

## Performance Modeling of Computer Systems and Networks

*Prof. Vittoria de Nitto Personè*

### Priority scheduling

Università degli studi di Roma Tor Vergata  
Department of Civil Engineering and Computer Science Engineering

Copyright © Vittoria de Nitto Personè, 2021  
<https://creativecommons.org/licenses/by-nc-nd/4.0/>



1

Analytical models  
*priority scheduling*

### Service classes

- (Multimedia traffic)
- Quality of Service (QoS)
- Penalties

The proper scheduling policy can improve  
performance of a server tremendously.  
It costs nothing to alter your scheduling policy  
(no money, no new hardware), so the  
performance gain comes for free.

prof. Vittoria de Nitto Personè

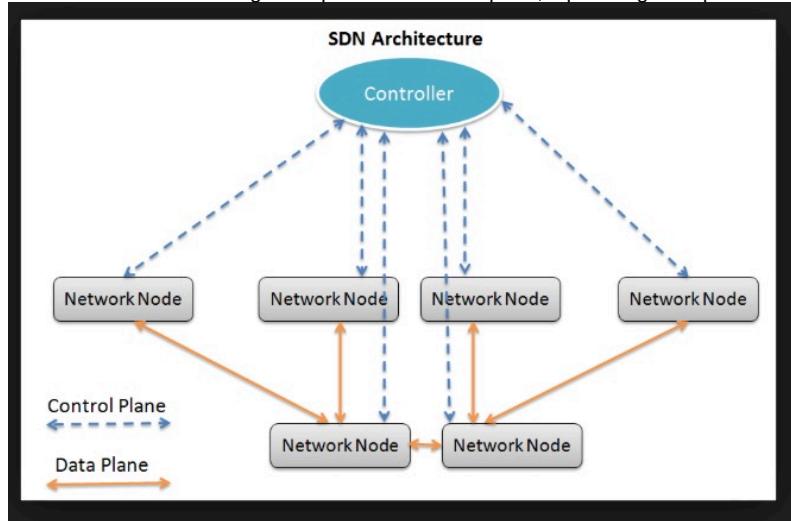
2

2

1

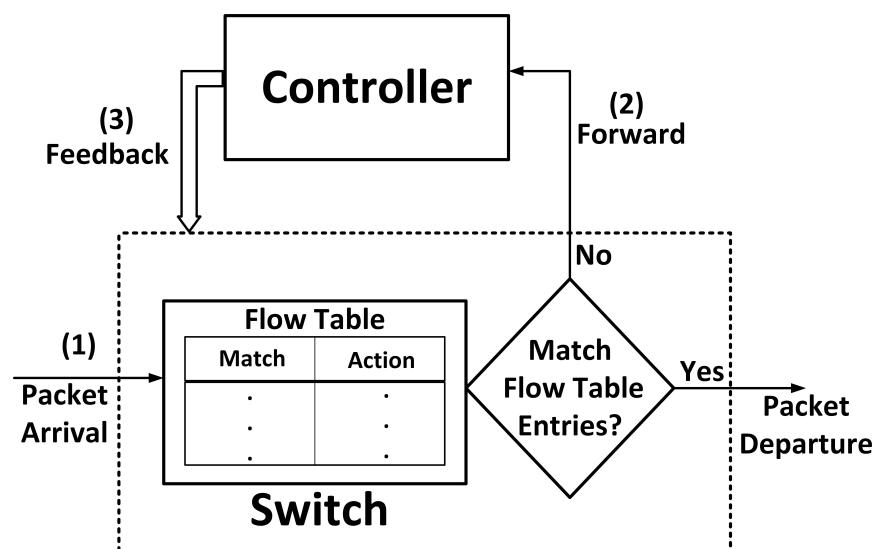
## SDN

architettura per la realizzazione di reti di telecomunicazioni  
il piano di controllo della rete e quello di trasporto dei dati sono  
separati logicamente  
distingue tra piano fisico di trasporto, e parte logica di pacchetti.



3

Se da (1) arriva pacchetto non presente in tabella, lo mando al controllore (2), che fa operazioni di aggiornamento tabella, rimandandolo allo switch (3). Quando lo rimanda allo switch ha priorità maggiorata (poiché non essendo inviato ha accumulato ritardo).



prof. Vittoria de Nitto Personè

4

4

2

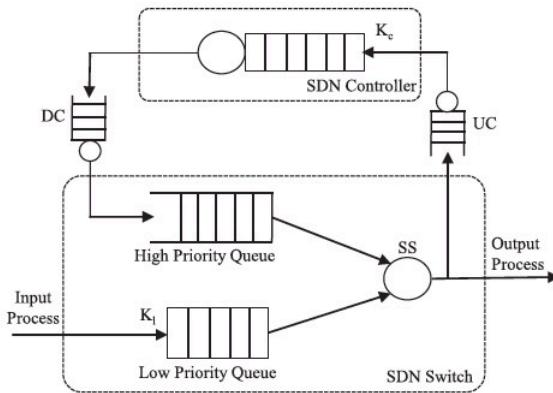


Fig. 1. The PQ-based SDN system architecture.

Miao W., Min G., Wu Y., Wang H. and Hu J., 2016. Performance Modelling and Analysis of Software-Defined Networking under Bursty Multimedia Traffic. *ACM Trans. Multimedia Comput. Commun. Appl.*, Vol. 12, No. 5s, Article 77.

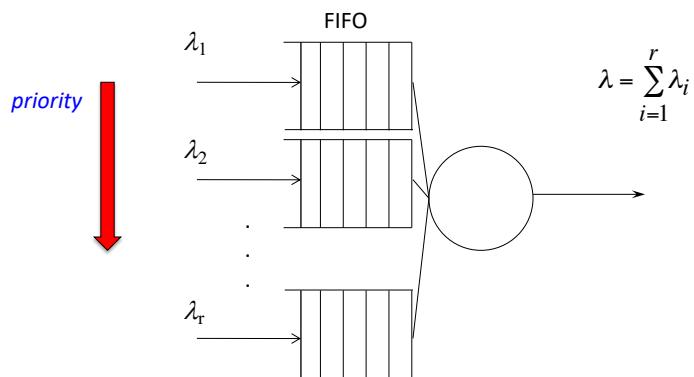
prof. Vittoria de Nitto Personè

5

5

Analytical models  
priority scheduling

## Priority classes



Coda 1 priorità max, coda 'r' priorità minima. Il flusso totale lamda è la somma di questi 'r' flussi, e il servente qui è SINGOLO.

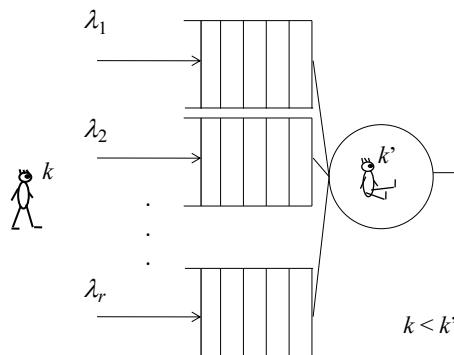
prof. Vittoria de Nitto Personè

6

6

3

## Priority classes



*abstract*

(non dipende da quanta è la size del job)

*size based*

(danno ordine in base alla quantità di lavoro chiesto)

*preemptive*

servizio interrompibile o meno!

*not-preemptive*

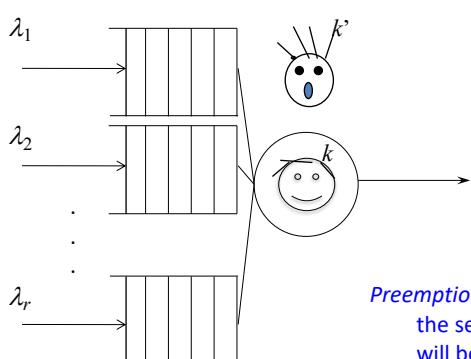
In servizio c'è "k'", questo vuol dire che tutte le code da 1 a "k' - 1" sono vuote, sennò toccava a loro in quell'istante. Se arriva 'k' più importante, e c'è PRELAZIONE, interrompo k' in favore di k.

prof. Vittoria de Nitto Personè

7

7

## Priority classes with preemption



*Preemption without loss:*

the service of the interrupted job  
will be resumed from the  
interruption point

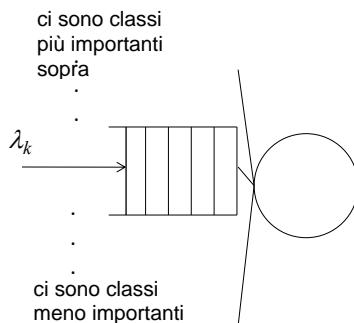
interrompo solo se posso salvare stato  
e riprenderlo da istante interruzione.  
SI parla di "interruzione senza perdita".

prof. Vittoria de Nitto Personè

8

8

## Abstract priority without preemption



$S_k$  in "generale"

$$E(S_k) = \frac{1}{\mu_k} \quad \sigma^2(S_k)$$

$$\rho_k = \lambda_k E(S_k) \quad \rho = \sum_{i=1}^r \rho_i \text{ totale}$$

abstract

tempo di servizio uguale per tutti.

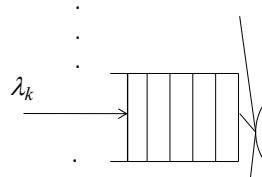
$$E(S_k) = E(S) = \frac{1}{\mu}, \quad \sigma^2(S_k) = \sigma^2(S), \forall k$$

Nel caso size-based,  
la coda-k ha un certo tempo di risposta  
diverso da altre code (ordino in base alla size).

quanto arriva\* quanto si ferma

## Abstract priority without preemption

local performance measures



$$E(T_{Q_k})?$$

Arrival of  $u_k$

$t_a$

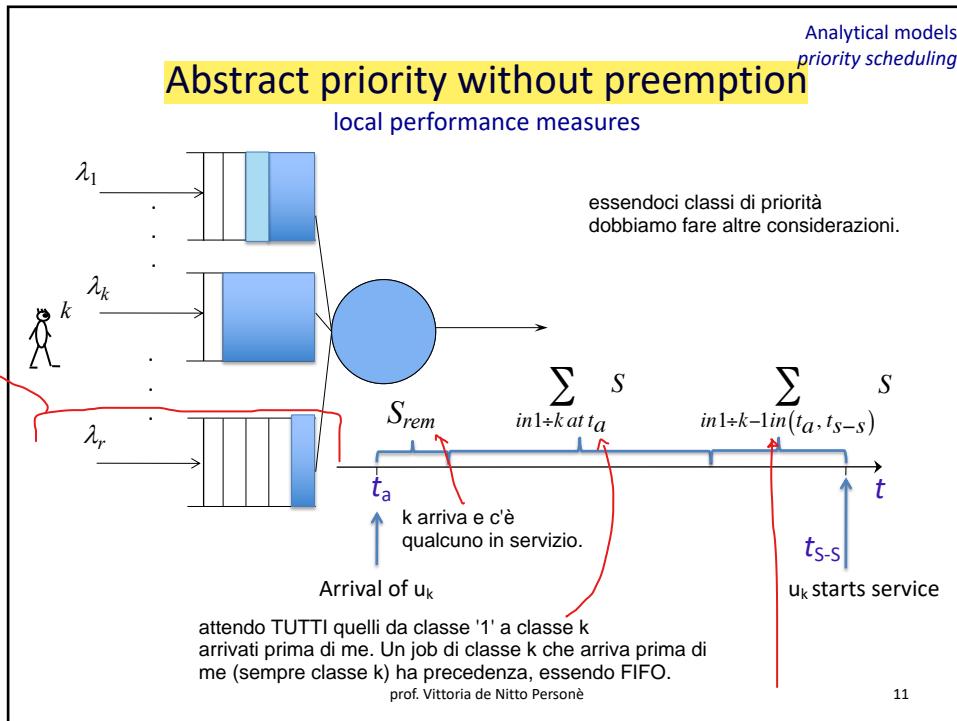
$u_k$  Starts service

$t$

## Abstract priority without preemption

local performance measures

per "k", quello che c'è sotto non interessa.



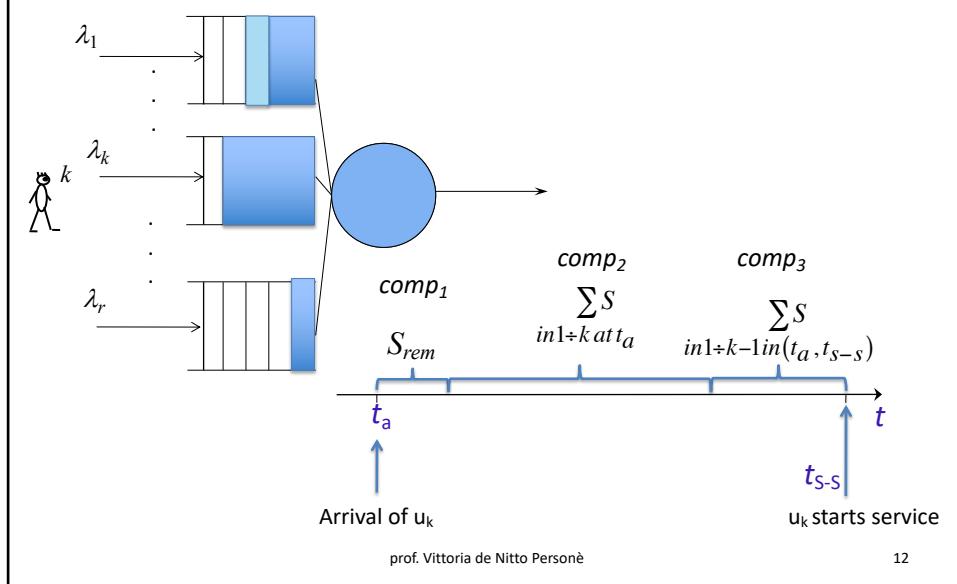
attendo i job che sono più importanti (classi 1 a  $k-1$ ) di me, ma che sono arrivati dopo di me e prima che io prenda servizio. Un job di classe  $k$  che arriva dopo di me (io sono classe  $k$ ) non ha precedenza, essendo FIFO.

11

11

## Abstract priority without preemption

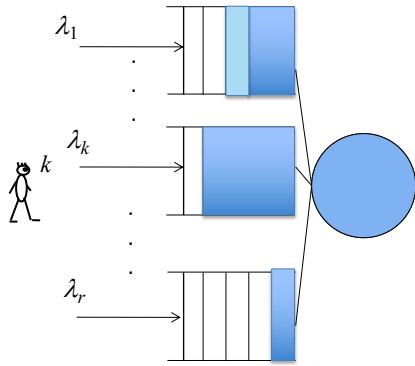
local performance measures



12

## Abstract priority without preemption

local performance measures



$$comp_1: E(S_{rem}) = \frac{\lambda}{2} E(S^2)$$

$$comp_2: \text{proportional to the load of queues } 1:k = \frac{1}{1 - \sum_{i=1}^k \rho_i}$$

nel caso singolare era:  $\frac{1}{1 - \rho}$

$$comp_3: \text{proportional to the load of queues } 1:k-1 = \frac{1}{1 - \sum_{i=1}^{k-1} \rho_i}$$

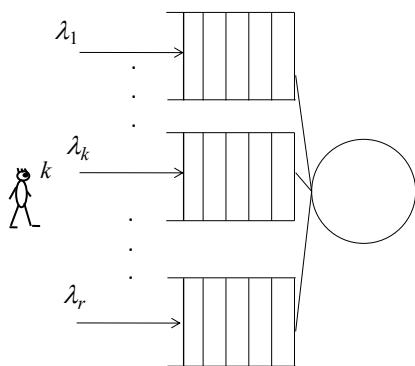
prof. Vittoria de Nitto Personè

13

13

## Abstract priority without preemption

local performance measures



$$E(T_{Q_k})^{NP\_priority} = \frac{\lambda}{2} E(S^2) / \left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k-1} \rho_i\right)$$

classi di priorità senza prelazione!

$$E(T_{Q_k}) \leq E(T_{Q_{k+1}})$$

abbastanza banale.

prof. Vittoria de Nitto Personè

14

14

## Abstract priority without preemption

local performance measures

$$E(T_{Q_k}) \leq E(T_{Q_{k+1}})$$

$$\frac{\frac{\lambda}{2} E(S^2)}{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k-1} \rho_i\right)} \leq \frac{\frac{\lambda}{2} E(S^2)}{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k+1} \rho_i\right)}$$

$$\frac{1}{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k-1} \rho_i\right)} \leq \frac{1}{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k+1} \rho_i\right)}$$

prendo solo i denominatori:

$$\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k-1} \rho_i\right) \geq \left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k+1} \rho_i\right)$$

## Abstract priority without preemption

local performance measures

$$\cancel{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k-1} \rho_i\right)} \geq \cancel{\left(1 - \sum_{i=1}^k \rho_i\right) \left(1 - \sum_{i=1}^{k+1} \rho_i\right)}$$

$$\cancel{1 - \sum_{i=1}^{k-1} \rho_i} \geq \cancel{1 - \sum_{i=1}^{k+1} \rho_i}$$

$$\sum_{i=1}^{k+1} \rho_i \geq \sum_{i=1}^{k-1} \rho_i \quad \rho_i \geq 0, \forall i$$

$$E(T_{Q_k}) \leq E(T_{Q_{k+1}})$$

## Abstract priority without preemption

local performance measures

Prestazioni locali, rispetto alla classe

indipendente dalla classe

$$E(T_{S_k}) = E(T_{Q_k}) + E(S) \quad E(T_{S_k}) \leq E(T_{S_{k+1}})$$

usando Little:

$$E(N_{Q_k}) = \lambda_k E(T_{Q_k})$$

$$E(N_{S_k}) = \lambda_k E(T_{S_k}) \quad E(N_{S_k}) = E(N_{Q_k}) + \rho_k$$

rho specifico  
della classe

prof. Vittoria de Nitto Personè

17

17

## Abstract priority without preemption

global performance measures

And the “global” performance? Devo fare una media pesata.

$$E(T_Q)^{NP\_priority} = E(E(T_{Q_k})) = \sum_{k=1}^r p_k E(T_{Q_k})$$

probabilità della classe k rispetto alla totale

$$p_k = \frac{\lambda_k}{\lambda}$$

% di traffico su una coda / tutto il traffico

and similarly for  $E(T_S)^{NP\_priority}$

$$E(T_S)^{NP\_priority} = E(T_Q)^{NP\_priority} + E(S)$$

prof. Vittoria de Nitto Personè

18

18

## Abstract priority without preemption

probabilità

$$\lambda_k = p_k \lambda$$

rho:  $\rho_k = \lambda_k E(S) = p_k \lambda E(S) = p_k \rho$   
 rho\_k = probabilità di essere di quella classe \* rho\_totale

prof. Vittoria de Nitto Personè

19

19

## priority vs no-priority

How are the performance improved in respect of a simple abstract scheduling not considering the priority classes? quante classi vanno meglio? quali peggio?

$$E(T_{Q_k})^{NP\_priority} = \frac{\frac{\lambda}{2} E(S^2)}{\left(1 - \sum_{i=1}^k \rho_i\right)\left(1 - \sum_{i=1}^{k-1} \rho_i\right)} \quad ? \quad E(T_Q)^{KP} = \frac{\frac{\lambda}{2} E(S^2)}{1 - \rho}$$

The highest priority class:

$$E(T_{Q_1})^{NP\_priority} = \frac{\frac{\lambda}{2} E(S^2)}{(1 - \rho_1)} \leq E(T_Q)^{KP} \quad \smiley$$

prima classe va sicuramente meglio (vede solo se stessa), meglio rispetto a vedere tutti mischiati.  
 rho\_1 è più grande di rho???

prof. Vittoria de Nitto Personè

20

20

## priority vs no-priority

How are the performance improved in respect of a simple abstract scheduling not considering the priority classes?

The lowest priority class:

$$E(T_{Qr})^{NP\_priority} = \frac{\frac{\lambda}{2} E(S^2)}{(1-\rho)(1 - \sum_{i=1}^{r-1} \rho_i)} \geq E(T_Q)^{KP}$$


And what about the "global" performance?

$$E(T_Q)^{NP\_priority} = E(T_Q)^{KP}$$

↓

se vado a fare somme pesate,  
non ho vantaggi. Le classi più basse  
annullano i vantaggi delle classi più alte.

$$E(T_S)^{NP\_priority} = E(T_S)^{KP}$$

prof. Vittoria de Nitto Personè

21

21

Se pensiamo a KP, scheduling astratti. Ma anche NP\_priority ha uno scheduling astratto, e quindi globalmente non è nient'altro che una KP.

## priority vs no-priority

$$E(T_Q)^{NP\_priority} = E(E(T_{Qk})) = \sum_{k=1}^r p_k E(T_{Qk}) = E(T_Q)^{KP}$$

$$E(T_Q) = p_1 E(T_{Q1}) + p_2 E(T_{Q2}) = p_1 \frac{\frac{\lambda}{2} E(S^2)}{(1-\rho_1)} + p_2 \frac{\frac{\lambda}{2} E(S^2)}{(1-\rho)(1-\rho_1)}$$

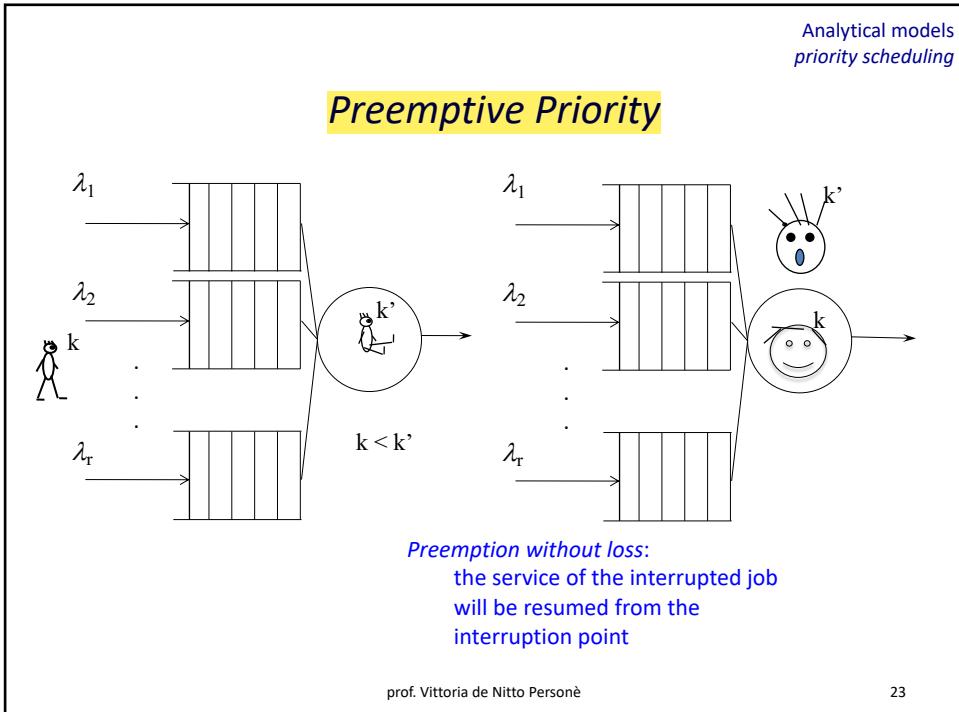
$$= \frac{\lambda}{2} E(S^2) \left[ \frac{p_1}{(1-\rho_1)} + \frac{p_2}{(1-\rho)(1-\rho_1)} \right] = \frac{\lambda}{2} E(S^2) \frac{p_1(1-\rho) + p_2}{(1-\rho)(1-\rho_1)} = \frac{\frac{\lambda}{2} E(S^2)}{1-\rho}$$

$$\begin{aligned} & p_1 - p_1 * \cancel{\rho} + p_2 \\ & \downarrow \\ & p_1 + \cancel{\rho_1} + p_2 = 1 + \cancel{\rho_1}, \text{ si annulla con denominatore.} \end{aligned}$$

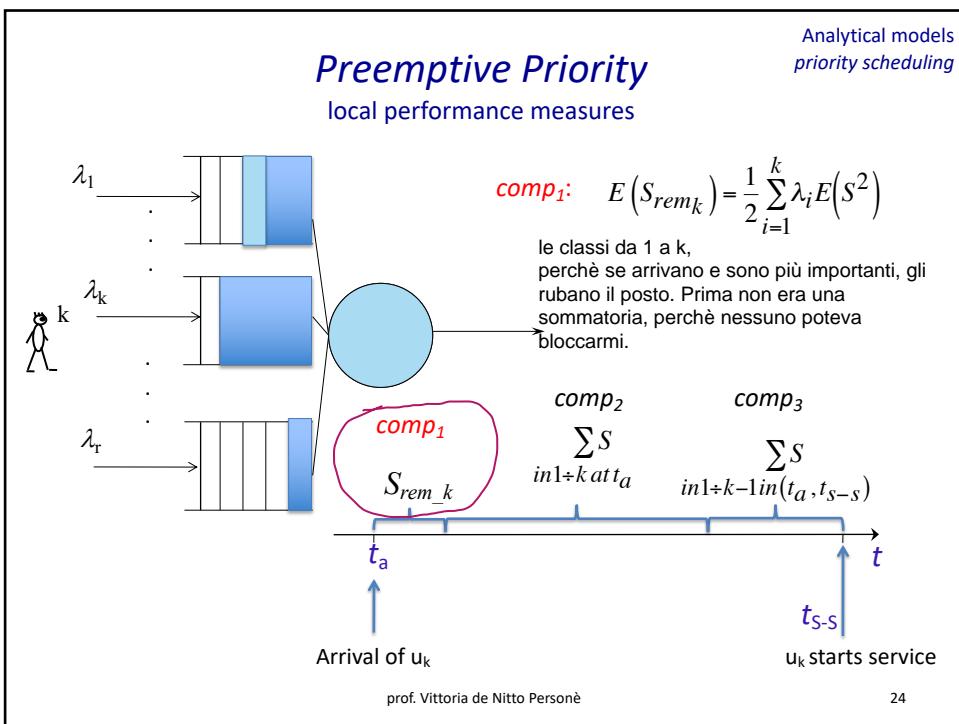
prof. Vittoria de Nitto Personè

22

22



23



24

Analytical models  
priority scheduling

### Preemptive Priority

quando non c'è prelazione, c'è tutto lambda.

$$E(T_{Q_k})^{P\_priority} = \frac{\frac{1}{2} \sum_{i=1}^k \lambda_i E(S^2)}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i)}$$

$$E(T_{Q_k})^{P\_priority} \leq E(T_{Q_{k+1}})^{P\_priority}$$

$$E(T_{Q_k})^{P\_priority} \leq E(T_{Q_k})^{NP\_priority}$$

$$E(T_Q)^{P\_priority} \leq E(T_Q)^{NP\_priority} = E(T_Q)^{KP}$$

per P\_priority ho al numeratore una sommatoria di lambda da 1 a k.

per NP\_priority ho tutto lambda, quindi è sempre maggiore del primo caso (sono uguali se prendo l'ultima classe di importanza).

prof. Vittoria de Nitto Personè      25

25

Ho guadagnato qualcosa per alcune classi, perché il tempo con prelazione globale è minore uguale rispetto alla variante senza prelazione. Ho finalmente guadagnato sulla KP. Nel modello SENZA Prelazione, se c'è in servizio 'k', e arriva un 'k-1', deve aspettare. Qui, con prelazione, sostituisco subito, quindi posso dire ancora meglio che NON VEDO CLASSI DI PRIORITÀ INFERIORI. Per questo c'è guadagno.

Analytical models  
priority scheduling

### Preemptive Priority

$$\frac{\frac{1}{2} \sum_{i=1}^k \lambda_i E(S^2)}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i)} \leq \frac{\frac{1}{2} \sum_{i=1}^{k+1} \lambda_i E(S^2)}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k+1} \rho_i)}$$

$$\frac{\sum_{i=1}^k \lambda_i}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i)} \leq \frac{\sum_{i=1}^{k+1} \lambda_i}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k+1} \rho_i)}$$

$$\frac{\sum_{i=1}^k \lambda_i}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i)} \leq \frac{\sum_{i=1}^{k+1} \lambda_i}{(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i)}$$

$$(1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k-1} \rho_i) \geq (1 - \sum_{i=1}^k \rho_i)(1 - \sum_{i=1}^{k+1} \rho_i)$$

stai sereno che lo vedo

prof. Vittoria de Nitto Personè      26

26

## Preemptive Priority

$$E(T_Q)^{X\_priority} = E(E(T_{Q_k})) = \sum_{k=1}^r p_k E(T_{Q_k}) \\ = p_1 E(T_{Q_1}) + p_2 E(T_{Q_2}) + \dots + p_r E(T_{Q_r})$$

$$E(T_Q)^{NP\_priority} = p_1 E(T_{Q_1}) + p_2 E(T_{Q_2}) + \dots + p_r E(T_{Q_r}) \\ = p_1 E(T_{Q_1}) + p_2 E(T_{Q_2}) + \dots + p_r E(T_{Q_r})$$

$$E(T_Q)^{P\_priority} \leq E(T_Q)^{NP\_priority} = E(T_Q)^{KP}$$

prof. Vittoria de Nitto Personè

27

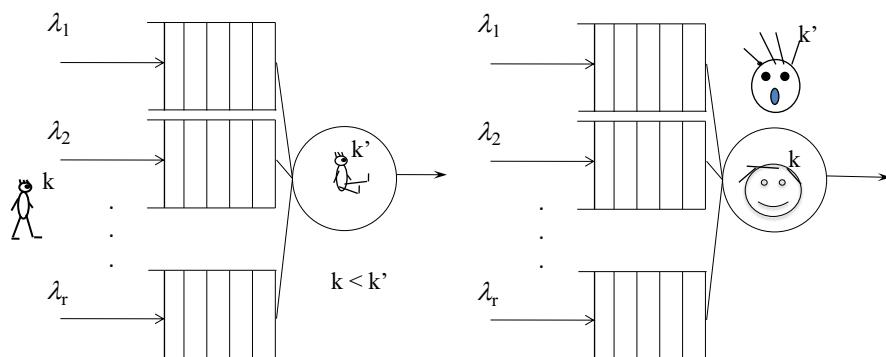
27

Se parliamo di servizio, un job che viene buttato fuori e poi ritorna, perde più tempo. La sua classe è sempre k, quindi perde molto tempo.  
Tempo attesa: tempo arrivo - tempo presa servizio. Adesso specifichiamo "tempo presa servizio" con "tempo presa servizio la prima volta".

## Preemptive Priority

Analytical models  
priority scheduling

Due to preemption, the service time of a job of class k may increase for the arrivals of higher priority classes



prof. Vittoria de Nitto Personè

28

28

## Preemptive Priority

Due to preemption, the service time of a job of class  $k$  may increase for the arrivals of higher priority classes

Essere buttato fuori dal servizio perchè arriva un job prioritario, rallenta il mio tempo di servizio.  
Lo posso immaginare come  $E[S] +$  attese per le interruzioni

*Virtual service time*  $E(S_{virt\_k})$

$$E(S_{virt\_k}) = \frac{E(S)}{1 - \sum_{i=1}^{k-1} \rho_i}$$

viene allungato di un fattore proporzionale a tutti i rho delle classi che mi possono interrompere. La sommatoria arriva a "k-1", una classe "k" non può infatti essere interrotta da sè stessa.

$$E(T_{S_k})^{P\_priority} = E(T_{Q_k})^{P\_priority} + E(S_{virt\_k})$$

AI                          IV

$$E(T_{S_k})^{NP\_priority} = E(T_{Q_k})^{NP\_priority} + E(S)$$

prof. Vittoria de Nitto Personè

29

29 Con priority miglioro l'attesa in coda, ma peggioro il tempo di servizio.

Non posso dire a priori chi sia più grande, c'è chi va meglio e chi peggio.  
Per la classe 1,  $E(S_{virt\_k}) = E[S]$ , perchè classe 1 non può essere buttata fuori.  
A parte classe 1, le altre non posso dire nulla: quante classi ci sono? quanto sono grandi i flussi?

## Preemptive Priority

global response time

$$E(T_{Q_k})^{P\_priority} + E(S_{virt\_k})$$

And what about the "global" performance?

$$E(T_S)^{P\_priority} = E(E(T_{S_k})) = \sum_{k=1}^r p_k E(T_{S_k})$$

$$\begin{aligned} E(T_S)^{P\_priority} &= \sum_{k=1}^r p_k [E(T_{Q_k}) + E(S_{virt\_k})] \\ &= \sum_{k=1}^r p_k E(T_{Q_k}) + \sum_{k=1}^r p_k E(S_{virt\_k}) \\ &= E(T_Q)^{P\_priority} + \sum_{k=1}^r p_k E(S_{virt\_k}) \end{aligned}$$

non è più  $E[S]$ , lo è solo per la classe 1.

prof. Vittoria de Nitto Personè

30

30

## preemption vs no-preemption

$$E(T_S)^{P\_priority} = E(T_Q)^{P\_priority} + \sum_{k=1}^r p_k E(S_{virt\_k})$$

\wedge \quad \quad \quad \vee

?

$$E(T_S)^{NP\_priority} = E(T_Q)^{NP\_priority} + E(S) = E(T_S)^{KP}$$

anche in termini globali non posso confrontarli, perchè per i singoli "pezzi" posso dire chi è più piccolo, ma mettendoli insieme non so dire se ho un guadagno o meno.

In general

$$E(T_S)^{P\_priority} \quad ? \quad E(T_S)^{KP}$$

For exponential service time

$$E(T_S)^{P\_priority} = E(T_S)^{KP} \quad !!!$$

La memoryless annulla tutto il guadagno, ovvero ciò che guadagno in attesa è compensato da ciò che perdo nel servizio.

prof. Vittoria de Nitto Personè

31

31

## preemption vs no-preemption

$r=2$

tutto viene dall'ESPONENZIALITA'.

$$\begin{aligned} E(T_S)^{P\_priority} &= p_1 E(T_{S1}) + p_2 E(T_{S2}) \\ &= p_1 \left[ \frac{\lambda_1}{2} \frac{E(S^2)}{(1-\rho_1)} + E(S) \right] + p_2 \left[ \frac{\lambda}{2} \frac{E(S^2)}{(1-\rho)(1-\rho_1)} + \frac{E(S)}{1-\rho_1} \right] \end{aligned}$$

tempo virtuale  
prima classe,  
non ritardabile  
seconda classe

from the expo assumption

$$\begin{aligned} &= p_1 \left[ \frac{\rho_1 E(S)}{(1-\rho_1)} + E(S) \right] + p_2 \left[ \frac{\rho E(S)}{(1-\rho)(1-\rho_1)} + \frac{E(S)}{1-\rho_1} \right] \\ &= E(S) \left\{ p_1 \left[ \frac{\rho_1 + 1 - \rho_1}{(1-\rho_1)} \right] + p_2 \left[ \frac{\rho + (1-\rho)}{(1-\rho)(1-\rho_1)} \right] \right\} \end{aligned}$$

prof. Vittoria de Nitto Personè

32

32

## preemption vs no-preemption

$r=2$

$$\begin{aligned} E(T_S)^{P\_priority} &= p_1 E(T_{S1}) + p_2 E(T_{S2}) \\ &= E(S) \left[ \frac{p_1}{(1-\rho_1)} + \frac{p_2}{(1-\rho)(1-\rho_1)} \right] \\ &= E(S) \frac{p_1(1-\rho) + p_2}{(1-\rho)(1-\rho_1)} = \frac{E(S)}{1-\rho} = E(T_S)^{KP} \end{aligned}$$

La proprietà Memoryless semplifica moltissimo i fenomeni.

Priorità astratte: criteri indipendenti da quanto chiedono di servizio.

Tempi di attesa in coda:

Se c'è prelazione, ho vantaggi locali e globali su  $E[Tq]$  rispetto alla NonPreemptive e rispetto a KP.  
Senza prelazione, localmente ho vantaggi tra due classi, ma globalmente ciò si traduce in vantaggio per una classe e svantaggio per un'altra, ottenendo stesse prestazioni della KP.

prof. Vittoria de Nitto Personè

33

33

Tempi di risposta:

Sia preemptive che Non, sia locale che non, non posso mai dire con certezza se preemptive vada meglio, anche se è meglio rispetto  $E[Tq]$  per questo vantaggio nel calcolo del tempo di servizio.  
(sia localmente che globalmente).

Con tempo esponenziale torno globalmente al caso della KP.

20/04/2023 osservazione: perché l'occupazione delle classi si vede usando i "rho"?

Il tempo di servizio rimanente È GENERALMENTE  $E[S_{rim}] = \frac{\lambda * E[S^2]}{2}$ .

Abbiamo visto, per la classe k, che  $E[Tq\_k] =$

$$E[S_{rim}] \cdot \left( 1 - \sum_{i=1}^k p_i \right) \cdot \left( 1 - \sum_{i=1}^{k-1} p_i \right)$$

Quante classi ci guadagnano? cioè quante volte questa media del tempo di attesa è  $\frac{E[S_{rim}]}{1-\rho}$ ?

ovvero caso coda unica KP. Stiamo lavorando nel caso senza prelazione.

Tutto si riduce a vedere il denominatore, ovvero:

$$\left( 1 - \sum_{i=1}^k p_i \right) \left( 1 - \sum_{i=1}^{k-1} p_i \right) > 1 - \rho = 1 - \sum_{i=1}^k p_i = 1 - \left( \sum_{i=1}^k p_i + \sum_{i=k+1}^n p_i \right)$$

Ho riscritto prima rho come somma di tutte le r classi, poi ho diviso questa ultima sommatoria in due componenti.  
Svolgendo il prodotto delle componenti a sinistra otteniamo:

$$1 - \sum_{i=1}^k p_i - \cancel{\sum_{i=1}^k p_i} + \sum_{i=1}^k p_i \cdot \sum_{i=1}^{k-1} p_i > 1 - \sum_{i=1}^k p_i - \underbrace{\sum_{i=k+1}^n p_i}_{< 1}$$

Porto la sommatoria in verde a sinistra, e tutto il resto a destra: sto confrontando a sinistra le ultime  $k+1$  classi con quelle più importanti.

$$\sum_{i=k+1}^n p_i > \sum_{i=1}^{k-1} p_i - \sum_{i=1}^k p_i \cdot \sum_{i=1}^{k-1} p_i = \sum_{i=1}^{k-1} p_i \left( 1 - \underbrace{\sum_{i=1}^k p_i}_{< 1} \right)$$

se livello occupazione delle classi più basse è > livello occupazione classi più alte, allora le prestazioni delle prime  $k$  classi sono migliori. La parte in rosso è  $< 1$ , quindi stiamo parlando in "percentuale", perchè questo valore è moltiplicato per le prime  $k-1$  classi. Tanto più sono le "k classi" considerate, tanto più quelle sotto vengono svantaggiate.