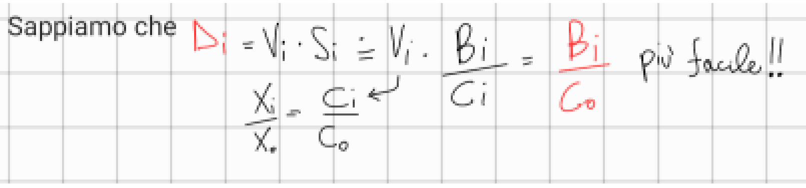
Bound & Bottleneck

Possiamo ragionare sia su sistemi chiusi sia su sistemi aperti. Il bottleneck è il *centro che limita le prestazioni*.

La domanda è $D_i = V_i \cdot S_i$, dove S_i è il tempo di servizio del dispositivo i, V_i sono le visite al dispositivo i ,mentre D_i è il tempo totale speso in quel dispositivo. La domanda è tuttavia calcolabile in un modo più semplice, poichè scritta in questo modo dipende da V_i, cioè rapporto tra completamenti. Possiamo usare la **legge del flusso forzato**, cioè riscriviamo $V_i = \frac{X_i}{X_0}$. Nel sistema aperto è ciò che entra e ciò che esce, nel sistema chiuso, poichè non esce nulla, si calcola rispetto ad un punto di riferimento.

Altra formula è $X_i = \frac{C_i}{T}$, questo ci permette di scrivere $V_i = \frac{C_i}{T} \cdot \frac{T}{C_0} = \frac{B_i}{C_0}$ Visivamente:



Questa versione è più semplice perchè sia *busy time* sia *completamenti* sono più facili da calcolare. Dalla **legge dell'utilizzazione** $U_i = X_i \cdot S_i = X_0 \cdot V_i \cdot S_i$ grazie alla **legge del flusso forzato**. Il tutto è riscrivibile come $U_i = X_0 \cdot D_i$, ottenendo quindi una visione sul *sistema intero*. $U_i \to 1$ se $X_0 \to \lambda_{saturazione}$, allora $D_i \to D_b$ cioè domanda di livello *bottleneck*. Ciò ci permette di dire: $1 = \lambda_{sat} \cdot D_b$ e quindi $\lambda_{sat} = \frac{1}{D_b}$, non vado oltre!

L'analisi del *bottleneck* cerca il *centro di domanda massima*, ovvero:
$max\{V_1S_1, V_2S_2, ..., V_kS_k\} = D_b = V_bS_b$

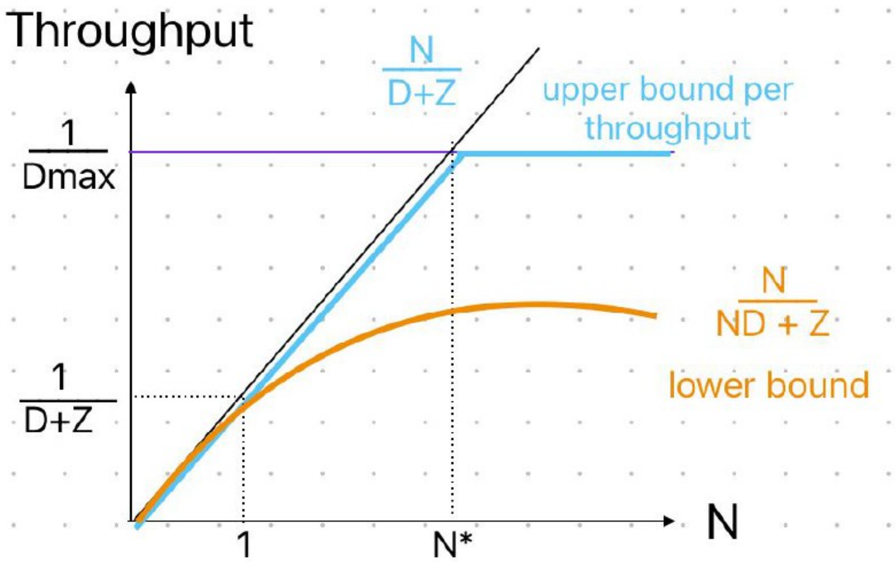
In ottica di tempo di risposta? Innanzitutto il minimo tempo di risposta possibile è quello che il job chiede. Se Il mio task richiede 5s, non posso metterci meno di questo tempo.

In un sistema chiuso: Il job va un numero di volte V_i nel centro i e chiede un tempo di servizio S_i. Il caso semplice prevede solo un job, che non aspetta nessuno e fa quello che vuole senza aspettare. La domanda totale è $D = \sum_{i=1}^{k} D_i$, cioè quanto chiede in ogni centro. Se considero anche il *think time Z*, quindi in un contesto *interattivo* e *sistema aperto* (infatti se $Z = 0$ il sistema è chiuso in quanto non c'è interazione con utenti), allora il throughput è $\frac{1}{D+Z}$, cioè l'inverso di quanto chiede + think time. Un singolo job avente interferenza massima ha throughput $\frac{1}{ND+Z}$. Se ci sono N job:

- Qual è il caso pessimo? Lo si ha se tutti gli N job vanno nello stesso centro ogni volta, quindi l'ultimo job della "fila" deve sempre aspettare tutti gli altri, cioè $\frac{N}{ND+Z}$

- Qual è il caso ottimo? Lo si ha se ogni job si muove in modo tale che quando visita un centro ci sia solo lui, allora qui il throughput è $\frac{N}{D+Z}$ (ovvero i vari job non interferiscono tra di loro).

Graficamente avremmo:



Quindi il nostro *throughput* cade tra la linea arancione e quella celeste. N^* è il punto di *saturazione del sistema*. Il suo valore è dato da $\frac{1}{D_{max}} = \frac{N^*}{D+Z}$, ovvero $N^* = \frac{D+Z}{D_{max}}$

Facciamo altre osservazioni:

- Il boind pessimistico è: $\frac{N}{ND+Z} \leq X(N) \leq min(\frac{1}{D_{max}}, \frac{N}{D+Z})$

  La parte di *destra* dipende la nostra posizione rispetto N^*:

  - prima di N^* l'andamento è dato da $\frac{N}{D+Z}$,
  - dopo N^* non posso andare oltre $\frac{1}{D_{max}}$

- Se $Z = 0$ allora il sistema è *chiuso* con *k* centri connessi e con *N* job.

- La legge del tempo *interattivo* (tempo di risposta interattivo) è: $R = \frac{N}{X_0} - Z$, ovvero tale tempo è *inversamente proporizionale al throughput*, cioè ho R minimo se massimizzo X_0, ovvero se $X_0 = \frac{1}{D_{max}}$

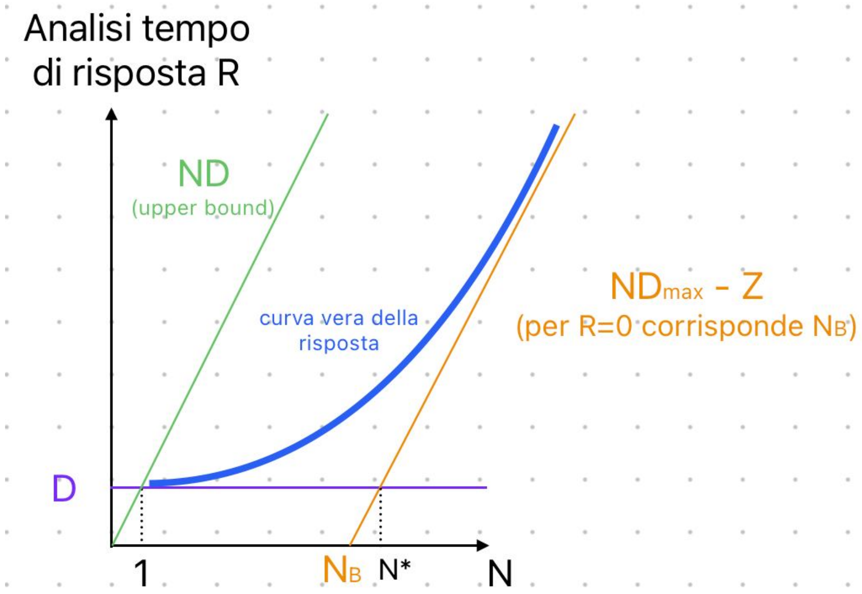
- Per R troviamo i seguenti bond: $ND \geq R \geq max\{D, \frac{N}{\frac{1}{D_{max}}} - Z\} = max\{D, ND_{max} - Z\}$

  Ovvero: ND è un upper bound (non posso fare peggio del caso in cui un job aspetta tutti gli N precedenti), l'altro è un lower bound, se prendo D ad esempio, un job non può metterci di meno del tempo che chiede lui stesso! Inoltre, se non ci fossero job, allora $R = 0$, quindi il tempo di risposta è minimo, allora il throughput (l'inverso) è il massimo, ma quanto è questo massimo? $ND_{max} - Z$, allora anche questo è un bound! Ho due lower bound e un upper bound.

Esaminiamo questi tempi in funzione dei job presenti:

- Se ci sono 0 job, allora il tempo minimo di risposta è 0 (nessuno chiede nulla), ma allora $R = 0 = ND_{max} - Z$ da cui $N_b = \frac{Z}{D_{max}}$, ovvero moltiplico il tempo *Z* per il *flusso entrante* $1/D_{max}$
- Se ci fosse 1 job, il tempo di risposta è D.
- Possiamo dire che $N^* = \frac{D+Z}{D_{max}} = \frac{Z}{D_{max}} + \frac{D}{D_{max}} = N_b + \frac{D}{D_{max}}$

  ovvero ho suddiviso in *chi sta nel think time* e *chi sta nel centro*.
- In corrispondenza del punto di saturazione N^*, il valore della popolazione nel centro "terminali", cioè il numero di terminali che stanno "pensando" quando il sistema è saturo è N_b, mentre il sottosistema centrale ho $\frac{D}{D_{max}}$



Esercizi di esempio

Esercizio 1

Abbiamo: T = 5 min, $U_{cpu} = 0.3$, $D_{disk} = 0.4s$=domanda al disco $V_{disk} = 10$, numero di operazione IO/singola transazione $U_{disk} = 0.4$, utilizzazione disco $R = 15s$, tempo risposta. $N = 50$,numero di utenti. *Quale è il think time medio per 50 utenti?* Sappiamo che il tempo di risposta interattivo è $R = \frac{N}{X_0} - Z$, e noi stiamo cercando $Z = \frac{N}{X_0} - R$, necessitiamo di X_0.

Sfruttando il legame tra l'uso di una risorsa rispetto al sistema totale, abbiamo $X_0 = \frac{U_{disk}}{D_{disk}} = 0.4/0.4 = 1$ transazione/secondo. Allora ho tutto per trovare Z $= 35s$

Esercizio 2

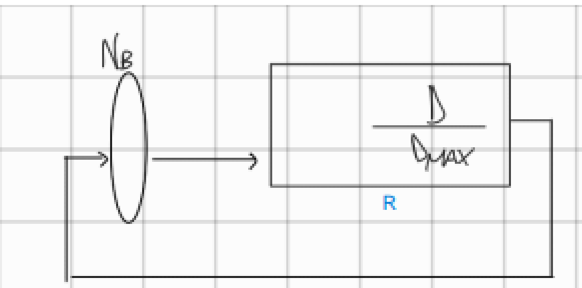
Prendiamo un sistema avente due risorse, di cui sappiamo che: $R_1 = 10s$, cioè il tempo di risposta della risorsa 1 $R_2 = 1s$, cioè il tempo di risposta della risorsa 2 $X_1 = 4\ trans/s$, cioè il throughput della risorsa 1 $X_2 = 8\ trans/s$, cioè il throughput della risorsa 2 $X_0 = 4\ trans/s$, cioè il throughput del sistema. *Quale è il tempo di risposta del sistema?* Per definizione, $R = \sum_{i=1}^{N} v_i R_i$ A noi servono le visite, cioè usando il *flusso forzato* $\frac{X_1}{X_0} = V_1 = 1$ e $\frac{X_2}{X_0} = 2$, allora $R = 12s$.

Esercizio 3

Siano dati $D_{cpu} = 4s$ = domanda cpu, $U_{cpu} = 0.5$= uso CPU, $R = 15s$ = tempo di risposta, $Z = 25s$. *Quale è il numero di utenti?* Sappiamo che $N = (R+Z)X_0 = (R+Z) \cdot (\frac{U_{cpu}}{D_{cpu}}) = 5s$

Esercizio 4

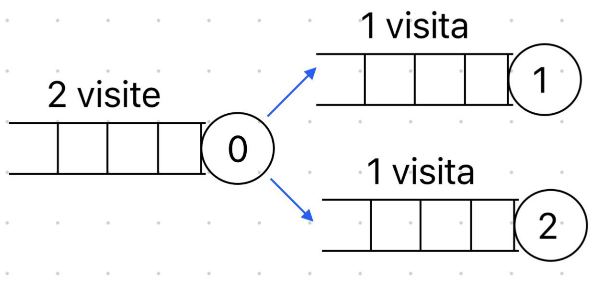
Siano dati $T = 1$ ora, cioè il tempo di osservazione, $M = 80$ utenti, $R = 5s$ tempo di risposta, $C = 60/$ ora il numero di transazioni completate, $U_{cpu} = 0.8$, $U_{disk1} = 0.5$, $U_{disk2} = 0.5$, *Quanto vale Z?* Sappiamo che le risorse che *non pensano* sono date da $N_{notThinking} = R \cdot X = R \cdot \frac{C}{T} = 5$, allora $N_{thinking} = \quad = M - N_{notThinking} = 80 - 5 = 75$
Questi sono il numero di utenti che pensano, non mi dice QUANTO PENSANO, cioè Z.
In questi esercizi, se si parla di think time, si fa riferimento a questo tipo di sistema:



da LITTLE vale sempre che:
n° utenti : tasso uscita (throughput) * tempo, qui nel nostro caso sarà:
n° utenti pensanti: Xo * Z
cioè quanti ne 'escono' per quanto tempo pensano, allora : 1 * 75 = 75 utenti

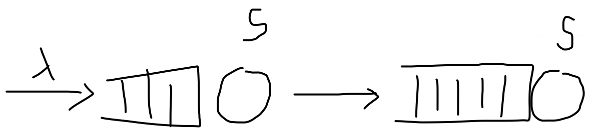
---

Esercizio 5

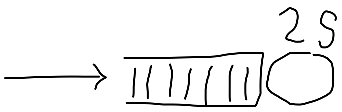


Rete aperta, se sono noti i tempi di risposte R_0, R_1, R_2, e volessimo sapere il tempo di risposta totale? Esso corrisponde a $R = 2R_0 + R_1 + R_2$. Anche se il tempo è un pò vago, le visite risolvono ogni dubbio, e ci permette di applicare la la legge del tempo di risposta. Un'idea della risposta è questa: visito il centro 0, e vado poi nel centro 1. Poi ritorno al centro 0, ma stavolta visito al centro 2. Quanto ci ho messo? il tempo espresso sopra!

Esercizio 6



sistema A; con due S separati.

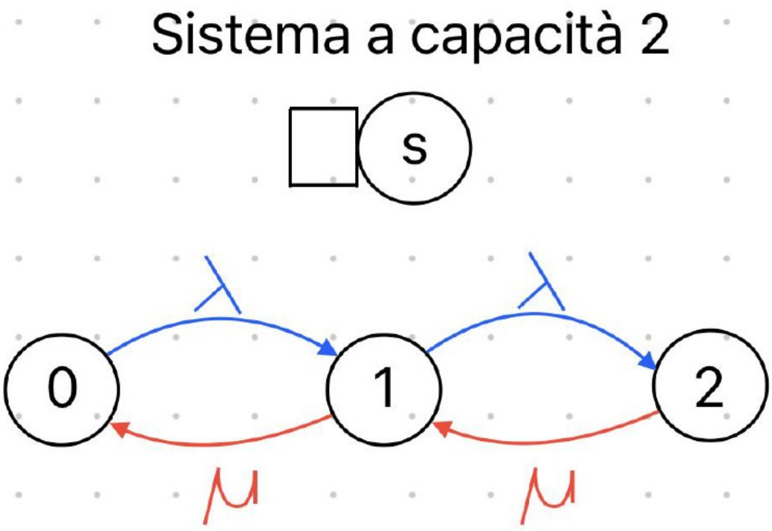


sistema B; con un unico S che va al doppio di quelli di prima!

Nel primo sistema viene specificato che la coda è di *Jackson*. Questa informazione è fondamentale, perchè essendo di Jackson, quindi centri *M/M/1* esponenziali, vale il teorema di *Burke*, ovvero ciò che esce dal primo centro entra tutto nel secondo centro. Senza questi ipotesi, non posso dirlo con tale facilità. I dati sono: $S = 0.5\ s$, $\lambda = 0.4\ trans/s$. *Quale è il valore del rapporto* $\frac{R_A}{R_B}$? Nel contesto appena descritto, il tempo per ogni singolo λ è $R = \frac{1}{\mu-\lambda} = \frac{1}{\frac{1}{s}-\lambda} = \frac{1}{2-0.4} = 5/8\ s$ per ognuno dei due centri. Allora $R_A = 2R = 5/4\ s$ Per il secondo sistema si ha $R_B = \frac{1}{1-0.4} = 5/3$ Mettendo a rapporto si ha $\frac{R_A}{R_B} = \frac{3}{4}$ Quindi $R_A = 0.75 \cdot R_B$, cioè il sistema A ha un R minore, perchè c'è meno congestione. *Due sistemi in serie usati a metà regime smaltiscono meglio di uno singolo che lavora a pieno regime.*

Esercizio 7

Sia data una coda *M/M/1/2*, ovvero di capacità 2, ovvero composta da un servente e un posto in coda. I dati sono: $\lambda = 0.5\ req/s$, $s = 0.5\ s$ *Quale è il numero medio di richiesto nel centro? E la varianza?* Trattandosi di capacità finita, si ha *stazionarietà*. Se non ci avessi fatto caso, in questo specifico esempio. non avrei avuto comunque problemi, perchè $\mu = 1/0.5 = 2$, quindi smaltisco tutto il carico. Quando la coda é **FINITA**, il mio pensiero deve andare **SEMPRE ALLA CATENA DI MARKOV**. In questo caso ci sono tre stati, come a seguire:



Dobbiamo risolvere il sistema: $\pi_0 \lambda = \pi_1 \mu \rightarrow \pi_1 = \frac{\pi_0 \cdot \lambda}{\mu} = \pi_0 \cdot 0.25$

Analogamente, seguendo gli stessi step:
$\pi_2 = \pi_1 \frac{\lambda}{\mu} = \pi_0 \frac{\lambda^2}{\mu^2} = \pi_0 \cdot 0.625$

π_0 la ottengo applicando la *Normalizzazione*, altrimenti non potrei trovarlo! $\pi_0 + \pi_1 + \pi_2 = 1 \rightarrow \pi_0 = \frac{1}{1+0.25+0.625} = 0.7619$

Allora $\pi_1 = 0.1905$ e $\pi_2 = 0.0476$

numero medio di richieste: $0 \cdot \pi_0 + 1 \cdot \pi_1 + 2\pi_2 = 0.2857 \doteq E[N_{req}]$

varianza $E[N_{req}^2] - (E[N_{req}])^2 = (1^2 \pi_1 + 2^2 \cdot \pi_2) - (0.2857)^2 = 0.29928$