

Protokoli z drsečim oknom – pregled mehanizmov, razvrstitev protokolov in izzivi za nadaljnje raziskave

Drago Hercog¹

¹ Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, SI-1000 Ljubljana, Slovenija
E-pošta: Drago.Hercog@fe.uni-lj.si

Povzetek. Članek podaja pregled osnovnih mehanizmov, ki jih mora uporabljati logično pravilen protokol z drsečim oknom. Nato predlagamo razvrstitev protokolov glede na štiri merila: širini oddajnega in sprejemnega okna, vrsto potrditev, število uporabljenih časovnikov in obseg funkcionalnosti protokola v protokolnem skladu. Po pregledu običajne uporabe protokolov v sedanjih praksi in novejših raziskovalnih izsledkih na tem področju ob koncu članka navedemo še nekatere zanimive raziskovalne izzive s področja protokolov z drsečim oknom: uporabo posplošenega protokola v praksi, uporabo individualnih in intervalnih potrditev, uporabo več časovnikov, vpliv medsebojne (ne)odvisnosti sosednjih slojev na učinkovitost protokola s selektivnim ponavljanjem in vpliv časa izteka časovnika na učinkovitost protokola s ponavljanjem zaporedja in posplošenega protokola.

Ključne besede: telekomunikacije, prenos podatkov, komunikacijski protokoli, protokoli z drsečim oknom

Sliding Window Protocols – Overview of Communication Mechanisms, Protocol Classification and Prospects of Further Research

Extended abstract. Automatic Repeat reQuest (ARQ) protocols are needed to provide for reliable transport of data. Sliding window protocols are the most usually used family of the ARQ protocols. Although they have already been used for decades [1,2], the research in the sliding window protocols is still important, as the data transfer reliability is becoming crucial in radio-based mobile networks [3,4], in spite of the availability of Hybrid ARQ (HARQ) protocols [5,6].

To provide for logical correctness of a sliding window protocol, channel coding of messages (allowing for error detection), positive acknowledgments of correctly received messages, sequence numbering of messages, one or more timer(s), and memory buffers at both the sending and receiving side must be used [7,8]. The number of messages that can be stored into send and receive buffers are referred to as send (W_s) and receive (W_r) window widths, respectively. Sequence numbering must be done using modulo M arithmetic, where M conforms to Eq. (1). Additionally, some auxiliary mechanisms, such as negative acknowledgments and enquiries, can be used only to improve the protocol efficiency.

In this paper, a sliding window protocol classification is proposed according to the following criteria: send and receive window widths, acknowledgments, number of timers and protocol stack layer functionalities. According to the send and receive window widths criterion, one can distinguish the generalised sliding window protocol (with any combination of W_s and W_r) [9,10] and its special cases stop-and-wait protocol ($W_s=W_r=1$), go-back-N protocol ($W_s>W_r=1$) and selective-repeat protocol ($W_s=W_r>1$). In case of a lossless channel, the protocol efficiency is given by Eq. (2) and can be 1 if the send window width is large enough ($W_s > t_r/t_t$) where t_t is the time needed to transmit a message and t_r is the round-trip time. In case of a real channel, the efficiency is lower. Fig. 1 shows a typical dependence of the efficiency on the bit error rate for the generalised protocol with the following protocol/channel

parameters: bit rate 1 Mb/s, user message length 200 octets, overhead 5 octets, channel delay 5 ms, send window width 10, and receive window widths 1, 2, 5 and 10. It can be seen that the selective-repeat ($W_r=10$) efficiency is higher than that of go-back-N ($W_r=1$), while efficiencies at $W_r=10$ and $W_r=5$ are almost equal. According to the acknowledgments criterion, individual, cumulative and block acknowledgments can be used [8]. According to the timer(s) criterion, one or several timers (as much as W_s) can be used. According to the protocol stack layer functionalities, some protocols cover functionalities of a single layer, while the others merge functionalities of two adjacent layers (usually data-link and physical), thus improving efficiency or even preventing interlayer congestion [13].

In the current practice, go-back-N protocols are mostly used in the data-link layer [14-17], while selective-repeat protocols are mostly used in the transport layer [18], especially above the connectionless network layer. Furthermore, protocols using cumulative acknowledgments and a single timer are mostly used. Transport layer protocols are mostly specified and implemented independently of adjacent layers. Data-link layer protocols are often implemented in hardware jointly with the physical layer; however, they usually do not rely on this cross-layer design.

Recently, researchers have mostly been active in the areas of formal specification and correctness proofs of protocols [19,20], as well as protocol efficiency modeling and analysis [20], generalisation of sliding window protocol [9], timeout recovery strategies [21], dependence of protocol efficiency on the timer expiration time [22,23], and advantages of the cross-layer protocol design [13,23]. However, several problems still remain to be solved, including the use of the generalised sliding window protocol in practice, the use of individual and block acknowledgments, the use of W_s timers (possibly with individual acknowledgments), the advantage of the cross-layer design of the selective-repeat protocol, and the impact of timer expiration time on the efficiency of the generalised and go-back-N protocols. Any of these problems must first be solved theoretically and based on simulations which will require the use of powerful and flexible simulators [24,25].

Keywords: Telecommunications, Data Communications, Communication Protocols, Sliding Window Protocols

1 Uvod

Sodobni komunikacijski sistemi morajo zvečine zagotavljati zanesljiv prenos podatkov (pri tem s podatki mislimo na informacijo, ki nima časovne dimenzije, torej ni generirana sinhrono, in je zato ni potrebno prenašati v realnem času). Zanesljiv prenos pomeni, da se vsa sporočila prenesejo brez napak, izgub, dvojnikov, pa tudi v enakem vrstnem redu, kot so bila oddana. Zanesljiv prenos je mogoče zagotoviti edinole s protokoli s ponavljanjem, ki so v literaturi znani tudi kot protokoli ARQ (ang. Automatic Repeat reQuest).

Najpogosteje uporabljeni protokoli s ponavljanjem so protokoli z drsečim oknom (ang. sliding window protocols). Tovrstni protokoli se uporabljajo že nekaj desetletij [1,2], zato je bilo do danes definiranih in uporabljenih mnogo različic teh protokolov. Razvoj je vseskozi potekal v želji po čim večji učinkovitosti (torej po doseganju zadanih ciljev s čim manjšo porabo sistemskih virov, kot so npr. čas, hitrost prenosa ter poraba spomina in procesorske moči).

Čeprav se na prvi pogled zdi, da je bilo o protokolih z drsečim oknom povedano že skoraj vse in da se je raziskovanje tega področja upočasnilo ali celo ustavilo, pa po drugi strani tovrstne raziskave postajajo čedalje pomembnejše; učinkovitost protokolov ARQ namreč postaja kritična, saj se težišče problemov podatkovnih omrežij preveča od krmiljenja pretoka in zamašitev v ožičenih paketno komutiranih ali usmerjenih omrežjih proti zagotavljanju prenosa brez napak in izgub v sodobnih mobilnih omrežjih, ki deloma temeljijo na radijskem prenosu v fizičnem sloju [3,4]. V takih omrežjih se vse bolj uveljavljajo tudi hibridni protokoli ARQ (HARQ) [5,6], kar pa ne zmanjšuje pomena čim večje učinkovitosti osnovnih protokolov ARQ.

Osnovni namen tega članka je podati pregled trenutnega stanja protokolov z drsečim oknom in perspektive nadaljnjih raziskav na tem področju.

V nadaljevanju članka bomo sledili naslednjim smernicam. V drugem poglavju bomo podali pregled mehanizmov, ki jih protokoli z drsečim oknom uporabljajo za zagotavljanje zanesljivega prenosa. V tretjem poglavju bomo protokole razvrstili glede na merila, ki se nanašajo na mehanizme protokolov. V četrtem poglavju bomo spregovorili o uporabi protokolov z drsečim oknom v sedanji praksi. V petem poglavju pa bomo skušali nakazati smernice nadaljnjih raziskav na tem področju.

2 Mehanizmi protokolov z drsečim oknom

Mehanizme, ki jih protokoli z drsečim oknom uporabljajo za zanesljiv prenos informacije, lahko razdelimo na osnovne mehanizme, ki so nujno potrebni za zagotavljanje zanesljivega prenosa sporočil in za logično pravilnost protokola, ter pomožne mehanizme, ki sicer za logično pravilnost protokola niso potrebni, lahko pa pripomorejo k učinkovitejšemu prenosu sporočil [7,8].

2.1 Osnovni mehanizmi

Osnovni mehanizmi, ki jih mora uporabljati vsak protokol z drsečim oknom, da je lahko logično pravilen, so kanalsko kodiranje, uporaba pozitivnih potrditev, številčenje sporočil, uporaba časovnika in uporaba vmesnika za shranjevanje sporočil (spomina) na oddajni in sprejemni strani.

V nadaljnjem besedilu bomo tista protokolna sporočila, ki v sebi nosijo uporabniško informacijo, imenovali informacijska protokolna sporočila, ona, ki uporabniške informacije ne vsebujejo, pa nadzorna protokolna sporočila. Tipično nadzorno protokolno sporočilo je potrditev.

Čeprav so protokoli z drsečim oknom največkrat simetrični, kar pomeni, da lahko uporabniško informacijo prenašajo v obeh smereh, bomo v nadaljnjem besedilu zaradi lažjega razmišljanja imenovali tisti protokolni osebki, ki oddaja informacijska protokolna sporočila, oddajnik, onega, ki taka sporočila sprejema, pa sprejemnik.

Kanalsko kodiranje vseh oddanih sporočil omogoča sprejemniku odkrivanje ali popravljanje napak, ki nastanejo pri prenosu sporočila skozi kanal. Pri protokolih ARQ sprejemnik napake le odkriva, pokvarjena sporočila pa v praksi največkrat zavrže.

Sprejemnik mora za vsa informacijska protokolna sporočila, ki jih sprejme od oddajnika tako, da lahko njihovo uporabniško vsebino preda uporabniku, poslati oddajniku potrditev. To seveda pomeni, da za kakršen koli protokol s ponavljanjem potrebujemo izmenično enosmerni (poldupleksni) ali dvosmerni (dupleksni) kanal.

Potrditve se vedno nanašajo na določeno sporočilo oziroma na določeno zaporedje sporočil. Da so ta sporočila enoumno določena, morata oba protokolna osebka (oddajnik in sprejemnik) vsa sporočila številčiti. Ker morata oba protokolna osebka te številke hraniti v spominu, številke pa nosijo s seboj tudi vsa sporočila, mora biti množica vseh številke omejena; štetje sporočil torej poteka po nekem modulu. S tem modulom je seveda povezano število bitov, ki jih uporabimo za zapis številke. Zaporedno številčenje sporočil, ki vsebujejo uporabniško informacijo, omogoča tudi odkrivanje izgubljenih sporočil in urejanje sporočil v pravem vrstnem redu.

Katerokoli protokolno sporočilo (informacijsko ali nadzorno) se lahko pokvari ali izgubi (če se to ne bi

moglo zgoditi, tovrstnega protokola ne bi potrebovali). Zato oddajnik uporablja enega ali več časovnikov. Vsako že oddano, a še ne potrjeno informacijsko protokolno sporočilo mora biti zavarovano z aktivnim časovnikom; če se le-ta izteče, je treba sporočilo ponovno poslati.

Seveda mora imeti oddajnik sporočila, ki jih je že oddal, a še ni prejel njihove potrditve, shranjena v oddajnem vmesniku, saj pred sprejemom ustrezne potrditve ne more vedeti, ali bo katero od njih moral ponovno oddati. Število sporočil, ki jih oddajnik lahko hkrati shrani v oddajnem vmesniku, imenujemo širina oddajnega okna (v nadaljevanju članka jo bomo označevali z W_s), oddajno okno pa pomeni zaporedje sporočil, ki jih je oddajnik bodisi že oddal, pa še ni prejel njihove potrditve, ali pa jih v nekem trenutku še lahko odda. Spomin v obliki sprejemnega vmesnika uporablja tudi sprejemnik, da lahko hrani in po potrebi ureja sporočila, ki jih ni sprejel v pravem vrstnem redu. Število informacijskih protokolnih sporočil, ki jih sprejemnik lahko sprejme v svoj sprejemni vmesnik, imenujemo širina sprejemnega okna (v nadaljevanju W_r). Sprejemno okno pa pomeni zaporedje sporočil, ki jih v danem trenutku sprejemnik lahko sprejme v sprejemni vmesnik; nekatera od njih je morda že sprejel, a še niso urejena po vrsti.

Med potekom komunikacijskega procesa oddajno okno s sprejemanjem novih potrditev drsi po številski premici. Sprejemno okno pa drsi po številski premici s sprejemanjem informacijskih protokolnih sporočil, ki sprejemniku omogočajo predajo uporabniških sporočil uporabniku v pravem vrstnem redu. Drsenje obeh oken daje ime družini protokolov – protokoli z drsečim oknom.

Vsota širin oddajnega in sprejemnega okna omejuje število sporočil, ki so lahko v najneugodnejšem primeru hkrati prisotna v celotnem komunikacijskem sistemu. Po eni strani to dejstvo zahteva, da je modul štetja sporočil enak ali večji od vsote širin obeh oken [9]

$$M \geq W_s + W_r, \quad (1)$$

kar zagotavlja, da je katerokoli sporočilo v sistemu kadarkoli enoumno določljivo. Po drugi strani pa tovrstno omejevanje števila sporočil daje določene možnosti za krmiljenje pretoka in zamašitev, kar pa ni več tema tega članka.

2.2 Pomožni mehanizmi

Protokol z drsečim oknom lahko poleg osnovnih mehanizmov uporablja tudi nekatere dodatne mehanizme, ki sicer niso nujno potrebni za zagotavljanje zanesljivega prenosa, lahko pa izboljšajo učinkovitost protokola. Najpogosteje uporabljana tovrstna mehanizma sta negativna potrditev in poizvedba.

Z negativno potrditvijo lahko sprejemnik oddajniku izrecno sporoči, da se je neko sporočilo pokvarilo ali izgubilo; taka negativna potrditev namreč lahko pride

do oddajnika precej prej, kot pa bi se mu iztekel časovnik.

Oddajnik, ki se mu izteče časovnik, seveda ne more natanko vedeti, kaj se je zgodilo: se je izgubilo sporočilo, ki ga je sam poslal, ali pa se je izgubila potrditev. Zato lahko sprejemniku pošlje poizvedbeno sporočilo; odgovor sprejemnika mu omogoči, da se na iztek časovnika pravilno odzove in ne izgublja časa z delovanjem, ki ne bi prineslo nobene koristi.

3 Razvrstitev protokolov z drsečim oknom

Preden se lotimo sistematičnega razvrščanja protokolov z drsečim oknom, si moramo postaviti merila, glede na katera bomo protokole razvrščali. Uporabili bomo naslednja merila:

- širina oddajnega in sprejemnega okna,
- vrsta potrditev,
- število časovnikov in
- obseg funkcionalnosti protokola v protokolnem skladu.

Seveda pa ne smemo pričakovati, da bo kakršna koli kombinacija protokolnih mehanizmov glede na to razvrstitev nujno tudi smiselna.

3.1 Razvrstitev glede na širino oddajnega in sprejemnega okna

Načeloma je mogoča in lahko logično pravilna katera koli kombinacija širin oddajnega in sprejemnega okna. Vendar pa je bilo v [9] pokazano, da primer $W_r > W_s$ ni smiseln, saj bi bil v tem primeru del sprejemnega okna ves čas neizkoriščen. Protokol s kakršno koli kombinacijo širin oken imenujemo posplošeni protokol z drsečim oknom (ang. Generalised Sliding Window Protocol), prvič pa je bil predlagan v [10] in prvič formalno specificiran v [9]. V teh člankih je bilo tudi pokazano, da so različice, ki jih bomo tu navedli, le posebni primeri posplošenega protokola.

- Protokol s čakanjem (ang. Stop-and-Wait Protocol) je protokol, za katerega velja $W_s = W_r = 1$. Širina oddajnega okna 1 pomeni, da lahko oddajnik odda le eno sporočilo, čaka na potrditev in šele nato odda naslednje sporočilo. Protokol je zelo preprost, vendar neučinkovit, saj oddajnik izgublja čas s čakanjem na potrditev. V skladu z enačbo (1) lahko sporočila štejemo po modulu 2, zato tak protokol včasih imenujemo tudi protokol z alternirajočim bitom (ang. Alternating Bit Protocol).
- Protokol s ponavljanjem zaporedja (ang. Go-Back-N Protocol) [11] je protokol, za katerega velja $W_s > W_r = 1$. Ker je širina oddajnega okna večja od 1, lahko oddajnik odda več sporočil zapovrstjo, ko pa dobiva od sprejemnika potrditve, pomika oddajno okno naprej. Sprejemnik zaradi širine okna 1 sporočil ne more urejati po vrsti, zato vsa sporočila, ki ne prispejo v pravem vrstnem redu (npr. ker se je eno sporočilo vmes izgubilo), zavrne; če torej oddajnik ugotovi, da se je neko sporočilo izgubilo,

ponovno pošlje to in tudi vsa naslednja že poslana sporočila (od tod ime protokola).

- Protokol s selektivnim ponavljanjem (ang. Selective-Repeat Protocol) je protokol, za katerega velja $W_s = W_r > 1$. Če izgub ni, ta protokol deluje popolnoma enako kot protokol s ponavljanjem zaporedja. Ker pa sprejemnik v tem primeru razpolaga z enako dolgim vmesnikom kot oddajnik, lahko vanj shrani tudi sporočila, ki ne prispejo v pravem vrstnem redu, oddajnik pa, ko ugotovi, da se je neko sporočilo izgubilo, ponovno pošlje le tistega.
- Posplošeni protokol se pri izgubah odzove na iztek časovnika s strategijo, ki je kombinacija strategij protokola s ponavljanjem zaporedja in protokola s selektivnim ponavljanjem. Zato formalna specifikacija posplošenega protokola velja tudi za vse tri posebne primere.

Če v kanalu ne bi imeli izgub (v tem primeru protokola ARQ seveda sploh ne bi potrebovali), bi bila učinkovitost protokola z drsečim oknom enaka

$$\eta = \frac{W_s \cdot t_i}{t_r} \quad (2)$$

kjer je t_i čas, potreben za oddajo informacijskega protokolnega sporočila, t_r pa čas do potrditve (ang. round trip time), ki mine od oddaje sporočila do sprejema potrditve (ta formula seveda velja le, če da rezultat, ki je manjši od ena, sicer je učinkovitost ena). Iz tega je jasno, da lahko protokol z drsečim oknom doseže pri brezizgubnem kanalu učinkovitost 1, če je oddajno okno dovolj veliko ($W_s > t_r/t_i$). V realnih razmerah je seveda učinkovitost vedno manjša od tiste, ki jo podaja enačba (2), in nikoli ne doseže vrednosti 1. V večini primerov je učinkovitost protokola s selektivnim ponavljanjem večja od učinkovitosti protokola s ponavljanjem zaporedja, učinkovitost protokola s čakanjem pa je še precej manjša.

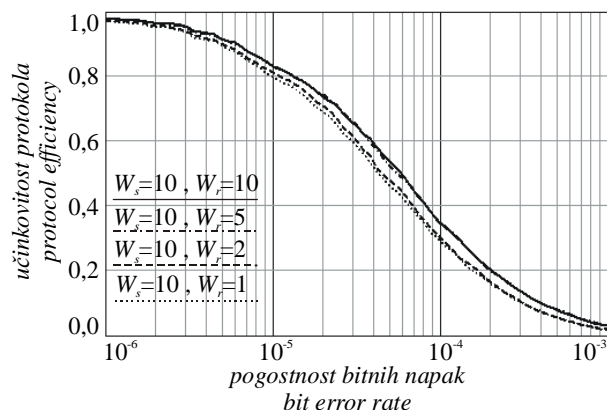
Učinkovitost posplošenega protokola lahko doseže enako vrednost kot učinkovitost protokola s selektivnim ponavljanjem, vendar ob manjši porabi spomina na sprejemni strani. Na sliki 1 vidimo značilen primer odvisnosti učinkovitosti posplošenega protokola od pogostnosti bitnih napak pri hitrosti prenosa 1 Mb/s, dolžini uporabniških sporočil 200 oktetov, dolžini režije informacijskih protokolnih sporočil in potrditev 5 oktetov, zakasnitvi v kanalu 5 ms, širini oddajnega okna $W_s = 10$ ter širinah sprejemnega okna $W_r = 1, 2, 5$ oziroma 10. V grafu vidimo, da je učinkovitost pri protokolu s ponavljanjem zaporedja ($W_s = 1$) manjša kot pri protokolu s selektivnim ponavljanjem ($W_s = 10$), učinkovitost pri $W_s = 5$ pa je skoraj enaka kot pri $W_s = 10$.

3.2 Razvrstitev glede na uporabljene potrditve

Protokoli z drsečim oknom lahko uporabljajo individualne, kumulativne in intervalne potrditve [8]. Individualna potrditev potrjuje sprejem nekega določenega sporočila. Kumulativna potrditev potrjuje

sprejem nekega sporočila in vseh, ki so bila poslana pred njim. Intervalna potrditev pa potrjuje sprejem zaporedja sporočil v določenem intervalu, od enega do drugega.

Protokoli, ki uporabljajo različne vrste potrditev, se seveda lahko različno obnašajo, porabljajo različne množine sistemskih virov in imajo različne učinkovitosti.



Slika 1. Učinkovitost posplošenega protokola
Figure 1. Generalised Protocol Efficiency

3.3 Razvrstitev glede na število uporabljenih časovnikov

Oddajnik lahko uporabi za vsako oddano, a še ne potrjeno sporočilo poseben časovnik; v tem primeru mora seveda razpolagati s toliko časovniki, kolikor je širina oddajnega okna. Lahko pa tudi uporablja en sam časovnik, ki mora biti aktiven, kadarkoli ima oddajnik v svojem vmesniku vsaj eno že oddano, a še ne potrjeno sporočilo. V tem primeru oddajnik sproži časovnik (če ta ni že aktiven), ko odda sporočilo. Časovnik ustavi, ko sprejme neko novo potrditev, a ga takoj spet požene, če ima v oddajnem vmesniku še kakšno sporočilo, ki čaka na potrditev. Če se časovnik izteče, oddajnik predpostavi, da se je izgubilo tisto sporočilo, ki že najdlje čaka na potrditev.

3.4 Razvrstitev glede na obseg funkcionalnosti v protokolnem skladu

V skladu z načeli razslojevanja komunikacijskega sistema v smislu protokolnega sklada naj bo protokol v vsakem sloju čim bolj neodvisen od protokolov v ostalih slojih [12]; osebke v sosednjih slojih povezujejo točke dostopa do storitve (ang. Service Access Point – SAP), prek katerih ti osebki sodelujejo. Pogosto je v teh točkah treba uporabiti tudi čakalne vrste. Medsebojna neodvisnost sosednjih slojev pripomore k lažjemu načrtovanju komunikacijskega sistema po načelu »deli in vladaj«, vendar pa lahko zahteva ta neodvisnost tudi svojo ceno v smislu manjše učinkovitosti ali celo možnosti zamašitev med slojema [13]. Lahko pa dva

sosednja sloja tudi združimo v enega samega, ki pa je zaradi tega seveda bolj kompleksen, a lahko učinkoviteje opravlja svojo nalogo. Takšno združevanje funkcionalnosti je mogoče predvsem, kadar želimo protokol z drsečim oknom uporabiti v povezavnem (ang. data-link) sloju po referenčnem modelu OSI. Govorimo lahko torej o troslojnem modelu komunikacijskega sistema (uporabnika protokola z drsečim oknom – protokol z drsečim oknom v povezavnem sloju – prenos v fizičnem sloju) ali o dvoslojnem modelu komunikacijskega sistema (uporabnika protokola z drsečim oknom – protokol z drsečim oknom v povezavnem sloju, združen s prenosom v fizičnem sloju).

4 Običajna uporaba protokolov z drsečim oknom v praksi

Protokoli z drsečim oknom se največ uporabljajo v povezavnem (ang. data-link) in transportnem (ang. transport) sloju protokolnega sklada po referenčnem modelu OSI, redko tudi više.

Zaradi svoje preprostosti pa tudi zaradi zgodovinskih razlogov se v povezavnem sloju pogosto uporablja protokol s ponavljanjem zaporedja. Z zgodovinskimi razlogi mislimo na evolucijo protokolov v zaporedju HDLC (ISO protokol) [14], LAPB (za dostop do paketno komutiranih omrežij X.25) [15], LAPD (za signalizacijo v lokalni zanki ISDN) [16], LAPDm (za signalizacijo prek radijskega vmesnika GSM) [17], ki so vsi protokoli povezavnega sloja. Čeprav je HDLC lahko deloval kot protokol s ponavljanjem zaporedja ali kot protokol s selektivnim ponavljanjem, so vsi njegovi omenjeni nasledniki protokoli s ponavljanjem zaporedja.

Protokoli z drsečim oknom, ki se (lahko) uporabljajo v transportnem sloju nad nepovezavno organiziranim omrežnim slojem, so protokoli s selektivnim ponavljanjem, saj ti protokoli niso občutljivi na morebitno spreminjanje vrstnega reda prenesenih sporočil v nepovezavnem omrežju; primer takega protokola je TCP [18].

V dejansko uporabljenih protokolih se večinoma uporabljajo kumulativne potrditve in en sam časovnik, spet najbrž predvsem zaradi svoje preprostosti, pa tudi iz zgodovinskih razlogov: številni protokoli nastanejo s prirejanjem že obstoječih in uporabljenih protokolov.

Protokoli transportnega in višjih slojev so ponavadi specifikirani, pa tudi implementirani samostojno, ne glede na protokole nižjega sloja. Protokoli povezavnega sloja so pogosto izvedeni strojno, torej skupaj s protokoli fizičnega sloja, vendar to dejstvo pri specifikaciji protokolov ni vedno upoštevano in izrabljeno v prid učinkovitosti protokola.

5 Novejši izsledki in izzivi za nadaljnje raziskave

V novejši strokovni literaturi so v zvezi s protokoli z drsečim oknom najpogostejši članki, ki obravnavajo formalno specifikacijo in preverjanje logične pravilnosti protokolov [19], pa tudi modeliranje in analizo učinkovitosti protokolov [20]. Nedolgo tega je bil predlagan in specifikiran posplošeni protokol z drsečim oknom z enim časovnikom in kumulativnimi potrditvami [9], ki predstavlja teoretično osnovo za razvoj novih protokolov. Članek [21] obravnava strategijo odzivanja oddajnega protokolnega osebka na iztek časovnika. Da je lahko učinkovitost protokola močno odvisna tudi od časa izteka časovnika, preberemo v [22] in [23]. Vpliv medsebojne (ne)odvisnosti sosednjih slojev protokolnega sklada na učinkovitost protokola z drsečim oknom pa je obravnavan v [13] in [23].

Nekaj vprašanj, povezanih s protokoli z drsečim oknom, pa je ostalo še (vsaj delno) nerazrešenih. Ta vprašanja so torej danes izziv za nadaljnje raziskovalno delo. Pri tem se moramo zavedati, da je treba tovrstne probleme najprej rešiti na ravni osnovnih protokolov, rešitve pa preveriti z obsežnimi simulacijami. V ta namen je seveda potrebno razpolagati z zmogljivim in prilagodljivim simulatorjem [24,25]. V nadaljevanju bomo nakazali nekatere odprte probleme, ki smo jih vsaj deloma obravnavali že v prejšnjih raziskavah, predvsem pa ostajajo predmet prihodnjih raziskav.

Posplošeni protokol z drsečim oknom je bil doslej teoretično obdelan, formalno specifikiran in preverjen s simulacijami, ni pa še bil uporabljen v praksi.

Omenili smo že, da se v praksi uporabljajo zvečine protokoli s kumulativnimi potrditvami. Ve se tudi, kaj so in kako se uporabljajo individualne in intervalne potrditve; potrebno pa bo še preveriti in s simulacijami potrditi, kako uporaba tovrstnih potrditev vpliva na učinkovitost protokolov.

Odperto ostaja tudi vprašanje protokolov z drsečim oknom, kjer oddajnik uporablja več časovnikov (toliko kot je širina oddajnega okna). Ob hkratni uporabi individualnih potrditev in pri velikih pogostnostih napak bi taki protokoli utegnili biti učinkovitejši.

Ugotovljeno je že bilo, da protokol s ponavljanjem zaporedja ni primeren, če so sloji protokolnega sklada med seboj neodvisni [13]. Preveriti pa bi bilo še potrebno, kako (če sploh) medsebojna neodvisnost slojev vpliva na učinkovitost protokolov s selektivnim ponavljanjem.

Opravljen je že bila raziskava vpliva časa izteka časovnika na učinkovitost protokola s selektivnim ponavljanjem [23]. Nekaj podobnega bi bilo treba storiti še za primer protokola s ponavljanjem zaporedja in za posplošeni protokol.

6 Sklep

V članku smo prikazali pregled komunikacijskih mehanizmov protokolov z drsečim oknom. Glede na uporabo teh mehanizmov smo protokole razvrstili v različne kategorije in na osnovi take razvrstitve opisali danes ustaljeno uporabo teh protokolov. Ob koncu članka smo pregledali zanimive teme raziskav na tem področju v zadnjem času, nato pa smo navedli nekaj še neobdelanih tem, ki bi utegnile prinesiti spodbudne rezultate pri bodočih raziskavah tega področja. Članek torej lahko zaključimo z ugotovitvijo, da nam bodo nadaljnje raziskave protokolov z drsečim oknom morda prinesle še nove, učinkovitejše protokole.

7 Zahvala

Raziskavo je delno finančno podprlo Ministrstvo za visoko šolstvo, znanost in tehnologijo Republike Slovenije (programska skupina P2-0246).

8 Literatura

- [1] U. D. Black, 'Data link controls: the great variety calls for wise and careful choices [and data transmission equipment buyer's guide]', *Data Communications*, 112-29, mid-Nov. 1982.
- [2] A. U. Shankar, S. S. Lam, 'Construction of sliding window protocols', *TR-86-09*, Univ. Texas at Austin, TX, USA, March 1986.
- [3] D. Hercog, S. Bešlagić, 'Zasnova modelov za primerjavo učinkovitosti prenosa podatkov v transportnem sloju kombiniranih fiksno-mobilnih omrežij', B. Zajc (ur.), A. Trost (ur.). Zbornik petnajste mednarodne Elektrotehniške in računalniške konference ERK 2006, zv. A, str. 179-182, Portorož, Slovenija, 2006.
- [4] M. Taferner, *Wireless Internet Access over GSM and UMTS*, Berlin, Springer, 2002.
- [5] K. Dongmyoung et al., 'A MAC/PHY Cross-Layer Design for Efficient ARQ Protocols', *IEEE Communications Letters*, Vol. 12, No. 12, pp. 909-11, dec. 2008.
- [6] I. Stanojev et al., 'Energy Efficiency of Non-Collaborative and Collaborative Hybrid-ARQ Protocols', *IEEE Transactions on Wireless Communications*, Vol. 8, No. 1, pp. 326-35, Jan. 2009.
- [7] R. Sharp, *Principles of Protocol Design*, Berlin, Springer-Verlag, 2008.
- [8] M. G. Gauda, *Elements of Network Protocol Design*, New York, John Wiley & Sons, 1998.
- [9] D. Hercog, 'Generalization of the basic sliding window protocol', *International Journal of Communication Systems*, vol. 18, no. 1, pp. 57-75, 2005.
- [10] D. Hercog, 'Generalised sliding window protocol', *Electronics Letters*, vol. 38, no. 18, pp. 1067-1068, 2002
- [11] T. Vidmar, *Informacijsko-komunikacijski sistem*, Ljubljana, Pasadena, 2002.
- [12] F. Halsall, *Data Communications, Computer Networks and Open Systems*, 4th Ed., Harlow, Addison-Wesley, 1996.
- [13] D. Hercog, '(Ne)primernost protokola s ponavljanjem N sporočil v protokolnem skladu z med seboj neodvisnimi sloji', B. Zajc (ur.), A. Trost (ur.), *Zbornik šestnajste mednarodne Elektrotehniške in računalniške konference ERK 2007*, zv. A, str. 84-87, Portorož, Slovenija, 2007.
- [14] ISO, 'Information technology -- Telecommunications and information exchange between systems -- High-level data link control (HDLC) procedures', *ISO/IEC standard 13239:2002*.
- [15] ITU-T, *Recommendation X.25*, Geneva, International Telecommunication Union, 1993 .
- [16] ITU-T, *Recommendation Q.921*, Geneva, International Telecommunication Union, 1993.
- [17] ETSI, *EN 300 938 V8.0.1 (GSM 04.06)*, Sophia-Antipolis, European Telecommunications Standards Institute, 2000.
- [18] J. Postel, 'Transmission Control Protocol, DARPA Internet Program, Protocol Specification', *RFC 793*, IETF, 1981.
- [19] E. Madelaine, D. Vergamini, 'Specification and verification of sliding window protocol in LOTOS', *IFIP Transactions C (Communication Systems)*, v C-2, pp. 495-510, 1992.
- [20] T. Ikegawa, Y. Takahashi, 'Sliding window protocol with selective-repeat ARQ: performance modeling and analysis', *Telecommunication Systems*, vol. 34, No. 3-4, pp. 167-80, April 2007.
- [21] D. Hercog, 'Timeout recovery strategies in datalink layer protocols at high bit error rates', *Electronics Letters*, Vol. 32, No. 20, pp. 1864-65, 1996.
- [22] D. Hercog, 'Čas izteka časovnika pri protokolih s selektivnim ponavljanjem', B. Zajc (ur.), A. Trost (ur.), *Zbornik sedemnajste mednarodne Elektrotehniške in računalniške konference ERK 2008*, zv. A, str. 73-76, Portorož, Slovenija, 2008.
- [23] D. Hercog, 'Timer Expiration Time of Selective Repeat Protocol', sprejeto za objavo v *Electronics Letters*, 2009.
- [24] D. Hercog, 'Simulacija zmogljivosti telekomunikacijskih protokolov, specificiranih v jeziku SDL', *Elektrotehniški vestnik*, letn. 68, št. 1, str. 13-19, 2001.
- [25] A. S. McGough, I. Mitrani, 'Efficient parallel simulation of a sliding window protocol', *Performance Evaluation*, vol. 48, pp. 237-246, 2002.

Drago Hercog je diplomiral leta 1974, magistriral leta 1984 in doktoriral leta 1989, vse na Fakulteti za elektrotehniko Univerze v Ljubljani. Zaposlen je v isti ustanovi kot izredni profesor, uči pa tudi na Visoki šoli za inženirje elektrotehnike ESIGELEC v Rouenu v Franciji. Predava več različnih predmetov s področja telekomunikacij na dodiplomskem in podiplomskem študiju. Raziskovalno se ukvarja s telekomunikacijskimi protokoli, simulacijo diskretnih sistemov in z metodami poučevanja bodočih inženirjev. Področje, na katero se je še posebej osredotočil, pa so protokoli z drsečim oknom.