benefits and tradeoffs *Journal of Selected Areas in Communication*, vol. 15, no. 8, pp. 1511, October 1997.

[3] J. Stankovic, and K. Ramamritham, Hard real-time systems, Tutorial text, *IEEE Computer press*, Wash. DC 1988.

[4] C. M. Aras, J. F. Kurose, D.S. Reevs and H. Schulzinne, Real-time communication in packet-switched networks, *Proc. of the IEEE Vol 82, No 1*, Jan 1994.

[5] IEEE standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specification 1-55937-935-9

[6] R. Steel, *Mobile Radio Communication*, Wiley & Sons 1996, ISBN 07273-1406-8.

[7] O. Bridal, L-Å. Johansson, J. Ohlsson, M. Rimen, B. Rostamzadeh, R. Snedsböl and J. Torin, *DACAPO: A dependable distributed computer architecture for control of applications with periodic operation, TR-165*, Department of Computer Engineering, Chalmers University of Technology, 1993

[8] K. Nilsson, B. Svensson and P-A. Wiberg, A modular, massively parallel computer architecture for trainable real-time control systems, Presented at *AARTC'92: 2nd IFAC Workshop on Algorithms and Architectures for Real-Time Control*, Seoul, Korea, August 31 - September 2, 1992. Published in *Control Engineering Practice*, Vol 1, No 4, pp.655-66

[9] M. Jonsson, K. Börjesson and M. Legardt, Dynamic time-deterministic traffic in a fiber optic WDM star network, *Proc. of 9th Euromicro Workshop on Real-Time Systems*, Toledo, Spain, June 11-13, 1997.

[10] K.S.Gilhousen, I.M. Jacobs, R. Padovani, A.J. Viterbi, L. A. Weaver, Jr. and C. E. Wetherly III, On the capacity of cellular CDMA Systems, *IEEE Transactions on Vehicular Technology*, Vol. 40, No 2, May 1991.

[11] J. Huang, Study of feedback CDMA interface cancellation receiver for combat the near-far problem *Proc. of IEEE 6th International Conference on Universal Personal Communications, ICUPC'97.* Part 2, San Diego, CA, USA 1997.

[12] T.W Chen, J.T. Tsai and M. Gerla, QoS routing performance in multihop, multimedia, wireless networks, *Wireless Personal Communications,* Kluver Academic Publications ISBN 0-7923-8017-7.

[13] J. Haartsen, Bluetooth- The universal radio interface for ad hoc, wireless connectivity, *Ericsson Review*, No. 3, 1998, pp 110-117.

[14] S.B. Wicker, *Error Control Systems for Digital Communication and Storage,* Prentice Hall Inc. 1995

[15] K. Fehrer, *Wireless Digital Communications – Modulation & Spread Spectrum Applications,* Prentice Hall Inc. 1995.

[16] H. Kopetz, A. Damm, C. Koza, M. Mulazzani, W. Schwabl, C. Senft and R. Zainlinger, Distributed fault-tolerant real-time systems: The Mars approach, *IEEE Micro* Vol. 9, No. 1 February 1989, pp 25-40.

[17] M. Jonsson Two fiber-ribbon ring network for parallel and distributed computing system, *Optical Engineering*, Vol. 37 No. 12, December 1998, pp 3196-3204.

[18] J.K. Strosnider, T.E. Marchok, Responsive, Deterministic IEEE 802.5 token ring scheduling, *J. Real-Time Systems*, Vol. 1, No. 2, 1989, pp 133-158.

[19] W. Zhao, J.A. Stankovic and K. Ramamritham, A window protocol for transmission of time constrained messages, *IEEE Transactions on Computers*, Vol. 39, No. 9, September 1990, pp 1186-1203.

[20] T. Aulin, *Ericsson Internal Report,* Document no. L/BK 6176, December 1986.

Authorized licensed use limited to: Halmstad Hogskola. Downloaded on January 27, 2010 at 07:04 from IEEE Xplore. Restrictions apply.