

# Reti di Calcolatori



**Prof. Roberto Canonico**

**Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione**

**Corso di Laurea in Ingegneria Informatica**

## Trasmissione di flussi multimediali in Internet

**I lucidi presentati al corso sono uno strumento didattico  
che NON sostituisce i testi indicati nel programma del corso**



## Nota di Copyright

Questo insieme di trasparenze è stato ideato e realizzato dai ricercatori del Gruppo di Ricerca COMICS del Dipartimento di Informatica e Sistemistica dell'Università di Napoli Federico II. Esse possono essere impiegate liberamente per fini didattici esclusivamente senza fini di lucro, a meno di un esplicito consenso scritto degli Autori. Nell'uso dovranno essere esplicitamente riportati la fonte e gli Autori. Gli Autori non sono responsabili per eventuali imprecisioni contenute in tali trasparenze né per eventuali problemi, danni o malfunzionamenti derivanti dal loro uso o applicazione.

### Autori:

Simon Pietro Romano, Antonio Pescapè, Stefano Avallone,  
Marcello Esposito, Roberto Canonico, Giorgio Ventre

# Trasferimento di informazioni multimediali su rete



- **Problema**: trasferire informazioni multimediali (audio, video, ...) da una sorgente ad uno o più ricevitori attraverso una rete
- Per ridurre la quantità di informazioni trasferita sulla rete, il **trasmettitore** effettua una compressione mediante un'opportuna tecnica (MPEG 1-2-4, MJPEG, MP3, ...)
- Sulla **rete** l'informazione è trasferita **a pacchetti**
- Il **ricevitore** recupera l'informazione originaria dalla sequenza di pacchetti ricevuti, mediante un'operazione inversa a quella di compressione e una successiva trasformazione in forma sonora o in forma di video (sequenza di fotogrammi)

# Trasferimento di informazioni multimediali su rete (2)



- Nel caso di **informazioni *live***, l'informazione è prodotta dalla sorgente mediante un apposito sistema di acquisizione (microfono + scheda audio, telecamera + video capture board), opportunamente compressa (in tempo reale) e trasmessa sulla rete ai ricevitori
- Nel caso di **informazioni pre-registrate**, l'informazione è già registrata in formato compresso (MPEG, MJPEG, MP3, ...) in un file memorizzato su memoria di massa (hard-disk, CDROM, DVD, ...)

# Informazioni multimediali pre-registrate



- **Trasferimento dell'intero file da sorgente a ricevitore e successiva riproduzione: *file transfer***
  - La riproduzione può iniziare solo al termine del trasferimento del file (ritardo proporzionale alla dimensione del file)
  - E' necessaria una adeguata capacità di memorizzazione (su memoria di massa) da parte del ricevitore
  - Questa soluzione è idonea solo per documenti di piccole dimensioni (audio-clip e/o video-clip)
- **Riproduzione progressiva del contenuto multimediale durante il trasferimento dell'informazione: *streaming***
  - Il ricevitore memorizza l'informazione ricevuta in un buffer (*playout buffer*) che viene continuamente alimentato dai dati ricevuti dalla rete e svuotato progressivamente
  - La riproduzione può iniziare non appena il buffer si è "sufficientemente" riempito
  - Il ricevitore non deve memorizzare l'intero file
  - La qualità della riproduzione può degradare se la rete non mantiene la continuità temporale del flusso di informazioni trasmesso dalla sorgente (*sensibilità al jitter*)

# Informazioni multimediali *live*



- Nel caso di informazioni *live*, la sorgente produce un **flusso** continuo di informazioni
- Questo flusso di informazioni è spezzato in **pacchetti** che sono trasmessi individualmente sulla rete: trasmissione in **streaming**

# Sensibilità dello streaming alla QoS



- Il ricevitore riceve i pacchetti, recupera l'informazione originaria e la riconverte in forma audio/video
- Il **ricevitore** riesce a recuperare la **continuità del flusso di informazioni** prodotto dalla sorgente se tutti i pacchetti arrivano a destinazione, con la stessa tempificazione relativa
- La **rete** può alterare la continuità temporale del flusso di informazioni in due modi:
  - Facendo occasionalmente perdere dei pacchetti
  - Consegnando i pacchetti al ricevitore con una tempificazione relativa diversa da quella con cui sono stati trasmessi (*jitter*)
- Perchè la rete possa effettivamente supportare la trasmissione di flussi multimediali occorre che alcuni parametri di Qualità del Servizio (QoS) siano soddisfatti
  - Percentuale di perdita di pacchetti, latenza, jitter, ...

# Degradazione introdotta dalla rete



- Gli effetti sono diversi a seconda della natura del media (audio/video), a seconda della tecnica di compressione utilizzata ed a seconda del grado di alterazione introdotto
  - nel caso di flusso audio, vengono percepite dei “disturbi” (*hiccup*s)
  - nel caso di flusso video, si hanno dei disturbi (*glitches*) che possono essere più o meno localizzati nel tempo e nello spazio
- Sia audio che video possono in genere tollerare una parziale degradazione, ma quando si oltrepassano dei valori di soglia l'informazione diventa inintelligibile



# Esempio di distorsione video prodotta da errori di trasmissione

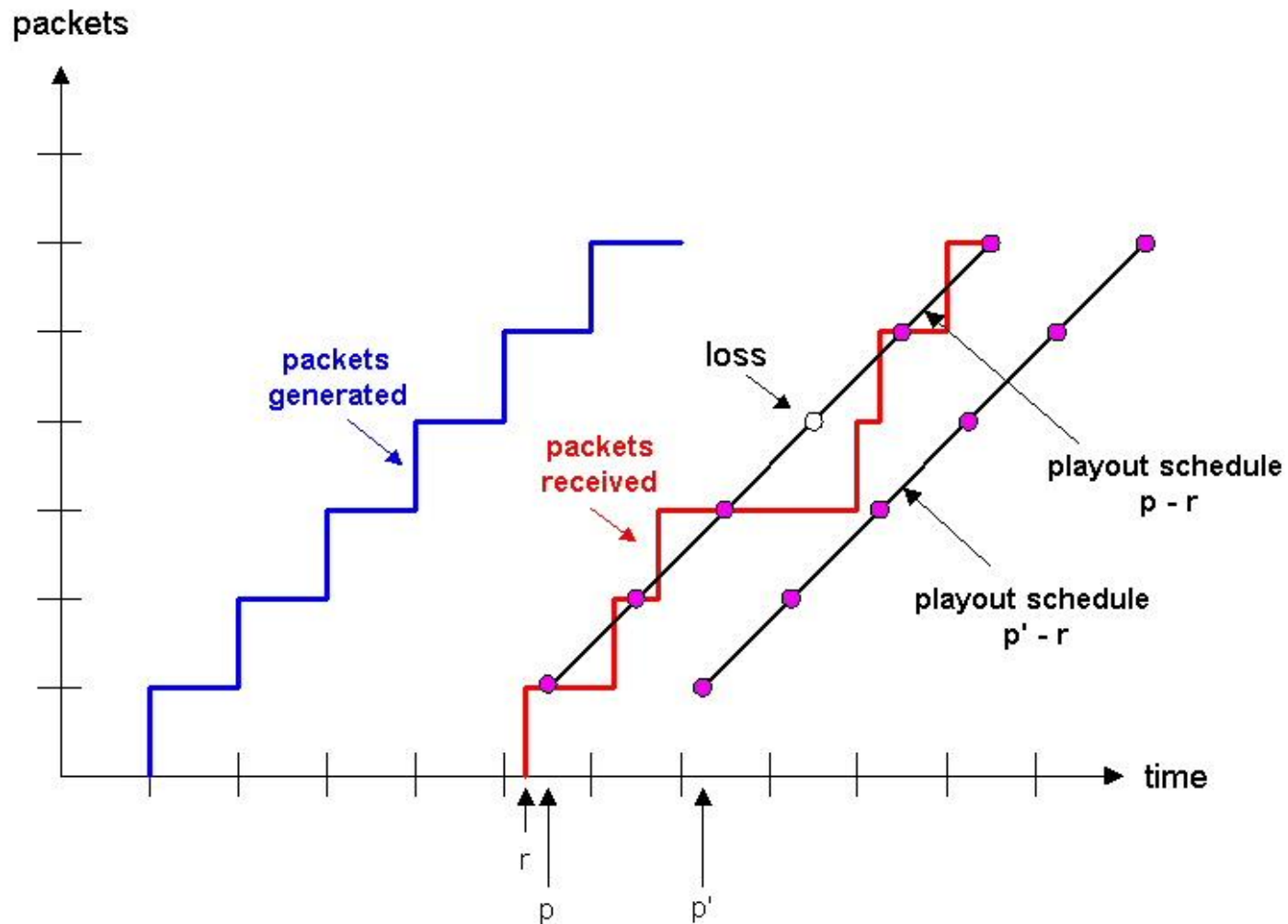




# Contromisure

- Rispetto alla perdita occasionale di pacchetti, ci si difende mediante l'adozione di tecniche di compressione **robuste**, per le quali l'informazione audio/video ricostruita non è sensibilmente degradata quando occasionalmente si perde un pacchetto
  - In alcuni casi si adottano tecniche di *Forward Error Correction* (FEC)
  - L'adozione di tecniche basate sulla ritrasmissione (alla TCP) non sono considerate idonee per lo streaming
- Per limitare gli effetti del *jitter* si adotta una strategia di bufferizzazione: un buffer in ricezione fa da volano e compensa (introducendo un ritardo extra) la variabilità del ritardo di attraversamento della rete
- **NOTA:** non sarebbe necessario introdurre delle contromisure se la rete fosse in grado di offrire servizi a **qualità garantita**
  - **Internet offre un servizio best-effort !**

# Bufferizzazione con ritardo di riproduzione costante



# Bufferizzazione con ritardo di riproduzione costante (2)

