



Universitatea Tehnică „Gheorghe Asachi” din Iași
Facultatea de Automatică și Calculatoare
Specializarea Calculatoare și Tehnologia Informației
Rețele de calculatoare – proiect

Transfer de fișiere – implementare printr-un protocol cu fereastră glisantă

Coordonator: prof.dr.ing. Nicolae-Alexandru Botezatu

Studenti: Duminică Ionel-Sergiu, grupa 1306A

Blanaru Ioana-Bianca, grupa 1306A

Cuprins

- 1 Introducere
- 2 Protocolul cu fereastră glisantă
- 3 Metoda întoarcerii cu n înapoi (*Go Back N*)
- 4 Componente (clase)
 - 4.1 Emițător
 - 4.2 Receptor
 - 4.3 Timer
 - 4.4 Handler de pachete
 - 4.5 UDP
- 5 Git repository
- 6 Bibliografie

1 Introducere

În informatică și telecomunicație, un protocol de comunicații este un set de reguli și norme care permite ca două sau mai multe entități dintr-un sistem de comunicații să comunice între ele prin transmiterea de informație printr-un mediu de orice tip.

Protocoalele de comunicații sunt clasificate după nivelul la care operează, cele mai cunoscute ierarhii de protocoale fiind modelul TCP/IP și modelul OSI. Pentru ca o comunicație să transmită unul sau mai multe pachete, este necesară folosirea de protocoale de la toate nivelele uneia dintre ierarhii.

TCP (*Transmission Control Protocol*) este un protocol standard orientat spre conexiune ce asigură considerabil de multe facilități cum ar fi recuperarea erorilor, controlul transmisiei și siguranța acestora etc.

2 Protocolul cu fereastră glisantă

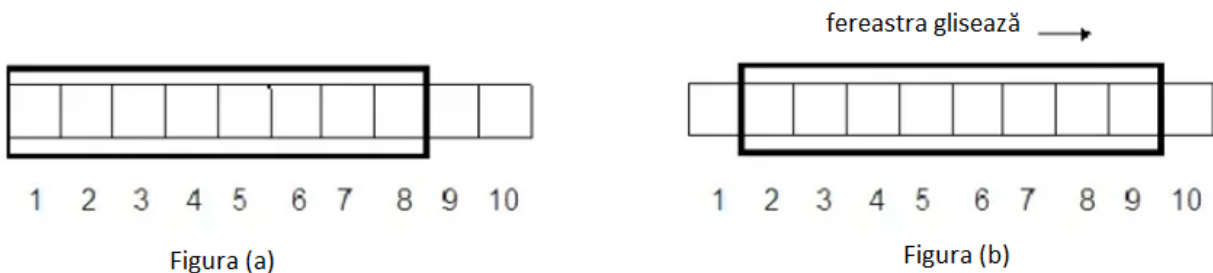
Protocolul cu fereastră glisantă este folosit acolo unde este necesară livrarea fiabilă în ordine a pachetelor, reprezentând astfel modelul TCP. Acest protocol face ca transmisia de flux să fie eficientă.

Scopul aplicației noastre este de a transfera în mod fiabil pachete prin comunicație UDP (*User Datagram Protocol*) folosind algoritmul *Go Back N*.

Pentru a realiza siguranța, emițătorul transmite un pachet și apoi așteaptă o confirmare înainte de a trimite altul. Datele călătoresc între mașini doar într-o singură direcție la un moment de timp chiar dacă rețeaua este capabilă să susțină o comunicație

simultană în ambele direcții. Rețeaua poate fi complet nefolosită în timpul în care mașina așteaptă răspunsul.

Pentru început, emițătorul asignează câte un număr de secvență pentru fiecare pachet ce dorește a fi trimis. Acesta poate transmite pachetele din fereastră fără să recepționeze o confirmare (*ACK*, în engleză: *acknowledgement*), dar pornește câte un cronometru (*Timer*) care va semnaliza depășirea unui interval de timp prestabilit pentru fiecare dintre pachete. Receptorul trebuie să confirme fiecare pachet primit, indicând numărul de secvență al ultimului pachet recepționat corect, urmând ca emițătorul să deplaseze fereastra (să o gliseze) la fiecare mesaj de confirmare.



Un protocol cu fereastră glisantă întotdeauna știe care pachete au fost confirmate și păstrează un timer separat pentru fiecare pachet neconfirmat. Dacă un pachet este pierdut, timerul expiră și emițătorul retransmite acel pachet.

3 Metoda întoarcerii cu n înapoi (*Go Back N*)

Go Back N este o instanță specifică a protocolului ARQ (*Automatic Repeat Protocol*), în care procesul de trimitere continuă să trimită un număr de pachete specificat de dimensiunea unei ferestre chiar și fără a primi o confirmare (*ACK*) de la receptor. Este

un caz special al protocolului general cu fereastră glisantă, cu dimensiunea ferestrei de transmisie de N și dimensiunea ferestrei de recepție de 1. Poate transmite N pachete către receptor înainte de a primi confirmarea.

Dacă un pachet este pierdut în tranzit sau sosește, dar nu sosește în forma sa corectă, toate pachetele următoare sunt șterse până când pachetul lipsă este retransmis, ceea ce implică o întârziere minimă de timp dus-întors și un timeout. În consecință, acest algoritm nu este eficient pe conexiunile care suferă pierderi frecvente de pachete.

4 Componente (clase)

4.1 Emițător

Clasa emițătorului trebuie să rețină variabile precum dimensiunea ferestrei, numărul ultimei secvențe ce a fost trimisă, numărul ultimei confirmări și numărul secvenței ce urmează a fi trimisă. Când ajunge o confirmare, emițătorul mută fereastra spre dreapta transmițând un nou pachet și asociază un nou *timer* pentru acel pachet.

4.2 Receptor

Clasa receptorului trebuie să rețină variabile precum dimensiunea ferestrei care este egală cu 1, numărul de secvență al celui mai mare cadru acceptabil și numărul de secvență al ultimului pachet primit.

4.3 Timer

Această clasă este folosită și de emițător și de receptor pentru a ține evidența timpului de expirare a ferestrei și timpul maxim dintre primirea a două pachete consecutive.

4.4 Handler de pachete

În această clasă vom găsi structura pachetelor, conversia pachetelor din date în bytes (și invers) și verificările aferente pachetului (verificăm dacă este complet).

4.5 UDP

Clasa UDP va retrace trimiterea și primirea pachetelor în funcție de verificarea pe care handler-ul de pachete o va returna.

5 Git repository

<https://github.com/SergiuDuminica/Duminica-Blanaru-RC-P>

6 Bibliografie

- www.cnic.ro/telecom/transm_ctrl_prot_tcp.htm
- www.scrigroup.com/calculatoare/c/Protocoale-cu-fereastra-glisan51993.php
- en.wikipedia.org/wiki/Go-Back-N_ARQ
- www.baeldung.com/cs/networking-go-back-n-protocol