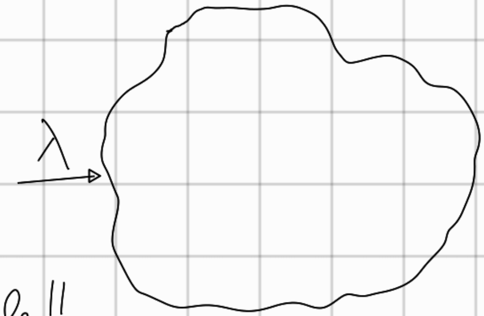
BOUND E PRESTAZIONI

SISTEMA APERTO

Ho la reseguente rete con flusso lambda, supporto che ci sia ancora capacità non usata: $U_i < 1$

Se lambda cresce, posso ancora 'accontetare il flusso',
ma poi arrivo ad un punto che: $U_i \to 1$ si satura



$\lambda_{SAT} \doteq$ il sistema diventa non stabile e saturo.

Sappiamo che $D_i = V_i \cdot S_i \doteq V_i \cdot \frac{B_i}{C_i} = \frac{B_i}{C_0}$ più facile!!

$\frac{X_i}{X_0} = \frac{C_i}{C_0}$

$X_i = X_0 V_i \overset{\text{SIST}}{\underset{\text{APERTO}}{\iff}} \lambda V_i$

uguale per tutti

$$U_i \doteq X_i \cdot S_i = \lambda V_i \cdot S_i = \lambda \cdot D_i$$

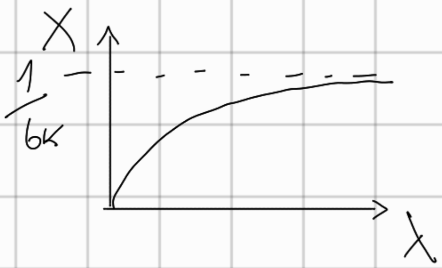
chi lo ha massimo

Bottleneck

$$V_B \cdot S_B = \max\{V_1 S_1, V_2 S_2, \dots, V_K \cdot S_K\} = D_B$$, colui che manda ad 1 l'utilizzazione

$\underset{1}{U_B} = \underset{\lambda_{SAT}}{\lambda} \underset{(V_B \cdot S_B)}{D_B}$ nel Bottleneck $\rightsquigarrow \lambda_{SAT} = \frac{1}{V_B S_B} = \frac{1}{D_B}$, non vado oltre!

graficamente ho:

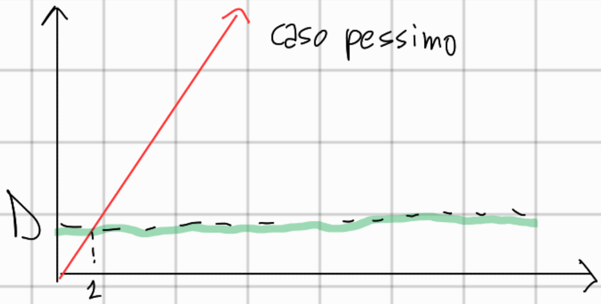


In termini di tempo di risposta? Innanzitutto il minimo possibile è quello che il job chiede!

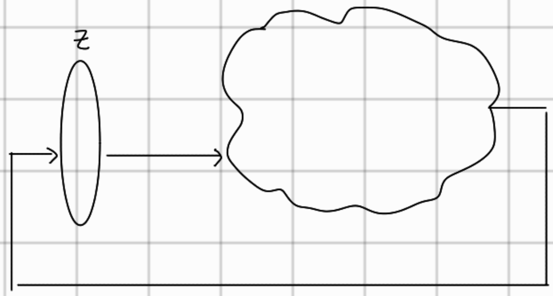
Il job ad ogni centro va Vi volte e chiede Si.
In questo caso è cose ci fosse un solo job,
non deve aspettare nessuno e fa quello che vuole senza aspettare.

$$D = \sum_{i=1}^{K} D_i$$

Il caso pessimo è quando N job vanno tutti ALLO STESSO CENTRO, quindi l'ultimo JOB deve sempre aspettare tutti gli altri (fila indiana).
Non ho un upper bound, perchè potrei avere quanti job voglio.
Quindi possiamo studiare poco qui.



sistema chiuso



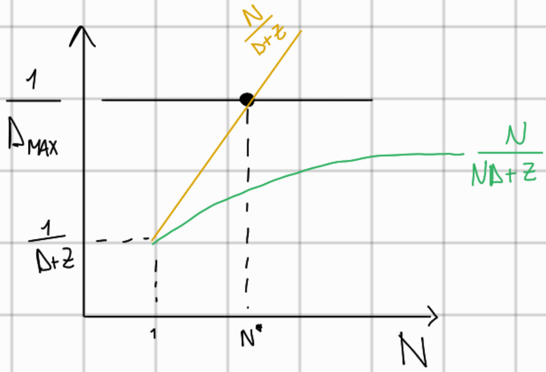
- il throughput max del sistema è dato da:

$$X_0 = \frac{X_B}{V_B} \xrightarrow{V \to 1} \frac{1}{D_B}$$

$$D_B = V_B \cdot S_B \quad \xrightarrow[\text{forzato}]{\text{flusso}} \quad X_B = V_B \cdot X_0$$

- caso carico leggero (ottimistico per carico leggero):

$$\text{throughput} = \frac{1}{D+Z} \xrightarrow[\text{non si danno fastidio}]{N\ \text{Job}} \frac{N}{D+Z}$$



$$N^* = \frac{D+Z}{D_{MAX}} = N \text{ per cui il sistema si satura}$$

Il caso peggiore si ha quando questi N job interferiscono tutti, e l'n-esimo deve attendere tutti i precedenti.

Per un job: $\frac{1}{ND+Z}$ quando sono N: $\frac{N}{ND+Z}$

quindi mi aspetto un throughput in mezzo ai bound gialli e verdi in figura.

Il bound pessimistico è : $\frac{N}{ND+Z} \leq X(N) \leq \min\left(\underbrace{\frac{1}{D_{MAX}}}_{\text{dopo } N^*}, \underbrace{\frac{N}{D+Z}}_{\text{prima di } N^*}\right)$

N* è punto di saturazione.

Il tempo si calcola con Little sfruttando questi bound.

La legge del tempo interattivo è : $R = \frac{N}{X_0} - Z$

(tempo risposta)

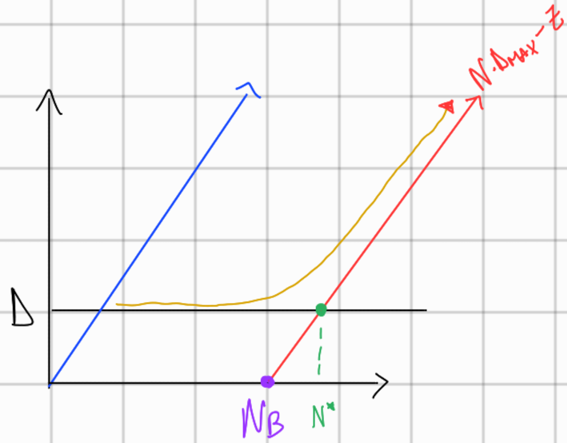
Tempo risposta inversamente proporzionare al throughput, quindi R minimo per Xo massimo, cioè Dmax.

$$N \cdot D \geq R \geq \max\left\{ D, \frac{N}{\left(\frac{1}{D_{MAX}}\right)} - Z \right\} = \max\left\{ D, N D_{MAX} - Z \right\}$$

minimo di ognuno

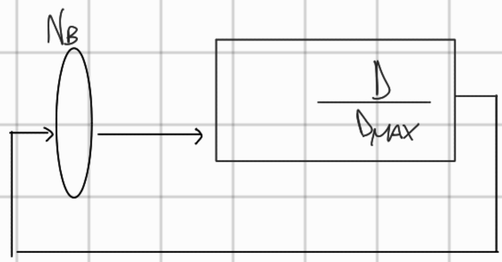
$$0 = N D_{MAX} - Z \quad (\text{retta tocca } x = \emptyset)$$

$$N_B = \frac{Z}{D_{MAX}}$$



$$N^* = \frac{D + Z}{D_{MAX}} = \underbrace{\frac{Z}{D_{MAX}}}_{N_B} + \frac{D}{D_{MAX}}$$

In corrispondenza del punto di saturazione N*, il valore di popolazione nel centro terminali è Nb, cioè numero di terminali che stanno 'pensando' quando il sistema è saturo. Il sottosistema centrale contiene $\frac{D}{D_{MAX}}$



NB: nell'analisi operazione si è usato il simbolo M = numero terminali.
Qui abbiamo sostituito N con M, N è il numero totale in questi casi.
Quando sistema è saturo sono cosi ripartiti: D/Dmax nel sistema centrale, e Z/Dmax == Nb nei terminali. Essi stanno 'pensando' mentre il sistema è saturo.