혼잡제어 구현

(컴퓨터 네트워크)

제출일: 2020. 12. 21

담당 교수: 정의훈 교수님

학번: 2016154004

이름: 권오제

**목 차**

1. 개요
2. 요구사항
3. 서버와 클라이언트 동작 방식
4. 주요 코드
5. 실행 화면

**1. 개요**

네트워크의 혼잡 상태가 감지되었을 때, 최대한 회피하기 위해서 송신 측의 윈도우 크기를 조절하여 데이터의 전송량을 줄이는 것을 혼잡 제어라고 한다.

이런 혼잡 제어를 서버와 클라이언트 간의 UDP통신을 통해서 구현한 프로젝트이다.

**2. 요구사항**

**요구사항 1. 데모 프로그램의 구조**

서버(수신 측)와 클라이언트(송신 측) 프로그램으로 구성하여 서로 혼잡 제어 알고리즘에 의해 전송하는 데이터를 제어한다.

**요구사항 2. Slow start 적용**

서버와 클라이언트의 연결이 시작되면 윈도우 크기인 CongWin 값을 1부터 시작하여 2배씩 증가한다.

**요구사항 3. 혼잡 사건 정의**

Timeout과 3개의 중복 ACK에 대한 사건들을 정의한다

**요구사항 4. 혼잡 윈도우(CongWin) 크기 조절**

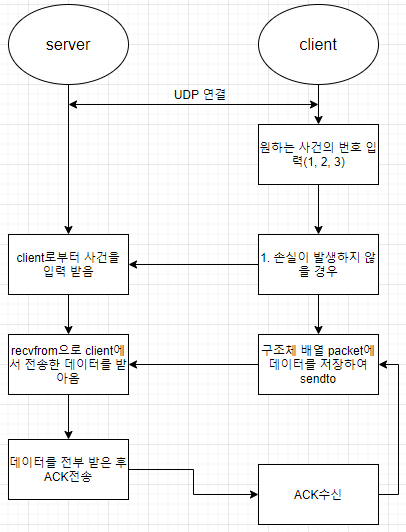
혼잡 상태가 발생했을 때, 각각의 사건에 맞게 윈도우의 크기를 설정한다.

- Timeout: 현재의 CongWin값의 1/2을 임계치(ThreshOld)로 설정하고, CongWing은 1로 설정한다.

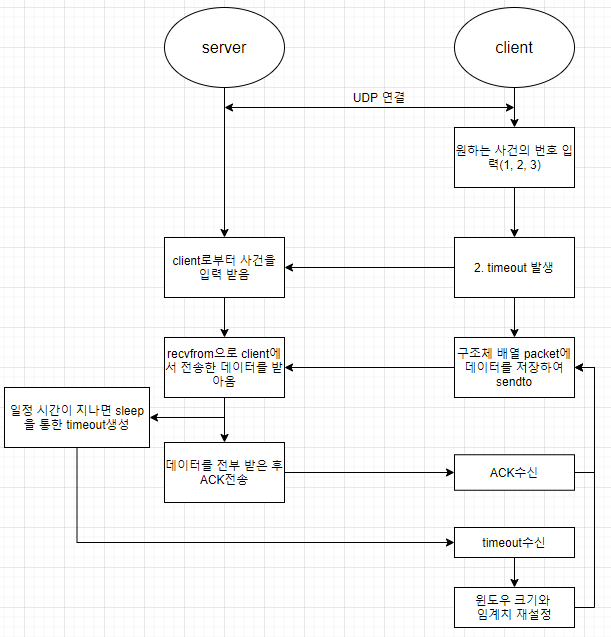
- 3dup ACK: 현재의 CongWing값을 1/2로 설정하고, 이 값을 임계치(ThreshOld)로 설정한다.

**3. 서버와 클라이언트 동작 방식**

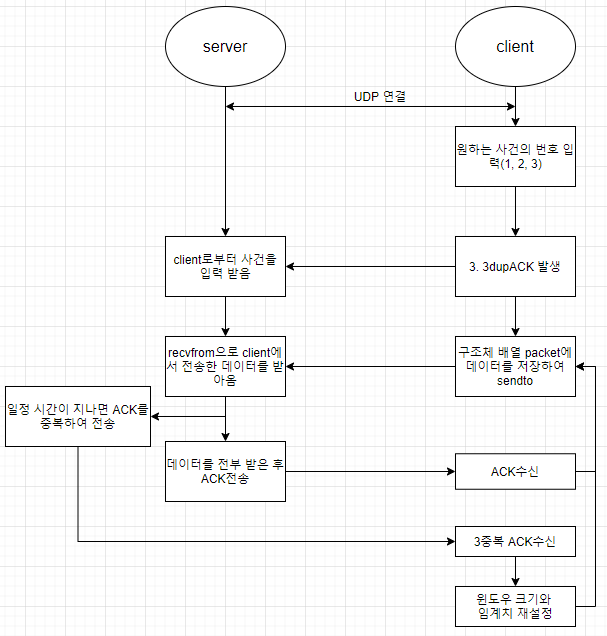
**(1) 손실이 발생하지 않을 경우**



**(2) Timeout이 발생한 경우**



**(3) 3dupACK가 발생한 경우**



**4. 주요 코드**

|  |
| --- |
|  |
|  |
| 위의 사진은 Client 측에서 server로 메시지를 보내는 함수이다. for문을 사용하여 CongWin크기만큼 데이터를 전송하게 하였다.  아래의 server 측 또한 CongWin크기만큼 데이터를 받는다. |
|  |
| 손실이 발생하였을 때, 상태(state)를 확인하여 그에 맞는 윈도우 크기와 임계치를 설정해준다. |
|  |
| CongWin값은 ThreshOld에 도달하기 전에는 2배씩 증가하다가, 임계치에 도달하게 되면 1씩 증가한다. |

**5. 실행 화면**

|  |
| --- |
| **(1) 손실이 발생하지 않을 경우** |
|  |
| Client 측에서 1번을 선택한다. |
|  |
|  |
| 임의로 ThreshOld값을 10으로 설정하여 처음에는 CongWin값이 1부터 시작하여 2배씩 증가하다가 ThreshOld에 도달해서 1씩 증가함을 볼 수 있다. |
| **(2) timeout이 발생한 경우** |
|  |
| Client 측에서 2번을 선택한다 |
|  |
| 1번과 동일하게 동작하다가 sleep이 걸려서 timeout이 발생한 사진이다. timeout발생 이후에는 ThreshOld값을 CongWin의 1/2로 줄이고, CongWin값을 1로 설정한 모습이다. |
| **(3) 3dupACK가 발생한 경우** |
|  |
| Client 측에서 3번을 선택한다. |
|  |
| 1번과 동일하게 동작하다가 client 측에서 중복된 ACK를 3번 수신 받으면, CongWin값을 1/2로 줄이고, ThreshOld의 값도 동일하게 설정해준다. |