

# Reti di Calcolatori

## Esercizi

Univr - Dipartimento di Informatica

**Fabio Irimie**

1° Semestre 2024/2025

# Indice

<b>1</b>	<b>Indirizzamento</b>	<b>2</b>
1.1	Esercizio 1 . . . . .	2
1.1.1	Risoluzione . . . . .	2
1.2	Esercizio 2 . . . . .	2
1.2.1	Risoluzione . . . . .	2
1.3	Esercizio 3 . . . . .	3
1.3.1	Risoluzione . . . . .	3
1.4	Esercizio 4 . . . . .	4
1.4.1	Risoluzione . . . . .	4
1.5	Esercizio 5 . . . . .	6
1.5.1	Risoluzione . . . . .	6
1.6	Esercizio 6 . . . . .	7
1.6.1	Risoluzione . . . . .	7
1.7	Esercizio 7 . . . . .	9
1.7.1	Risoluzione . . . . .	10
<b>2</b>	<b>TCP</b>	<b>11</b>
2.1	Esercizio 1 . . . . .	11
2.1.1	Risoluzione . . . . .	12
2.2	Esercizio 2 . . . . .	14
2.2.1	Risoluzione . . . . .	14
2.3	Esercizio 3 . . . . .	16
2.3.1	Risoluzione . . . . .	17
2.4	Esercizio 4 . . . . .	18
2.4.1	Risoluzione . . . . .	18
2.5	Esercizio 5 . . . . .	21
2.5.1	Risoluzione . . . . .	21
<b>3</b>	<b>Livello 2</b>	<b>22</b>
3.1	Esercizio 1 (Aloha) . . . . .	23
3.1.1	Risoluzione . . . . .	23
3.2	Esercizio 2 (CSMA persistent) . . . . .	23
3.2.1	Risoluzione . . . . .	23
<b>4</b>	<b>Esercizi con Bridge (switch a 2 porte)</b>	<b>24</b>
4.1	Esercizio 1 (Aloha) . . . . .	24
4.1.1	Risoluzione . . . . .	24
4.2	Esercizio 2 (CSMA Persistent) . . . . .	25
4.2.1	Risoluzione . . . . .	25

# 1 Indirizzamento

## 1.1 Esercizio 1

Qual'è l'indirizzo di rete se ho il seguente indirizzo IP:

140.120.84.20/20

### 1.1.1 Risoluzione

L'indirizzo di rete corrisponde ai primi 20 bit dell'indirizzo IP, quindi bisogna passare alla notazione binaria:

140.120.84.20 → 10001100 01111000 01010100 00010100

I primi 20 bit sono assegnati al prefisso:

10001100 01111000 0101 0100 00010100  
Prefisso Suffisso

Per ottenere l'indirizzo di rete bisogna azzerare i bit del suffisso:

10001100 01111000 0101 0000 00000000  
Prefisso Suffisso

che in notazione decimale puntata diventa:

140.120.80.0

La maschera di questo IP è:

11111111 11111111 1111 0000 00000000  
Prefisso Suffisso

che in notazione decimale puntata diventa:

255.255.240.0

## 1.2 Esercizio 2

Si hanno 3 LAN. All'insieme delle 3 LAN è stato assegnato il blocco:

165.5.1.0/24

Creare 3 sottoreti per le 3 LAN in modo che abbiano tutte lo stesso numero di host.

### 1.2.1 Risoluzione

Per prima cosa si trasforma l'indirizzo IP in notazione binaria:

10100101 00000101 00000001 00000000  
Prefisso Suffisso

Per poter ottenere 3 sottoreti di dimensione servono 2 bit che vengono presi dal suffisso per identificare ciascuna delle 3 reti:

10100101 00000101 00000001 00 000000  
Prefisso Sottorete Suffisso

Le combinazioni possibili sono:

- $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{00}_{\text{Sottorete}} \underbrace{000000}_{\text{Suffisso}}$
- $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{01}_{\text{Sottorete}} \underbrace{000000}_{\text{Suffisso}}$
- $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{10}_{\text{Sottorete}} \underbrace{000000}_{\text{Suffisso}}$
- $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{11}_{\text{Sottorete}} \underbrace{000000}_{\text{Suffisso}}$

Ci troviamo con 4 sottoreti con lo stesso numero di indirizzi ( $2^6 = 64$ ). Di queste 4 sottoreti ne utilizziamo 3 e l'ultima rimane libera per utilizzi futuri.

Traducendo i blocchi in notazione decimale puntata si ha:

165.5.1.0/26 → LAN 1  
 165.5.1.64/26 → LAN 2  
 165.5.1.128/26 → LAN 3  
 165.5.1.192/26 → Libero

### 1.3 Esercizio 3

Usando lo stesso blocco dell'esercizio 2 si modifichi la LAN 1 affinché abbia il doppio degli indirizzi rispetto a quelli assegnati alle altre 2 LAN.

#### 1.3.1 Risoluzione

Il blocco di partenza in notazione binaria è:

10100101 00000101 00000001 00000000

Per ottenere il doppio degli indirizzi rispetto alle altre 2 LAN bisogna prendere un bit dal suffisso e assegnarlo al prefisso ottenendo così 2 reti /25.

$\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{0}_{\text{Sottorete}} \underbrace{0000000}_{\text{Suffisso}}$   
 $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{1}_{\text{Sottorete}} \underbrace{0000000}_{\text{Suffisso}}$

Dalla rete si fa la stessa operazione separando un bit dal suffisso e ottenendo altri 2 blocchi da /26.

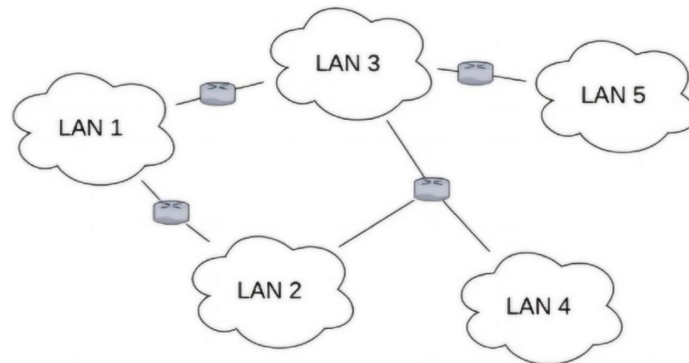
$\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{0}_{\text{Lan 1}} \underbrace{0000000}_{\text{Suffisso}}$   
 $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{10}_{\text{Lan 2}} \underbrace{000000}_{\text{Suffisso}}$   
 $\underbrace{10100101 \ 00000101 \ 00000001}_{\text{Prefisso}} \underbrace{11}_{\text{Lan 3}} \underbrace{000000}_{\text{Suffisso}}$

Traducendo i blocchi in notazione decimale puntata si ha:

Lan 1: 165.5.1.0/25  
 Lan 2: 165.5.1.128/26  
 Lan 3: 165.5.1.192/26

## 1.4 Esercizio 4

Si consideri la seguente rete suddivisa in 5 sottoreti:



Ci sono due indirizzi già assegnati alla rete:

- 101.75.79.255
  - 101.75.80.0
1. Qual'è il blocco **CIDR** più piccolo (con il minor numero di indirizzi) che contiene tali indirizzi?
  2. Dato il blocco **CIDR** della domanda precedente, si creino 5 sottoreti con i seguenti vincoli:
    - **LAN 1**: deve essere una sottorete /21
    - **LAN 2**: deve ospitare fino a 1000 host
    - **LAN 3**: deve essere una sottorete /23
    - **LAN 4**: deve ospitare fino a 400 host
    - **LAN 5**: deve ospitare metà host rispetto al blocco iniziale

### 1.4.1 Risoluzione

1. Convento entrambi gli indirizzi in notazione binaria:

101.75.79.255 → 01100101 01001011 01001111 11111111

101.75.80.0 → 01100101 01001011 01010000 00000000

Siccome i due IP sono uguali fino al 19° bit a partire da sinistra, si può dire che il blocco CIDR più piccolo che contiene entrambi gli indirizzi sia quello della rete:

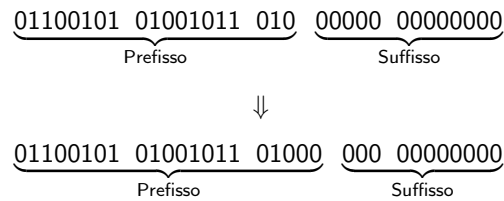
01100101 01001011 010 00000 00000000  
Prefisso Suffisso

che in notazione intera puntata è il seguente:

101.75.64.0/19

2. • **LAN 1:**

Per avere una sottorete /21 basta spostare i bit del prefisso:

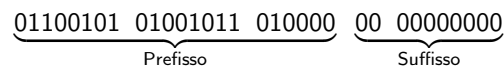


che in notazione intera puntata risulta:

101.75.64.0/21

• **LAN 2:**

1000 host sono circa  $2^{10}$ , di conseguenza per avere un blocco che possa ospitare fino a 1000 host esso deve avere almeno 10 bit di suffisso:



che in notazione intera puntata risulta:

101.75.64.0/22

• **LAN 3:**

Per avere una sottorete /23 basta spostare i bit del prefisso:

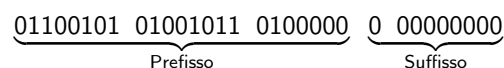


che in notazione intera puntata risulta:

101.75.64.0/23

• **LAN 4:**

400 host sono circa  $2^9$ , di conseguenza per avere un blocco che possa ospitare fino a 400 host esso deve avere almeno 9 bit di suffisso:



che in notazione intera puntata risulta:

101.75.64.0/23

- **LAN 5:**

Il blocco iniziale riesce ad ospitare  $2^{13}$  host, quindi per creare una rete che ne ospiti la metà bisogna avere  $\frac{2^{13}}{2} = 2^{13-1} = 2^{12}$  12 bit di suffisso:

$\underbrace{01100101\ 01001011\ 0100}_{\text{Prefisso}}\ \underbrace{0000\ 00000000}_{\text{Suffisso}}$

che in notazione intera puntata risulta:

101.75.64.0/20

## 1.5 Esercizio 5

Si hanno 3 LAN con i seguenti numeri di host:

1. LAN 1: 300 host
2. LAN 2: 40 host
3. LAN 3: 90 host

L'indirizzo di broadcast della LAN 3 è:

148.12.79.255

1. Trovare il blocco CIDR totale da assegnare all'intera rete
2. Partendo da tale blocco suddividerlo in sottoreti da assegnare alle 3 LAN

### 1.5.1 Risoluzione

1. Per trovare il blocco CIDR totale bisogna trovare il blocco che riesce a contenere il numero di host totale (in base 2) delle 3 LAN:

$$512 + 64 + 128 = 704$$

Il blocco CIDR che riesce a contenere 704 host è:

$$2^{10} = 1024$$

Di conseguenza il blocco CIDR totale dovrà avere 10 bit di suffisso e l'indirizzo di rete si ottiene convertendo l'indirizzo di broadcast in notazione binaria e azzerando i bit del suffisso:

$\underbrace{10010100\ 00001100\ 010011}_{\text{Prefisso}}\ \underbrace{00\ 00000000}_{\text{Suffisso}}$

che in decimale risulta:

148.12.76.0/22

2. Per suddividere il blocco CIDR in 3 sottoreti bisogna trovare il numero di bit di suffisso necessari per contenere il numero di host di ciascuna LAN:

- LAN 1: 300 host,  $2^9 = 512$  quindi 9 bit di suffisso

- LAN 2: 40 host,  $2^6 = 64$  quindi 6 bit di suffisso
- LAN 3: 90 host,  $2^7 = 128$  quindi 7 bit di suffisso

Quindi il blocco CIDR totale:

$\underbrace{10010100 \ 00001100 \ 010011}_{\text{Prefisso}} \ \underbrace{00 \ 00000000}_{\text{Suffisso}}$

verrà suddiviso in:

LAN 1:  $\underbrace{10010100 \ 00001100 \ 010011}_{\text{Prefisso}} \ \underbrace{0}_{\text{Lan 1}} \ \underbrace{0 \ 00000000}_{\text{Suffisso}}$   
 LAN 2:  $\underbrace{10010100 \ 00001100 \ 010011}_{\text{Prefisso}} \ \underbrace{11 \ 01}_{\text{Lan 2}} \ \underbrace{000000}_{\text{Suffisso}}$   
 LAN 3:  $\underbrace{10010100 \ 00001100 \ 010011}_{\text{Prefisso}} \ \underbrace{11 \ 1}_{\text{Lan 3}} \ \underbrace{0000000}_{\text{Suffisso}}$

che in notazione puntata risultano:

LAN 1: 148.12.76.0/23  
 LAN 2: 148.12.79.64/26  
 LAN 3: 148.12.79.128/25

## 1.6 Esercizio 6

Si hanno 4 lan che devono contenere il seguente numero di host:

LAN 1: 130 host  
 LAN 2: 270 host  
 LAN 3: 65 host  
 LAN 4: 35 host

La LAN 1 contiene l'indirizzo 46.144.141.41.

1. Calcolare il blocco CIDR totale.
2. Quali sono gli indirizzi di rete delle 4 LAN?

### 1.6.1 Risoluzione

1. Per trovare il blocco CIDR totale bisogna trovare il blocco di indirizzi che contiene tutti gli indirizzi delle LAN (in base 2):

LAN 1:  $130 \rightarrow 256 = 2^8$   
 LAN 2:  $270 \rightarrow 512 = 2^9$   
 LAN 3:  $65 \rightarrow 128 = 2^7$   
 LAN 4:  $35 \rightarrow 64 = 2^6$

il blocco totale sarà:

$$256 + 512 + 128 + 64 = 960 \rightarrow 1024 = 2^{10}$$



Avremo bisogno di 10 bit per il suffisso per poter indirizzare 1024 host.

Convertiamo l'indirizzo contenuto dalla LAN 1 in binario per poter ricavare l'indirizzo di rete:

46.144.141.41

↓

00101110 10010000 10001101 00101001

Sappiamo che 10 bit vanno dedicati al suffisso, di conseguenza 22 saranno dedicati al prefisso, quindi l'indirizzo di rete è:

00101110 10010000 10001100 00000000  
Prefisso Suffisso

che in notazione decimale diventa:

46.144.140.0/22

2. Per trovare gli indirizzi di ognuna delle LAN dobbiamo vedere quanti bit bisogna assegnare ad ognuna di esse per indirizzare tutti gli host richiesti. Abbiamo visto nella domanda precedente che gli indirizzi richiesti per ogni lan sono i seguenti:

LAN 1:  $130 \rightarrow 256 = 2^8$

LAN 2:  $270 \rightarrow 512 = 2^9$

LAN 3:  $65 \rightarrow 128 = 2^7$

LAN 4:  $35 \rightarrow 64 = 2^6$

- La LAN 1 richiede 8 bit di suffisso e sappiamo che contiene il seguente indirizzo:

00101110 10010000 100011 01 00101001  
Prefisso LAN 1 Suffisso

di conseguenza l'indirizzo di rete della LAN 1 è:

00101110 10010000 100011 01 00000000  
Prefisso LAN 1 Suffisso

↓

46.144.141.0/24

- La LAN 2 richiede 9 bit di suffisso:

00101110 10010000 100011 1 0 00000000  
Prefisso LAN 2 Suffisso

↓

46.144.142.0/23

- La LAN 3 richiede 7 bit di suffisso:

$$\underbrace{00101110}_{\text{Prefisso}} \underbrace{10010000}_{\text{Prefisso}} \underbrace{10001100}_{\text{LAN 3}} \underbrace{10000000}_{\text{Suffisso}}$$

$$\downarrow$$

$$46.144.140.128/25$$

- La LAN 4 richiede 6 bit di suffisso:

$$\underbrace{00101110}_{\text{Prefisso}} \underbrace{10010000}_{\text{Prefisso}} \underbrace{10001100}_{\text{LAN 4}} \underbrace{01000000}_{\text{Suffisso}}$$

$$\downarrow$$

$$46.144.140.64/26$$

## 1.7 Esercizio 7

Si hanno 3 lan con i seguenti host:

LAN 1: 400 host

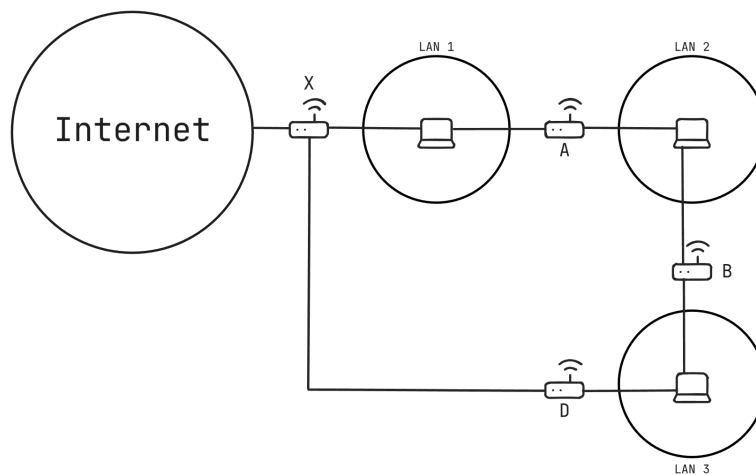
LAN 2: 300 host

LAN 3: 1200 host

La LAN 1 contiene un host con indirizzo:

178.242.85.168

Si consideri la seguente topologia:



Ogni host è collegato tramite un cavo coassiale a tutti gli altri host e anche al router e quindi li raggiunge entrambi con 1 hop.

1. Calcolare il blocco CIDR totale.
2. Calcolare gli indirizzi di rete delle 3 LAN.
3. Trovare la tabella di routing del router A considerando come metrica il numero di hop e assumendo che il router X abbia annunciato di raggiungere tutti gli host su Internet con 5 hop.

### 1.7.1 Risoluzione

1. Per trovare il blocco CIDR totale bisogna trovare il blocco di indirizzi che contiene tutti gli indirizzi delle LAN (in base 2):

$$\text{LAN 1: } 400 \rightarrow 512 = 2^9$$

$$\text{LAN 2: } 300 \rightarrow 512 = 2^9$$

$$\text{LAN 3: } 1200 \rightarrow 2048 = 2^{11}$$

il blocco totale sarà:

$$512 + 512 + 2048 = 3072 \rightarrow 4096 = 2^{12}$$

Avremo bisogno di 12 bit per il suffisso per poter indirizzare 4096 host.

Convertiamo l'indirizzo contenuto dalla LAN 1 in binario per poter ricavare l'indirizzo di rete:

178.242.85.168

↓

10110010 11110010 01010101 10101000

Il blocco CIDR totale sarà:

10110010 11110010 0101 0000 00000000  
Prefisso Suffisso

2. • La LAN 1 richiede 9 bit di suffisso e visto che sappiamo uno degli indirizzi contenuti all'interno della lan possiamo derivare l'indirizzo di rete:

10110010 11110010 0101 0101 10101000  
Prefisso Suffisso

↓

10110010 11110010 0101 010 0 00000000  
Prefisso LAN 1 Suffisso

In notazione decimale puntata:

178.242.84.0/23

- La LAN 2 richiede 9 bit di suffisso:

10110010 11110010 0101 011 0 00000000  
Prefisso LAN 2 Suffisso

In notazione decimale puntata:

178.242.86.0/23

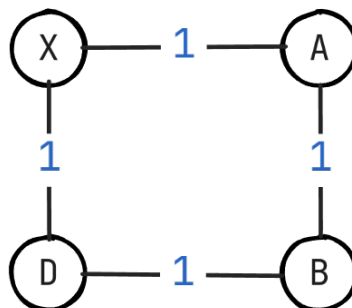
- La LAN 3 richiede 11 bit di suffisso:

10110010 11110010 0101 1 000 00000000  
Prefisso LAN 3 Suffisso

In notazione decimale puntata:

178.242.88.0/21

3. La topologia in forma di grafo sarà la seguente:



Il router X comunica ai router vicini i suoi distance vector:

Dst	Cst
Internet	5
A	1
D	1

Il router A aggiornerà la sua tabella di routing con i distance vector ricevuti da X e conoscerà la distanza per raggiungere Internet:

Dst	Next Hop	Cst
X	X	1
B	B	1
D	X	2
Internet	X	6

## 2 TCP

L'obiettivo di questi esercizi è quello di vedere come si comporta l'algoritmo in situazioni particolari.

### 2.1 Esercizio 1

Un'applicazione A deve trasferire verso un'applicazione B 96000byte. Si suppone che la connessione sia già stata instaurata. I dati sono i seguenti:

- $mss = 1000$  byte
- $rcvwnd = 32000$  byte, costante per l'intero trasferimento dei dati
- $ssthresh = \frac{rcvwnd_{iniziale}}{2}$
- $rtt =$  costante, pari a 0.5 secondi
- $rto = 2 \cdot rtt$ , raddoppia in caso di perdite sequenziali

- Down di rete (rete fuori uso, in cui tutti i segmenti vengono persi) =

$$t_1 = 3 \rightarrow t_2 = 3,5$$

$$t_3 = 7 \rightarrow t_4 = 7,5$$

Lo scopo è quello di valutare l'evoluzione temporale della `cwnd` fino a fine trasmissione.

### 2.1.1 Risoluzione

Il numero di segmenti da trasmettere sono:

$$\frac{\text{byte da trasmettere}}{\text{mss}} = \frac{96000\text{byte}}{1000\text{byte}} = 96 \text{ segmenti}$$

La `rcvwnd` iniziale vale:

$$\text{rcvwnd}_{\text{iniziale}} = \frac{32000\text{byte}}{1000\text{byte}} = 32 \text{ segmenti}$$

La `ssthresh` vale:

$$\text{ssthresh}_{\text{iniziale}} = \frac{32}{2} = 16 \text{ segmenti}$$

La `cwnd` iniziale vale 1:

$$\text{cwnd}_{\text{iniziale}} = 1$$

L'andamento della trasmissione è il seguente:

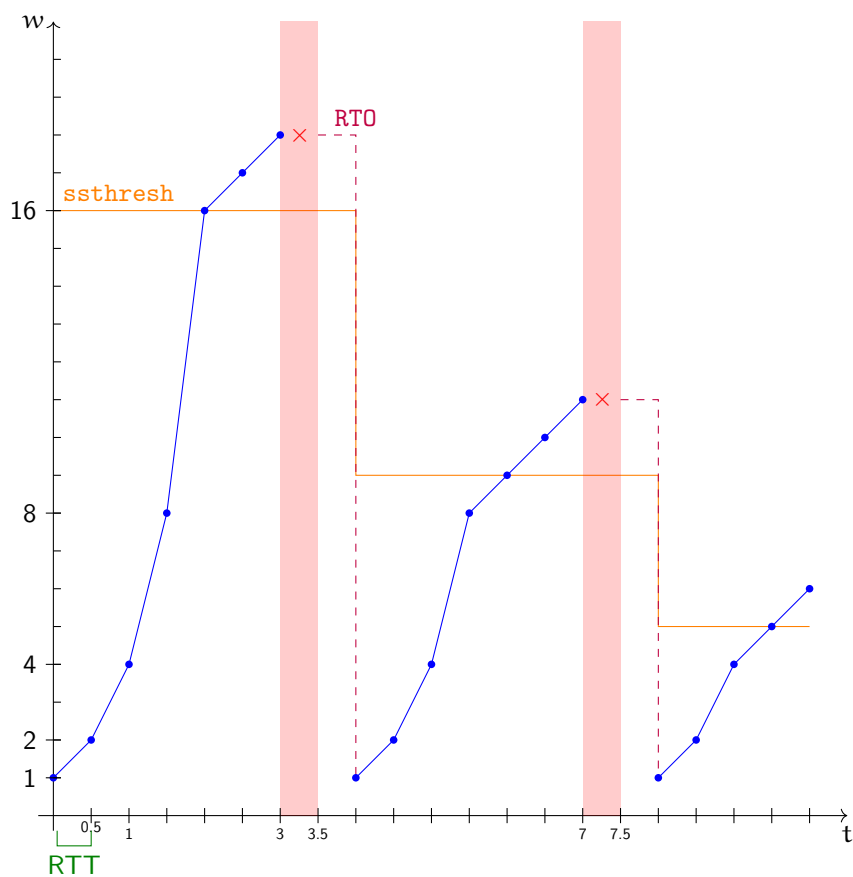


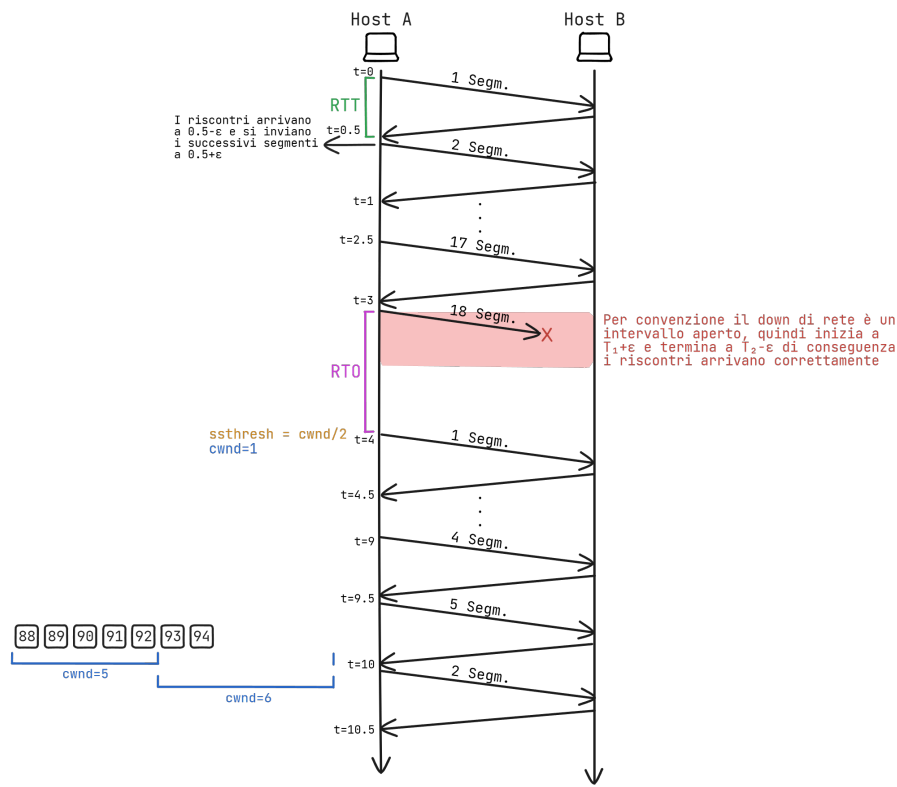
Figura 1: Andamento di cwnd in funzione del tempo

Il numero di segmenti trasmessi è:

$$\begin{aligned} \#seg &= 1 + 2 + 4 + 8 + 16 + 17 + 18 + 1 + 2 + 4 + 8 + 9 + 10 + 11 + 1 + 2 + 4 + 5 + 6 \\ &= 96 \end{aligned}$$

All'ultimo RTT si trasmettono soltanto 2 segmenti al posto di 6 perchè nonostante la finestra sia grande 6, il numero di pacchetti rimasti da trasmettere sono soltanto 2.

Un'altra possibile rappresentazione è la seguente:



## 2.2 Esercizio 2

Abbiamo un'applicazione A che trasferisce 46500byte verso un'applicazione B.

$$MSS = 1500 \text{ byte}$$

$$RCVWND_{\text{iniziale}} = 24000 \text{ byte} \rightarrow \text{costante}$$

$$SSTHRESH = \frac{RCVWND_{\text{iniziale}}}{2}$$

$$RTT = 0.5 \text{ secondi} \rightarrow \text{costante}$$

$$RTO = 2 \cdot RTT \rightarrow \text{raddoppia in caso di perdite consecutive}$$

$$\text{Down di rete} = [1.5 \rightarrow 3.5], [7 \rightarrow 7.5]$$

### 2.2.1 Risoluzione

Il numero di segmenti da trasmettere sono:

$$\frac{46500}{1500} = 31 \text{ segmenti}$$

$$RCVWND_{\text{iniziale}} = \frac{24000}{1500} = 16 \text{ segmenti}$$

$$SSTHRESH = \frac{16}{2} = 8 \text{ segmenti}$$

L'evoluzione della finestra di congestione è la seguente:

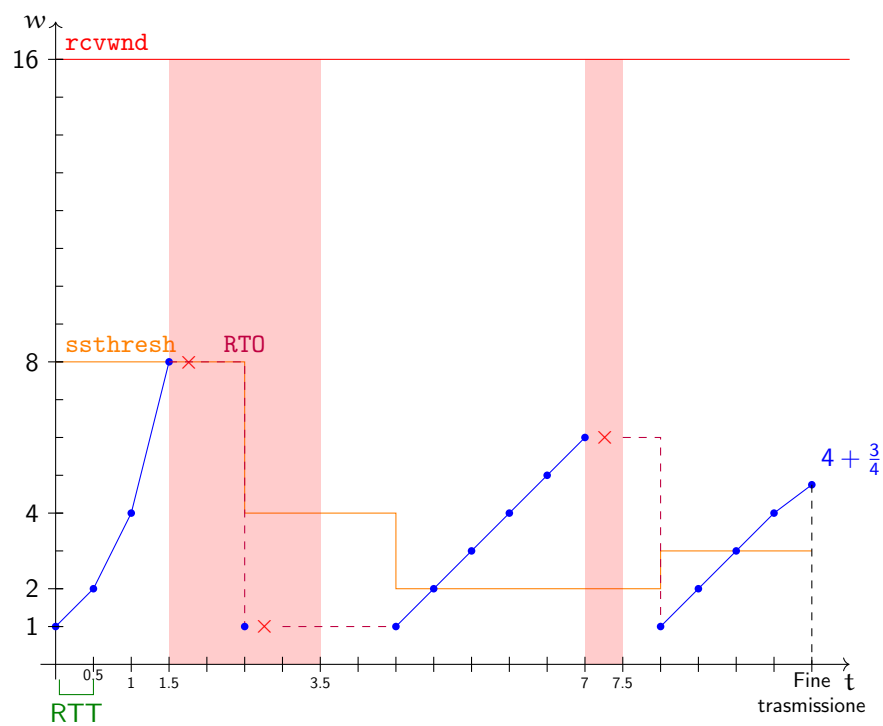


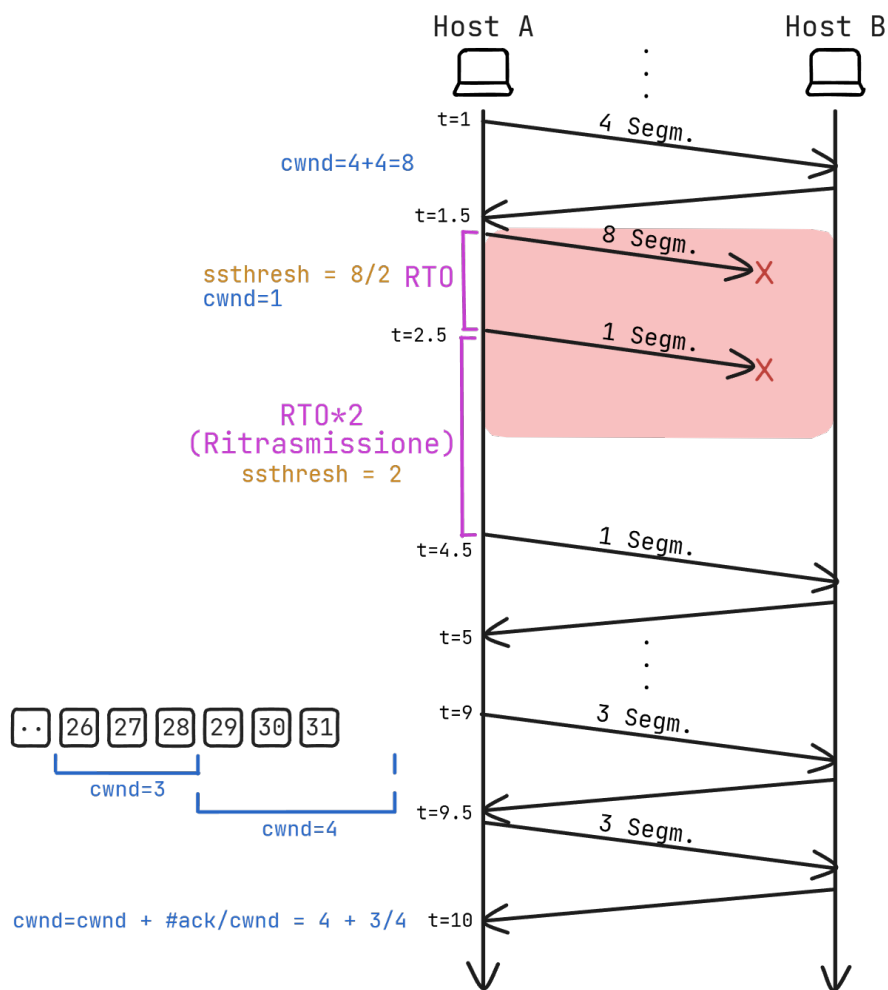
Figura 2: Andamento di cwnd in funzione del tempo

Il numero di segmenti trasmessi è:

$$\begin{aligned} \#seg &= 1 + 2 + 4 + \cancel{8} + \cancel{1} + 1 + 2 + 3 + 4 + 5 + \cancel{6} + 1 + 2 + 3 + \cancel{4}^3 \\ &= 31 \end{aligned}$$

Un'altra possibile rappresentazione è la seguente:





### 2.3 Esercizio 3

Abbiamo un'applicazione A che trasferisce 104000byte verso un'applicazione B.

MSS = 1200 byte

RCVWND<sub>iniziale</sub> = 24000 byte → costante

SSTHRESH = RCVWND<sub>iniziale</sub>

RTT = 0.5 secondi → costante

RTO = 2 · RTT → raddoppia in caso di perdite consecutive

Down di rete = [3.5 → 4.5], [6.5 → 10.5]

### 2.3.1 Risoluzione

Il numero di pacchetti inviati è:

$$\frac{104000}{1200} = 87 \text{ segmenti}$$

$$\text{RCVWND}_{\text{iniziale}} = \frac{24000}{1200} = 20$$

$$\text{SSTHRESH} = 20 \text{ segmenti}$$

L'evoluzione della finestra di congestione è la seguente:

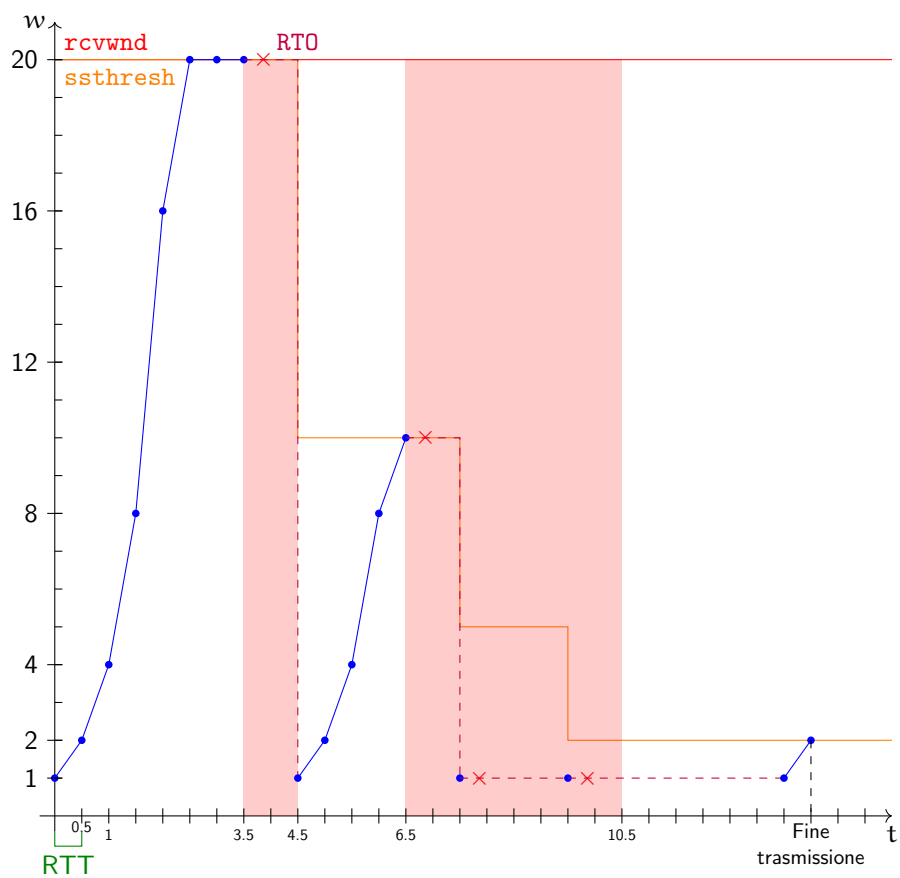


Figura 3: Andamento di cwnd in funzione del tempo

Il numero di segmenti trasmessi è:

$$\#seg = 1 + 2 + 4 + 8 + 16 + 20 + 20 + 16 + 1 + 2 + 4 + 8 + 1 + 1 + 1 + 1$$

$$= 87$$

## 2.4 Esercizio 4

Un'applicazione A trasferisce 104400 byte verso un'applicazione B.

$$MSS = 1200 \text{ byte}$$

$$RCVWND_{\text{iniziale}} = 9600 \text{ byte}$$

$$SSTHRESH = RCVWND_{\text{iniziale}}$$

$$RTT = 1 \text{ secondo} \rightarrow \text{costante}$$

$$RTO = 2 \cdot RTT \rightarrow \text{raddoppia in caso di perdite consecutive}$$

$$\text{Down di rete} = [11.5 \rightarrow 12.5]$$

A partire dall'istante  $t_a > 4 \text{ sec}$  la destinazione annuncia una

$$RCVWND = 14400 \text{ byte}$$

A partire dall'istante  $t_b > 9 \text{ sec}$  la destinazione annuncia una

$$RCVWND = 7200 \text{ byte}$$

Bisogna tenere in considerazione il ritardo di ricezione dell'annuncio della nuova RCVWND

### 2.4.1 Risoluzione

Il numero di segmenti da trasmettere sono:

$$\frac{104400}{1200} = 87 \text{ segmenti}$$

$$RCVWND_{\text{iniziale}} = \frac{9600}{1200} = 8 \text{ segmenti}$$

$$SSTHRESH = 8 \text{ segmenti}$$

$$t_a^{\text{dst}} > 4 = 12 \text{ segmenti}$$

$$t_b^{\text{dst}} > 9 = 6 \text{ segmenti}$$

L'evoluzione della finestra di congestione è la seguente:

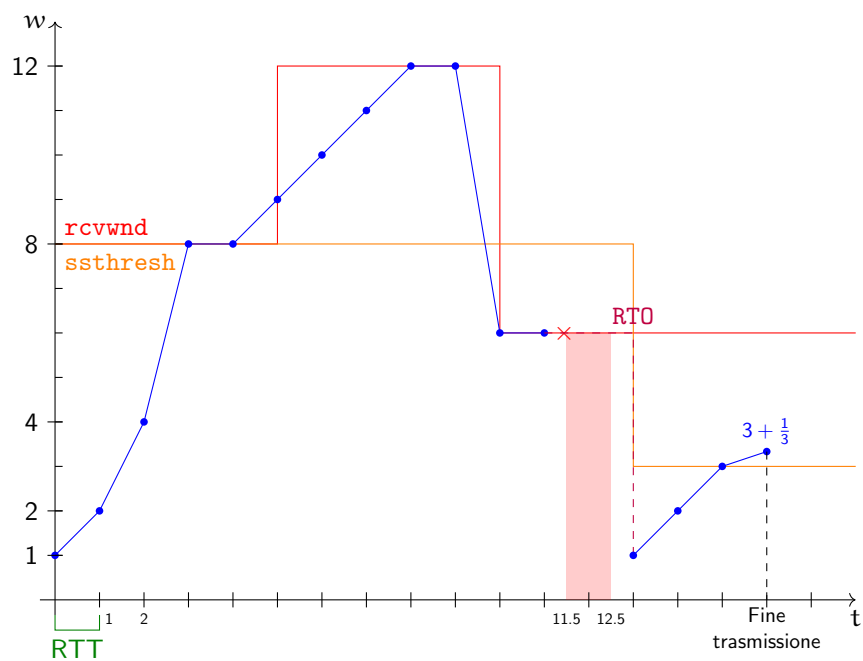
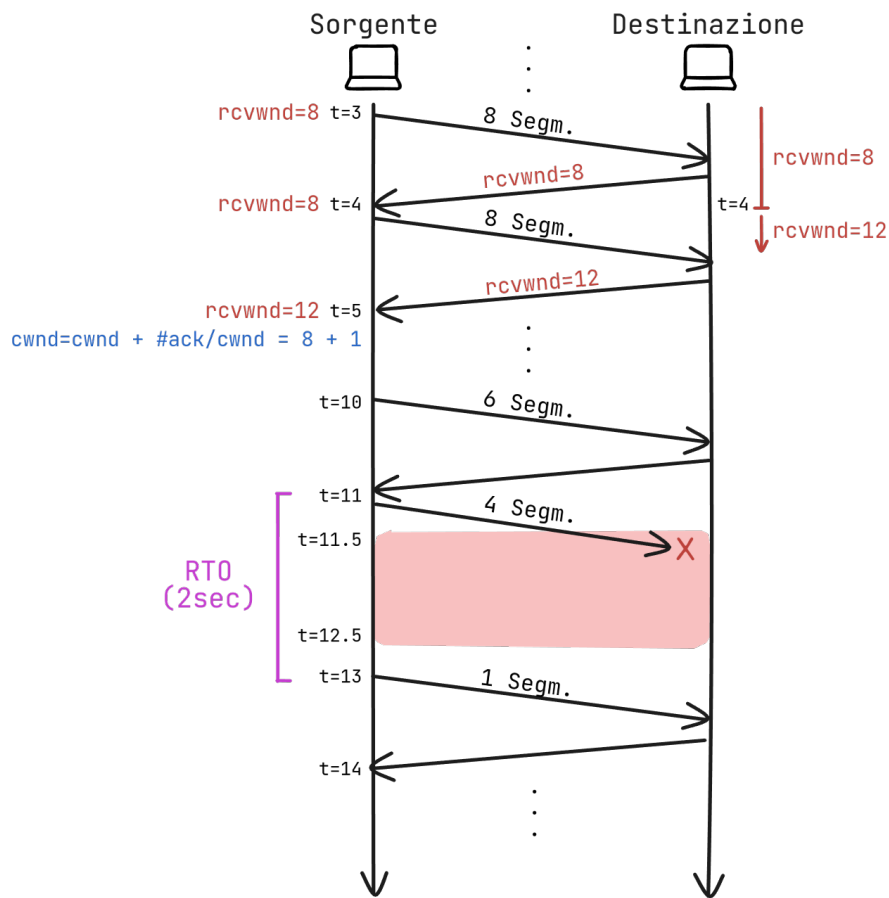


Figura 4: Andamento di `cwnd` in funzione del tempo

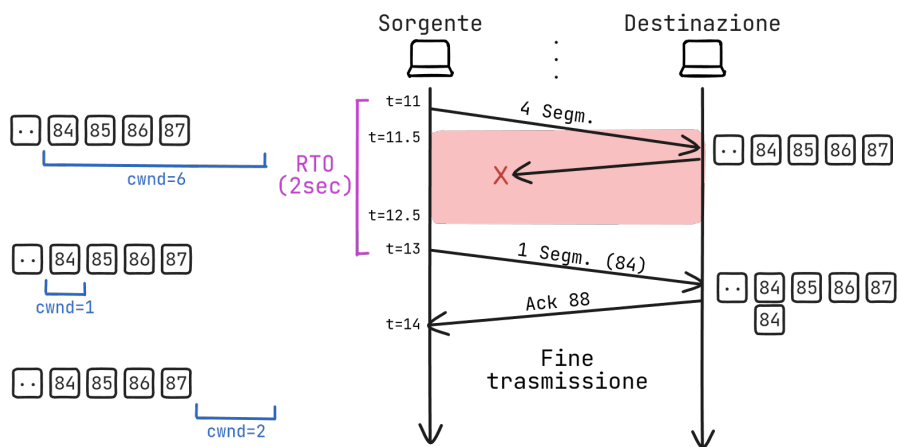
Il numero di segmenti trasmessi è:

$$\#seg = 1 + 2 + 4 + 8 + 8 + 9 + 10 + 11 + 12 + 12 + 6 + 4 + 1 + 2 + \frac{1}{3} = 87$$

Un'altra possibile rappresentazione è la seguente:



Se al posto di perdere il pacchetto si fosse perso il riscontro, avremmo avuto la seguente rappresentazione:



Anche questa rappresentazione è corretta, in quanto il numero di segmenti trasmessi è sempre 87.

## 2.5 Esercizio 5

Un applicazione A trasferisce 77500 byte verso un'applicazione B.

$$MSS = 1250 \text{ byte}$$

$$RCVWND_{\text{iniziale}} = 10000 \text{ byte}$$

$$SSTHRESH = RCVWND_{\text{iniziale}}$$

$$RTT = 1 \text{ second} \rightarrow \text{costante}$$

$$RTO = 2 \cdot RTT \rightarrow \text{raddoppia in caso di perdite consecutive}$$

$$\text{Down di rete} = [8 \rightarrow 10], [13.5 \rightarrow 14.5]$$

A partire dall'istante  $t_a > 4 \text{ sec}$  la destinazione annuncia una

$$RCVWND = 17500 \text{ byte}$$

A partire dall'istante  $t_b > 8 \text{ sec}$  la destinazione annuncia una

$$RCVWND = 12500 \text{ byte}$$

### 2.5.1 Risoluzione

Il numero di segmenti da trasmettere sono:

$$\frac{77500}{1250} = 62 \text{ segmenti}$$

$$RCVWND_{\text{iniziale}} = \frac{10000}{1250} = 8 \text{ segmenti}$$

$$SSTHRESH = 8 \text{ segmenti}$$

$$t_a^{\text{dst}} > 4 \quad RCVWND = 14 \text{ segmenti}$$

$$t_b^{\text{dst}} > 8 \quad RCVWND = 10 \text{ segmenti}$$

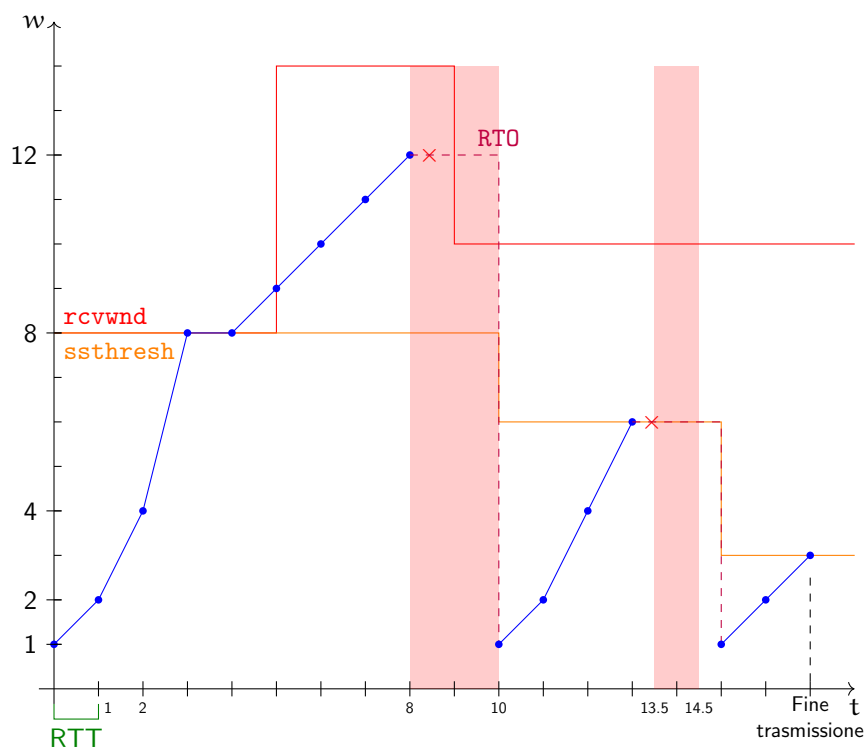


Figura 5: Andamento di  $w$  in funzione del tempo

Il numero di segmenti trasmessi è:

$$\#seg = 1 + 2 + 4 + 8 + 8 + 9 + 10 + 11 + 12 + 1 + 2 + 4 + 6 + 1 + 2 = 62$$

### 3 Livello 2

Sono esercizi su algoritmi di accesso al mezzo condiviso, cioè Aoha e CSMA, in cui lo scopo è quello di determinare l'evoluzione della trasmissione date una serie di trame generate dalle stazioni. Per questi esercizi al posto di estrarre un numero pseudocasuale bisogna usare una versione semplificata che non garantisce le proprietà statistiche di uniformità del campione, quindi il numero deve essere **deterministico**.

Per calcolare il numero casuale

1. Si usano le cifre che compongono l'istante di inizio trasmissione
2. Si sommano
3. Si moltiplicano per il numero di collisioni consecutive
4. Si aggiunge il tempo di trama

**Esempio 3.1.** Consideriamo una trama trasmessa a  $t = 512\text{ms}$  che ha già subito 1 collisione (questo è il secondo tentativo di trasmissione) e il tempo di trama è  $10\text{ms}$

$$z = (5 + 1 + 1) \cdot 2 + 10 = 26$$

### 3.1 Esercizio 1 (Aloha)

Ci sono 3 stazioni collegate da un mezzo condiviso, con ritardo di propagazione nullo  $\tau = 0$ , che generano le seguenti trame:

$$A \rightarrow t_A = 410\text{ms}$$

$$B \rightarrow t_B = 418\text{ms}$$

$$C \rightarrow t_C = 454\text{ms}$$

La velocità di trasmissione  $v = 2.5 \frac{\text{Mbit}}{\text{s}}$  e la lunghezza delle trame  $L = 3750\text{byte}$ .

Determinare l'evoluzione della trasmissione.

#### 3.1.1 Risoluzione

Il tempo di trama è:

$$T = \frac{L}{v} = \frac{3750 \cdot 8\text{bit}}{2.5 \cdot 10^6 \frac{\text{bit}}{\text{s}}} = 12\text{ms}$$

Alla prima collisione:

$$z_A = (4 + 1 + 0) \cdot 1 + 12 = 17\text{ms} \rightarrow 422 + 17 = 439\text{ms}$$

$$z_B = (4 + 1 + 8) \cdot 1 + 12 = 25\text{ms} \rightarrow 430 + 25 = 455\text{ms}$$

Il tentativo di ritrasmissione parte dalla fine della trasmissione.

Alla seconda collisione:

$$z_B = (4 + 5 + 5) \cdot 2 + 12 = 40\text{ms} \rightarrow 467 + 40 = 507\text{ms}$$

$$z_C = (4 + 5 + 4) \cdot 1 + 12 = 25\text{ms} \rightarrow 466 + 25 = 491\text{ms}$$

### 3.2 Esercizio 2 (CSMA persistent)

Ci sono 2 stazioni collegate da un mezzo condiviso, con ritardo di propagazione  $\tau = 2\text{ms}$ . La velocità di trasmissione è  $v = 2.5 \frac{\text{Mbit}}{\text{s}}$  e la lunghezza delle trame è  $L = 3750\text{byte}$ . Il tempo di trama quindi è  $T = 12\text{ms}$ . Le trame generate sono:

$$A \rightarrow 2 \text{ trame} \rightarrow \begin{cases} A \rightarrow t_{A_1} = 225\text{ms} \\ A \rightarrow t_{A_2} = 240\text{ms} \end{cases}$$

$$B \rightarrow 1 \text{ trama} \rightarrow t_B = 228\text{ms}$$

#### 3.2.1 Risoluzione

Alla prima collisione:

$$z_A = (2 + 4 + 0) \cdot 1 + 12 = 18\text{ms} \rightarrow 252 + 18 = 270\text{ms}$$

$$z_B = (2 + 3 + 9) \cdot 1 + 12 = 26\text{ms} \rightarrow 251 + 26 = 277\text{ms}$$



## 4 Esercizi con Bridge (switch a 2 porte)

In questo tipo di esercizi viene sempre riportato il comportamento del bridge che può essere:

- **Store and forward:** prima memorizza completamente la trama nella sua memoria interna, poi trasferisce la trama sull'altra interfaccia e gestisce la trasmissione come se la trama fosse stata generata dal bridge in quel momento. Se la destinazione di una trama non si trova sul lato opposto del bridge, la trama viene memorizzata e poi scartata.
- **Il comportamento nei due segmenti è indipendente:** Ci possono essere trasmissioni contemporanee nei due segmenti perchè il bridge li isola.

### 4.1 Esercizio 1 (Aloha)

Il tempo di propagazione è  $\tau_1 = \tau_2 = 0\text{ms}$  e la velocità di trasmissione è

$$v_1 = v_2 = 1.6 \frac{\text{Mbit}}{\text{s}}$$

La lunghezza delle trame è  $L = 800$  e le trame generate da 3 stazioni sono:

$$\begin{aligned} A &\rightarrow \begin{cases} t_{A_1} = 216\text{ms} \\ t_{A_2} = 256\text{ms} \end{cases} && \text{entrambe dirette a B} \\ B &\rightarrow t_B = 233\text{ms} && \text{diretta ad A} \\ C &\rightarrow t_C = 219\text{ms} && \text{diretta ad A} \end{aligned}$$

#### 4.1.1 Risoluzione

Il tempo di trama è:

$$T = \frac{L}{v} = \frac{800 \cdot 8\text{bit}}{1.6 \cdot 10^6 \frac{\text{bits}}{\text{s}}} = 4\text{ms}$$

Alla prima collisione:

$$\begin{aligned} z_{br} &= (2 + 2 + 0) \cdot 1 + 4 = 8\text{ms} \rightarrow 224 + 8 = 232\text{ms} \\ z_C &= (2 + 1 + 9) \cdot 1 + 4 = 16\text{ms} \rightarrow 223 + 16 = 239\text{ms} \end{aligned}$$

Alla seconda collisione:

$$\begin{aligned} z_{br} &= (2 + 3 + 2) \cdot 2 + 4 = 18\text{ms} \rightarrow 236 + 18 = 254\text{ms} \\ z_B &= (2 + 3 + 3) \cdot 1 + 4 = 12\text{ms} \rightarrow 237 + 12 = 249\text{ms} \end{aligned}$$

Alla terza collisione:

$$\begin{aligned} z_{br} &= (2 + 5 + 3) \cdot 1 + 4 = 14\text{ms} \rightarrow 257 + 14 = 271\text{ms} \\ z_A &= (2 + 5 + 6) \cdot 1 + 4 = 17\text{ms} \rightarrow 260 + 17 = 277\text{ms} \end{aligned}$$

## 4.2 Esercizio 2 (CSMA Persistent)

Ci sono 2 segmenti di rete con una stazione per ogni segmento di rete. Il protocollo utilizzato è CSMA Persistent con tempi di propagazione:

$$\tau_{br-A} = 1\text{ms}$$

$$\tau_{br-B} = 2\text{ms}$$

La lunghezza delle trame è  $L = 1500\text{byte}$  e la velocità di trasmissione è  $v = 1.5 \frac{\text{Mbit}}{\text{s}}$  su entrambi i segmenti. Le trame generate sono:

$$A : \begin{cases} t_{A_1} = 713\text{ms} & \text{diretta a B} \\ t_{A_2} = 715\text{ms} & \text{diretta a B} \end{cases}$$
$$B : \begin{cases} t_{B_1} = 719\text{ms} & \text{diretta a A} \\ t_{B_2} = 730\text{ms} & \text{diretta a A} \end{cases}$$

### 4.2.1 Risoluzione

Il tempo di trama è:

$$T = \frac{L}{v} = \frac{1500 \cdot 8\text{bit}}{1.5 \cdot 10^6 \frac{\text{bit}}{\text{s}}} = 8\text{ms}$$

Alla collisione tra  $A_1$  e  $B_1$ :

$$z_{A_1} = (7 + 1 + 4) \cdot 1 + 8 = 20\text{ms} \rightarrow 722 + 20 = 742\text{ms}$$

$$z_{B_1} = (7 + 1 + 9) \cdot 1 + 8 = 25\text{ms} \rightarrow 727 + 25 = 752\text{ms}$$

Alla collisione tra  $A_2$  e  $B_2$ :

$$z_{A_2} = (7 + 2 + 2) \cdot 1 + 8 = 19\text{ms} \rightarrow 730 + 19 = 749\text{ms}$$

$$z_{B_2} = (7 + 3 + 0) \cdot 1 + 8 = 18\text{ms} \rightarrow 738 + 18 = 756\text{ms}$$

Alla seconda collisione tra  $A_2$  e  $B_1$ :

$$z_{A_2} = (7 + 5 + 0) \cdot 2 + 8 = 32\text{ms} \rightarrow 758 + 32 = 790\text{ms}$$

$$z_{B_1} = (7 + 5 + 2) \cdot 2 + 8 = 36\text{ms} \rightarrow 760 + 36 = 796\text{ms}$$