

6/04/2023

## Performance Modeling of Computer Systems and Networks

Prof. Vittoria de Nitto Personè

### The multi-server queue

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  
the multiserver queue

### Erlang, 1917 M/M/m abstract scheduling

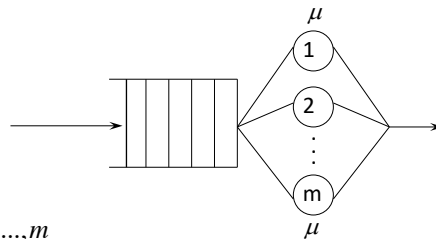
$E(N_Q)_{Erlang}$

$$p(n) = \begin{cases} \frac{1}{n!} (m\rho)^n p(0) & \text{for } n = 1, \dots, m \\ \frac{m^m}{m!} \rho^n p(0) & \text{for } n > m \end{cases}$$

ci sono 'n' job

$$p(0) = \left[ \sum_{i=0}^{m-1} \frac{(m\rho)^i}{i!} + \frac{(m\rho)^m}{m!(1-\rho)} \right]^{-1}$$

probabilità che il sistema sia vuoto



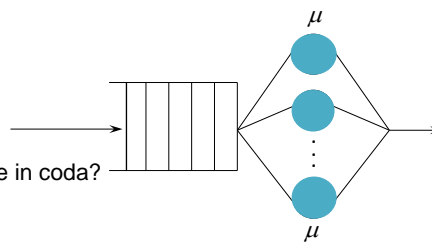
Prof. Vittoria de Nitto Personè

2

2

## The Erlang-C formula

probabilità che quando un job arriva finisce in coda?



$$\begin{aligned}
 P_Q &\equiv \Pr\{n \geq m\} = \sum_{n=m}^{\infty} p(n) \\
 &= \sum_{n=m}^{\infty} \frac{m^m}{m!} \rho^n p(0) = \frac{m^m}{m!} p(0) \sum_{n=m}^{\infty} \rho^n \\
 &= \frac{m^m}{m!} p(0) \sum_{n=0}^{\infty} \rho^{n+m} = \frac{m^m}{m!} p(0) \rho^m \sum_{n=0}^{\infty} \rho^n
 \end{aligned}$$

serie nota  $\frac{1}{1-\rho}$

Prof. Vittoria de Nitto Personè

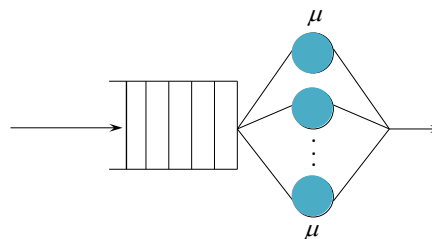
3

3

## The Erlang-C formula

probabilità che siano tutti pieni,  
dipende da 'm' e da 'rho'

$$P_Q = \frac{(m\rho)^m}{m!(1-\rho)} p(0)$$



$$E(N_Q)_{Erlang} = P_Q \frac{\rho}{1-\rho}$$

simile al caso servente singolo

Little's law

$$E(T_Q) = \frac{E(N_Q)}{\lambda} \quad E(T_Q) = P_Q \frac{\rho}{\lambda(1-\rho)} = \frac{P_Q E(S)}{1-\rho}$$

$$E(N_S) = P_Q \frac{\rho}{1-\rho} + m\rho$$

sommo quelli serviti mediamente

Prof. Vittoria de Nitto Personè

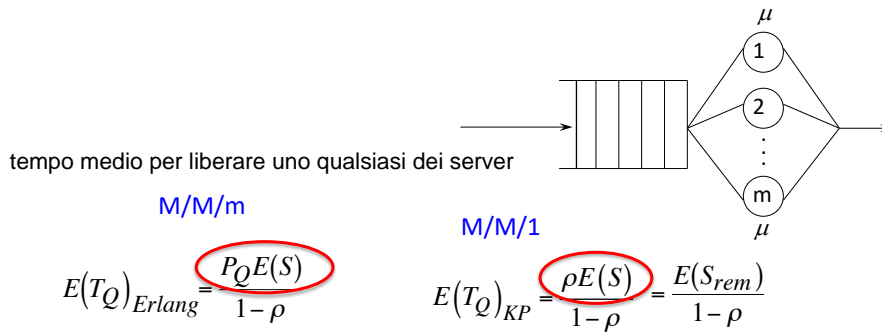
4

4

NB: vale solo per distribuzioni esponenziali, mentre la KP è più generica

Analytical models  
the multiserver queue

## The Erlang formula



tempo per far sì che se ne liberi uno, devo metterci lei!

$$E(S) = \frac{E(S_i)}{m}$$

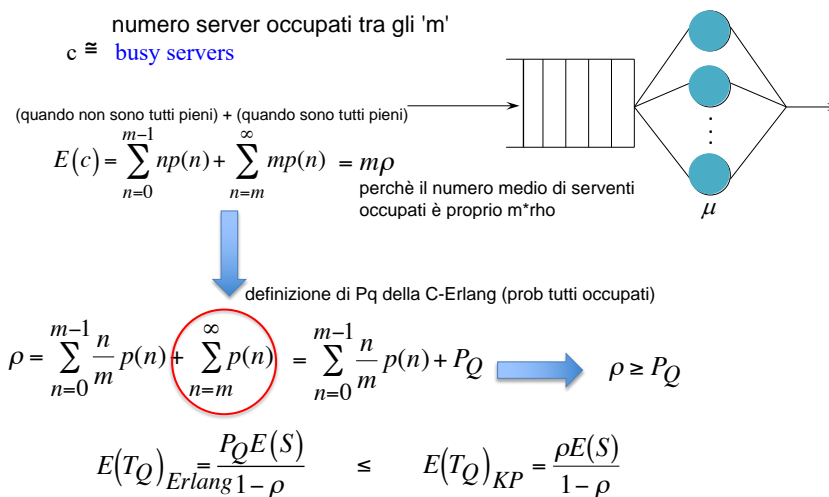
Prof. Vittoria de Nitto Personè

5

5

Analytical models  
the multiserver queue

## The Multi Server Queue



dato un certo carico,  $\lambda$  e  $\mu$ , la probabilità che siano tutti pieni è più piccola della probabilità che sia pieno solo uno.

Prof. Vittoria de Nitto Personè

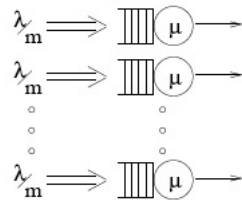
6

6

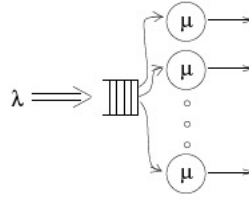
Nel multiserver ho più "sedie" su cui far sedere i job, se devo ottimizzare l'attesa, conviene distribuire la capacità, avere ad esempio 10 server meno potenti che uno 10 volte più potente, perchè dal punto di vista dell'attesa  $\rho > P_Q$ . Se devo minimizzare tempi di attesa è meglio la soluzione distribuita!

## Server Organizations

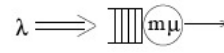
m server disgiunti che si dividono il traffico, potresti avere punti vuoti e punti in cui si creano code.



m server divisi ma con traffico convogliato tutto insieme, quindi non posso avere punti vuoti e punti con code



server m volte più veloce che si prende tutto lambda.



$$\rho = \frac{\lambda}{m\mu}$$

Prof. Vittoria de Nitto Personè

7

7

## Communication systems



communication line



independent Poisson packet streams

each with an arrival rate of  $\lambda/m$   
packets per second

the transmission  
time for each packet Exponential( $1/\mu$ )

Frequency-division Multiplexing

Statistical Multiplexing

Prof. Vittoria de Nitto Personè

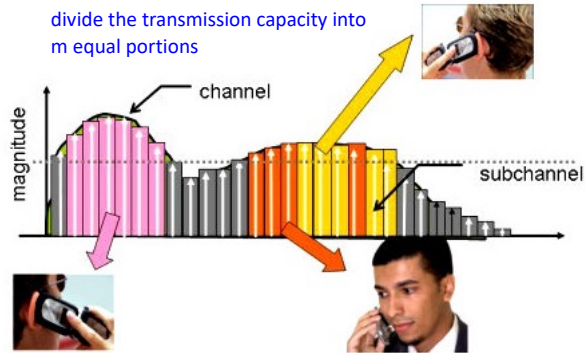
8

8

## Communication systems

### Frequency-division Multiplexing

1  
2  
3  
m  
separated streams



tengo gli 'm' flussi separati quando si divide la capacità trasmissiva in 'm' porzioni uguali.

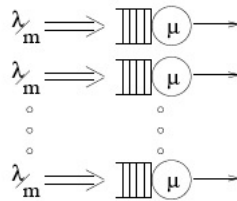
Prof. Vittoria de Nitto Personè

9

9

## Communication systems

### Frequency-division Multiplexing



Prof. Vittoria de Nitto Personè

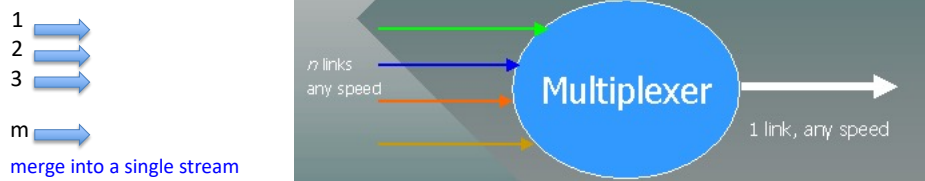
10

10

## Communication systems

### Statistical Multiplexing

keep the transmission capacity as a whole



Qui gli ' $m$ ' flussi vengono convogliati in un unico flusso, fa un 'merge' e li manda su un unico link.

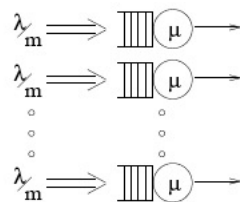
Prof. Vittoria de Nitto Personè

11

11

## Communication systems

### Frequency-division Multiplexing



### Statistical multiplexing



How do the two approaches compare with  
respect to mean response time?

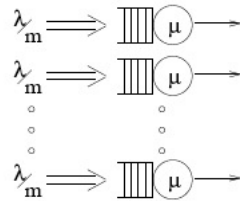
Prof. Vittoria de Nitto Personè

12

12

## Communication systems

### Frequency-division Multiplexing



$$E(T_S) = \frac{\rho E(S)}{1-\rho} + E(S) = \frac{E(S)}{1-\rho}$$

$$E(T_S) = \frac{1}{\mu \left(1 - \frac{\lambda}{\mu}\right)} = \frac{1}{\mu - \lambda}$$

### Statistical multiplexing



M/M/1

$\lambda \quad \mu$

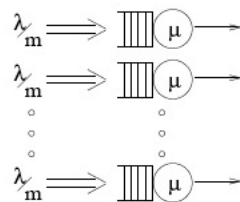
Prof. Vittoria de Nitto Personè

13

13

## Communication systems

### Frequency-division Multiplexing



$$E(T_S)^{FDM} = \frac{1}{\mu - \frac{\lambda}{m}} = \frac{m}{m\mu - \lambda}$$

### Statistical multiplexing



$$E(T_S)^{SM} = \frac{1}{m\mu - \lambda}$$

**FDM shows a response time m times greater than for SM !**

Prof. Vittoria de Nitto Personè

14

14

FDM però garantisce a ciascun flusso una specifica frequenza di servizio!

Sia 'C' la capacità operativa fisica, essa è fissa, non probabilistica Si esprime in [oper/sec].  
 C'è anche 'Z': la domanda di servizio media del job, espressa in [oper/job].  
 Quando parlavamo di E[S] 'univamo' questi due concetti, infatti  $E[S] = Z/C$ , espressa in [sec/job]  
 allora  $\mu$ , che è l'inverso, è espresso in [job/sec]  
 Questo vuol dire che, se dovessero essere forniti C e Z. allora devo ricavarli E[S]

Analytical models  
server organizations

## Communication systems

Frequency-division Multiplexing

Statistical multiplexing

1 QoS guarantee for each stream:  
a specific service rate to each stream

No QoS guarantee

se avessi Poisson, riottenrei Poisson

2 If the original m streams were very regular (not Poisson), i.e., they were much less variable than Poisson, by merging them, we introduce lots of variability into the arrival stream.  
 This leads to problems if the application requires a low variability in delay, e.g., voice or video.

15

15

11/04/2023

Analytical models  
server organizations

## Server Organizations

rho = rapporto frequenza arrivo / massima frequenza di servizio  
 devo sempre ragionare a parità di condizione!

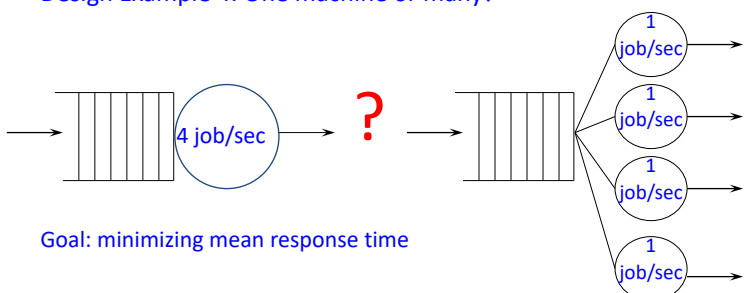
16

16




**Modelling power: Design tool**

Design Example 4: One machine or many?



Goal: minimizing mean response time

Assumption: jobs *non-preemptible*  
each job must be run to completion

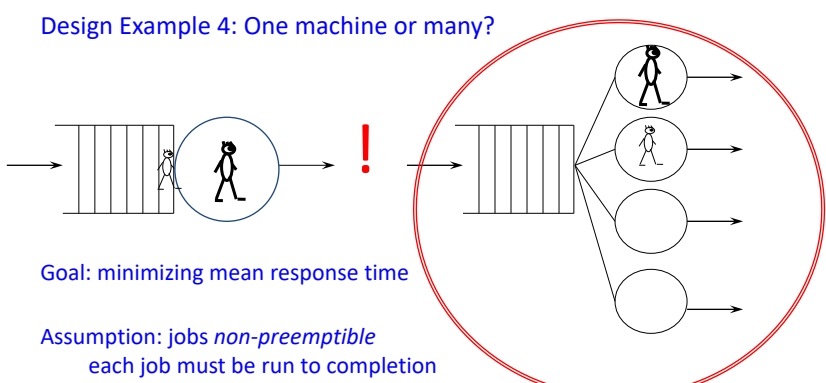
high variability 

Prof. Vittoria de Nitto Personè 17

17


**Modelling power: Design tool**

Design Example 4: One machine or many?



Goal: minimizing mean response time

Assumption: jobs *non-preemptible*  
each job must be run to completion

high variability 

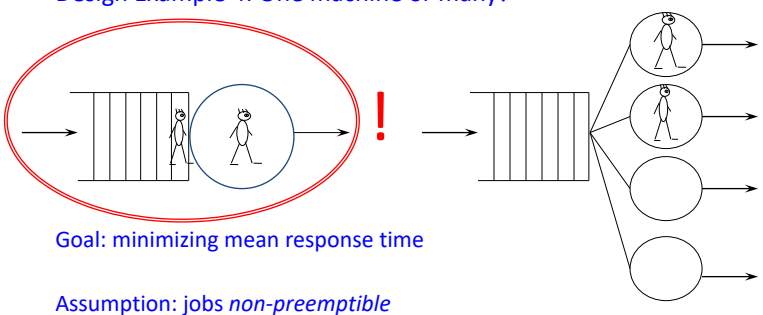
meglio dividere!

Prof. Vittoria de Nitto Personè 18

18


**Modelling power: Design tool**

Design Example 4: One machine or many?



Goal: minimizing mean response time

Assumption: jobs *non-preemptible*  
each job must be run to completion

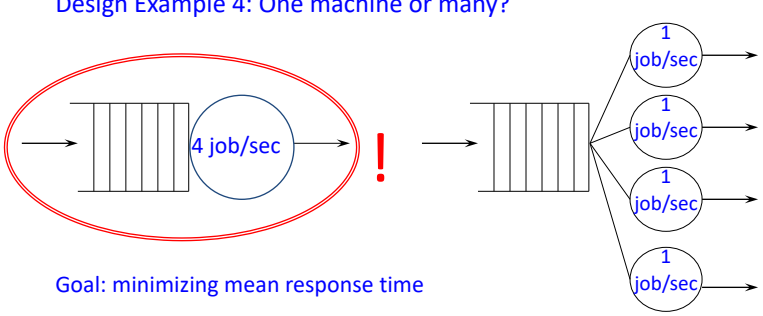
low variability  variabilità bassa, attesa proporzionale per tutti.  
Meglio uno più potente.

Prof. Vittoria de Nitto Personè 19

19

**Modelling power: Design tool**

Design Example 4: One machine or many?



Goal: minimizing mean response time

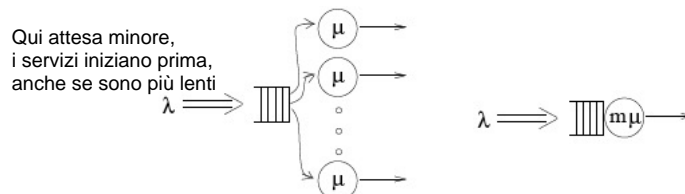
Assumption: jobs *preemptible*  
each job can be stopped and restarted where they left off

Prof. Vittoria de Nitto Personè 20

20

## Server Organizations

Stiamo parlando di esponenziale, perchè uso Erlang. La variabilità dell'esponenziale è medio/bassa.



$$E(T_Q)_{Erlang} = \frac{P_Q E(S)}{1 - \rho} \leq E(T_Q)_{KP} = \frac{\rho E(S)}{1 - \rho}$$

$\rho \geq P_Q$   
rho, cioè prob che UNO SIA PIENO  
=> prob. che siano tutti pieni!

from the waiting time perspective the distributed capacity solution produces an improvement in the user perceived QoS

Prof. Vittoria de Nitto Personè

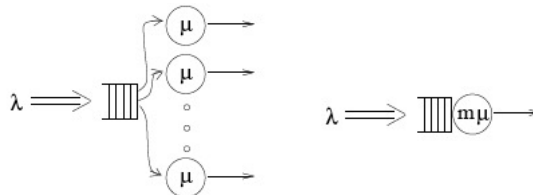
21

Per l'organizzazione a sinistra, essendo un multiserver, devo usare Erlang, il quale modella tale organizzazione.  
A destra, avendo un server singolo, non serve erlang ma uso KP.  
In entrambi i casi ho esponenziali però!

21

## Server Organizations

a sinistra i server sono 'm' volte più lenti



What about the response time perspective??

$$E(T_S)_{Erlang} = \frac{P_Q E(S)}{1 - \rho} + E(S_i) \quad E(T_S)_{KP} = \frac{\rho E(S)}{1 - \rho} + E(S)$$

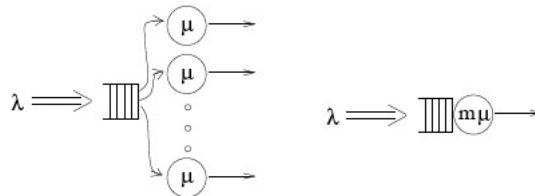
$$E(S_i) = \frac{1}{\mu} = m \frac{1}{m\mu} = mE(S)$$

Prof. Vittoria de Nitto Personè

22

22

## Server Organizations



What about the response time perspective??

$$E(T_S)_{Erlang} = \frac{P_Q E(S)}{1 - \rho} + mE(S)$$

Decreasing less than linear  $\Delta$   $\nabla$  linear growth

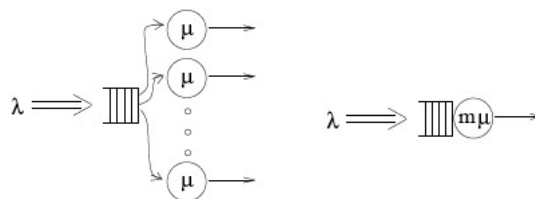
$$E(T_S)_{KP} = \frac{\rho E(S)}{1 - \rho} + E(S)$$

Prof. Vittoria de Nitto Personè

23

23

## Server Organizations



Performance goal:

Waiting time perspective

Distributed  
capacity

$\rho \rightarrow 0$

distrib. capac. gives  
an  $m$  times slower  
organization

Response time  
perspective

$\rho \rightarrow 1$

approximately the  
same response time

Prof. Vittoria de Nitto Personè

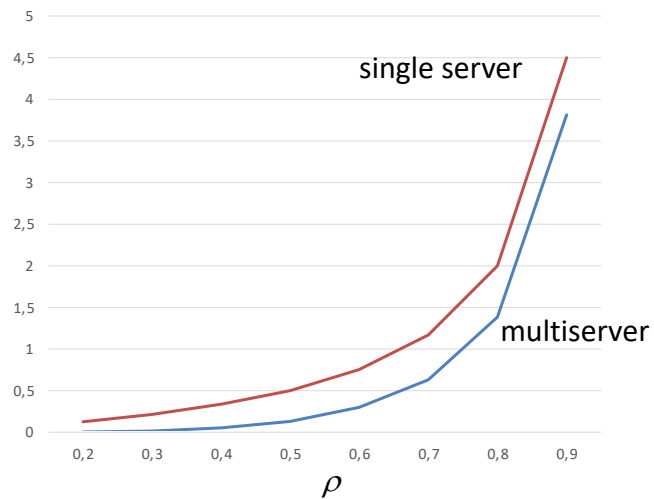
24

24

I casi qui esaminati valgono solo per l'ESPOENZIALE, se così non fosse?

## Waiting time perspective

$E(S)=0,5$  s



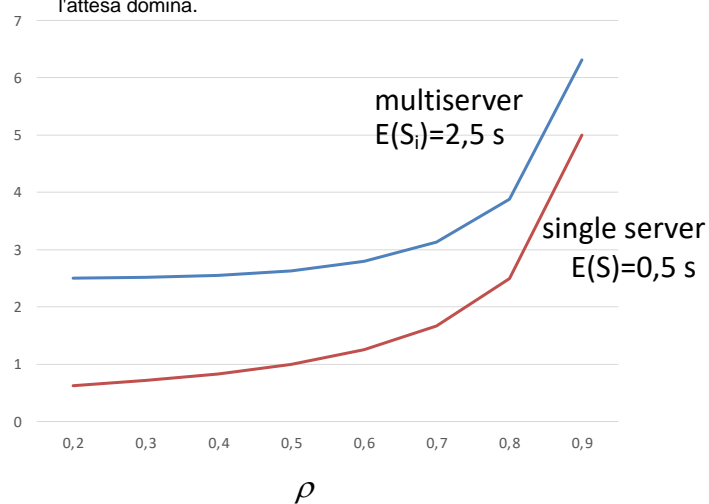
Prof. Vittoria de Nitto Personè

25

25

## Response time perspective

Qui va meglio il multiserver, più  $\rho$  cresce, più si avvicinano, perchè l'attesa domina.



Prof. Vittoria de Nitto Personè

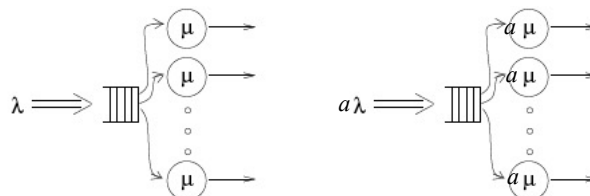
26

26

## Scaling factor

Problema del fattore di scala:

Se faccio crescere dello stesso fattore sia tasso di arrivo sia tasso di servizio, cosa succede alle prestazioni?



Se lambda aumenta? Vorrei che i clienti non se ne rendessero conto!

What about waiting and response time? a = fattore di scala

$$\rho = \frac{\lambda}{m\mu}$$

$$E(S_i) = \frac{1}{\mu}$$

$$\rho = \frac{a\lambda}{ma\mu} = \frac{\lambda}{m\mu}$$

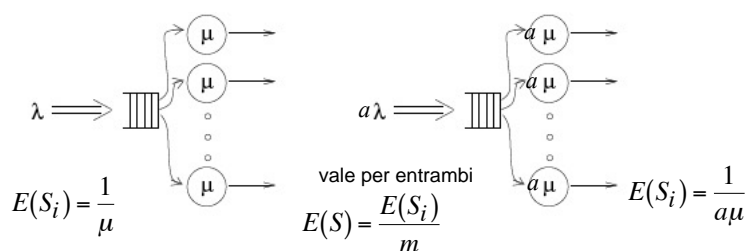
$$E(S_i) = \frac{1}{a\mu} \quad E(S) = \frac{E(S_i)}{m}$$

Prof. Vittoria de Nitto Personè

27

27

## Scaling factor



Mean waiting time

tempo servizio medio per liberarne uno qualsiasi

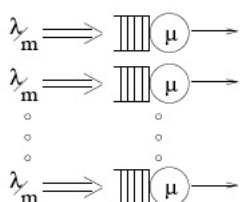
$$E(T_Q)_{m,a} = \frac{P_Q E(S)_{m,a}}{1-\rho} = \frac{P_Q}{ma\mu(1-\rho)} = \frac{1}{a} \frac{P_Q E(S)_{m,1}}{(1-\rho)} = \frac{1}{a} E(T_Q)_{m,1}$$

$P_Q$  è in funzione di 'rho' ed 'm', non viene variata da lambda.

Prof. Vittoria de Nitto Personè

28

28 Quando faccio il tempo di servizio devo sommarvi  $E[S_i]$ , perchè è ciò che sperimenta un singolo job.

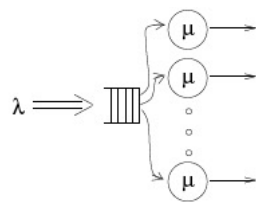


$\lambda = 4 \text{ j/s}, m=4, \mu=1.5 \text{ j/s} \quad E(S)=0.666667 \text{ s}$

$\rho = 0.666667 \quad E[S] \cdot \lambda / m$

$E(T_S) = \frac{1}{\mu - \lambda} = 2 \text{ s} \quad \text{di ciascuna linea}$

$E(T_Q) = \frac{\rho E(S)}{1 - \rho} = 1.3336$   
potevo anche fare  $E[T_S] - E[S]$



$E(S) = \frac{E(S_i)}{4} = 0.1667$

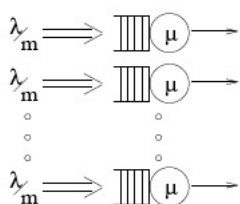
$\rho = 0.666667 \quad \text{come prima}$

$p(0) = \left[ \sum_{i=0}^3 \frac{(4\rho)^i}{i!} + \frac{(4\rho)^4}{4!(1-\rho)} \right]^{-1}$

$= \left[ 1 + 4\rho + \frac{(4\rho)^2}{2} + \frac{(4\rho)^3}{6} + \frac{(4\rho)^4}{24(1-\rho)} \right]^{-1} = 0.059857$

Prof. Vittoria de Nitto Personè

29

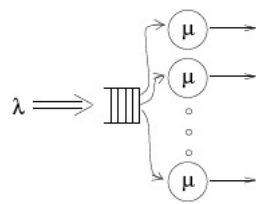


$\lambda = 4 \text{ j/s}, m=4, \mu=1.5 \text{ j/s} \quad E(S)=0.666667 \text{ s}$

$\rho = 0.666667$

$E(T_S) = \frac{1}{\mu - \lambda} = 2 \text{ s}$

$E(T_Q) = \frac{\rho E(S)}{1 - \rho} = 1.3336$



$E(S) = \frac{E(S_i)}{4} = 0.1667$

$E(S_i) = 0.666667 \text{ s}$

$\rho = 0.666667$

$P_Q = \frac{(4\rho)^4}{4!(1-\rho)} p(0) = 0.37847 \leq \rho$

$E(T_S) = \frac{P_Q E(S)}{1 - \rho} + E(S_i) = 0.855992$

$E(T_Q) = 0.189292$

Prof. Vittoria de Nitto Personè

30

$\lambda = 4 \text{ j/s}, \quad m\mu = 4 \times 1.5 = 6 \text{ j/s} \quad E(S) = 0.166667 \text{ s}$

$\rho = 0.666667$

$E(T_S) = \frac{1}{m\mu - \lambda} = 0.5 \quad \text{1/4 di quello prima}$

$E(T_Q) = 0.3334$

Prof. Vittoria de Nitto Personè
31

31

$\rho = 0.666667$

$E(T_S) = \frac{1}{\mu - \lambda} = 2$   
 $E(T_Q) = \frac{\rho E(S)}{1 - \rho} = 1.3336$

$E(T_S) = 0.855992$   
 $E(T_Q) = 0.189292$

$E(T_S) = \frac{1}{\mu - \lambda} = 0.5 \text{ s}$   
 $E(T_Q) = 0.3334$

cosa sarebbe successo con variabilità non esponenziale? con un altro 'm'?

Esercizio proposto:  $\rho = 0.533334 \quad \rho = 0.8$

Prof. Vittoria de Nitto Personè
32

32

Le popolazioni del sistema si mantengono costanti, anche se sembra controintuitivo, ma alla fine con Little moltiplico  $N = \lambda \cdot T$ , e quindi se  $T$  diventa  $T/a$ , alla fine si semplifica e resta un qualcosa di costante. Resta costante perchè i due termini analiticamente si elidono, concettualmente è proprio il fatto che la popolazione rimanga costante che ci dà l'idea che il tempo sia minore. Con Little abbiamo popolazione in media nel sistema, che è diverso dal traffico in entrata che è cresciuto da 'a'.  $N = \lambda \cdot T = a \cdot \lambda \cdot T/a$ , non cambia nulla.