Communication Networks Spring 2021





Rüdiger Birkner

https://comm-net.ethz.ch/

ETH Zürich March 25 2021

Register your group until Friday, April 9 2021

the groups have to consist of three students check #group_search for a group or a member

please register using your nethz usernames invalid registrations will be removed

comm-net.ethz.ch/registration/

registration opens at 11am

Communication Networks Exercise 4

Overview current assignment

Time for you to solve the tasks

Solutions will be published next week



Task 1: Max-Min Fair Allocation



Slides: Week 4, 68–81

Max-min fair allocation can easily be computed

step 1	Sta
step 2	Inc a n
step 3	Hol tha
step 4	Go

art with all flows at rate 0

crease the flows until there is new bottleneck in the network

Id the fixed rate of the flows at are bottlenecked

to step 2 for the remaining flows

Done!

E-F			
C-D			
C-B			
A-B-D			
B-D			
B-E			
C-D-E			
B-D-E			
D-E-F			



E-F			
C-D			
C-B			
A-B-D			
B-D			
B-E			
C-D-E			
B-D-E			
D-E-F			



E-F			
C-D			
C-B			
A-B-D			
B-D			
B-E			
C-D-E			
B-D-E			
D-E-F			



E-F	1		
C-D	1		
C-B	1		
A-B-D	1		
B-D	1		
B-E	1		
C-D-E	1		
B-D-E	1		
D-E-F	1		



E-F	1		
C-D	1		
C-B	1		
A-B-D	1		
B-D	1		
B-E	1		
C-D-E	1		
B-D-E	1		
D-E-F	1		



E-F	1		
C-D	1		
C-B	1		
A-B-D	1		
B-D	1		
B-E	1		
C-D-E	1		
B-D-E	1		
D-E-F	1		



E-F	1		
C-D	1		
C-B	1		
A-B-D	1		
B-D	1		
B-E	1		
C-D-E	1		
B-D-E	1		
D-E-F	1		



E-F	1	3		
C-D	1			
C-B	1	3		
A-B-D	1	3		
B-D	1	3		
B-E	1	3		
C-D-E	1			
B-D-E	1	3		
D-E-F	1	3		



E-F	1	3		
C-D	1			
C-B	1	3		
A-B-D	1	3		
B-D	1	3		
B-E	1	3		
C-D-E	1			
B-D-E	1	3		
D-E-F	1	3		



E-F	1	3		
C-D	1			
C-B	1	3		
A-B-D	1	3		
B-D	1	3		
B-E	1	3		
C-D-E	1			
B-D-E	1	3		
D-E-F	1	3	 	



E-F	1	3		
C-D	1			
C-B	1	3		
A-B-D	1	3		
B-D	1	3		
B-E	1	3		
C-D-E	1			
B-D-E	1	3		
D-E-F	1	3	 	



E-F	1	3		
C-D	1			
C-B	1	3		
A-B-D	1	3		
B-D	1	3		
B-E	1	3		
C-D-E	1			
B-D-E	1	3		
D-E-F	1	3		



E-F	1	3	7	
C-D	1			
C-B	1	3		
A-B-D	1	3	7	
B-D	1	3	7	
B-E	1	3	7	
C-D-E	1			
B-D-E	1	3	7	
D-E-F	1	3	7	



E-F	1	3	7	
C-D	1			
C-B	1	3		
A-B-D	1	3	7	
B-D	1	3	7	
B-E	1	3	7	
C-D-E	1			
B-D-E	1	3	7	
D-E-F	1	3	7	



E-F	1	3	7	
C-D	1			
C-B	1	3		
A-B-D	1	3	7	
B-D	1	3	7	
B-E	1	3	7	
C-D-E	1			
B-D-E	1	3	7	
D-E-F	1	3	7	



E-F	1	3	7	
C-D	1			
C-B	1	3		
A-B-D	1	3	7	
B-D	1	3	7	
B-E	1	3	7	
C-D-E	1			
B-D-E	1	3	7	
D-E-F	1	3	7	



E-F	1	3	7	11	
C-D	1				
C-B	1	3			
A-B-D	1	3	7		
B-D	1	3	7		
B-E	1	3	7	11	
C-D-E	1				
B-D-E	1	3	7		
D-E-F	1	3	7	11	



E-F	1	3	7	11	
C-D	1				
C-B	1	3			
A-B-D	1	3	7		
B-D	1	3	7		
B-E	1	3	7	11	
C-D-E	1				
B-D-E	1	3	7		
D-E-F	1	3	7	11	



E-F	1	3	7	11	15
C-D	1				
C-B	1	3			
A-B-D	1	3	7		
B-D	1	3	7		
B-E	1	3	7	11	
C-D-E	1				
B-D-E	1	3	7		
D-E-F	1	3	7	11	15



E-F	1	3	7	11	15
C-D	1				
C-B	1	3			
A-B-D	1	3	7		
B-D	1	3	7		
B-E	1	3	7	11	
C-D-E	1				
B-D-E	1	3	7		
D-E-F	1	3	7	11	15



For an extra challenge, try the real exam task (2019 - 4c) and do it in reverse

observation

hint Start with the flow(s) with the lowest bandwidth

More flows than links

Some links need to be the bottleneck for multiple flows

Task 2 & 3: Go-Back-N



```
if(nextseqnum<base+N){</pre>
   sndpkt[nextseqnum]=make_pkt(nextseqnum,data,checksum)
   udt_send(sndpkt[nextseqnum])
   if(base==nextseqnum)
      start_timer
```

```
timeout
                        start_timer
                        udt_send(sndpkt[base])
          Wait
                        udt_send(sndpkt[base+1])
                        . . .
                        udt_send(sndpkt[nextseqnum-1])
rdt_rcv(rcvpkt) && notcorrupt(rcvpkt)
```

```
base=getacknum(rcvpkt)+1
If(base==nextseqnum)
```

start_timer

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender Iossy link Receiver



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions



Sender



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions



Receiver

4 5 6 7 8 9

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender









a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions



5 6 7 8 9





a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions

Sender



a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions



Sender

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions











a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions







When a timeout occurs, the sender retransmits all segments in the window.

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions





Receiver

When a timeout occurs, the sender retransmits all segments in the window.

a simple reliable transport protocol with a sliding window, cumulative ACKs, timeouts and retransmissions





Communication Networks Exercise 4

Overview current assignment

Time for you to solve the tasks

Solutions will be published next week

