

Selective-Repeat Protocol with Multiple Retransmit Timers and Individual Acknowledgments

Drago Hercog

*University of Ljubljana, Faculty of Electrical Engineering, Tržaška 25, 1000 Ljubljana, Slovenia
E-mail: Drago.Hercog@fe.uni-lj.si*

Abstract. A new variant of the selective-repeat protocol is presented. While the protocols used until now have been based on cumulative acknowledgments and a single retransmit timer, a protocol based on individual acknowledgments and a separate timer for each outstanding packet is proposed here. In spite of a slightly increased complexity of the proposed protocol, when comparing it to the basic selective-repeat protocol, it is still relatively simple to be implemented; furthermore, it seems to be even less complex than the selective-reject protocol. The simulation results presented in the paper show that the new protocol is more efficient not only than the basic selective-repeat protocol, but also than the selective-reject protocol. Because the basic protocol is especially nonefficient when several packets in line are lost, the advantage of the proposed protocol is especially distinctive in case of heavy losses in the channel, but also in case of long channel delays. The relative efficiency of the proposed protocol is almost the same as the relative efficiencies of the basic and selective-reject protocols. Therefore, the new protocol seems to be most suitable in nonmultiplexing environments, especially in the data-link layer where cumulative acknowledgment single timer go-back-N protocols have mostly been used until now.

Keywords: selective-repeat protocol, selective-reject protocol, retransmit timer, individual acknowledgment, protocol efficiency

Protokol s selektivnim ponavljanjem ter več časovniki za ponovno oddajo in individualnimi potrditvami

V članku je predstavljena nova različica protokola s selektivnim ponavljanjem. Medtem ko smo doslej uporabljali protokol s kumulativnimi potrditvami in enim samim časovnikom za ponovno oddajo, tu predlagamo uporabo protokola z individualnimi potrditvami in posebnim časovnikom za vsako že oddano, a še ne potrjeno sporočilo. Čeprav je predlagani protokol nekoliko kompleksnejši kot osnovni protokol, ga je danes vseeno mogoče dovolj preprosto implementirati; kaže pa, da je predlagani protokol celo manj kompleksen kot protokol z negativnimi potrditvami. Rezultati simulacij kažejo, da je tak protokol učinkovitejši ne le od osnovnega protokola s selektivnim ponavljanjem, ampak tudi od protokola, ki poleg pozitivnih uporablja tudi negativne potrditve. Ker osnovni protokol deluje neučinkovito predvsem takrat, ko se zaporedoma izgubi več sporočil, je prednost predlaganega protokola še zlasti izrazita pri velikih pogostnostih izgub, pa tudi pri velikih zakasnitvah v kanalu. Relativna učinkovitost predlaganega protokola pa je približno enaka kot relativna učinkovitost osnovnega protokola in protokola z negativnimi potrditvami. Zato se zdi novi protokol še posebej primeren v okoljih, kjer ne uporabljamo multipleksiranja, še zlasti v povezanem sloju referenčnega modela OSI, kjer smo doslej največ uporabljali protokole s ponavljanjem zaporedja s kumulativnimi potrditvami in enim časovnikom za ponovno oddajo.

1 INTRODUCTION

The problems that usually occur in a telecommunication channel include packet corruption, loss, duplication and reorder. All these problems can be solved using the so-called ARQ (Automatic Repeat reQuest) protocols [1] which therefore provide for the reliable message transfer service to the users. The ARQ protocols are most usually used in the data-link and transport layers of the OSI reference model.

An ARQ protocol is most usually implemented as a sliding window protocol. Although a generalised sliding window protocol was already specified in the literature [2, 3], three special cases are usually used in practice, namely stop-and-wait, go-back-N, and selective-repeat protocol [4].

Of course, a logical correctness is the most important property that must be possessed by any protocol. However, the protocol efficiency is also very important; it indicates the share of the channel resources that a protocol is capable to use to provide for a service to users. The protocol efficiency is most often defined as the ratio of the highest possible information transfer rate that is useful for users to the nominal transfer rate of the channel. In [5], the relative protocol efficiency was also defined as the ratio of the information transfer rate that is useful for users to the transfer rate of all the messages

that are transmitted into the channel. While the protocol efficiency is the most useful performance measure in nonmultiplexing environments, the relative efficiency is especially important in multiplexing environments.

The goal of this paper is to propose a new variant of the selective-repeat protocol that has not yet been proposed or even used up to now. The protocol efficiency of this variant is superior to the efficiency of the basic sliding window protocol, although the implementation of the new variant seems to be quite simple.

2 SELECTIVE-REPEAT PROTOCOL

Both in theory and in practice, all the three special cases of the sliding window protocols were already carefully examined. In this paper, the selective-repeat protocol and its improvement will be investigated.

The selective-repeat protocol is known to be the most efficient among the three; furthermore, its relative efficiency is much higher than the relative efficiency of the go-back-N protocol [5]. The advantage of the selective-repeat protocol is also its ability that even misordered messages can be stored at the receiving side, and yet they are passed to the receiving user in the correct order.

The selective-repeat protocol must use the same basic mechanisms as all the other sliding window protocols to be logically correct: all transmitted packets must be numbered which allows the receiver to detect the packet loss or packet disorder; all packets must be channel-coded before transmission to allow the receiver to detect and discard corrupted packets; all correctly received packets must be acknowledged by sending positive acknowledgments to the sender; all packets that have already been received but not in a right order must be stored in the receive buffer; all outstanding (already transmitted but not yet acknowledged) packets must be stored in the transmit buffer; all outstanding packets must be guarded by an active timer. The sizes of the transmit and receive buffers are referred to as the transmit and receive window widths, respectively. In the case of the selective-repeat protocol, the sizes of the transmit and receive buffers are the same.

Three kinds of acknowledgments can be used [6,7]. An individual acknowledgment acknowledges a single packet to have been received. A block acknowledgment acknowledges the reception of all the packets in a specified interval of sequence numbers. A cumulative acknowledgment acknowledges all the packets up to a specific one.

The transmitter can use a single or several retransmit timers. A single timer is used if cumulative acknowledgements are also used. The timer must be running whenever there is at least one outstanding packet in the transmit buffer; when a new packet is transmitted, the timer is activated (if not already active); when a new acknowledgment is received, the timer is stopped and immediately reactivated if there are still

any unacknowledged packets in the transmit buffer. On the other hand, a separate retransmit timer can be used for each outstanding packet; in this case, the number of the necessary timers evidently equals the transmit window width. Although the transmitter usually uses several timers for different purposes, the retransmit timer is the most important; therefore the term timer will always indicate the retransmit timer in this text.

In practice, cumulative acknowledgments and a single retransmit timer are almost exclusively used. Concerning the implementation complexity of the transmit and receive protocol entities, such implementation has been considered the most simple; furthermore, when using cumulative acknowledgments, several packets can be acknowledged with a single acknowledgment, thus imposing a lower load upon the network.

In addition to the basic mechanisms mentioned above which are necessary for a logically correct protocol operation, some additional mechanisms can also be used to improve the protocol efficiency. In practice, negative acknowledgments (rejects) are almost always used. The receiver sends a reject message when it receives an unexpected packet, considering the previous packet to have been lost. Usually, the goal of a protocol designer is to resolve most losses with negative acknowledgments, with the timer serving as a backup mechanism to provide a correct operation for the cases when rejects themselves are not sufficient.

3 SELECTIVE-REPEAT PROTOCOL WITH MULTIPLE RETRANSMIT TIMERS

Nowadays, it is quite difficult to pretend that a specific protocol is superior only because it is simple to be implemented and requires lower amounts of the processor and memory resources, as the processor power and memory size are steadily growing while their prices are steadily dropping. Furthermore, the implementation of timers is nowadays very simple, as they are often implemented simply by counting clock ticks. It is therefore quite probable that modern protocols are still based on cumulative acknowledgments and a single retransmit timer more or less for historic reasons, but also because other possibilities have not yet been enough researched.

In this paper, a topic will be tackled that has been left untouched until now: a selective-repeat protocol with individual acknowledgements and multiple retransmit timers will be described. One should be aware that the use of multiple timers is compatible with individual acknowledgments as well as the use of a single timer is compatible with cumulative acknowledgments.

The only difference in the receiver's behaviour between the basic selective-repeat protocol (i. e., the selective-repeat protocol with cumulative acknowledgments and a single retransmit timer) and the proposed multiple-timer selective-repeat protocol is that

whenever the receiver receives a new packet that can be stored into its receive buffer according to the current position of the receive window it sends the individual acknowledgment of this packet, instead of letting the transmitter know where the beginning of its receive window is currently positioned.

When the transmitter transmits a packet, it also activates the timer associated with the sequence number of this packet. If the timer expires, the packet is retransmitted. The transmitter records which packets in its transmit window have already been acknowledged. After a new acknowledgment has been received, the transmitter inspects which packets have already been received and moves (if necessary) the transmit window so that it again begins with the unacknowledged packet with the lowest sequence number.

As has been just indicated, the usage of the timers slides along the number scale concurrently with the usage of the sequence numbers and the usage of the memory in the transmit buffer, just like in any sliding window protocol. Hence, the number of the retransmit timers that are necessary equals exactly the transmit window width.

In a sense, the operation of the selective-repeat protocol with multiple timers and individual acknowledgments is similar to parallel operation of the W stop-and-wait protocols, where W is the window width.

4 EXPECTATIONS

Although the selective-repeat protocol is more efficient than the other two types of the sliding window protocols and its relative efficiency is substantially higher than the relative efficiency of the go-back-N protocol [5], it encounters problems if two or more packets in a sequence are lost; in practice, such a situation occurs quite frequently at intermediate and even more at high bit error rates, as bit errors often occur in bursts. In spite of such problems, the basic protocol is logically correct, only its efficiency drops. The efficiency decrease is expected to be more distinctive at higher bit error/loss rates.

In Fig. 1, an example operation of the basic selective-repeat protocol, using only the basic mechanisms, a single timer and cumulative acknowledgments, is shown for the case when two packets in line (with the sequence numbers 1 and 2) were lost. In this figure, I,x indicates the information (i. e., transferring user information) packet with sequence number x , and A,y indicates the cumulative acknowledgment acknowledging all the packets up to the sequence number $y-1$, inclusive. The vertical lines at the left edge of the figure show the timer operation, the symbol \rightarrow indicating the timer expiration, and the symbol \times indicating stopping the timer. The arrows at the right edge show how the user information is passed up to the receiving user: $IND(z)$ indicates that the user contents of the packet $I(z)$ have been passed to the user (of course, users have no idea

about the sequence numbers). In the scenario shown in Fig. 1, the left protocol entity (indicated as a transmitter in the figure) only sends information packets and receives acknowledgments, while the right protocol entity (receiver) only receives the information packets and sends acknowledgments. In this scenario, the transmitter transmits six different information packets (with the sequence numbers 0 through 5); hence, six user messages are transferred from the sending to the receiving user.

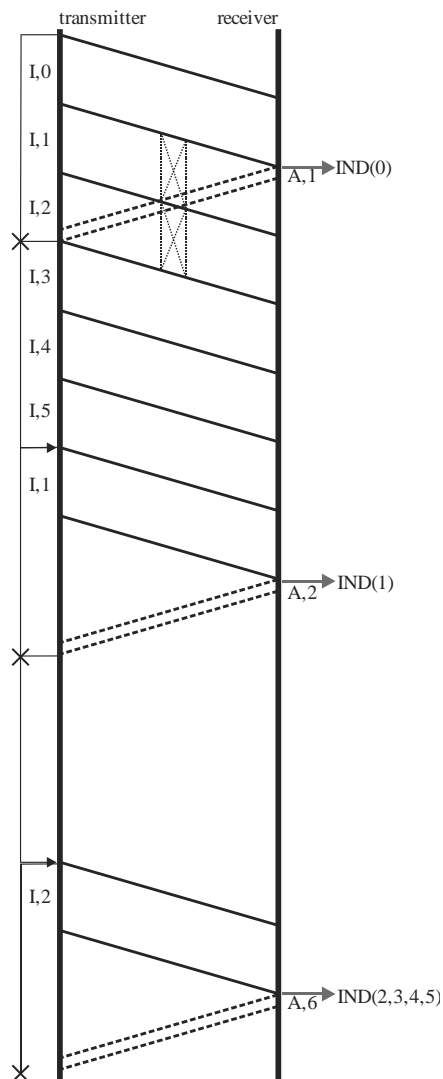


Figure 1. Example operation of the basic selective-repeat protocol, the case of two lost packets.

The operation shown in Fig. 1 is simple; the retransmit timer management is as described in Section 2, and the acknowledgments are cumulative. The problem in this scenario occurs because two packets have been lost. The transmitter becomes aware that something went wrong only one round trip time (i. e., one timer expiration time) after it had received A,1. But still at that moment, it is not aware that two packets have been lost; so it assumes that a single packet has

been lost (which is indeed more probable), retransmits I,1 and sets the timer again. Then one more round trip time must be elapsed before it is aware that the I,2 also has been lost! In this way the transmitter is losing the time instead of quickly reacting to the data loss. It is clear that the problem stems from the fact that the information brought by the cumulative acknowledgments and a single timer expiration is not enough specific and therefore not sufficient for an efficient operation (although it is sufficient for a logically correct operation). Evidently, the impact of the multiple-packets loss on the protocol efficiency is detrimental. The longer is the channel delay, the longer is the timer expiration time, so the problem can be expected to be more grave with long channel delays.

In Fig. 2, the same scenario (the packets with the sequence numbers 1 and 2 lost) is shown. However, the multiple timers protocol with individual acknowledgments is used here. In this case, any outstanding packet is guarded with a separate retransmit timer, and the timers are independent. In this figure, only timer expirations (\rightarrow) and stoppings of timers (\times) are shown. The timer labels indicate the sequence numbers with which the timers are associated. While I,x and IND,z have the same meaning as in Fig. 1, A,y indicates the individual acknowledgment of the information packet I,y specifically.

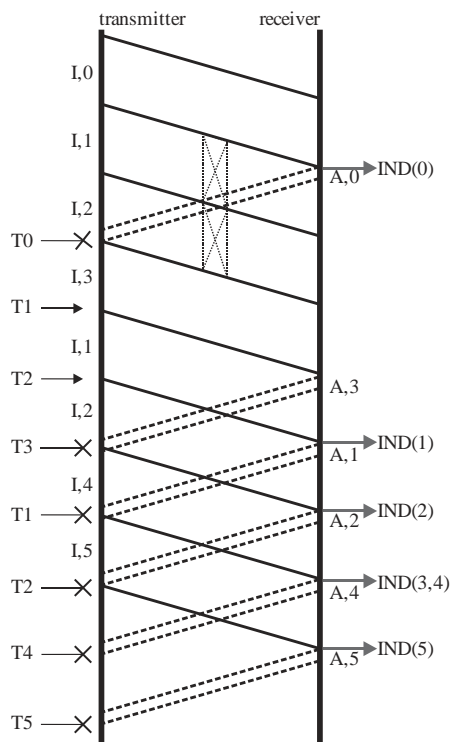


Figure 2. Operation of the multiple-timers selective-repeat protocol, the case of two lost packets.

Because the timers guarding different outstanding packets are independent in the case of this protocol, the transmitter is aware of the loss of I,2 as quickly as it is

aware of the loss of I,1; the time elapsed between a packet loss and the consequent reaction (retransmission) of the transmitter is the same with all the lost packets (that is the timer expiration time which must be slightly longer than the round trip time). Any timer expiration tells to the transmitter more specifically what has happened (although the transmitter still does not know if the information packet or its acknowledgment has been lost), so the transmitter loses no additional time before retransmission; hence, the usage of the time is better and consequently the efficiency is also better.

Comparing Figs. 1 and 2, it is easy to see that less time is needed to transfer six messages with the multiple-timers selective-repeat than with the basic selective-repeat protocol which consequently means that the multiple-timers protocol is more efficient than the basic selective-repeat protocol. However, this statement is based on the consideration of a special case and shall be confirmed by simulating many different scenarios with both variants of the protocol.

A similar communication scenario can be drawn (but is not shown here) for the selective-repeat protocol with negative acknowledgments (usually referred to as the selective-reject protocol). In case of the same scenario as shown in Figs. 1 and 2 (two lost packets), no difference in the efficiency between the selective-reject and multiple-timers protocols can be seen. However, in case that three packets are lost in line, the efficiency of the selective-reject protocol is lower than the efficiency of the multiple-timers protocol, but higher than the efficiency of the basic selective-repeat protocol.

Generalising the above special cases, it is possible to expect that the efficiency of the multiple-timers selective-repeat protocol will be higher than the efficiency of the basic selective-repeat protocol, with the efficiency of the selective-reject protocol in between.

5 SIMULATION RESULTS

Simulation models of each of the three variants of the selective-repeat protocol discussed in this paper (i. e. basic, selective-reject and multiple-timers selective repeat protocols) were developed according to the methodology published in [8]. The basic protocol uses only the basic mechanisms, cumulative acknowledgments and a single retransmit timer. The selective-reject protocol also uses cumulative acknowledgments and a single timer; in addition, whenever the receiver receives an out-of-sequence packet, it sends negative acknowledgments for all the packets that are missing in the receive window between the beginning of the window and the recently received packet; after reception of a negative acknowledgment, the transmitter retransmits the rejected packet. The multiple-timers protocol uses individual acknowledgments and a separate timer for each outstanding packet, as described in Section 3 and illustrated in Fig. 2. In all cases, the protocol efficiency

and relative protocol efficiency were the results of interest.

A number of scenarios, differing in the channel and protocol parameters, were simulated. Figs. 3 and 4 show the results of a characteristic scenario; however, the results of other scenarios were similar in principle, the differences being of a more quantitative than qualitative nature. The parameters of the scenario shown in Figs. 3 and 4 are the following: the user message length 100 octets, overhead and acknowledgment length 5 octets, nominal channel bit rate 1 Mb/s, channel delay 2 ms and window width 6; it should be noted that the window width 6 guarantees the protocol efficiency 1 in case the channel is lossless or the loss rate is very low (as can be well seen in Fig. 3).

In Fig. 3, the protocol efficiency is shown as a function of the bit error rate. As expected, the protocol efficiency decreases with the bit error rate. In conformance with the speculations in Section 4, the efficiency of the multiple-timers protocol is the highest and the efficiency of the basic protocol is the lowest.

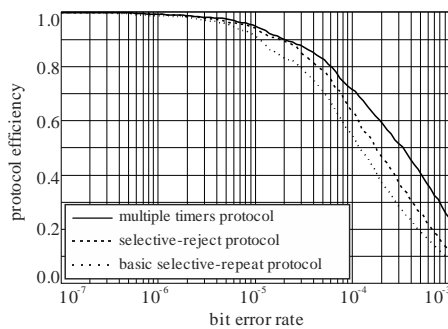


Figure 3. Protocol efficiency as a function of the bit error rate.

In Fig. 4, the ratios of the efficiencies of the selective-reject and multiple-timers protocols against the efficiency of the basic protocol are shown, also as functions of the bit error rate. As expected, this ratio is higher at higher bit error rates; in case of a lossless channel, all the three protocols function identically, but with the bit error rate increase, the probability of a loss of more packets in a sequence also increases, and hence the advantage of the multiple-timers protocol against the basic one also increases. It is evident that the increase in the efficiency at high bit error rates is especially significant in case of the multiple-timers protocol. It is also clear that introduction of the multiple timers improves the efficiency much more than introduction of the negative acknowledgments.

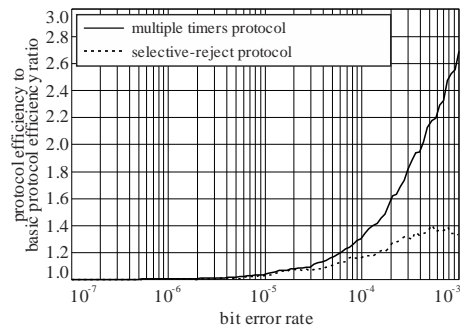


Figure 4. Ratio of the efficiency of the selective-reject and multiple-timers protocols to the efficiency of the basic protocol

Although the efficiency of the multiple-timers protocol seems to be always better than the efficiency of the basic selective-repeat protocol, the improvement depends on the channel and protocol parameters. It was found out that the improvement is more significant in case of longer channel delays (which also means longer round trip times and consequently longer timer expiration times). Such a result can be expected if the problems encountered by the basic protocol, as seen in the example of Fig. 1, are considered. In Fig. 5, the ratios of the efficiencies of the selective-reject and multiple-timers protocols to the efficiency of the basic protocol are shown as functions of the channel delay. The bit error rate in this diagram is 10^{-3} and the window width is always adjusted to the value which guarantees the 100 % efficiency in case of a lossless channel, while all the other parameters have the same values as in Figs. 3 and 4.

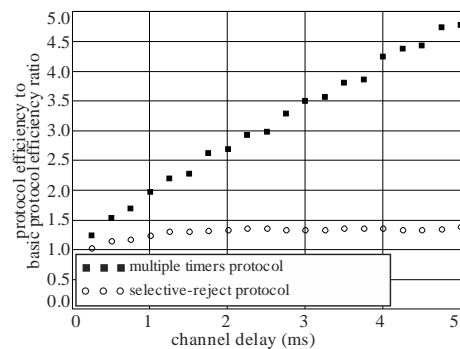


Figure 5. Ratio of the efficiency of the selective-reject and multiple-timers protocols to the efficiency of the basic protocol.

The advantage of the multiple timers protocol is better utilisation of the time; hence, the protocol efficiency is also better. However, the ratio of successfully transferred packets to all transferred packets is the same with all the three protocols, therefore, the relative efficiencies [5] of the three protocols are also the same; simulation results confirmed this fact.

6 CONCLUSIONS

The sliding window protocols are massively used in the telecommunication networks to provide for a reliable transfer of messages over unreliable channels which can corrupt, lose, reorder or duplicate packets. Three special cases of these protocols are usually used in practice, namely the stop-and-wait, go-back-N and selective-repeat protocols. The latter is known to be the most efficient among the three. In addition to the basic mechanisms which are mandatory for a logically correct operation of a protocol, negative acknowledgments are almost always used to improve the protocol efficiency. Although three types of the acknowledgments and one or several retransmit timers can be used, cumulative acknowledgments and a single timer are almost always used in practice.

In this paper, it was shown that in case when two or more packets in a sequence were lost, cumulative acknowledgments and expiration of a single timer do not provide for a sufficiently accurate information about what has happened in the channel to allow the transmitter to react efficiently. It is clear that the information provided by the expiration of a specific timer associated with a specific outstanding packet is more specific and accurate and therefore allows for a more efficient communication. Hence, a new protocol was defined in this paper; this is a selective-repeat protocol which uses individual acknowledgments and a separate timer for each outstanding packet; it should be noted that individual acknowledgments are more compatible with multiple timers operation than cumulative acknowledgments. This protocol shall be termed multiple-timers selective-repeat protocol. The simulation results show the expectations about the new protocol to be correct, as the efficiency of this protocol is substantially higher not only than the efficiency of the basic selective-repeat protocol, but also than the efficiency of the selective-reject protocol. The multiple-timers selective-repeat protocol seems to be better than the basic selective-repeat or selective-reject protocols not only because of its better efficiency, but also because its implementation seems to be more simple than the implementation of the selective-reject protocol (this statement is based on the comparison of the complexities of the simulation models of the selective-reject and multiple-timers selective-repeat protocols).

The advantage of the multiple-timers selective-repeat protocol over the other variants of the selective-repeat protocol is its higher efficiency; however, its relative efficiency is approximately the same as the relative efficiencies of the basic selective-repeat and the selective-reject protocols. Hence, the multiple-timers variant is suitable for the nonmultiplexing environments at the first place; in the multiplexing environments, the choice is not essential, as in such environments, the relative efficiency is more important. The new protocol should also be successful as a datalink layer protocol. Up to now, the go-back-N protocol has been mostly

used in the data-link layer, as it is simpler while only slightly less efficient as the basic selective-repeat protocol; however, the multiple-timers variant is substantially more efficient than the basic selective-repeat, and even more than the go-back-N protocol, especially at higher bit error rates.

Finally, it should also be emphasised that the use of individual acknowledgments and multiple timers is compatible with the selective-repeat protocol rather than with the go-back-N protocol. This is why the topic of the research presented in this paper is the selective-repeat protocol with multiple timers and individual acknowledgments.

REFERENCES

- [1] Stallings, William, *Data and Computer Communications, 9th Ed.*, Pearson, 2011
- [2] Hercog, Drago, "Generalised Sliding Window Protocol", *Electronics Letters*, 38(18), pp. 1067-8, 2002
- [3] Hercog, Drago, "Generalization of the basic sliding window protocol", *International Journal of Communication Systems*, 18(1), pp. 57-75, 2005
- [4] Tanenbaum, Andrew S., Wetherall, D.J., *Computer Networks, 5th Ed.*, Prentice-Hall, 2011
- [5] Hercog, Drago, "Performance of Communication Protocols Using Multiplexed Channels", *Elektrotehniški vestnik*, 82(1-2), pp. 51-54, 2015
- [6] Gouda, Mohamed G., *Elements of Network Protocol Design*, Wiley, 1998
- [7] Hercog, Drago, "Protokoli z drsečim oknom - pregled mehanizmov, razvrstitev protokolov in izzivi za nadaljnje raziskave", *Elektrotehniški vestnik*, 76(4), pp. 211-6, 2009
- [8] Hercog, Drago, "Mapping SDL protocol specification into protocol performance simulator based on Watkins simulation library", Proceedings of the 5th EUROSIM Congress on Modeling and Simulation, 6-10 September 2004, ESIEE Paris, Marne la Vallée, France

Drago Hercog received his dipl. ing., MSc. and DSc. degrees from the University of Ljubljana, Faculty of Electrical Engineering, Ljubljana, Slovenia, in 1974, 1984 and 1989, respectively. He is employed at that same institution as Associate Professor. He is teaching several telecommunication courses at the undergraduate and postgraduate level. He was teaching telecommunications courses at three graduate engineering schools in France for several years. His research interests are in telecommunication networks and telecommunication protocols with the emphasis on sliding window protocols. He is the author of many papers in the areas of telecommunication protocols, discrete event simulation and engineering education. He is also the author of the book »Telecommunication Networks« in the Slovenian language.