Netze und Verteilte Systeme

Programmierprojekt

Dmitrii Polianskii, Lukas Lamminger

Universität Salzburg

Description

Usage TX

C:

./TX portTX portRX packet_size packet_block_size send_delay file_name

java:

java TX portTX portRX packet_size packet_block_size send_delay file_name

- **portTX** - port to recieve acknowlegments (default: 4700)
- **portRX** - port to send datagrams (default: 4711)
- **packet_size** - size of a packet in Bytes (default: 1000)
- **packet_block_size** - amount of packets between delay (default: 100)
- **send_delay** - delay in microsec between blocks (default: 200)
- **file_name** - name of a file to transmit (default: to_send.jpg)

Usage RX

C:

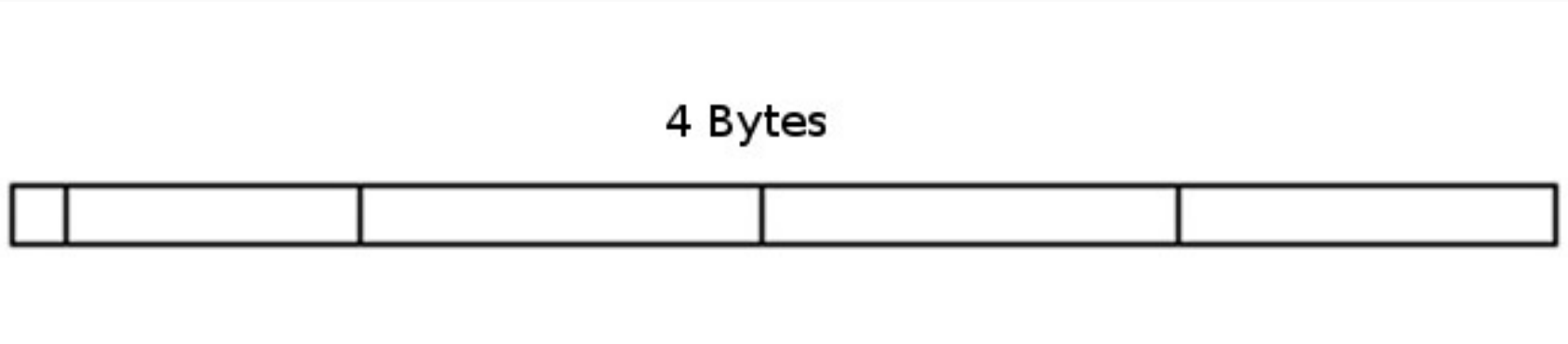
./RX portTX portRX

java:

java portRX portRX

- **portTX** - port to send acknowlegments (default: 4700)
- **portRX** - port to recieve datagrams (default: 4711)

Header structure



A header consists of 4 bytes.

First bit is used to indicate the last packet.

31 bits left for sequence number

TX description

1. Read file
   - 1.1 Read file in buffer
   - 1.2 Calculate CRC32 and add to filebytes
   - 1.3 Split filebytes in packages
2. Initialize UPD Socket
3. Initialize Acknowlegments array to obtain transmitted packets
4. Transmit one block of packets
   - 4.1 For every packet in block check if acknolegment was recieved, if not - send a packet.
   - 4.2 If last packet is reached, then start with first again,
5. Wait for acknowlegments {DELAY} microseconds.
   - 5.1 Write every sequence number from acknowlegment packet in Acknowlegments array,
   - 5.2 If all packets were acknowlegment - end transmission. Else - goto punkt 4.

RX description

RX description

1. Initialize UPD Socket
2. Listen for incomming packages
   - 2.1 Write databits from package in a memory
   - 2.2 If last-package-bit was seen, the size of file and Amount of packets can be defined.
   - 2.3 If not all of package were recieved, then goto punkt 2.
3. Assemble a file
4. Calculate CRC32 and compare with recieved one.

Tests

RX description

- For each set of parameters, the speed measurement is performed 10 times. After that, the average value is calculated
- The running time and the transfer rate are calculated only for the file transfer phase. Time for initialization and assembly / disassembly of the file is not taken into account.
- System Characteristics:
  - OS: Ubuntu 16.04 64-bit
  - Processor: Pentium T4200 @ 2.00GHz x 2
  - Memory: 4GB

Tests: C to C

TX.c to RX.c: Manipulate delays

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	200	0.023	62,8
100Kb	1000	100	50	0,009	111,1
100Kb	1000	100	10	0.01	124,2
1Mb	1000	100	200	0,118	76,5
1Mb	1000	100	50	0,087	103,1
1Mb	1000	100	10	0,127	71,3
10Mb	1000	100	200	1,863	45,7
10Mb	1000	100	50	1,618	52,1
10Mb	1000	100	10	1,813	48,8

TX.c to RX.c: Manipulate with size of packet

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	50	0,009	111,1
100Kb	10000	100	50	0.001	278,2
1Mb	1000	100	50	0,087	103,1
1Mb	10000	100	50	0,020	432,4
1Mb	65000	100	50	0,010	1003,7
10Mb	1000	100	50	1,618	52,1
10Mb	10000	100	50	0,170	498,1
10Mb	65000	100	50	0,079	1065,4

TX.c to RX.c: Manipulate with size of block

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	50	0,009	111,1
100Kb	1000	500	50	0,010	96,8
1Mb	1000	50	50	0,122	79,3
1Mb	1000	100	50	0,087	103,1
1Mb	1000	500	50	0,112	81,3
10Mb	1000	50	50	2,773	30,9
10Mb	1000	100	50	1,618	52,1
10Mb	1000	500	50	1,380	63,7
10Mb	1000	2000	50	1,307	67,7

TX.c to RX.c: Best results

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	65000	100	50	0,001	511,7
1Mb	65000	100	50	0,010	1003,7
10Mb	65000	200	50	0,069	1100,5

Tests: C to Java

TX.c to RX.java: Manipulate delays

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	200	0,135	6,4
100Kb	1000	100	1000	0,088	9,4
100Kb	1000	100	5000	0,082	9,9
1Mb	1000	100	200	0,708	13,2
1Mb	1000	100	1000	0,624	14,4
1Mb	1000	100	5000	0,622	14,4
10Mb	1000	100	200	3,020	27,8
10Mb	1000	100	1000	3,012	27,8
10Mb	1000	100	5000	3,104	27,5

TX.c to RX.java: Manipulate with size of packet

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	200	0,133	6,6
100Kb	10000	100	200	0,055	15,4
1Mb	1000	100	200	0,576	15,5
1Mb	10000	100	200	0,384	23,5
1Mb	65000	100	200	0,248	36.2
10Mb	1000	100	200	3,101	27,2
10Mb	10000	100	200	1,162	75,8
10Mb	65000	100	200	1,061	79,5

TX.c to RX.java: Manipulate with size of block

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	200	0,133	6,6
100Kb	1000	500	200	0,152	5.8
1Mb	1000	50	200	0,264	16,6
1Mb	1000	100	200	0,576	15,5
1Mb	1000	500	200	0,574	15,4
10Mb	1000	50	200	4,080	20,6
10Mb	1000	100	200	3,101	27,2
10Mb	1000	500	200	3,300	26,6
10Mb	1000	2000	200	3,94	21,4

TX.c to RX.java: Best results

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	65000	100	1000	0,040	99,0
1Mb	65000	100	1000	0,177	51,1
10Mb	65000	100	1000	1,748	63,2

Tests: Java to C

TX.java to RX.c: Manipulate delays

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,081	10,13
100Kb	1000	100	2000	0,080	10,14
1Mb	1000	100	1000	0,404	21,9
1Mb	1000	100	2000	0,448	19,7
10Mb	1000	100	1000	2,120	39,6
10Mb	1000	100	2000	2.151	38,8
10Mb	1000	100	5000	2.548	32,7

TX.java to RX.c: Manipulate with size of packet

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,081	10,13
100Kb	10000	100	1000	0,016	53,1
1Mb	1000	100	1000	0,404	21,9
1Mb	10000	100	1000	0,088	99,0
1Mb	65000	100	1000	0,020	376,9
10Mb	1000	100	1000	2,120	39,6
10Mb	10000	100	1000	0,482	183,8
10Mb	65000	100	1000	0.136	635,4

TX.java to RX.c: Manipulate with size of block

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,081	10,13
100Kb	1000	500	1000	0,013	54,8
1Mb	1000	50	1000	0,418	20,9
1Mb	1000	100	1000	0,404	21,9
1Mb	1000	500	1000	0,574	6,9
10Mb	1000	50	1000	2,312	36,2
10Mb	1000	100	1000	2,120	39,6
10Mb	1000	500	1000	3,288	26,2
10Mb	1000	2000	1000	31,190	2,8

TX.java to RX.c: Best results

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	500	1000	0,013	54,8
1Mb	65000	100	1000	0,020	376,9
10Mb	65000	100	1000	0.136	635,4

Tests: Java to Java

TX.java to RX.java: Manipulate delays

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,169	4,9
100Kb	1000	100	2000	0,218	4,3
1Mb	1000	100	1000	1,022	8,7
1Mb	1000	100	2000	0,900	9,7
10Mb	1000	100	1000	4,142	20,3
10Mb	1000	100	2000	4,148	20,2

TX.java to RX.java: Manipulate with size of packet

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,169	4,9
100Kb	10000	100	1000	0,092	8,9
1Mb	1000	100	1000	1,022	8,7
1Mb	10000	100	1000	0,710	13,2
1Mb	65000	100	1000	0,378	23,7
10Mb	1000	100	1000	4,142	20,3
10Mb	10000	100	1000	2,788	31,9
10Mb	65000	100	1000	1,91	77,5

TX.java to RX.java: Manipulate with size of block

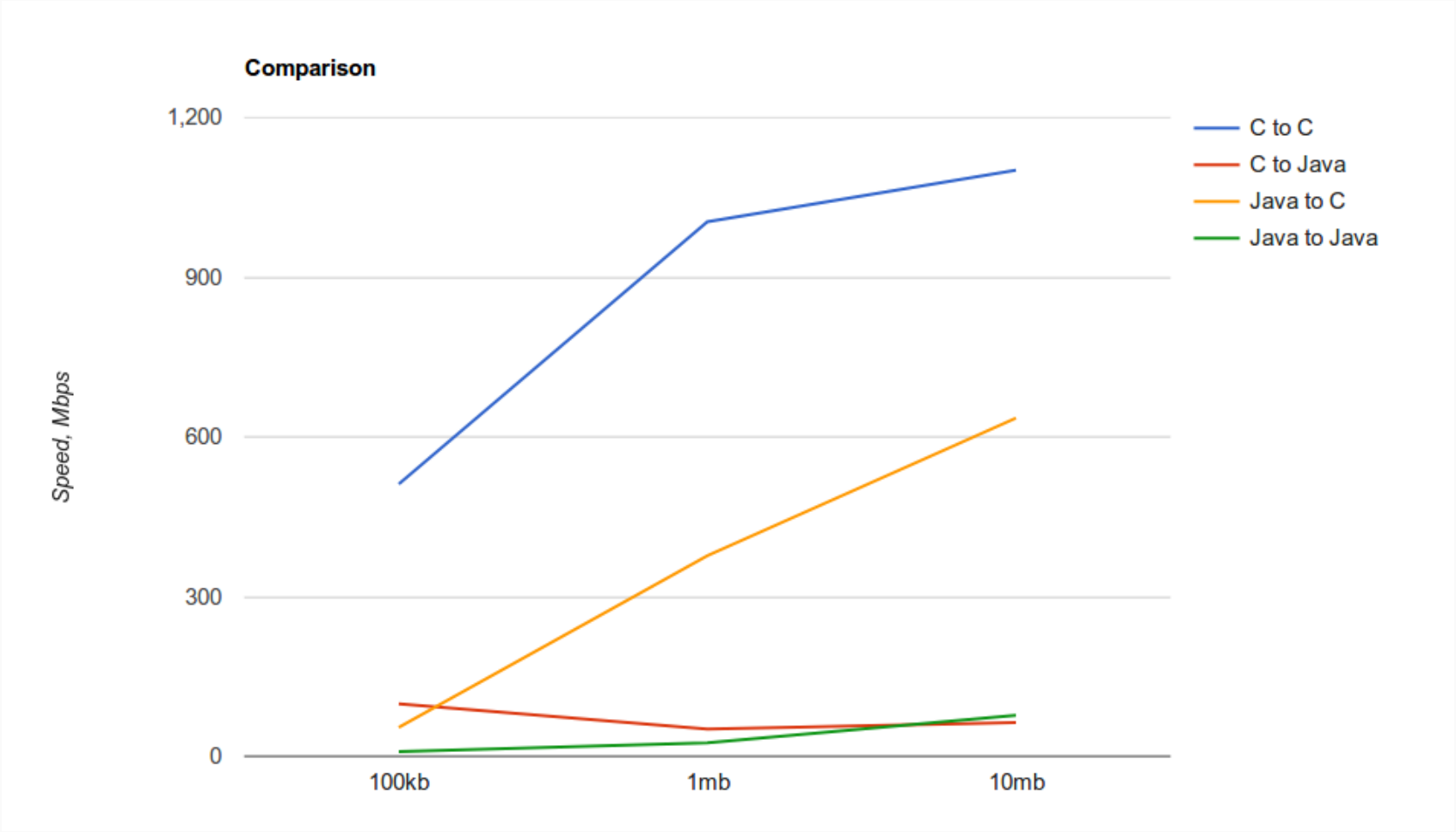
File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	1000	100	1000	0,169	4,9
100Kb	1000	500	1000	0,167	4,9
1Mb	1000	50	1000	1,010	8,8
1Mb	1000	100	1000	1,022	8,7
1Mb	1000	500	1000	1,188	7,5
10Mb	1000	50	1000	3,87	21,7
10Mb	1000	100	1000	4,142	20,3
10Mb	1000	500	1000	6,388	13,9

TX.java to RX.java: Best results

File size	Packet size (Bytes)	Block size (packets)	Delay (microseconds)	Elapsed time (s)	Speed (Mbps)
100Kb	10000	100	1000	0,092	8,9
1Mb	65000	100	5000	0,346	25,4
10Mb	65000	100	1000	1,91	77,5

Compare the best of results

Compare


Comparison
1,200
900
600
300
0
Speed, Mbps
100kb
1mb
10mb
C to C
C to Java
Java to C
Java to Java