

Queueing System with Potential for Recruiting Secondary Servers

Luca Lombardo

Contents

1	Introduzione	1
1.1	Queueing Theory	1
1.2	Obiettivi del paper	2
2	Modello Matematico	3
2.1	Markovian arrival process (MAP)	3
3	QBD Approach to the Steady State Analysis	4
3.1	Description of the QBD Process Governing the System and Its Generator	4
3.2	Ergodicity Condition of the QBD Process	6
3.3	Computation of the Performance Measures of the System	7
3.4	Computation of the Performance Measures of the System	8
4	GI/M/1 Approach	9
5	Risultati numerici	10
5.1	Primo esempio numerico	11
5.2	Secondo esempio numerico	15
5.3	Esempio numerico 3	18
6	Conclusioni	20

1 Introduzione

1.1 Queueing Theory

I modelli di coda sono utilizzati per rappresentare sistemi di risorse, tradizionalmente chiamati "server", che devono essere utilizzati da diversi utenti, chiamati "clienti". La terminologia deriva da applicazioni come gli sportelli bancari, le reception degli hotel, i caselli autostradali, e così via, dove i clienti effettivamente si mettono in coda finché non vengono serviti da un dipendente. Tuttavia, la teoria delle code viene utilizzata in contesti molto più diversi.

Le code semplici consistono di un solo server che attende un solo cliente alla volta, in ordine di arrivo, con l'aggiunta dell'assunzione che i clienti siano indefinutamente pazienti. Si assume che il tempo sia discretizzato in intervalli di lunghezza fissa, che un numero casuale di clienti si unisca al sistema durante ogni intervallo e che il server rimuova un cliente dalla coda alla fine di ogni intervallo, se presente. Definendo α_n come il numero di nuovi arrivi durante l'intervallo $[n-1, n)$ e X_n come il numero di clienti nel sistema al tempo n , abbiamo

$$X_{n+1} = \begin{cases} X_n + \alpha_{n+1} - 1 & \text{se } X_n + \alpha_{n+1} \geq 1 \\ 0 & \text{se } X_n + \alpha_{n+1} = 0 \end{cases} \quad (1)$$

Se α_n è una collezione di variabili casuali indipendenti, allora X_{n+1} è condizionalmente indipendente da X_0, \dots, X_{n-1} se X_n è noto. Se, inoltre, le α_n sono identicamente distribuite, allora X_n è omogenea. Lo spazio degli stati è \mathbb{N} e la matrice di transizione è

$$P = \begin{pmatrix} q_0 + q_1 & q_2 & q_3 & q_4 & \dots \\ q_0 & q_1 & q_2 & q_3 & \ddots \\ \vdots & q_0 & q_1 & q_2 & \ddots \\ 0 & & \ddots & \ddots & \ddots \end{pmatrix} \quad (2)$$

dove q_i è la probabilità $P[\alpha = i]$ che i nuovi clienti che entrino in coda durante un intervallo di un'unità di tempo, mentre α denota ognuna delle possibili distribuzioni di α_n identicamente distribuite. Le catene di Markov aventi matrice di transizione della forma

$$P = \begin{pmatrix} B_1 & B_2 & B_3 & B_4 & \dots \\ A_0 & A_1 & A_2 & A_3 & \ddots \\ & A_0 & A_1 & A_2 & \ddots \\ 0 & & \ddots & \ddots & \ddots \end{pmatrix} \quad (3)$$

dove $A_i, B_{i+1}, i \geq 0$ sono matrici non negative di dimensione $k \times k$ sono dette M/G/1-type Markov Chains e sono utilizzate per modellare svariati problemi di coda.

1.2 Obiettivi del paper

Il paper presenta un nuovo approccio per migliorare i modelli di coda con l'utilizzo di server secondari temporanei, reclutati tra i clienti stessi. Questi server secondari sono disponibili solo temporaneamente e forniranno servizi in gruppi di diverse dimensioni. Dopo aver servito esattamente un gruppo, i server secondari lasceranno il sistema, permettendo ai clienti di proseguire le loro attività senza essere tratti in considerazione. Il contributo principale del paper è l'introduzione del concetto di reclutamento di server secondari da parte dei clienti, in modo da aiutare il sistema. I risultati numerici indicano che il modello proposto funziona meglio del modello di coda classico corrispondente. Questo può aiutare i responsabili del sistema a reclutare server secondari quando necessario per migliorare le prestazioni del sistema.

Il paper analizza anche altri approcci di modelli di coda con server secondari presenti in letteratura. Tuttavia, andremo a considerare le due seguenti caratteristiche che sono intrinseche in alcuni sistemi del mondo reale e non sono state studiate in passato: (i) un server secondario verrà assegnato ad un gruppo (che non supererà una soglia finita prestabilita); questo server offrirà i servizi uno alla volta; e una volta che tutti i clienti assegnati sono stati serviti, il server secondario lascerà anche il sistema; e (ii) con una certa probabilità, un cliente servito da un server secondario diventa insoddisfatto e quindi torna al sistema principale per ottenere un nuovo servizio.

2 Modello Matematico

Consideriamo un sistema di coda a singolo server in cui gli arrivi avvengono secondo un processo di arrivo markoviano (MAP) con matrici di parametro (D_0, D_1) di ordine m . Il MAP generalizza alcuni dei processi puntiformi ben noti come Poisson, Poisson interrotto e rinnovamenti di tipo fase, tra gli altri. Inoltre, MAP è ideale in situazioni in cui può essere presente una correlazione nei tempi tra gli arrivi. Supponiamo che gli arrivi provengano da diverse fonti in un'area comune per il trattamento. Anche se tutte le singole fonti generano arrivi secondo processi di rinnovo, quello combinato potrebbe non essere necessariamente un processo di rinnovo (a meno che tutte le singole fonti siano processi di Poisson). Un'altra parte attraente dell'uso di MAP è che l'analisi richiede il formalismo delle matrici e le ragioni intuitive associate all'analisi.

2.1 Markovian arrival process (MAP)

Il generatore irriducibile del MAP è dato da $D_0 + D_1$. Sia δ il vettore invariante tale che

$$\delta(D_0 + D_1) = \mathbf{0}, \quad \delta e = 1 \quad (4)$$

Dove d'ora in poi, e è il vettore colonna di tutti gli elementi 1 con appropriata dimensione mentre $\mathbf{0}$ rappresenta il vettore riga di tutti zeri con dimensioni appropriate. La matrice D_0 governa le transizioni corrispondenti al generatore sottostante che non produce arrivi, mentre la matrice D_1 governa quelle transizioni corrispondenti agli arrivi nel sistema.

Il rate medio di arrivi (λ), la varianza degli tempi interni di arrivo (σ^2) e la correlazione (ρ_c) tra due successivi tempi interni di arrivo sono dati da

$$\lambda = \delta D_1 e, \quad \sigma^2 = \frac{2}{\lambda} \delta (-D_0)^{-1} e - \frac{1}{\lambda^2}, \quad \rho_c = \frac{\lambda \delta (-D_0)^{-1} D_1 (-D_0)^{-1} e - 1}{2 \lambda \delta (-D_0)^{-1} e - 1} \quad (5)$$

Il sistema ha un singolo server che offre servizi in modo FCFS. Questo server sarà chiamato server principale. I tempi di servizio sono esponenziali con parametro μ_1 . Con probabilità $p, 0 \leq p \leq 1$, un cliente servito può essere reclutato (o optato dal punto di vista del cliente servito) per servire altri clienti in attesa nel sistema (assumendo che la dimensione della coda sia positiva) a condizione che non ci sia già un altro server secondario che sta servendo. Un tale server è chiamato server secondario. In altre parole, una reclutamento avviene solo quando c'è almeno un cliente in attesa nella coda e quando non c'è altro server secondario presente nel sistema. Pertanto, il sistema può avere al massimo due server in qualsiasi momento. Si noti che con probabilità $q = 1 - p$, il cliente servito, che può diventare il server secondario, non accetta di farlo e lascia il sistema. Quando viene reclutato un server secondario, il server verrà assegnato a un gruppo di, diciamo, i clienti, dove $i = \min\{\text{numero nella coda}, L\}$, dove L è un pre-determinato positivo finito intero. In altre parole, $1 \leq L < \infty$. Il server secondario offrirà servizi ai clienti del gruppo uno alla volta e i tempi di servizio sono distribuiti in modo esponenziale con parametro μ_2 . Un cliente che riceve un servizio da un server secondario potrebbe non essere soddisfatto del servizio ricevuto e richiedere di essere servito di nuovo con probabilità $v, 0 \leq v \leq 1$, e con probabilità $\bar{v} = 1 - v$ lascerà il sistema. I clienti insoddisfatti sono reinseriti nel sistema. Una volta che il server secondario ha finito di servire tutti i clienti assegnati, il sistema rilascerà questo server. Si noti che prendendo $p = 0$ (in questo caso v non ha alcun ruolo e può essere ignorato), otteniamo il modello di coda a singolo server classico. Questo caso viene utilizzato solo come verifica dell'accuratezza nei calcoli numerici e non è altrimenti interessante. Il caso in cui $v = 1$ non è interessante poiché in questo caso ogni cliente servito da un server secondario viene reinserito nel sistema e l'assunzione di server secondari rallenta solo il sistema nell'offrire servizi.

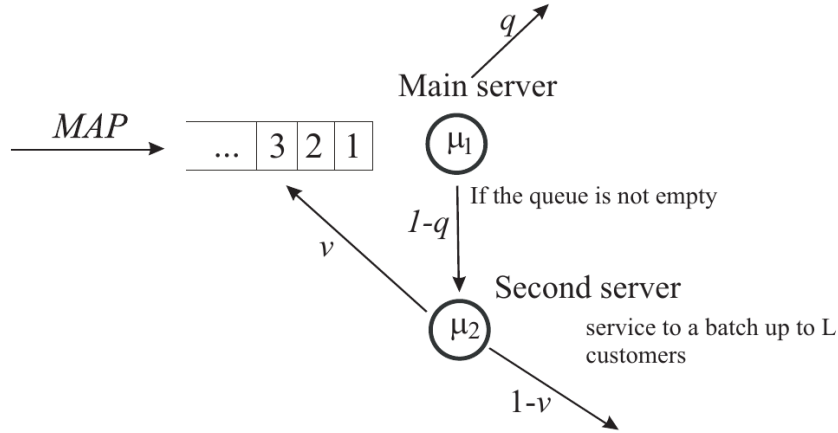


Figure 1: Structure of the system under study

3 QBD Approach to the Steady State Analysis

Analizzeremo il modello di coda in studio in stato stazionario. L'analisi può essere effettuata tramite il processo QBD o tramite un tipo GI/M/1. In questa sezione, adotteremo l'approccio QBD, mentre nella prossima sezione evidenzieremo brevemente l'altro approccio. Come è noto, il processo QBD è un caso particolare della catena di Markov a tempo continuo (CTMC). *Concetto dato per buono*

3.1 Description of the QBD Process Governing the System and Its Generator

Supponiamo che, al tempo $t \geq 0$, indichiamo:

- il numero di clienti nel sistema come $i_t \geq 0$;
- il numero di clienti in servizio al server secondario come $n_t \in \{0, \dots, \min(i_t, L)\}$ (notare che quando $n_t = 0$, il sistema non ha un server secondario);
- lo stato del processo sottostante del MAP che descrive gli arrivi dei clienti come $\xi_t = 1, \dots, m$.

Allora, il processo stocastico $\{\zeta_t = (i_t, n_t, \xi_t), t \geq 0\}$ che descrive il comportamento del modello in esame è un CTMC regolare e irriducibile. Enumerando gli stati del CTMC, $\{\zeta_t, t \geq 0\}$, in ordine lessicografico e indicando con i il livello, per $i \geq 0$, l'insieme di stati come $\{(i, n, k) : 0 \leq n \leq \min(i, L), 1 \leq k \leq m\}$, il generatore (infinitesimale), Q , di questo CTMC è dato dal seguente teorema.

Teorema 3.1. *Il generatore infinitesimale Q del processo stocastico CTMC $\{\zeta_t, t \geq 0\}$ ha una struttura a blocchi tridiagonale*

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & O & O & \dots & O & O & O & O & O & \dots \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & O & \dots & O & O & O & O & O & \dots \\ O & Q_{2,1} & Q_{2,2} & Q_{2,3} & \dots & O & O & O & O & O & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ O & O & O & O & \dots & Q_{L,L-1} & Q_{L,L} & Q^+ & O & O & \dots \\ O & O & O & O & \dots & O & Q^- & Q^0 & Q^+ & O & \dots \\ O & O & O & O & \dots & O & O & Q^- & Q^0 & Q^+ & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

Dove i blocchi $Q_{i,i}$ non nulli sono definiti come segue

$$\begin{aligned} Q_{0,0} &= D_0, \\ Q_{i,i} &= I_{i+1} \otimes D_0 + \nu \mu_2 E_i^- \otimes I_m - (\mu_1 \hat{I}_i + \mu_2 (I_{i+1} - \bar{I}_i)) \otimes I_m, \quad 1 \leq i \leq L, \\ Q_{i,i} &= Q^0 = I_{L+1} \otimes D_0 + \nu \mu_2 E_L^- \otimes I_m - (\mu_1 I_{L+1} + \mu_2 (I_{L+1} - \bar{I}_L)) \otimes I_m, \quad i > L, \\ Q_{i,i+1} &= E_i^+ \otimes D_1, \quad 0 \leq i \leq L-1, \\ Q_{i,i+1} &= Q^+ = I_{L+1} \otimes D_1, \quad i \geq L, \\ Q_{1,0} &= (1-\nu) \mu_2 \bar{E}_1^- \otimes I_m + \mu_1 I_1^- \otimes I_m, \quad 1 \leq i \leq L, \\ Q_{i,i-1} &= (1-\nu) \mu_2 \bar{E}_i^- \otimes I_m + q \mu_1 I_i^- \otimes I_m + (1-q) \mu_1 I_i^+ \otimes I_m, \quad 1 \leq i \leq L, \\ Q_{i,i-1} &= Q^- = (1-\nu) \mu_2 E_L^- \otimes I_m + q \mu_1 I_{(L+1)m}^- + (1-q) \mu_1 I^+ \otimes I_m, \quad i > L. \end{aligned}$$

Dove si usa la seguente notazione:

- O and I are, respectively, zero and identity matrices of appropriate dimensions as indicated in the suffix;
- \otimes indicates the Kronecker product of matrices (see, e.g., [51–54]);
- E_l^+ is a matrix of dimension $(l+1) \times (l+2)$ with $(E_l^+)_{k,k} = 1, 0 \leq k \leq l$, and all other entries are zero;
- E_l^- is a square matrix of dimension $l+1$ with $(E_l^-)_{k,k-1} = 1, 1 \leq k \leq l$, and all other entries are zero;
- \hat{I}_l is a square matrix of dimension $l+1$ with $(\hat{I}_l)_{k,k} = 1, 0 \leq k \leq l-1$, and all other entries are zero;
- I_l is a square matrix of dimension $l+1$ with $(I_l)_{0,0} = 1$, and all other entries are zero;
- \bar{E}_l^- is a matrix of dimension $(l+1) \times l$ with $(\bar{E}_l^-)_{k,k-1} = 1, 1 \leq k \leq l$, and all other entries are zero;
- I_l^- is the matrix of dimension $(l+1) \times l$ with $(I_l^-)_{k,k} = 1, 0 \leq k \leq l-1$, and all other entries are zero;
- I_l^+ is the matrix of dimension $(l+1) \times l$ with $(I_l^+)_{0,l-1} = 1, (I_l^+)_{k,k} = 1, 1 \leq k \leq l-1$, and all other entries are zero;
- I^+ is the matrix of dimension $(L+1) \times (L+1)$ with $(I^+)_{k,k} = 1, 1 \leq k \leq L, (I^+)_{0,L} = 1$, and all other entries are zero.

Proof. Immediata □

3.2 Ergodicity Condition of the QBD Process

Teorema 3.2. *Il processo stocastico CTMC $\{\zeta_t, t \geq 0\}$ è ergodico se e solo se vale la seguente disuguaglianza:*

$$\lambda < \mu_1 + \mu_2(1 - v) \frac{L(1 - q)\mu_1}{L(1 - q)\mu_1 + \mu_2} \quad (6)$$

Proof. È noto, grazie all'approccio matriciale-geometrico di Neuts (vedi, ad esempio, il riferimento [37]), che il criterio per l'ergodicità del QBD con il generatore di forma data in (3) è la soddisfazione dell'ineguaglianza:

$$yQ^-e > yQ^+e \quad (7)$$

dove il vettore y è l'unica soluzione del sistema

$$y(Q^- + Q^0 + Q^+) = \mathbf{0}, \quad ye = 1 \quad (8)$$

Si può inoltre verificare facilmente che

$$Q^- + Q^0 + Q^+ = I_{L+1} \otimes (D_0 + D_1) + S \otimes I_m \quad (9)$$

dove Usando la regole del mixed product per il prodotto di Kronecker, ed usando 4 si verifica che la

$$S = \begin{pmatrix} -\mu_1(1 - q) & 0 & 0 & \dots & 0 & \mu_1(1 - q) \\ \mu_2 & -\mu_2 & 0 & \dots & 0 & 0 \\ 0 & \mu_2 & -\mu_2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \mu_2 & -\mu_2 \end{pmatrix}.$$

soluzione del sistema di equazioni lineari è

$$y = x \otimes \delta \quad (10)$$

dove δ è come definita in 4 ed x è la soluzione del sistema

$$xS = 0, \quad xe = 1 \quad (11)$$

per sostituzione diretta, si verifica facilmente che le componenti del vettore $x = (x_0, x_1, \dots, x_L)$, corrispondenti alle uniche soluzioni del sistema 11, sono date da

$$x_0 = \frac{\mu_2}{L(1 - q)\mu_1 + \mu_2}, \quad x_i = \frac{\mu_1(1 - q)}{L(1 - q)\mu_1 + \mu_2}, \quad i = 1, \dots, L \quad (12)$$

La tesi segue dalle equazioni 7, 9 e 12 assieme alla definizione di λ . □

Osservazione 3.3. *La condizione di stabilità data nell'Equazione 7 può essere intuitivamente spiegata nel seguente modo. In generale, la condizione di ergodicità richiede che il tasso di arrivo dei clienti per unità di tempo debba essere inferiore al tasso di servizio che i clienti ricevono per unità di tempo quando il sistema è sovraccarico (nel senso che il numero di clienti presenti nel sistema è molto grande). Qui, il tasso di arrivo dei clienti è λ per unità di tempo. Il tasso di servizio dei clienti quando il sistema è sovraccarico è la somma di μ_1 (il tasso di servizio per unità di tempo fornito dal server principale) e il tasso di servizio (per unità di tempo) fornito dal server secondario. Quest'ultimo tasso di servizio è 0 quando il server secondario non è presente nel sistema, il che avviene con probabilità x_0 . Quando il server secondario è presente nel sistema,*

che avviene con probabilità $(1 - x_0)$, i clienti ricevono il servizio e lasciano il sistema ad un tasso di $\mu_2(1 - \nu)$ per unità di tempo. Pertanto, il tasso di servizio medio totale è dato da:

$$\mu = \mu_1 + \mu_2(1 - \nu) \frac{L(1 - q)\mu_1}{L(1 - q)\mu_1 + \mu_2} \quad (13)$$

da cui segue che la condizione di ergodicità vista in 7

Osservazione 3.4. La probabilità, x_0 , che il secondo server non sia presente nel sistema in un qualsiasi momento in cui il sistema è sovraccarico può essere facilmente calcolata dalla seguente considerazione. Si considerino i periodi in cui il server secondario non è presente nel sistema (ovvero, la durata media di questo periodo è $\frac{1}{\rho\mu_1}$) alternati ai periodi in cui il server secondario è presente nel sistema. Quando il sistema attiva un server secondario (quando il sistema è sovraccarico, il server secondario viene assegnato a gestire L per i servizi), la durata media del server secondario continuamente presente nel sistema è data da $\frac{L}{\mu_2}$. Pertanto, abbiamo:

$$x_0 = \frac{\frac{1}{\mu_1(1-q)}}{\frac{1}{\mu_1(1-q)} + \frac{L}{\mu_2}} = \frac{\mu_2}{L(1-q)\mu_1 + \mu_2} \quad (14)$$

che corrisponde all'espressione 12 vista in precedenza.

3.3 Computation of the Performance Measures of the System

Sotto l'assunzione che la condizione di ergodicità data dalla relazione 7 sia valida, esistono le seguenti probabilità stazionarie degli stati del CTMC ζ_t , $t \geq 0$:

$$\pi(i, n, \zeta) = \lim_{t \rightarrow \infty} P\{i_t = i, n_t = n, \zeta_t = \zeta, i \geq 0, n \in \{0, 1, \dots, \min\{i, L\}\}, \zeta \in \{0, \dots, n\} \quad (15)$$

Consideriamo i vettori riga delle probabilità di stato stazionario π_i come segue: il vettore riga $\pi(i, n)$ è dato da $\pi(i, n) = (\pi(i, n, 1), \dots, \pi(i, n, m))$ e

$$\pi_i = (\pi(i, 0), \dots, \pi(i, \min\{i, L\})), \quad i \geq 0 \quad (16)$$

È ben noto che i vettori di probabilità stazionari π_i , $i \geq 0$, soddisfano il sistema di equazioni algebriche lineari (equazioni di equilibrio):

$$(\pi_0, \pi_1, \pi, \dots)Q = 0 \quad (\pi_0, \pi_1, \pi, \dots)e = 1 \quad (17)$$

dove Q è la matrice di transizione del CTMC ζ_t , $t \geq 0$ e e è il vettore colonna di tutti gli elementi 1. La soluzione del problema di calcolo della distribuzione stazionaria di una QBD indipendente dal livello è ben nota; si veda [37]. Per i livelli in cui le transizioni della QBD non dipendono dal livello, i vettori di probabilità stazionaria sono trovati in forma matriciale geometrica. I vettori di probabilità stazionaria dei livelli di confine, in cui le transizioni della QBD dipendono dal livello, sono quindi direttamente trovati come soluzione del sistema di equazioni algebriche lineari. Tuttavia, se il numero di livelli di confine è grande (cosa che accade nel nostro modello se L è grande), questo sistema ha una grande dimensione. Qui presentiamo un algoritmo che sfrutta essenzialmente la struttura tridiagonale a blocchi ma dipendente dal livello del generatore per i livelli minori di $L + 1$. L'algoritmo utilizzato per risolvere il sistema infinito di equazioni di equilibrio è presentato nella seguente teorema:

Teorema 3.5. I vettori π_i , $i \geq 0$, sono trovati come soluzione del sistema di equazioni algebriche lineari:

$$\pi_i = \alpha_i \left(\sum_{l=0}^{\infty} \alpha_l e \right)^{-1}, \quad i \geq 0 \quad (18)$$

dove il vettore α_0 è calcolato come l'unica soluzione del sistema di equazioni

$$\alpha_0(Q_{0,0} + Q_{0,1}G_0) = 0, \quad \alpha_0 e = 1 \quad (19)$$

ed i vettori $\alpha_i, i \geq 1$, sono definiti come

$$\alpha_i = \alpha_0 \prod_{l=1}^i R_l, \quad i \geq 1 \quad (20)$$

o tramite la formula ricorsiva

$$\alpha_i = \alpha_{i-1} R_i, \quad i \geq 1 \quad (21)$$

dove

$$R = \begin{cases} -Q_{i-1,i}(Q_{i,i} + Q_{i,i+1}G_i)^{-1}Q & 1 \leq i \leq L-1 \\ -Q_{L-1,L}(Q_{L,L} + Q^+G)^{-1} & i = L \\ -Q^+(Q^0 + Q^+G)^{-1} = R & i > L \end{cases} \quad (22)$$

Le matrici stocastiche G_i sono calcolate utilizzando la seguente formula ricorsiva all'indietro:

$$\begin{aligned} G_L &= G \\ G_L - 1 &= -(Q_{L,L} + Q^+G_L)^{-1}Q_{L,L-1} \\ G_i &= -(Q_{i+1,i+1} + Q_{i+1,i+2}G_{i+1})^{-1}Q_{i+1,i}, \quad i = L-2, L-3, \dots, 0 \end{aligned} \quad (23)$$

dove la matrice G è la minima soluzione non negativa dell'equazione quadratica matriciale

$$Q^+G^2 + Q^0G + Q^- = 0 \quad (24)$$

Proof. non fornita □

Questo algoritmo è una modifica efficace dell'algoritmo per il calcolo della distribuzione stazionaria del CTMC asintoticamente quasi-Toeplitz (vedi, ad esempio, [31], pp. 145-146). In [31], i vettori π_i sono calcolati come $\pi_i = \pi_0 F_i, i \geq 0$, dove le matrici F_i sono ottenute dalla ricorsione di matrici simile all'Equazione 21. Utilizzando la ricorsione di vettori come indicato nell'Equazione 21 invece della ricorsione di matrici corrispondente, si ottiene una significativa riduzione della memoria del computer richiesta e del tempo di esecuzione. L'esistenza delle inverse delle matrici (tutte sub-generatori irriducibili) che appaiono nell'algoritmo sopra segue immediatamente, ad esempio, dal teorema di O. Tausska [55]. Inoltre, queste matrici sono semi-stabili (e quindi le inverse dei negativi di queste matrici sono non negative), risultando nella produzione di procedure ricorsive stabili nell'implementazione numerica dell'algoritmo.

Corollario 3.6. Per $i \geq L$ vale la seguente formula

$$\alpha_i = \alpha_L R^{i-L} \quad (25)$$

dove

$$\alpha_L = \alpha_0 \prod_{l=1}^L R_l \quad (26)$$

3.4 Computation of the Performance Measures of the System

Per studiare il modello di coda in questione qualitativamente e confrontarlo con la corrispondente coda classica MAP/M/1 per valutare l'impatto del processo di reclutamento, dobbiamo sviluppare alcune misure di performance chiave. Di seguito sono elencate alcune di queste insieme alle loro formule:

4 GI/M/1 Approach

In questa sezione, presentiamo brevemente come analizzare il sistema di coda in studio utilizzando code di tipo GI/M/1 a tempo continuo. Tenendo traccia del numero di clienti in attesa nella coda insieme allo stato del server principale (occupato o libero) e allo stato del server secondario (non presente o presente con un numero specificato di clienti assegnati), possiamo studiare il modello come una CTMC di tipo GI/M/1 come segue.

Definiamo come prima cosa lo spazio degli stati Ω del CTMC come:

$$\Omega = \{(i, j, k) : i \geq 0, 0 \leq j \leq K, 1 \leq k \leq m\} \quad (27)$$

In seguito, consideriamo e_r come un vettore colonna con 1 nella posizione r -esima e 0 altrove. Notare che quando necessario, indicheremo la dimensione tra parentesi. Ad esempio, $e(L+1)$ indicherà un vettore colonna di 1 con dimensione $L+1$. La "T" che appare come pedice in un vettore o una matrice sta per la notazione di trasposizione. Quindi, e^T indicherà un vettore riga di 1.

Definiamo il livello $\mathbf{i} = \{(i, j, k) : 0 \leq j \leq L, 1 \leq k \leq m\} = \{(\mathbf{i}, 0), \dots, (\mathbf{i}, L)\}, i \geq 0$. Notare che il livello (\mathbf{i}, \mathbf{j}) indica che il server principale è occupato (a patto che $i > 0$), ci sono $i-1$ clienti in attesa nella coda principale, il server secondario (a patto che $j > 0$) è occupato e il processo di arrivo si trova in varie fasi. Il livello $(\mathbf{0}, \mathbf{0})$ corrisponde al sistema inattivo con il processo MAP in una delle m fasi. Il generatore \tilde{Q} della CTMC che governa il sistema in studio è dato da:

$$\tilde{Q} = \begin{pmatrix} B_0 & A_0 & & & & & & & \\ B_1 & A_1 & A_0 & & & & & & \\ B_2 & A_2 & A_1 & A_0 & & & & & \\ \vdots & & \ddots & \ddots & \ddots & & & & \\ B_L & & & & A_2 & A_1 & A_0 & & \\ B_{L+1} & & & & A_2 & A_1 & A_0 & & \\ & A_{L+2} & & & & A_2 & A_1 & A_0 & \\ & & A_{L+2} & & & & A_2 & A_1 & A_0 \\ & & & \ddots & & & & \ddots & \ddots \end{pmatrix},$$

where

$$B_0 = \begin{pmatrix} D_0 & & & & \\ \tilde{v}\mu_2 I & D_0 - \mu_2 I & & & \\ & \tilde{v}\mu_2 I & D_0 - \mu_2 I & & \\ & & \ddots & \ddots & \\ & & & \tilde{v}\mu_2 I & D_0 - \mu_2 I \end{pmatrix},$$

$$A_0 = \begin{pmatrix} D_1 & & & & \\ v\mu_2 I & D_1 & & & \\ & v\mu_2 I & D_1 & & \\ & & \ddots & \ddots & \\ & & & v\mu_2 I & D_1 \end{pmatrix}, \quad A_1 = B_0 - \mu_1 I,$$

$$A_2 = \mu_1 \Delta(q, 1, \dots, 1), \quad B_1 = \mu_1 I,$$

$$B_r = p\mu_1(e_r^T \otimes e(L+1)), \quad 2 \leq r \leq L+1, \quad A_{L+2} = B_{L+1},$$

GUARDARE PAPER, TROPPO LUNGO

5 Risultati numerici

In questa sezione, forniamo alcuni esempi illustrativi utilizzando cinque diversi processi di arrivo. Di questi cinque, tre sono processi di rinnovo e due sono processi correlati. In particolare, prendiamo i cinque MAP come:

- **ERL:** Questo è un Erlang di ordine 5 con parametro 2.5 in ciascuno dei 5 stati. Notare che qui abbiamo $\lambda = 0.5, \sigma = 0.899427$ e $\rho_c = 0$.
- **EXP:** Questo è un esponenziale con una frequenza di 0.5. Notare che qui abbiamo $\lambda = 0.5, \sigma = 2$ e $\rho_c = 0$.
- **HEX:** Questa è una distribuzione iper-esponenziale con una probabilità di mixing data da (0.5, 0.3, 0.15, 0.04, 0.01) con i corrispondenti tassi della distribuzione esponenziale pari a (1.09, 0.545, 0.2725, 0.13625, 0.068125). Qui abbiamo $\lambda = 0.5, \sigma = 3.3942$ e $\rho_c = 0$.

I due processi correlati, negativo e positivo, sono i seguenti:

- **NCR:** Questo è un MAP negativamente correlato, con matrici di rappresentazione date da: dove

$$D_0 = \begin{pmatrix} -1.125 & 1.125 & 0. & 0. & 0. \\ 0. & -1.125 & 1.125 & 0. & 0. \\ 0. & 0. & -1.125 & 1.125 & 0. \\ 0. & 0. & 0. & -1.125 & 0. \\ 0. & 0. & 0. & 0. & -2.25 \end{pmatrix},$$

$$D_1 = \begin{pmatrix} 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. \\ 0.01125 & 0. & 0. & 0. & 1.11375 \\ 2.2275 & 0. & 0. & 0. & 0.0225 \end{pmatrix}.$$

abbiamo $\lambda = 0.5, \sigma = 2.02454$ e $\rho_c = -0.57855$.

- **PCR:** Questo è un MAP positivamente correlato, con matrici di rappresentazione date da: dove abbiamo $\lambda = 0.5, \sigma = 2.02454$ e $\rho_c = 0.57855$.

$$D_0 = \begin{pmatrix} -1.125 & 1.125 & 0. & 0. & 0. \\ 0. & -1.125 & 1.125 & 0. & 0. \\ 0. & 0. & -1.125 & 1.125 & 0. \\ 0. & 0. & 0. & -1.125 & 0. \\ 0. & 0. & 0. & 0. & -2.25 \end{pmatrix},$$

$$D_1 = \begin{pmatrix} 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. \\ 1.11375 & 0. & 0. & 0. & 0.01125 \\ 0.0225 & 0. & 0. & 0. & 2.2275 \end{pmatrix}.$$

Guardando le cinque MAP sopra riportate, è evidente che sono tutte qualitativamente diverse. È importante sottolineare che il processo di arrivo denominato PCR è ideale per situazioni in cui gli arrivi dei clienti sono altamente irregolari, con periodi alternati di congestione e di scarsità del sistema. Tali arrivi sono comuni nella pratica, specialmente nelle telecomunicazioni e nelle industrie dei servizi. È importante notare, inoltre, che il processo di arrivo denominato HEX è noto per presentare un comportamento irregolare simile nel senso che gli arrivi con tempi tra di essi più brevi sono separati da tempi più lunghi. Tuttavia, la differenza tra questi due processi sta nella correlazione positiva presente nel processo PCR. L'impatto di questa correlazione positiva, così come dell'elevata variabilità nei tempi tra gli arrivi, come nei due processi sopra citati, è stato ben documentato in letteratura (vedi, ad esempio, riferimenti [29,30]). Discutiamo tre esempi numerici rappresentativi e illustrativi per evidenziare la natura qualitativa del modello in studio.

5.1 Primo esempio numerico

Qui discutiamo l'impatto del parametro L su alcune misure di performance del sistema selezionate per tutte e cinque le MAP. Innanzitutto, fissiamo $\mu_1 = 1$, $\mu_2 = 0.5$, $q = 0.5$, e $\nu = 0.4$, e variamo L da 1 a 30.

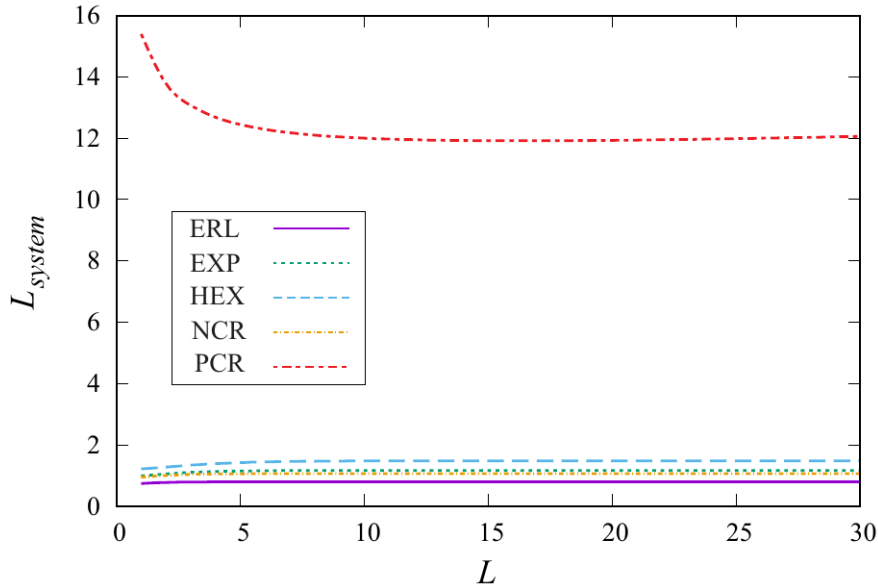


Figure 2: Impact of L on the average number of customers in the system L_{system} for different MAPs.

La Figura 2 illustra chiaramente l'effetto dell'irregolarità nel processo di arrivo, ovvero PCR. Il numero medio di clienti nel sistema nel caso di PCR è molte volte maggiore rispetto agli altri MAPs. Vale la pena sottolineare che per i primi quattro MAPs, la misura L_{system} è una funzione non decrescente di L , mentre per PCR si osserva un trend non crescente. Ciò spiega il ruolo della correlazione, soprattutto positiva, e non dovrebbe essere ignorato. Inoltre, un valore elevato di L indica che quando un server secondario viene reclutato, verranno assegnati più clienti e, a causa della lentezza del server secondario (rispetto al server principale), c'è una alta probabilità, soprattutto per i casi dei primi quattro MAPs, che il sistema abbia in media più clienti nel sistema. Similmente a quanto noto nella coda classica, ovvero il numero medio nel sistema aumenta con l'aumento della variabilità nei tempi di arrivo tra gli arrivi di rinnovo, vediamo che questo comportamento si verifica anche qui nei primi tre MAPs, che corrispondono agli arrivi di rinnovo.

Tuttavia, per quanto riguarda gli arrivi PCR, osserviamo un trend interessante ma opposto, ovvero un trend decrescente. Questo può essere intuitivamente spiegato come segue. Innanzitutto, si osserva che il sistema L ha un valore massimo di 15.3983 quando $L = 1$, il che può essere spiegato utilizzando il fatto che, quando $L = 1$, i server secondari lasciano il sistema dopo aver servito un solo cliente; con una probabilità del solo 0.5 di essere reclutati, la coda tende ad accumularsi rapidamente. Aumentando L , i server secondari sono maggiormente coinvolti nella pulizia della coda, soprattutto quando gli arrivi avvengono a sprazzi, e quindi L_{system} diminuisce. Raggiunge un valore minimo di 11.9757 quando $L = 16$ e poi inizia ad aumentare a causa della mancata possibilità di essere serviti dal server principale. Per $L = 30$, $L_{\text{system}} = 12.0605$

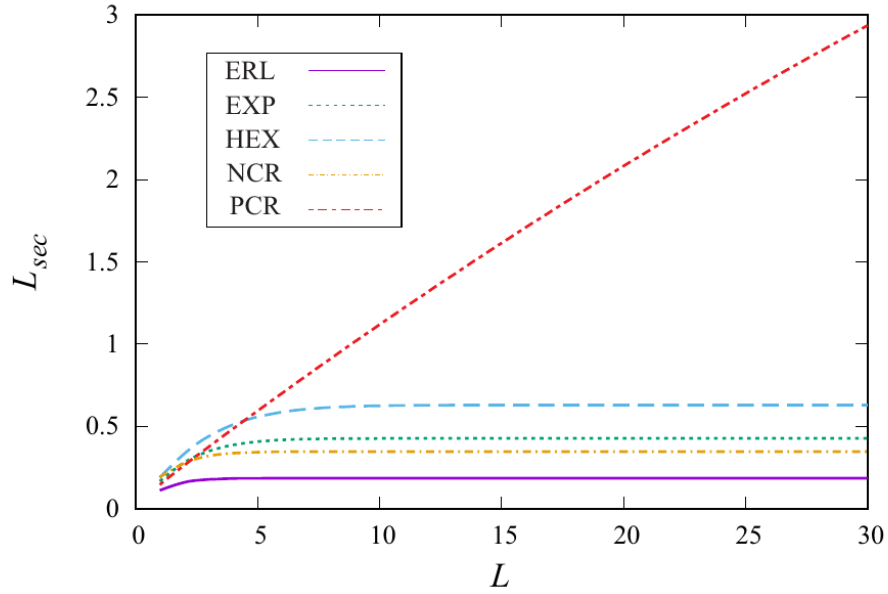


Figure 3: Dependence of the average number of customers with the secondary server L_{sec} on the parameter L for different MAPs.

La Figura 3 mostra il comportamento della media del numero di clienti con il server secondario L_{sec} .

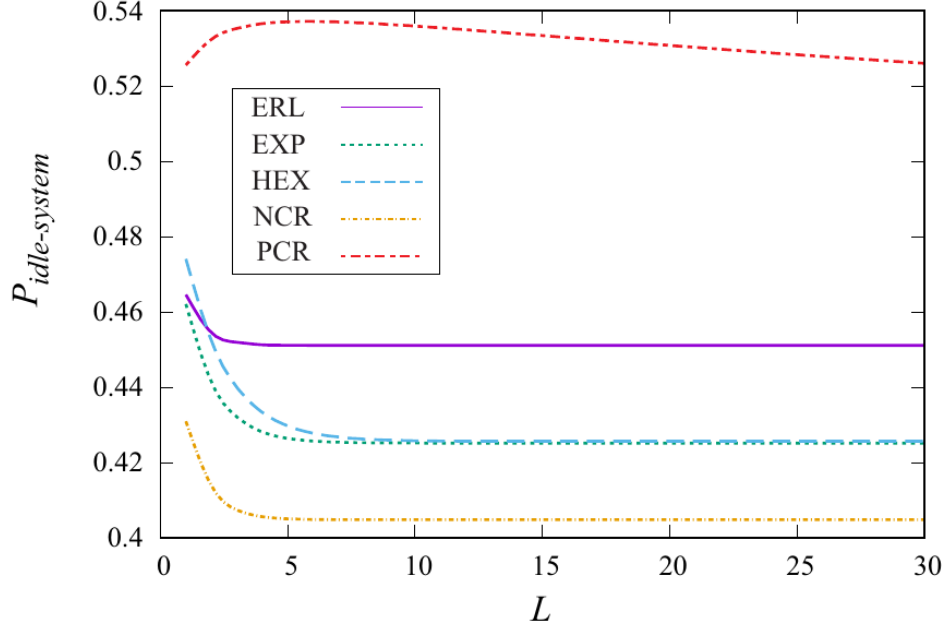


Figure 4: Dependence of the probability $P_{\text{idle-system}}$ that the system is idle at an arbitrary moment on the parameter L for different MAPs

La Figura 4 illustra il comportamento della probabilità, $P_{\text{idle-system}}$, che il sistema sia inattivo in un momento arbitrario. Questa figura corrisponde alla Figura 2 su due aspetti. Il primo è che mostra anche una grande differenza nella misura quando viene confrontata con vari MAPs. Quando si cerca di trovare un valore ottimale di L , è evidente che conta quale misura viene scelta come funzione obiettivo e il tipo di MAPs utilizzato quando tutti gli altri parametri sono fissati. Ad esempio, se si considera il processo di arrivo PCR, il valore ottimale di L è 16 se si cerca di minimizzare L_{system} . Tuttavia, se la misura $P_{\text{idle-system}}$ è l'obiettivo del problema di ottimizzazione, allora $L=6$ produce il valore più grande per questa misura.

Le Figure 5 e 6 illustrano il comportamento delle probabilità $P_{\text{idle-busy}}$ e $P_{\text{busy-idle}}$, che corrispondono rispettivamente al momento in cui il server principale è inattivo con il server secondario occupato, e al momento in cui il server principale è occupato con il server secondario inattivo, in un momento arbitrario. Mentre la prima probabilità aumenta all'aumentare di L , la seconda probabilità diminuisce. Da queste figure, si possono notare le differenze essenziali in queste probabilità in vari scenari.

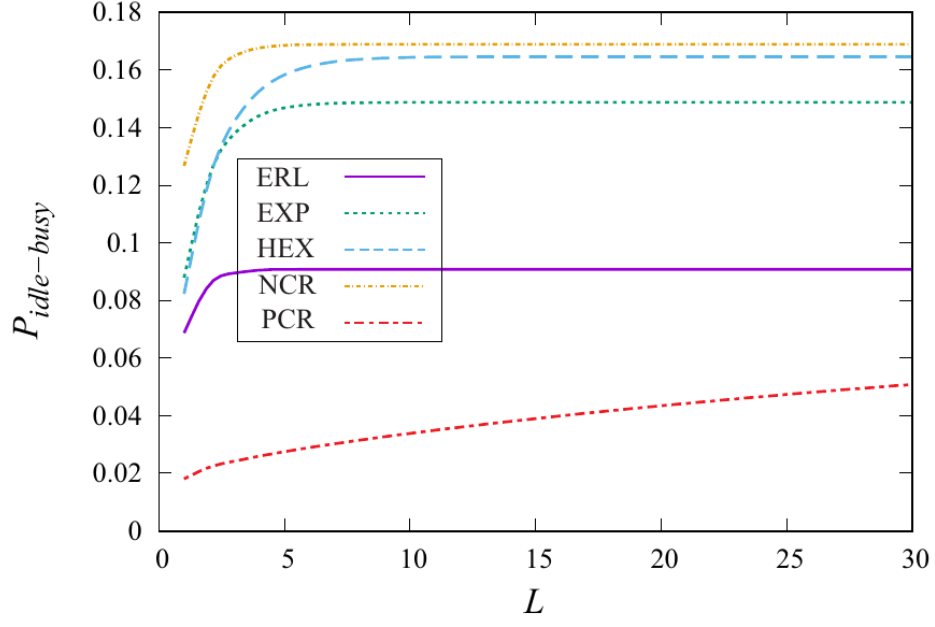


Figure 5: Dependence of the probability $P_{\text{idle-busy}}$ that the main server is idle while the secondary server is busy on the parameter L for different MAPs

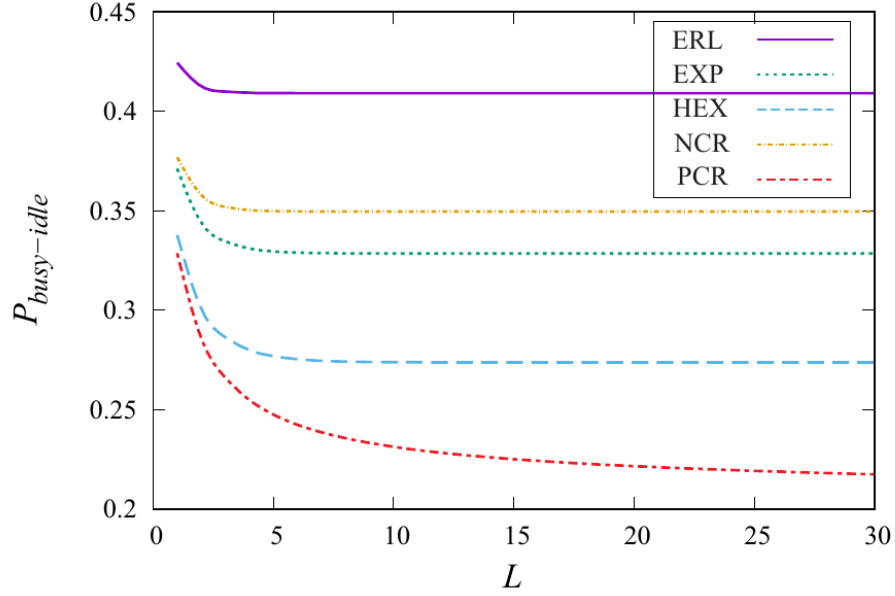


Figure 6: Dependence of the probability $P_{\text{busy-idle}}$ that the main server is busy while the secondary server is idle on the parameter L for different MAPs.

5.2 Secondo esempio numerico

Lo scopo di questo esempio è di indagare l'impatto dei parametri q (ricorda che questo è la probabilità che un cliente servito si rifiuti di agire come server secondario) e v (questa è la probabilità che un cliente servito da un server secondario sia insoddisfatto e torni al sistema). Fissiamo il valore di L a 10 (punto medio tra i due valori ottimali menzionati nel primo esempio). Fissiamo anche i tassi di servizio come $\mu_1 = 1$ e $\mu_2 = 0.5$ e indaghiamo la dipendenza di diverse misure di prestazione dalle probabilità q e v . Variamo i valori di queste probabilità da 0 a 1 con passo 0.05. Si noti che il valore $q = 1$ corrisponde al classico sistema MAP/M/1 con il tasso di servizio μ_1 .

In questo esempio ci concentriamo sul processo di arrivo etichettato come PCR, la cui scelta è basata sul comportamento di questo processo sulle misure evidenziato nel primo esempio illustrativo. Dalla Figura 7, che mostra la dipendenza del numero medio di clienti nel sistema L_{system} dai parametri q e v , deduciamo diverse osservazioni interessanti.

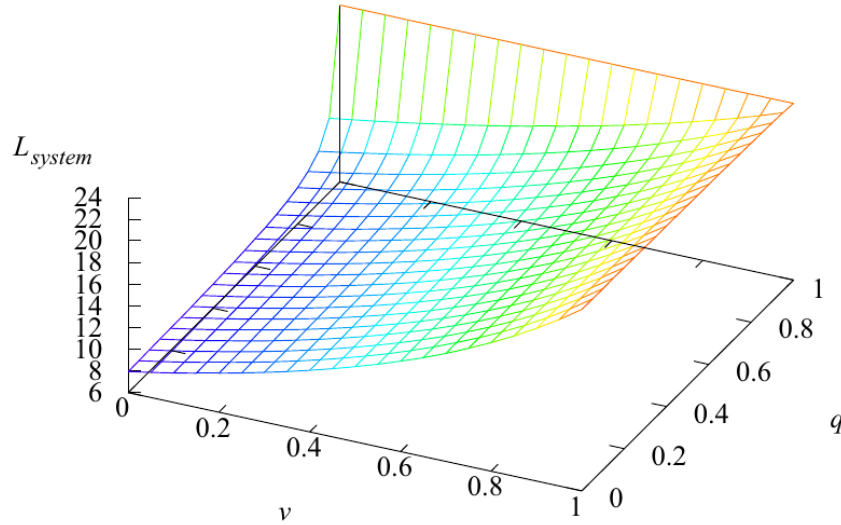


Figure 7: Dependence of the average number of customers in the system L_{system} on the parameters q and v .

Il valore di L_{system} è minimo con un valore di 7,9328 quando il cliente servito è sempre disponibile per essere reclutato (quando il sistema ne ha bisogno) e quando il cliente che riceve il servizio da un server secondario è sempre soddisfatto. Il valore minimo si ottiene quando $q = 0$ e $v = 0$. Questa misura aumenta quando aumenta q o v , e il tasso di aumento diventa più elevato quando uno o entrambi si avvicinano al valore 1. Quando $q = 1$, il sistema si trasforma nel corrispondente modello di coda MAP/M/1 classico e in un sistema senza l'uso del server secondario, e $L_{\text{system}} = 22,30425$ per tutti i valori di v (come è evidente). Quando $q = 0$, che corrisponde al caso in cui un cliente servito viene sempre reclutato (quando necessario), anche quando la probabilità di insoddisfazione è alta ($v = 0,5$), il valore di L_{system} è pari a 12,91247. Pertanto, l'uso di un server secondario riduce essenzialmente il numero medio di clienti nel sistema di più del 40%. Inoltre, abbiamo cercato il punto di interruzione, diciamo v^* , per una percentuale di insoddisfazione tale per cui il modello classico di coda è migliore del modello proposto qui. Per i parametri di questo esempio, il punto di interruzione è $v^* \sim 0,985$, nel senso che la percentuale di insoddisfazione deve essere superiore al 98,5% affinché il modello classico funzioni meglio.

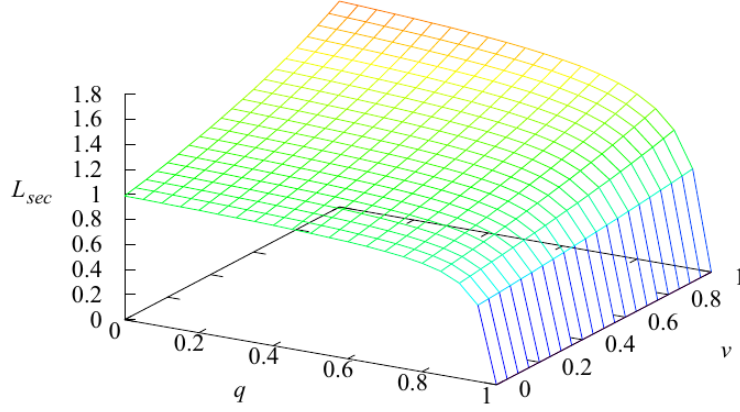


Figure 8: Dependence of the average number of customers with the secondary server L_{sec} on the parameters q and v

Per testare ulteriormente l'importo della riduzione nel numero medio, abbiamo aumentato λ del 50% a $\lambda = 0.75$. Mantenendo tutti gli altri parametri (ad eccezione della normalizzazione dei parametri del processo di arrivo per ottenere questo specifico λ) gli stessi, abbiamo ottenuto una percentuale di riduzione superiore al 52,8%. Pertanto, un aumento del carico del sistema beneficerà notevolmente dell'avere un server secondario per aiutare il sistema anche con un tasso di insoddisfazione del cliente del 50% con questo server secondario. La figura 8 mostra la dipendenza del numero medio di clienti con il server secondario L_{sec} dai parametri q e v . Questa probabilità diminuisce significativamente quando q si avvicina a 1 e quando i clienti sono raramente reclutati per diventare server secondari. L_{sec} ha il valore massimo quando q è uguale a zero, ovvero tutti i clienti vengono reclutati (quando necessario) per diventare server secondari, e quando v è vicino a 1. Ovviamente, in quest'ultimo caso, quasi tutti i clienti serviti da un server secondario devono essere rimandati al sistema a causa della loro insoddisfazione. Questo spiega la creazione di ulteriore lavoro per il sistema e dovrebbe essere scoraggiato ricorrendo alla coda classica anziché reclutare server secondari "scarsi". Vale la pena sottolineare che un sistema del genere (scadente) può riflettersi negativamente sul sistema stesso per la fornitura di servizi che non possono essere replicati da altri clienti serviti.

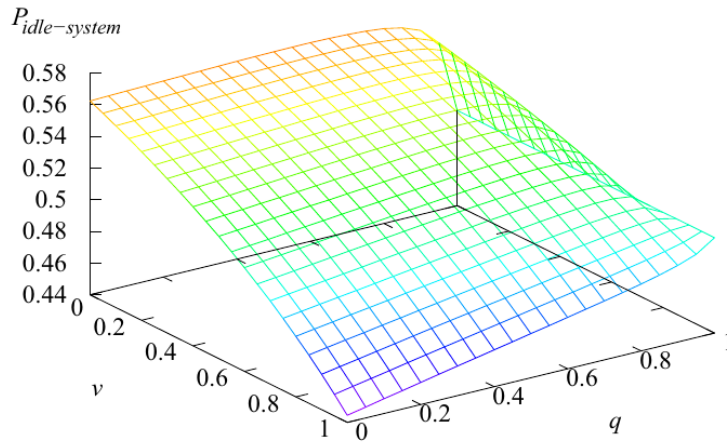


Figure 9: Dependence of the probability $P_{\text{idle-system}}$ that the system is idle at an arbitrary moment on the parameters q and v .

Nella Figura 9, è mostrato il comportamento della probabilità $P_{\text{idle-system}}$ che il sistema è inattivo in un momento arbitrario come funzione di q e ν . Questa probabilità ha il valore minimo di 0.4445 quando $\nu = 1$ e $q = 0$, il che è intuitivamente chiaro, poiché dover servire nuovamente i clienti dopo aver passato attraverso un server secondario mette un carico sul sistema. La probabilità che $P_{\text{idle-system}}$ aumenta quando q aumenta e/o ν diminuisce: il valore massimo 0.5652 di questa probabilità si ottiene quando $q = 0.65$ e $\nu = 0$. Nel corrispondente sistema MAP/M/1 classico, questa misura è $P_{\text{idle-system}} = 0.5$.

ALTRE COSE DA DIRE SU QUESTO ESEMPIO NUMERICO CHE SECONDO ME SI POSSONO SALTARE. TORNARE DOPO

5.3 Esempio numerico 3

In questo ultimo esempio, analizziamo l'impatto della variazione dei tassi di servizio μ_1 e μ_2 quando tutti gli altri parametri sono fissati. A tal fine, fissiamo $L = 10$, $q = 0.5$, $v = 0.4$ e $\lambda = 0.5$. I tassi μ_1 e μ_2 vengono variati da 0.25 a 2.0 con incrementi di 0.05. È importante menzionare che, per soddisfare la condizione di ergodicità (vedi Equazione 7), limitiamo ulteriormente il valore di μ_2 quando μ_1 è piccolo. In particolare, quando $\mu_1 = 0.25$, il valore minimo del tasso μ_2 (con il passo sopra descritto di 0.05) è scelto in modo tale da non essere inferiore a 0.65. Quando $\mu_1 = 0.3$, il tasso μ_2 è scelto in modo tale da non essere inferiore a 0.45. Quando $\mu_1 = 0.35$, il tasso μ_2 è scelto in modo tale da non essere inferiore a 0.3. Solo per $\mu_1 \geq 0.4$, il valore di μ_2 può essere variato da 0.25, come originariamente indicato.

Con le sopra descritte restrizioni sulla scelta di μ_1 e μ_2 , mostriamo nelle Figure 10 e 11 la dipendenza della misura L_{system} da μ_1 e μ_2 . Nella Figura 10, gran parte della superficie che mostra la dipendenza appare piatta. Ciò è dovuto al fatto che, per molte combinazioni dei valori dei parametri con un piccolo tasso μ_1 , la condizione di ergodicità viene violata e la misura L_{system} diventa molto grande. Pertanto, nella Figura 11, la dipendenza di L_{system} da μ_1 e μ_2 è mostrata solo per valori non piccoli di μ_1 . Chiaramente, si può notare una tendenza decrescente, poiché L_{system} diminuisce rapidamente quando μ_1 aumenta per μ_2 fissato e viceversa.

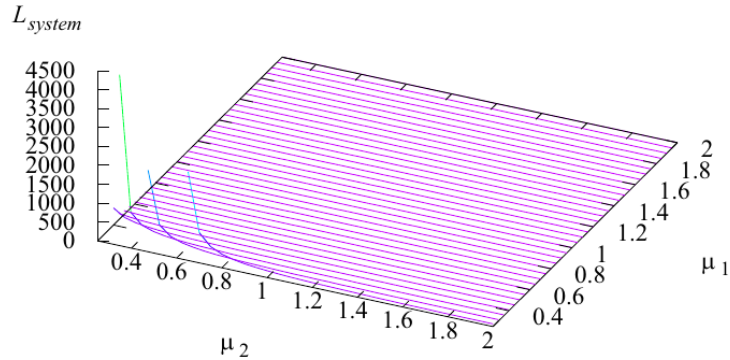


Figure 10: Dependence of the average number of customers in the system L_{system} on the parameters μ_1 and μ_2

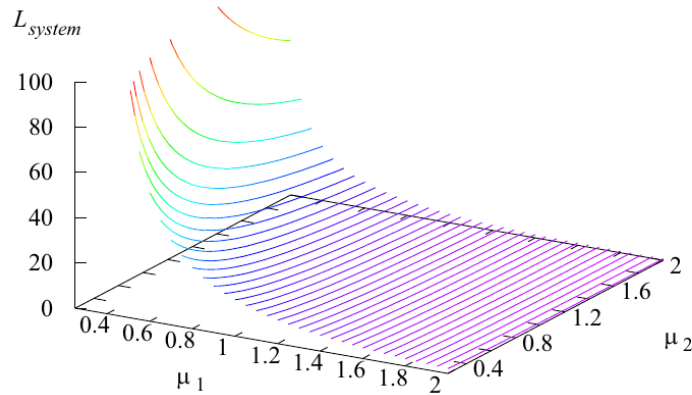


Figure 11: Dependence of the average number of customers in the system L_{system} on the parameters μ_1 and μ_2 .

La Figura 12 mostra il comportamento del numero medio di clienti con il server secondario L_{sec} . Il valore di L_{sec} è massimizzato con un valore di circa 5 quando μ_1 e μ_2 sono piccoli. Questo è intuitivamente chiaro poiché per valori piccoli di μ_1 e μ_2 , la condizione di ergodicità è vicina a essere violata, causando un alto tasso di reclutamento per i server secondari che, molto probabilmente prima di lasciare il sistema, serviranno un gruppo di dimensione $L = 10$. Pertanto, il numero medio di clienti in servizio in un momento arbitrario è di circa 5. Con un aumento di μ_1 e μ_2 , il valore di L_{sec} diminuisce come ci si aspetterebbe. Per valori piccoli di μ_1 , la diminuzione è significativa all'aumentare di μ_2 ; per valori più grandi di μ_1 , notiamo un tasso insignificante di diminuzione in L_{sec} con un aumento di μ_2 .

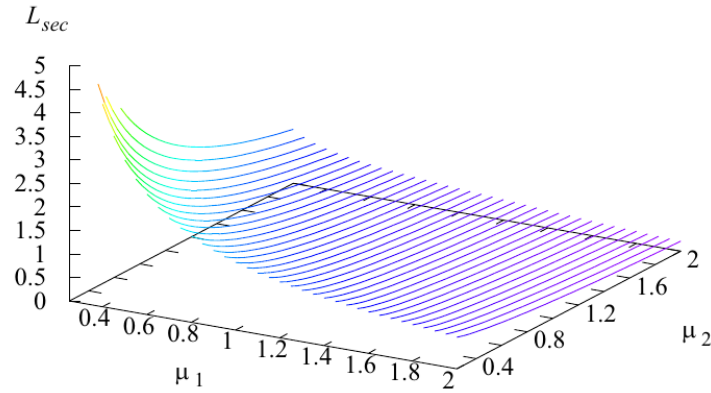


Figure 12: Dependence of the average number of customers with the secondary server L_{sec} on the parameters μ_1 and μ_2 .

ALTRE COSE DA DIRE SU QUESTO ESEMPIO NUMERICO CHE SECONDO ME SI POSSONO SALTARE. TORNARE DOPO

6 Conclusioni

In questo articolo, è stato analizzato un sistema di coda in cui c'è la possibilità di reclutare un cliente già servito come server secondario per aiutare il server principale assegnando un gruppo di clienti in attesa. Il processo di arrivo è stato modellizzato utilizzando un processo di punto Markoviano versatile, MAP. È stata presa in considerazione la possibilità di insoddisfazione dei clienti con il servizio fornito dal server secondario, causando il ritorno di quei clienti nel sistema. È stata implementata l'analisi dello stato stazionario della catena di Markov multidimensionale che descrive il comportamento del sistema e sono stati presentati risultati numerici illustrativi potenzialmente utili per prendere decisioni manageriali. Il modello studiato in questo articolo può essere generalizzato in diversi modi. Ad esempio,

1. il servizio fornito dal server secondario può essere effettuato in gruppi;
2. rilassare l'ipotesi di avere solo un server secondario a più di uno e vedere l'impatto dell'aumento a , diciamo, 2;
3. utilizzare servizi di tipo fase-possibilmente con rappresentazioni diverse per il server principale e secondario;
4. incorporare l'impazienza dei clienti sia nei buffer principali che secondari;
5. implementare un processo di reclutamento in base alla lunghezza della coda osservata basato su una politica di controllo di tipo soglia;
6. consentire arrivi di gruppo; e infine
7. incorporare la possibilità di reclutare molti server secondari con due tipi di clienti in modo che solo un tipo possa agire come server secondario.