Paolo Cremonesi

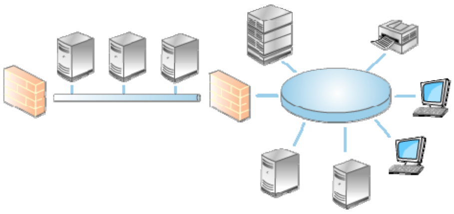
Impianti Informatici

POLITECNICO DI MILANO

Modelli a rete di code: modelli aperti e modelli chiusi

Distinzione tra modelli aperti, chiusi e misti

- Modelli aperti, classi aperte
  - Numero potenzialmente illimitato di richieste nel sistema
- Admission control



$$N = 1024$$

- Modelli chiusi, classi chiuse
- Modelli misti

La lezione che andremo a presentare ci consetira' di apprendere alcune semplici tecniche che, assieme alle leggi dell'analisi operazionale viste nella lezione precedente, consentono la risoluzione completa di un modello a reti di code.

•Prima di procedere, dobbiamo pero' operare una ulteriore distinzione tra la reti in forma prodotto che consideriamo

•Le reti che abbiamo considerato finora sono dette reti aperte, in quanto il nostro sistema comunica con il mondo esterno e da questi puo' ricevere

•un numero a priori illimitato di richieste da processare

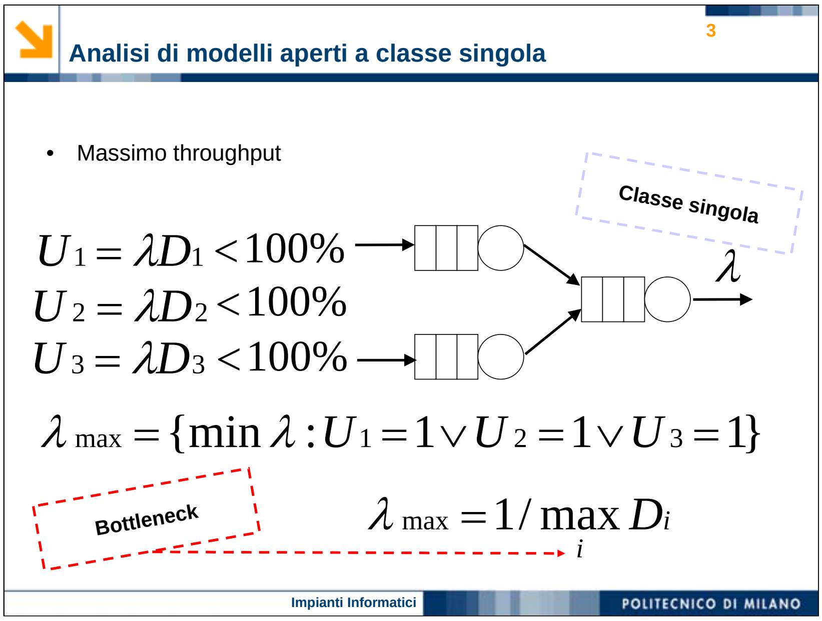
Tuttavia, un ampia famiglia di componenti e sistemi reali ha delle limitazioni circa il massimo numero di richieste presenti nel sistema, Tali vincoli, sono specificati tramite politiche di admission control che non permettono l'accessp a nuove richieste se il sistema e' pieno.

•ad esempio un router con un buffer di 1024 richieste, non puo' accettarne di nuove fino a quando non ha smaltite almeno una richiesta di quelle in attesa al suo interno. Inoltre, poiche' al liberarsi di un slot del buffer e' probabile che il router riceva subito un nuova richiesta da servire,

•il comportamento del router e' meglio descritto da una coda che processa una popolazione costante N=1024

•I modelli a reti di code che descrivono sistemi con numero di richiest al proprio interno costante sono detti modelli chiusi

•Infine chiamiamo modelli modelli misti quei modelli con piu' classi di richieste, alcune aperte, ovvero senza limiti al numero di richieste in attesa nel sistema, in parte chiuse, ovvero con popolazione costante,



•Cominciamo allora a vedere le tecniche risolutive esistenti per i modelli a reti di code aperti.

•Per non complicarci la vita, supporremo inizialmente che tutte le richieste appartengono ad un'unica classe, ovvero che il nostro modello sia a classe singola. Nei modelli a classe singola e' solitamente piu' comodo omettere gli indici c che denotano la classe delle richieste

•Ad esempio, lambda indichera' il tasso di arrivo dell'unica classe di servizio nella rete

•Ricordiamo, inoltre, che per la legge del bilanciamento di flusso lambda equivale anche al throughput della rete e, dunque, essendo lambda un parametro noto richiesto per la specifica del modello, possiamo sempre assumere di conoscere il throughput di un modello aperto

•La prima informazioni di performance che vogliamo ottenere e' il massimo throughput che la rete puo' sostenere, o equivalentemente, il massimo tasso di arrivo medio che puo' ricevere

In generale, prescidendo dal tempo di risposta sperimentato dalle richieste, l'unico vincolo esistente che limita la capacita' della rete deriva dalla legge dell'utilizzo

•Per le tre code del sistema rappresentato, la legge esprime per ciascuna il legame tra il throughput e la domanda globale di servizio alla stazione

•Ma poiché l'utilizzo e' la percentuale di tempo speso per servire le richieste, esso non può mai superare il 100%

•Avremo cioè che il fattore limitate del throughput è l'utilizzo delle stazioni, e il throughput massimo sia avra' per il primo valore di lambda che porta ad utilizzo 1, ovvero ad utilizzo del 100%, uno qualunque tra i server della rete. In questo caso si parla di saturazione di un server della rete.

•Con semplici passaggi algebrici si vede che il minimo dei lambda è dato dall'inverso del massimo valore delle Di, tale valore e' detto massimo throughput o tasso di arrivo di saturazione

•e la stazione i per la quale l'utilizzo sale al 100% è detta la stazione bottleneck della rete, cioè il collo di bottiglia delle prestazioni

•Nel caso in cui esistano più stazioni bottleneck con lo stesso valore delle D con i si parla di reti con bottleneck multipli

Calcolo del tempo di risposta di una rete aperta 4

- Misura additiva

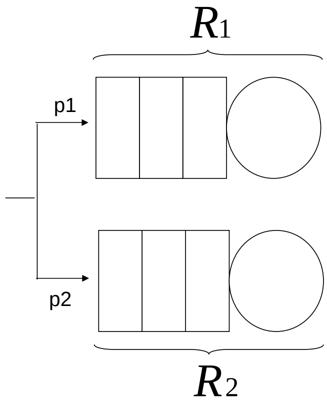
$$R = \sum_i R_i$$

- Pesi dei tempi di risposta

$$R = p_1 R_1 + p_2 R_2$$

- Probabilita' catturate dalle visite

$$R_1 = f(D_1)$$
$$R_2 = f(D_2)$$



Impianti Informatici POLITECNICO DI MILANO

Vediamo ora come affrontare il problema della determinazione dei tempi di risposta delle stazioni e del tempo di risposta medio della rete.

•Si noti che esiste un legame fra queste quantità, in quanto e' sufficiente andare a calcolare la somma dei tempi di attraversamento dei singoli server, cioè dei loro tempi di risposta Ri, per ottenere il tempo di risposta della rete

•In altri termini, il tempo di risposta della rete e' una misura additiva

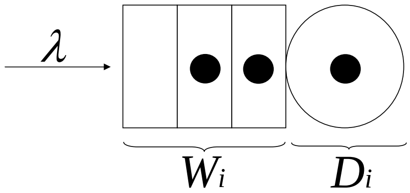
•Si noti anche che questa espressione non e' intuitiva, in quanto se ad esempio la rete e' costituita da due server in parallelo, il tempo di risposta dei job dipende da quale dei due server viene selezionato

•Ovvero, in media, si avrà R=pi*R1+p2*R2, dove R1 ed R2 sono i tempi di risposta dei singoli server

•Ma come abbiamo detto nella scorsa lezione, p1 e p2 sono catturate implicitamente nelle visite e dunque sarà sufficiente esprimere R1 ed R2 in funzione, rispettivamente, di D1 e D2 invece che di S1 e S2 per tenere conto anche dell struttura di rete

5

Calcolo del tempo di risposta dei server



$$R_i = D_i + W_i$$

$$W_i = D_i Q_i = D_i X R_i = D_i \lambda R_i$$

$$R_i = D_i + D_i \lambda R_i \qquad R_i = \frac{D_i}{1-\lambda D_i} = \frac{D_i}{1-U_i}$$

Impianti Informatici

POLITECNICO DI MILANO

Andiamo quindi a dettagliare come si possa effettuare il calcolo dei tempi di risposta di ciascun server.

•Consideriamo quindi un server i con una domanda di servizio D con I

•a cui giunga un flusso di richieste con tasso di arrivo lambda

•Il tempo di risposta sara’ in generale dato dal tempo necessario alla richiesta per attraversare il servente Di, piu’ il tempo speso ad aspettare che le richieste gia’ in coda siano tutte servite, tempo che indichiamo come W con I. Dobbiamo quindi cercare di esprimere Wi in funzione dei soli D con i e lambda

•Come intuitivo, se in media una richiesta trova al suo arrivo Qi richieste, ciascuna delle quali impiega Di per essere servita, allora il tempo Wi sara’ dato da D con i per Q con I

•Ma dalla legge di little abbiamo Qi = X*Ri e imponendo X e lambda, otteniamo che Wi e’ uguale a D con I per lambda Ri e reinserendo nella formula originale otteniamo Ri=Di+Dilambda Ri e portando Ri a primo membro concludiamo che Ri=Di/(1-lambda Di) che per la legge dell’utilizzo puo’ essere anche scritto come Di/1-Ui. Questo formula e’ sufficiente per calcolare il tempo di risposta del modello aperto dalla conoscenza delle Di di tutti I server e del tasso di arrivo lambda alla rete

6

Calcolo della lunghezza di coda

$$R_i = \frac{D_i}{1-U_i} \qquad Q_i = \lambda R_i = \frac{\lambda D_i}{1-U_i} = \frac{U_i}{1-U_i}$$

$$R(\lambda) = \sum_i R_i(\lambda) \qquad R_i(\lambda) = \frac{D_i}{1-U_i(\lambda)}$$

$$Q_i(\lambda) = \frac{U_i(\lambda)}{1-U_i(\lambda)}$$

Impianti Informatici POLITECNICO DI MILANO

La lunghezza di coda puo' immediatamente calcolata banalmente dalla legge di Little partendo

•dalla formula precedente.

•Poiche' dalla legge di Little abbiamo Qi=lambda*Ri, ricaviamo immediatamente Qi=lambda*Di/(1-Ui) e raccogliendo anche in questo caso con la legge dell'utilizzo abbiamo

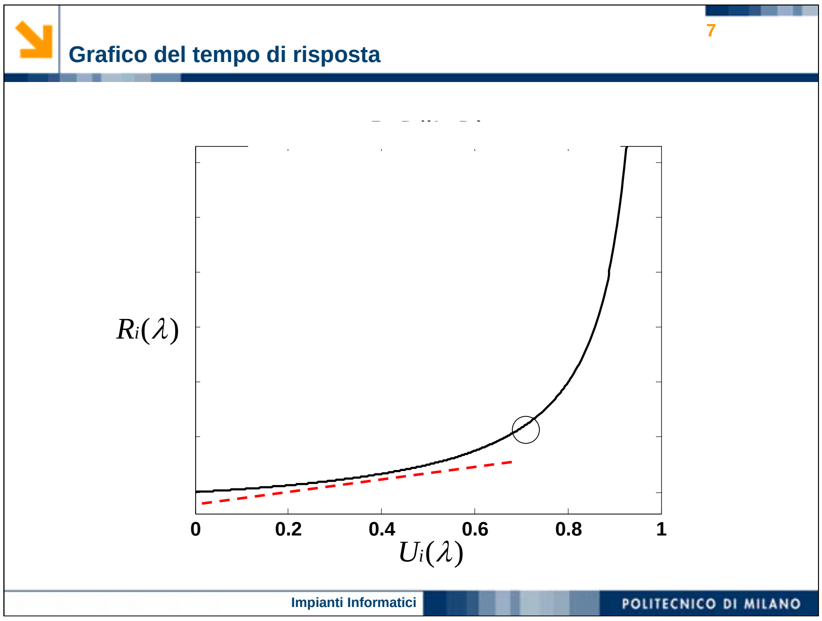
•Qi = Ui/(1-Ui)

•Ricaptilando, e esplicitando la dipendenza dei diversi termini dal tasso di arrivo lambda della rete, le formule che consentono la risoluzione di un modello aperto sono

•Tempo di risposta di rete R(lambda) uguale alla somma degli Ri(lambda) di tutte le stazioni

•Ri lambda = Di/(1-Ui di lambda) e

•Qi lambda = Ui/(1-Ui di lambda)



Grazie alle formule trovate possiamo disegnare l'andamento del tempo di risposta di una singolo server in funzione dell'utilizzo della stazione Ui(lambda).

•Quello che possiamo notare e' che inizialmente, all'aumentare dell'utilizzo, la curva subisce una crescita pressoche' lineare. Questo vuol dire che al raddoppiare del traffico in ingresso, si ha inizialmente un aumento dei tempi di risposta proporzionale al nuovo carico arrivato

•Tuttavia, quello che possiamo vedere dalla parte destra del grafico e' che successivamente la stazione non riesce piu' a smaltire le richieste in coda e il tempo di risposta esplode in modo incontrollato, tendendo verso un asintoto a + infinito quando il denominatore 1-Ui del tempo di risposta va a zero. Questo avviene quando l'utilizzo della stazione Ui si avvicina ad 1, cioe' al 100%, e questo effetto, come gia' visto nelle slide precedenti, e' detto saturazione del server i.

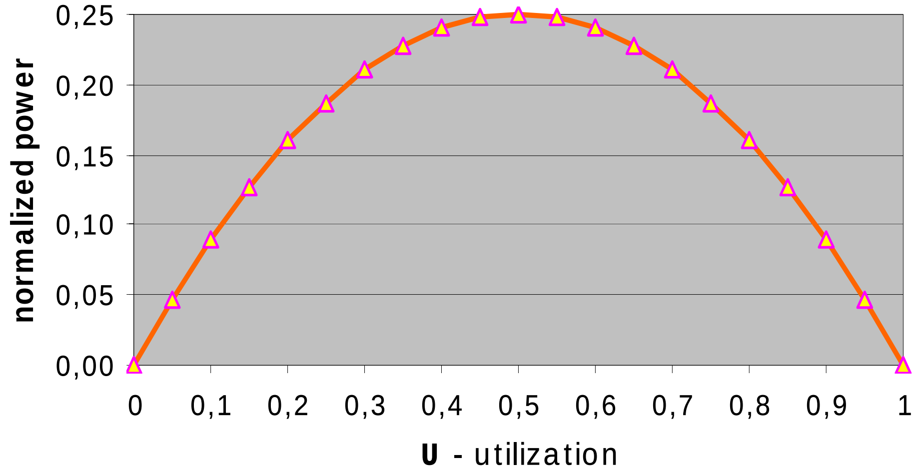
•Il punto in cui convenzionalmente si assume lo switching tra i due comportamenti e' quello in cui il server ha utilizzo pari al 75%. Dunque, componenti che abbiano utilizzi medi superiori a tale soglia sono da considerarsi un ostacolo alle prestazioni della rete.

Si tenga presente che il comportamento della lunghezza di coda e' analogo a quello del

Potenza

$$\Phi \equiv \frac{\lambda}{R} = \frac{\lambda(1-\lambda D)}{D}$$

$\phi' = 0$ - U = 50%



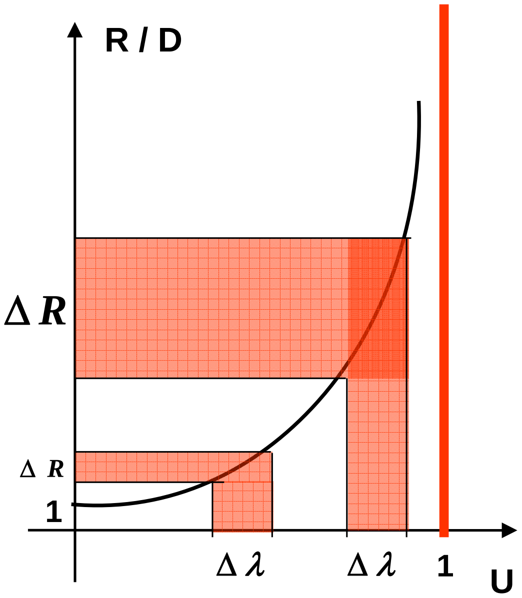
Asintoto di saturazione

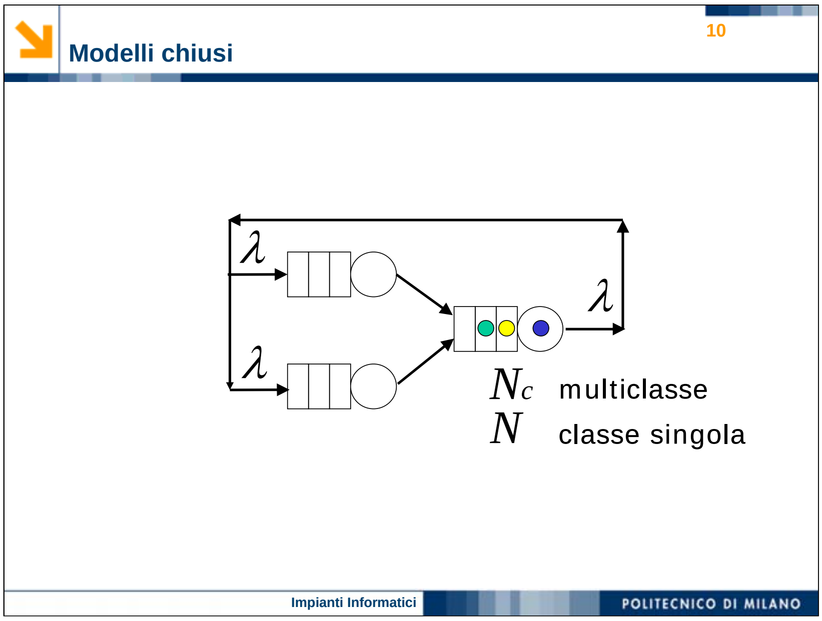
$$\Delta\lambda \Rightarrow \Delta R = \frac{D^2}{(1-U)^2}\Delta\lambda$$

$$U = 0.2 \Rightarrow \Delta R = 1.56\ D^2\Delta\lambda$$

$$U = 0.5 \Rightarrow \Delta R = 4\ D^2\Delta\lambda$$

$$U = 0.9 \Rightarrow \Delta R = 100\ D^2\Delta\lambda$$





In questa lezione consideriamo i modelli a reti di code chiusi.

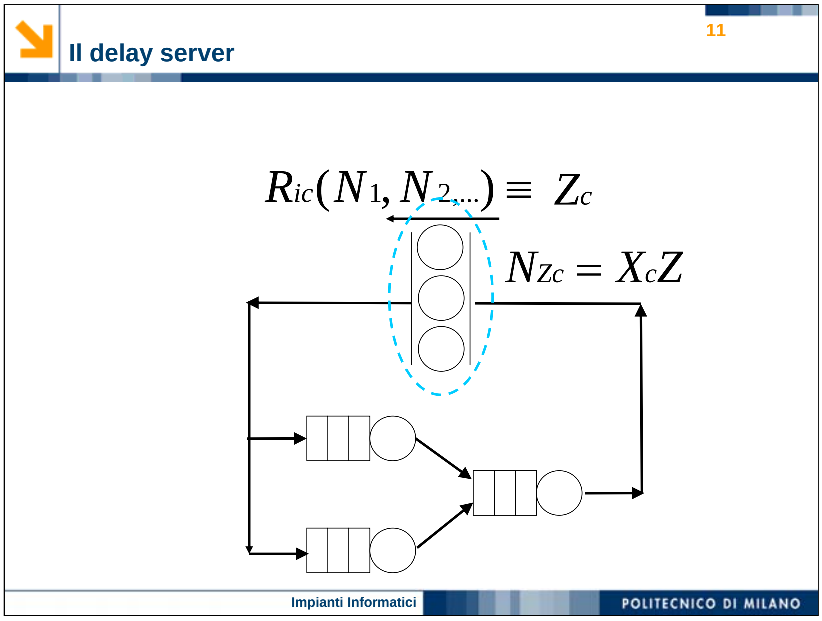
•Come visto, la principale differenza rispetto ai modelli aperti e' che nei modelli chiusi le richieste sono in numero costante e continuano a ciclare nella rete

•

•Il numero di richieste nella rete e' detto popolazione del modello

•e per una classe di richieste c viene indicato con la lettera Nc

•o con la lettera N per I modelli a classe singola



•In alcuni casi le richieste impiegano alcuni secondi prima di compiere un nuovo ciclo nel modello. Per modellare questo effetto si inserisce un delay nella rete.

•un delay e' uno speciale server dove il tempo di risposta ed il tempo di servizio coincidono sempre ovvero, dove nessuna richiesta si accoda in quanto viene sempre servita all'istanto di arrivo.

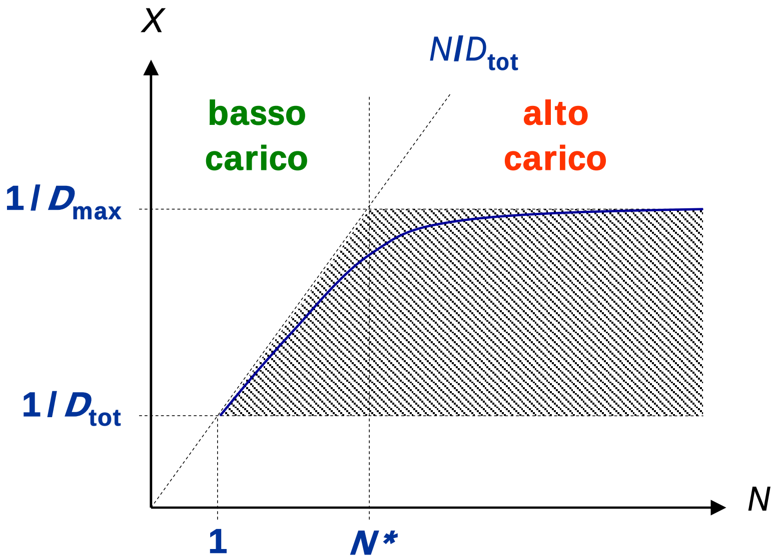
•La domanda di servizio del delay e' indicata con la lettera Zc per una classe c ed e' spessa detta think time delle richieste di classe c

•Infine si noti che per la legge di little il numero di richieste di classe c in attesa presso il delay e' Nzc = Xc(N)*Z

Tecniche approssimate

- Si applicano a **modelli chiusi** a **classe singola**
- D_{max} = tempo di servizio della risorsa più lenta
- D_{tot} = somma dei tempi di servizio di tutte le risorse
- N = numero di utenti presenti nel sistema

Limiti sul throughput



$N^* = D_{tot} / D_{max}$

Limiti sul tempo di risposta

$N^* = D_{tot} / D_{max}$