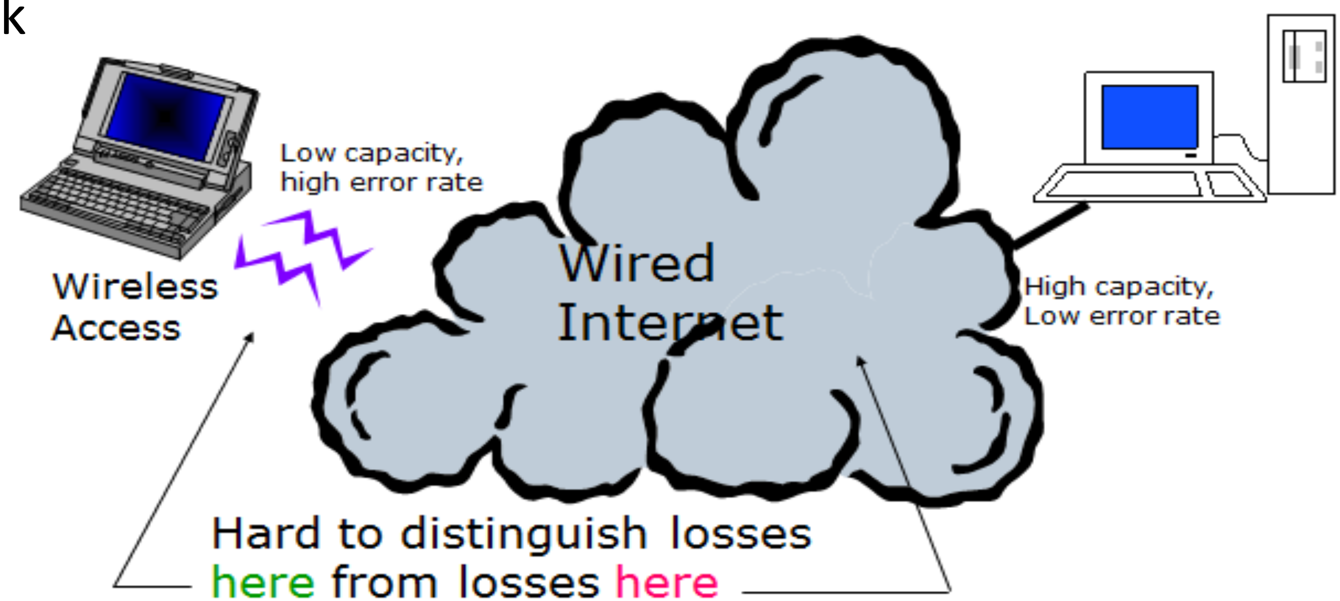
Combining TCP with coding in wireless network

Jinzhu Wang

Problem Statement

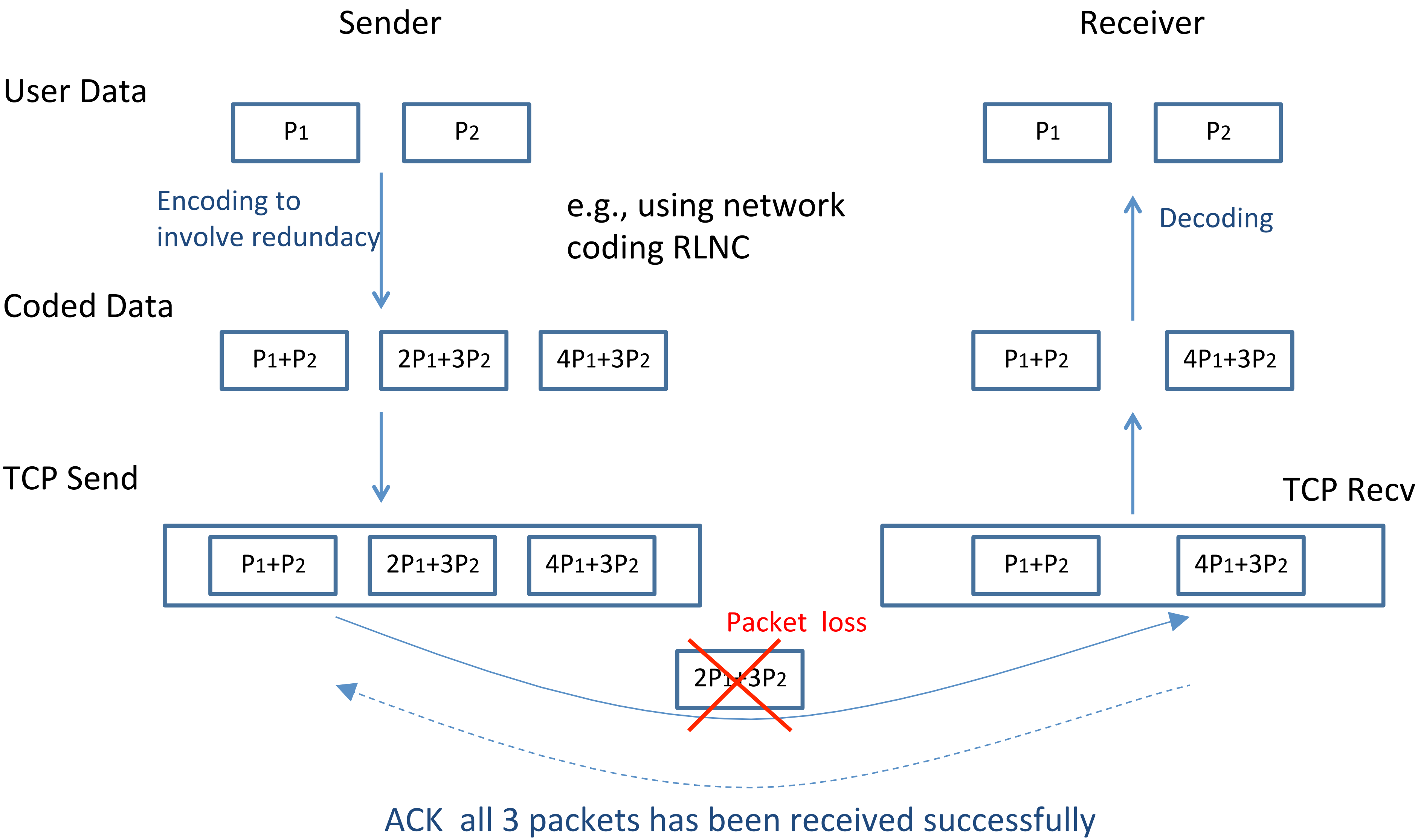
- Traditional TCP is inefficient in wireless network
  - TCP was primarily designed for the wired network with low BER(Bit error rate), where packet loss caused by network congestion
  - In wireless network, a lot of factors leads to unstable air-link, which causes high BER and thus high packet loss
    - weather conditions, urban obstacles, multi-path interferences, limited coverage, mobility of the handset
  - TCP mistakes the link error packet loss as the congestion packet loss and thus reduces the sending rate unnecessarily in wireless network



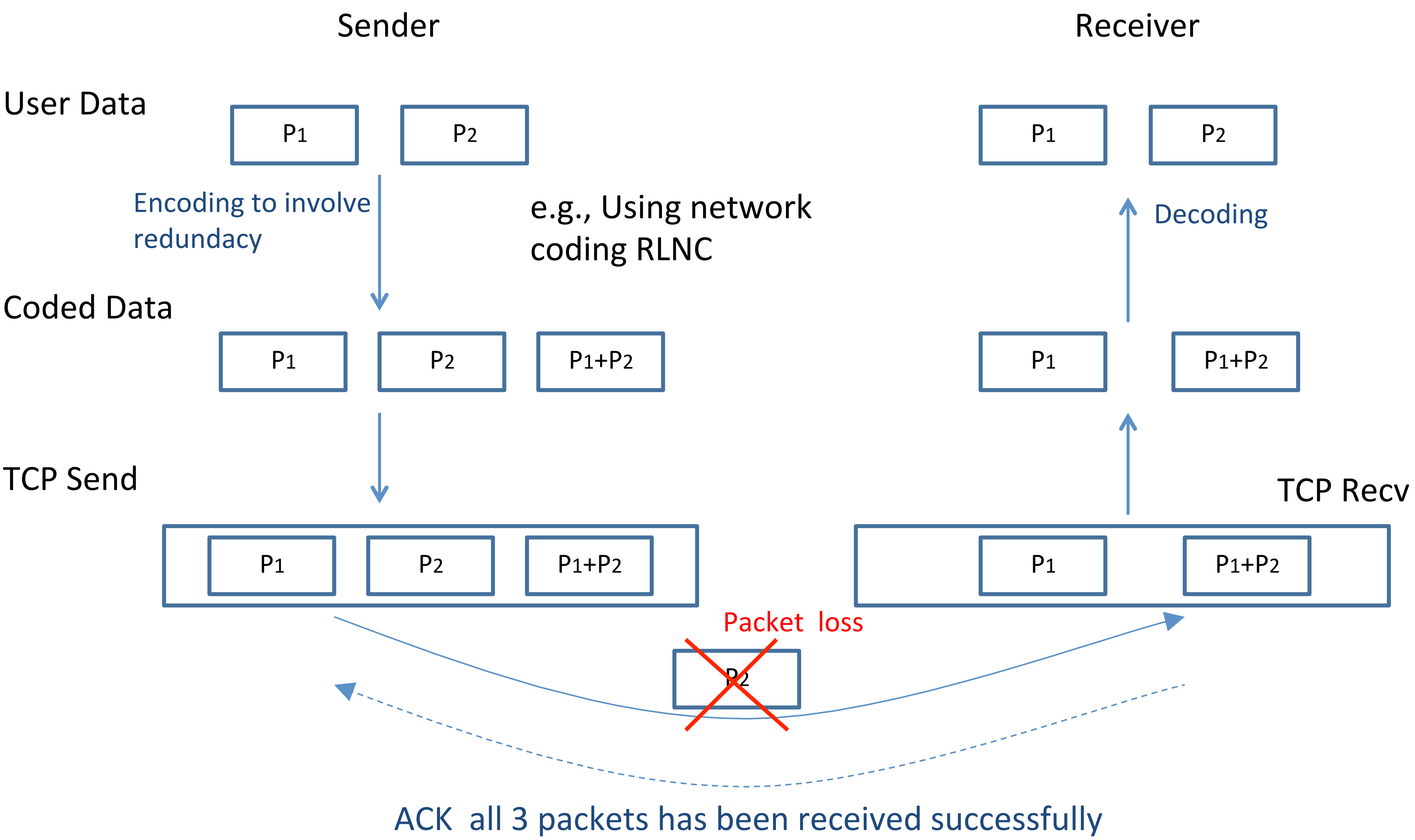
Problem Statement

- Test Packet loss in wireless network (static test)
  - WiFi
    - Test1: Public WiFi (location: Colleage campus)
      - Packet loss: 0.35% ;
    - Test2: Public WiFi (location: Business Center)
      - Packet loss: 1%;
    - Test3: Private WiFi (location: Home)
      - Packet loss: 0.15%;
  - LTE
    - Test4: CMCC LTE (location: Colleage campus)
      - Packet loss: 0%;
    - Test5 : CMCC LTE (Location: urban obstacles)
      - Packet loss: 0.75%;

Using coding to erase packet loss



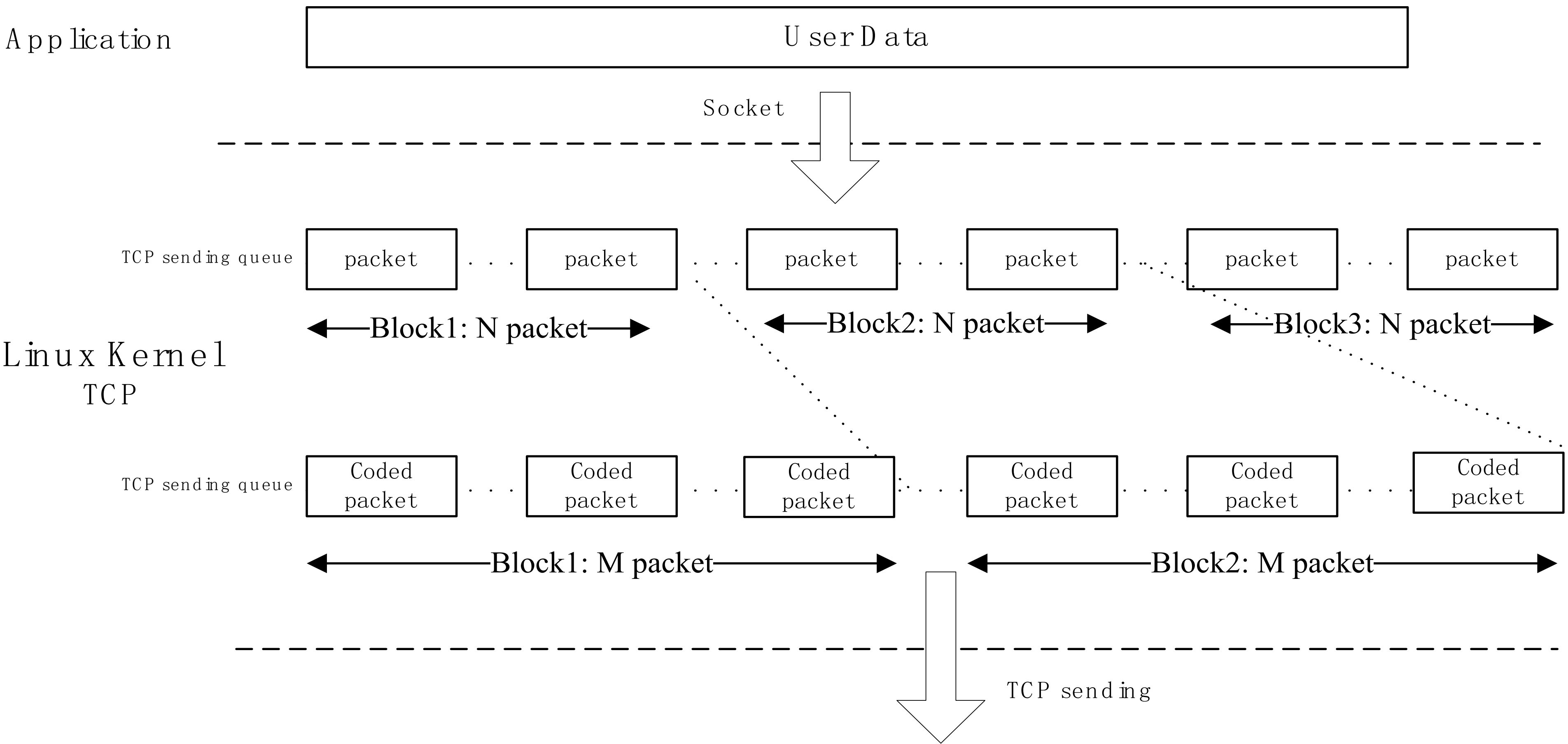
Using coding to erase packet loss (systematic codes)



Coded TCP archtecture

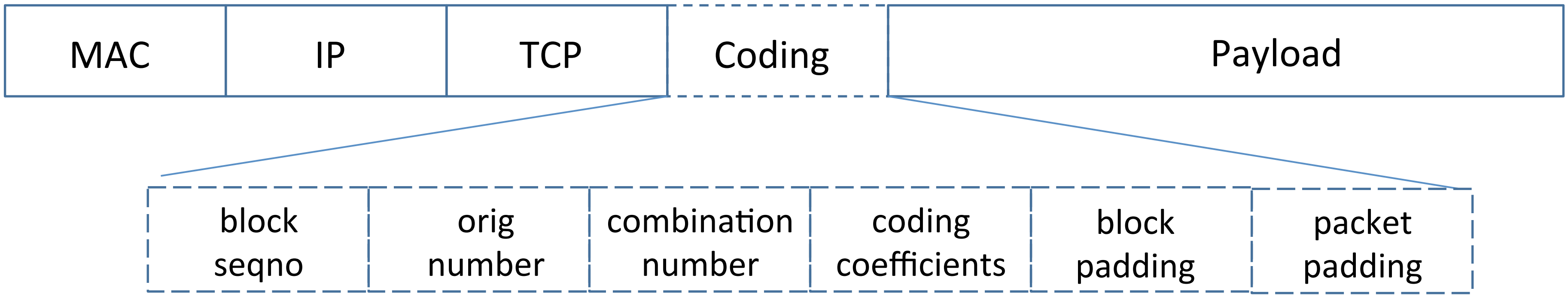
➢ Sender side

- Coding at each Block.
- M > N : coding involving redundancy
- Based on coding algorithm, make sure that any N of M coded packets received at the receiver side can result in decoding successfully



Coded TCP archtecture

Coding Header



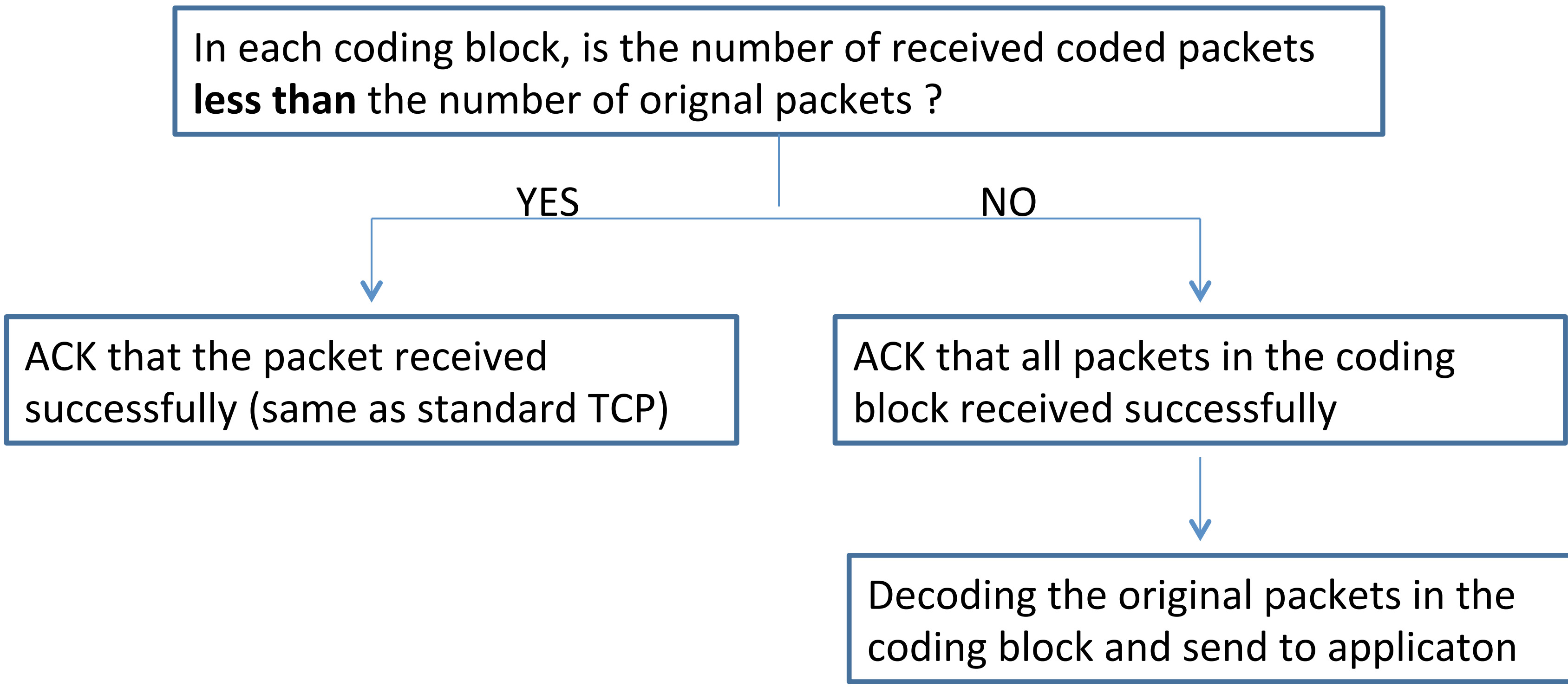
- **block seqno**: indicates the sequence number of the coding block
- **orig number**: the number of original packets in the coding block.
- **combination number**: the number of generated combinations (coded packets) in the coding block.
- **Coding coefficients**: the coefficients of original packets involved in combination. The coding coefficients are generated by coding algorithm and used by receiver to decode
- **Paddings**: the number of padding in the packet/block

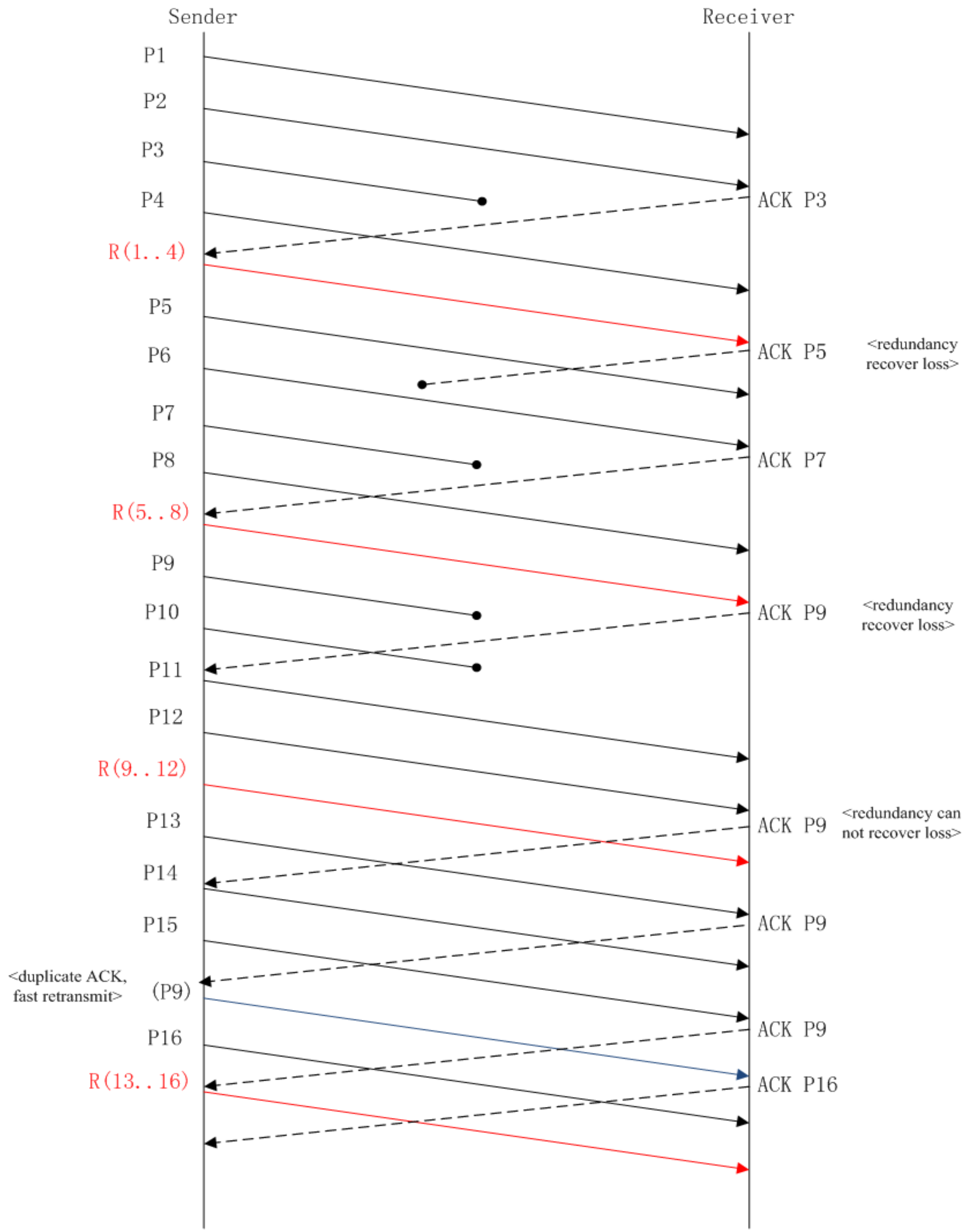
Coded TCP archtecture

➢ Receiver side

- Recod the the number of received coded packets (combinations) in each block.

- When packet arriving:

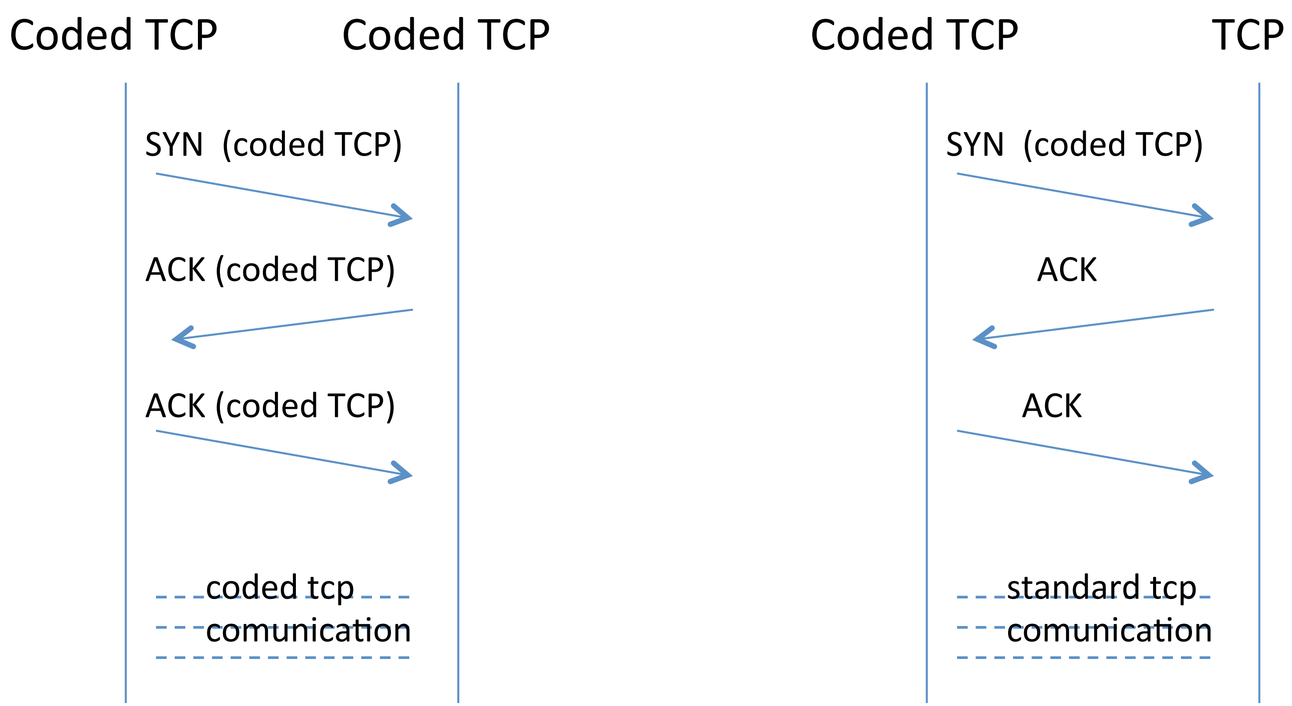




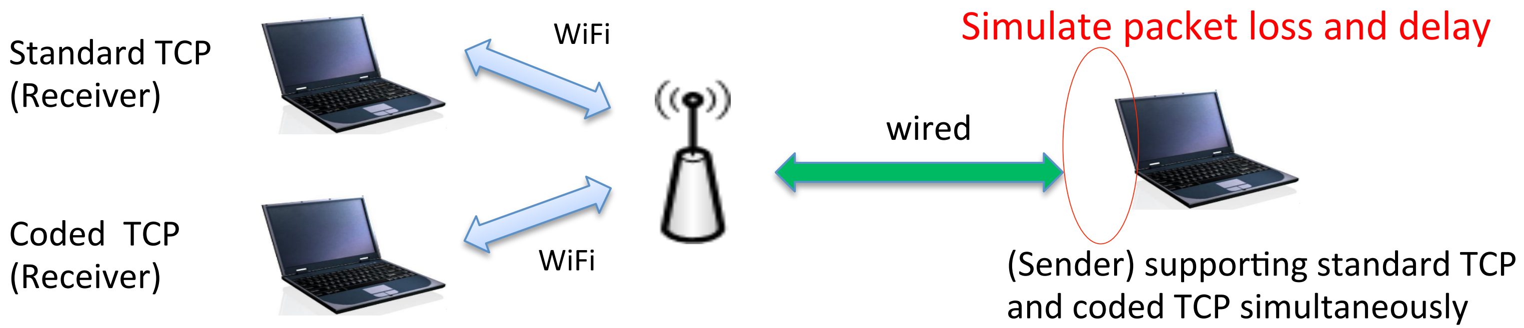
Sender
Receiver
P1
P2
P3
P4
ACK P3
R(1..4)
P5
P6
ACK P5
<redundancy recover loss>
P7
P8
ACK P7
R(5..8)
P9
P10
ACK P9
<redundancy recover loss>
P11
P12
R(9..12)
P13
ACK P9
<redundancy can not recover loss>
P14
P15
ACK P9
duplicate ACK, fast retransmit
(P9)
P16
ACK P9
R(13..16)
ACK P16

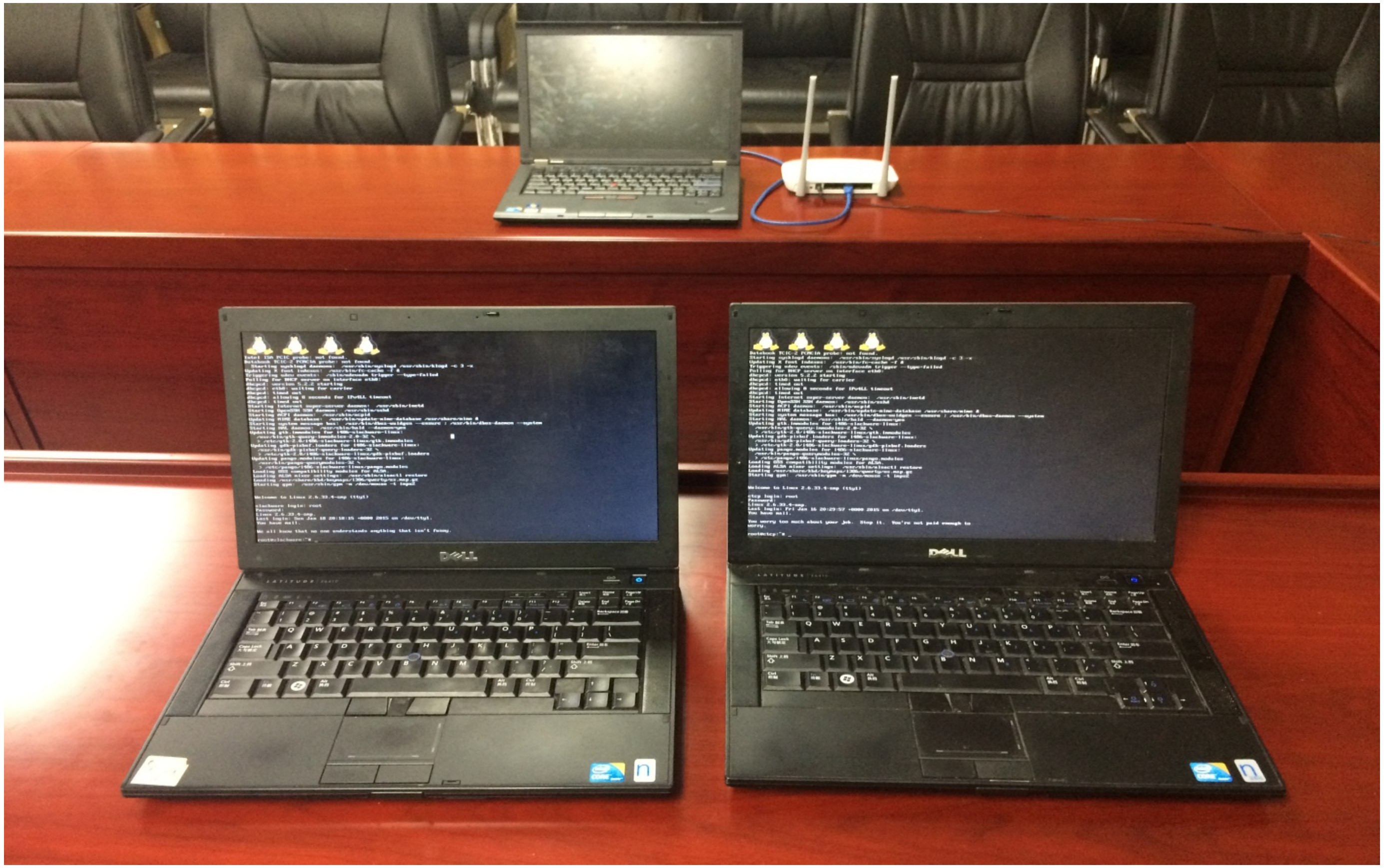
Interworking with standard TCP

- Coded TCP transmits the coded packets and cannot interwork with standard
- In order to interwork, Coded TCP SHOULD be able to choice whether or not enabling coding dynamically. If coding is disabled, coded TCP behaves like the standard TCP
- One bit in the reserved field of the TCP header is defined to identify whether transport uses the coded TCP or standard TCP



Experiment in lab





Experiment in lab

- Coded TCP: based on the Linux kernel 2.6.33.4, add coding to the standard TCP stack. The modified Linux kernel supports standard TCP and coded TCP simultaneously.

- Sender / Receiver: laptop with modified Linux kernel.
CPU: Intel(R) Core(TM) i5 CPU M540 @ 2.53GHz. Memory: 4G

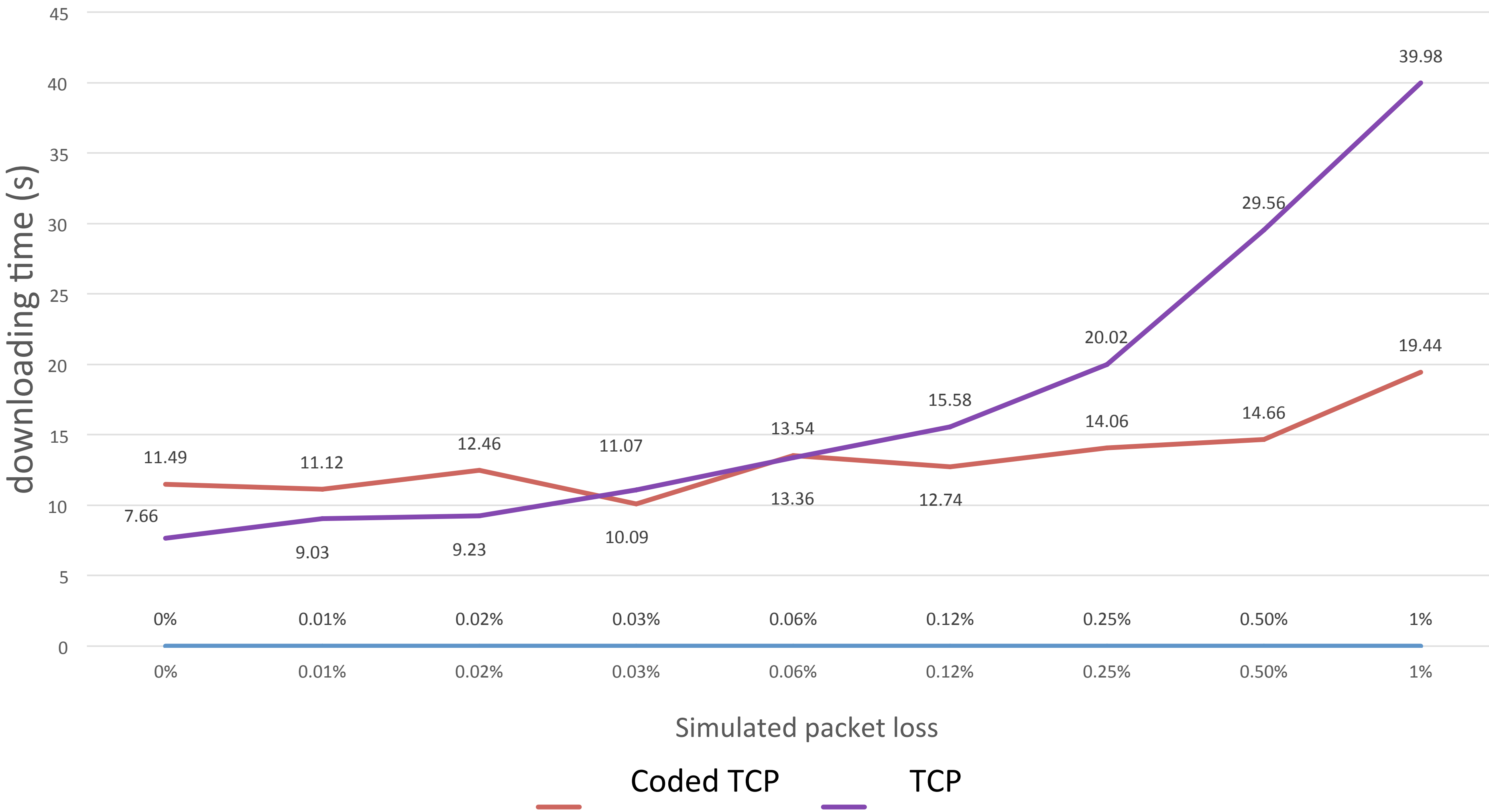
- WiFi: 802.11 b/g/n at 2.4 GHz

- Simulation software at Sender: Linux Traffic Control

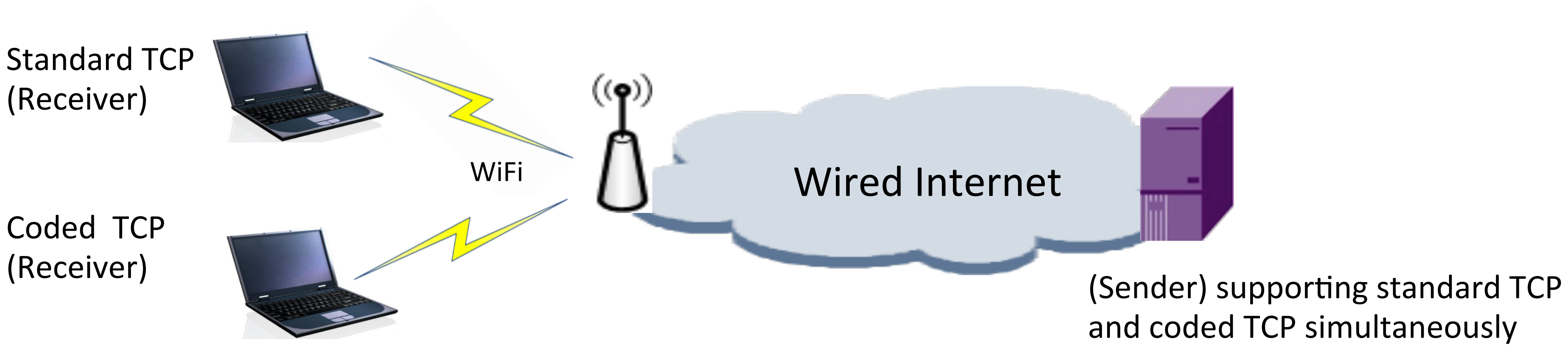
Experiment in lab

Wi-Fi， using FTP to transmit 20M file， simulated delay 20ms

Coded TCP: $\frac{coded\ packets}{original\ packets} = 1.2$



Experiment in public WiFi



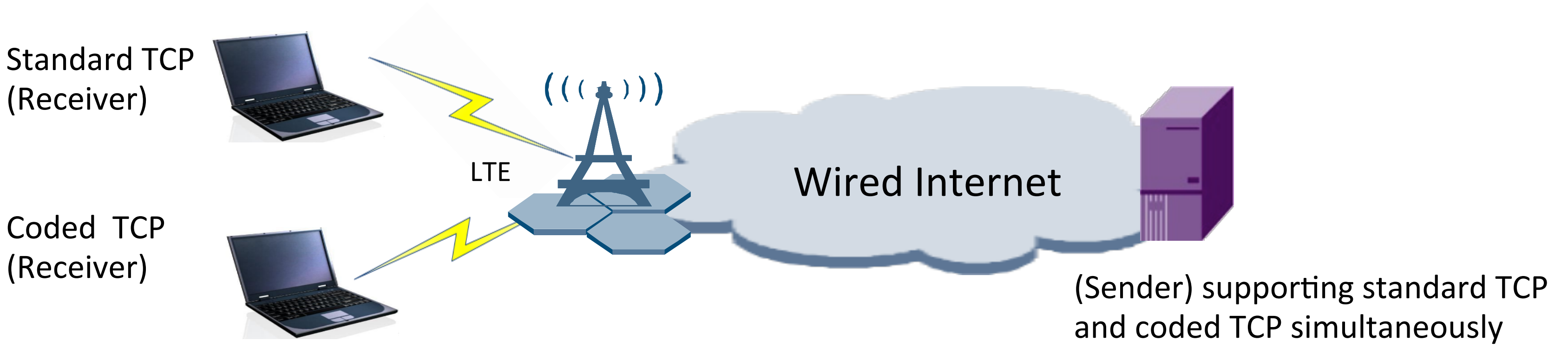
- using FTP to transmit 20M file
- CMCC Public WIFI at Colleage campus

Coded TCP: $\frac{coded\ packets}{original\ packets} = 1.2$

	Download time (s)
TCP	372
Coded TCP	263

- In Public WiFi, the coded TCP improves throughput significantly

Experiment in LTE



- using FTP to transmit 20M file
- CMCC LTE at Colleage campus

Coded TCP: $\frac{coded\ packets}{original\ packets} = 1.2$

	Download time (s)
TCP	33.8
Coded TCP	31.2

Use case: voice over WiFi

- In the test, we found that when using TCP to transmit voice/vedio, if the packet loss rate is greater than 2%, the quality of the voice/vedio will be degraded significantly
- Considering that many public WiFi suffer a non-negligible packet loss rate, the quality of voice over WiFi can not be guaranteed.
- Combing TCP with coding, the packet loss can be erased by coding and thus the quality of voice can be guaranteed in the WiFi.

Relationship with TCP congestion control

- Currently, coded TCP can not distinguish link error loss from congestion loss, thus the CWND will not be reduced if congestion loss can be recovered through coding. This will impact TCP congestion control
- On the other hand, In the condition of congestion loss, if coding can not recover the lost packets, coded TCP obeys TCP congestion control.
- we set the maximal coding ratio as 1.2 (20% redundant coded packets) in the coded TCP prototype. Thus:
  - case 1: if the packet loss is 0%, the coded tcp obeys the TCP congestion control with setting coding ratio as 1.
  - case2: if the packet loss is too high to be recovered by 1.2 coding ratio, the coded TCP obeys the TCP congestion control
  - case 3: if the packet loss is low so the 1.2 coding ratio can recoverr it, the coded TCP does not reduce the CWND. This leads to congestion increases and the 1.2 coding ratio can not recover. After that, the coded TCP obeys the TCP congestion control again.

  we plan to carefully evaluate whether case 3 can be acceptted in follow up work

Thanks