



***Masurarea lungimii de banda  
disponibile intr-o retea.  
Dezvoltarea de module si  
integrarea in MonALISA***

**Coordonator:**

**Prof. Dr. Ing. Valentin Cristea**

**Indrumator:**

**Prof. Dr. Fiz. Iosif Charles Legrand**

**Student:**

**Madalin Mihailescu**



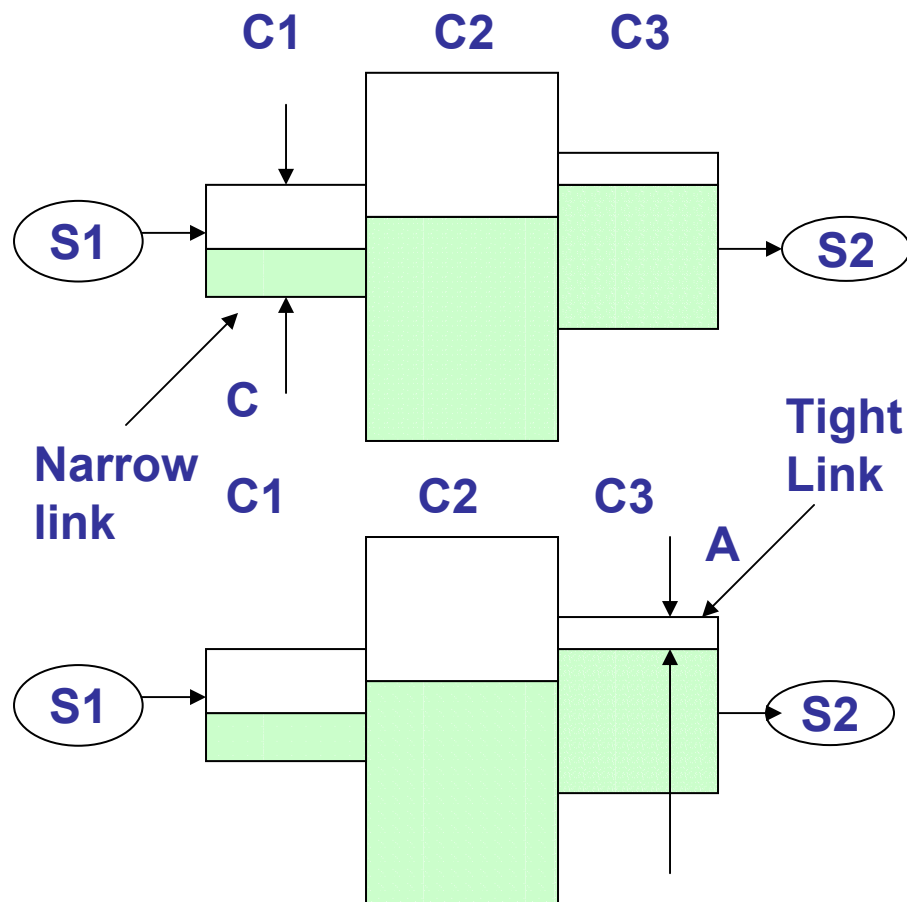
# Introducere

- Monitorizare retea
  - end-to-end
- Available bandwidth
- Probleme
  - Trade off acuratete - volum de date
  - Routeri? Nu: cost, timp -> aplicatie end-to-end



# Definitii

- $A = \min A_i = \min C_i(1-u_i)$
- $C = \min C_i$
- Narrow link
- Tight link





# VRVS

- Videoconferinta
  - marirea calitatii serviciului (calitatea conexiunilor alternative)
- Servere (reflectori)
  - streamuri audio/video
  - Rutare : cost conexiune (MST)
  - Cost : monABPing
    - RTT, jitter, procentul pachetelor pierdute



# SLoPS(1)

- Idee: Delay-urile „one-way” ale unui stream periodic de pachete au un trend crescator cand rata stream-ului este mai mare decat lungimea de banda disponibila
- Cale SND -> RCV cu H link-uri,  $i = 1 \dots H$ .
  - $C_i$  – capacitate,  $A_i$  - avail-bw,  $u_i = (C_i - A_i) / C_i$  – utilizare
- *SloPS cu cross-traffic fluid*
  - Cross trafic - model stationar (invariant in timp)
  - SND -> RCV : stream periodic
    - K pachete de L bytes, rata  $R_0$ , perioada  $T = L/R_0$
  - OWD - diferenta intre delay-urile a doua pachete succesive k si k+1
  - $R_0 > A$  -> cele K pachete ale stream-ului periodic vor sosi la RCV cu OWD-uri crescatoare
  - $R_0 \leq A$  -> pachetele vor avea OWD-uri egale.



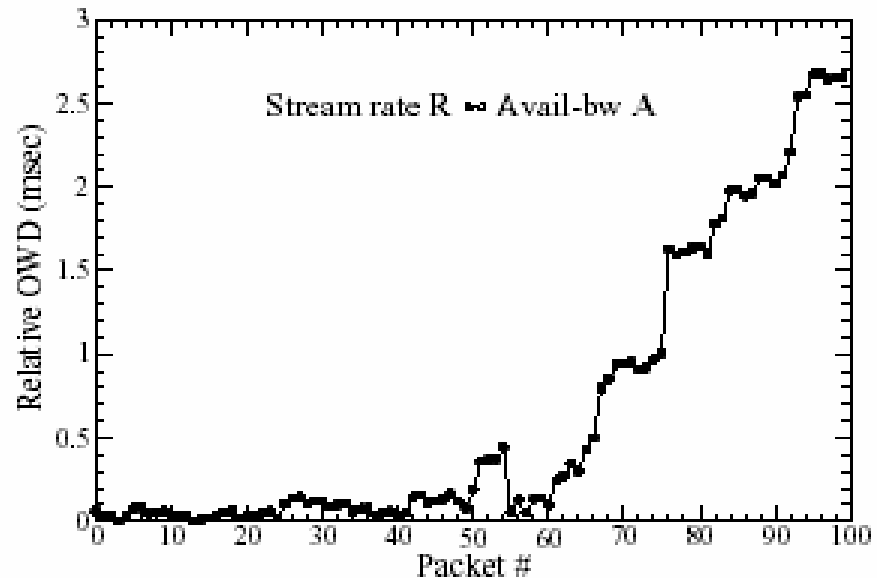
# SLoPS(2)

- Algoritm iterativ pentru masurarea end-to-end a lui A:
  - SND  $\rightarrow$  RCV : stream periodic  $n$  cu rata  $R(n)$
  - RCV analizeaza variatiile OWD ale stream-ului pentru a determina relatia  $R(n)$ ,  $A$
  - RCV  $\rightarrow$  SND relatia  $R(n)$ ,  $A$
  - Daca  $R(n) > A$ , SND  $\rightarrow$  RCV stream  $n+1$  cu rata  $R(n+1) < R(n)$   
Altfel,  $R(n+1) > R(n)$
  - $R(n) > A \rightarrow R_{max} = R(n)$  ;  $R(n) \leq A \rightarrow R_{min} = R(n)$
  - Initial  $R_{min} = 0$ ,  $R_{max}$  - suficient de mare pentru ca  $R_{max} > A$ .
  - $R(n+1) = (R_{max} + R_{min})/2$



# SLoPS(3)

- *SloPS cu cross-traffic real*
- La trafic fluid - A constanta pe timpul masurarii
- 2 modificari ale metodei:
  - prezenta unui trend crescator general pentru intreg stream-ul
  - A variaza in jurul lui R
  - nu este o ordine stricta intre R si A
  - > „regiunea gri”.





# Implementare(1)

- Client-server - java NIO
  - Thread pentru accept
  - ThreadPool pentru mesaje de control
- UDP – stream-urile periodice de pachete
- TCP – conexiune mesaje control
- JNI
- Monitoring Module
- AppControl



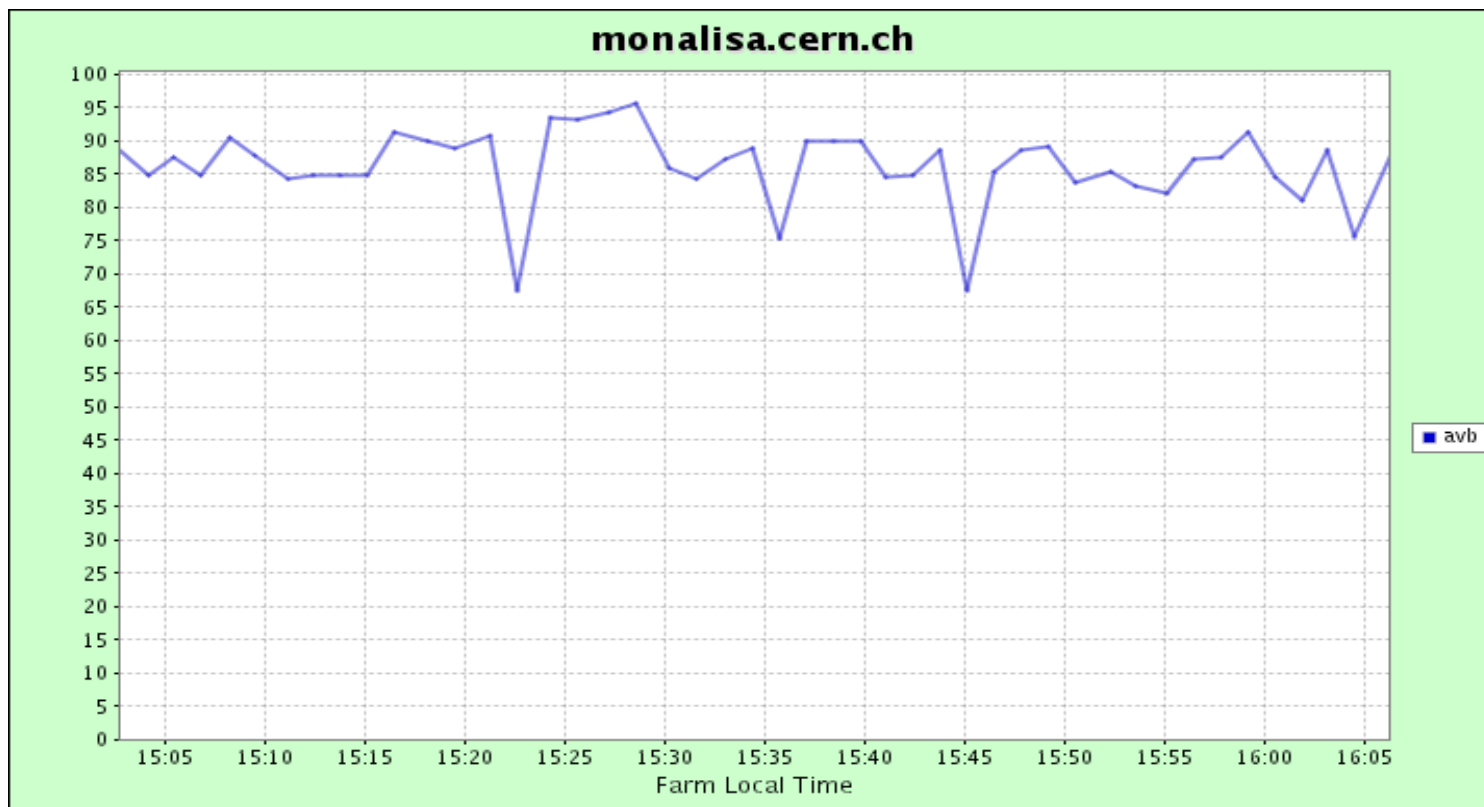


# Implementare(2)

- Sincronizare ceas SND-RCV
- Trenuri de streamuri
- Regiunea gri
- Detectie context-swicth
  - Sender
  - Receiver



# Teste si analiza



- Legatura – 11 hop-uri; Trafic generat: 10 M/1M; Timp: ~15s/100Mbps
- Paralel cu iperf -> scadere

# Concluzii si dezvoltari ulterioare



- Integrare in MonALISA -> mai mult decat o aplicatie end-to-end
- Criteriu VRVS: available bandwidth
- Imbunatatiri:
  - Simulare trenuri de pachete - semnale
  - Functie de transfer ( L , A )
  - Output real vs Output calculat



# MULTUMESC!

Madalin Mihailescu [[madalin@cs.pub.ro](mailto:madalin@cs.pub.ro)]