

# PERFORMANCE MODELING COMPUTER SYSTEMS AND NETWORKS

## Relazione Progetto

Leonardo Monnati  
0310864

Alessandro Corsico  
0311156

11 Settembre 2023

## Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
<b>2</b>	<b>Problematiche del sistema</b>	<b>5</b>
<b>3</b>	<b>Obiettivo</b>	<b>6</b>
<b>4</b>	<b>Modello Concettuale</b>	<b>6</b>
<b>5</b>	<b>Definizione Dello Stato</b>	<b>8</b>
<b>6</b>	<b>Modello di Specifiche</b>	<b>9</b>
6.1	Dati di input . . . . .	9
6.1.1	Distribuzioni . . . . .	11
6.1.2	Tempi medi dei sottosistemi . . . . .	11
6.1.3	Percentuali di utenti nel sistema . . . . .	13
6.2	Analisi Spese e Profitti . . . . .	13
6.2.1	Costi di gestione . . . . .	13
6.2.2	Ricavi . . . . .	14
6.3	Probabilità di Routing . . . . .	14
6.3.1	Tabella Di Routing . . . . .	16
6.3.2	Flussi In Ingresso: Caso Low . . . . .	16
6.3.3	Flussi In Ingresso: Caso Mid . . . . .	16
6.3.4	Flussi In Ingresso: Caso High . . . . .	17

<b>7</b>	<b>Modello computazionale</b>	<b>18</b>
7.1	Strutture Dati . . . . .	18
7.2	Generatori pseudocasuali . . . . .	19
7.3	Gestione degli eventi . . . . .	19
7.3.1	Arrivo . . . . .	20
7.3.2	Completamento . . . . .	20
7.3.3	Cambio di fascia oraria . . . . .	20
7.4	Simulazione a orizzonte infinito . . . . .	20
7.5	Simulazione ad orizzonte finito . . . . .	21
<b>8</b>	<b>Verifica</b>	<b>23</b>
<b>9</b>	<b>Validazione</b>	<b>26</b>
<b>10</b>	<b>Analisi dei risultati</b>	<b>28</b>
10.1	Analisi Collo di Bottiglia . . . . .	28
10.2	Analisi ad Orizzonte Infinito . . . . .	29
10.2.1	Analisi ad Orizzonte Infinito: Caso Low . . . . .	30
10.2.2	Analisi ad Orizzonte Infinito: Caso Mid . . . . .	32
10.2.3	Analisi ad Orizzonte Infinito: Caso high . . . . .	35
10.3	Analisi ad Orizzonte Finito . . . . .	37
10.3.1	Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11 . . . . .	38
10.3.2	Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11 . . . . .	39
10.3.3	Scenario 2 - 2,2,5,2, 2,5,20,5, 4,12,50,11 . . . . .	40
10.3.4	Scenario ottimo - 2,3,6,3, 2,6,23,9, 5,11,50,10 . . . . .	42
<b>11</b>	<b>Algoritmo Migliorativo</b>	<b>44</b>
11.1	Obiettivo . . . . .	44
11.2	Modello Concettuale . . . . .	44
11.3	Modello Specifiche . . . . .	45
11.4	Probabilità di Routing . . . . .	47
11.4.1	Tabella Di Routing . . . . .	47
11.4.2	Flussi In Ingresso: Caso Low . . . . .	47
11.4.3	Flussi In Ingresso: Caso Mid . . . . .	47
11.4.4	Flussi In Ingresso: Caso High . . . . .	48
11.5	Modello Computazionale . . . . .	48
11.6	Verifica . . . . .	48
11.7	Validazione . . . . .	50

<b>12 Analisi dei risultati</b>	<b>51</b>
12.1 Analisi Collo di Bottiglia . . . . .	51
12.1.1 Analisi ad Orizzonte Infinito: Caso Low . . . . .	53
12.1.2 Analisi ad Orizzonte Infinito: Caso Mid . . . . .	54
12.1.3 Analisi ad Orizzonte Infinito: Caso High . . . . .	55
12.2 Analisi ad Orizzonte Finito . . . . .	56
12.3 Conclusioni . . . . .	58
12.4 Bibliografia . . . . .	58

Il seguente studio si propone di esaminare le prestazioni del processo che si verifica dall'ingresso di un cliente in un ristorante fast food fino alla sua uscita. Dopo un'analisi dettagliata dei dati reali effettivamente registrati del sistema e una successiva fase di simulazione, l'obiettivo è identificare una strategia di ottimizzazione che possa migliorare la gestione complessiva dei clienti in modo più efficiente.

## 1 Introduzione

Il sistema che è stato analizzato prende in esame il McDonald di via Tuscolana i cui dati sono stati presi durante l'anno corrente.

Il McDonald considerato si trova lungo una delle vie più trafficate e frequentate della capitale, per i negozi di giorno e i locali di notte. Pertanto è logico aspettarsi una notevole affluenza durante l'intero arco della giornata e un orario di apertura che supera le 20 ore giornaliere.

Si vuole fornire una strategia di miglioramenti che porteranno una più efficiente gestione della clientela.

Il consumatore che vuole procurarsi il proprio pasto può decidere se entrare all'interno della sede e recarsi presso una delle casse automatiche per procedere con l'ordinazione e il pagamento, oppure fare la sua ordinazione comodamente da casa tramite sito web, oppure infine telefonando alla sede. I clienti che si recano in presenza devono poi attendere che la loro ordinazione sia presa in carico dalla cucina e, una volta preparata, possono recarsi nell'area tavoli adibita alla consumazione. Coloro che invece ordinano da casa usufruiscono del servizio delivery messo a disposizione dalla sede che, tramite riders, porta il pasto direttamente a casa del consumatore.

Gli utenti considerati nel sistema sono quindi di due tipi:

- Gli utenti "consumazione al tavolo" avranno accesso alle casse in cui potranno scegliere il cibo da consumare. Superata la cassa, l'ordinazione passerà alla cucina per infine essere consumata al tavolo.
- Gli utenti "ordinazioni delivery" avranno accesso diretto alla cucina. Una volta terminata la preparazione, essa sarà affidata al corriere.

Di seguito si mostra che l'unico nodo di intersezione fra i due tipi di utenti risulta essere quello della cucina.

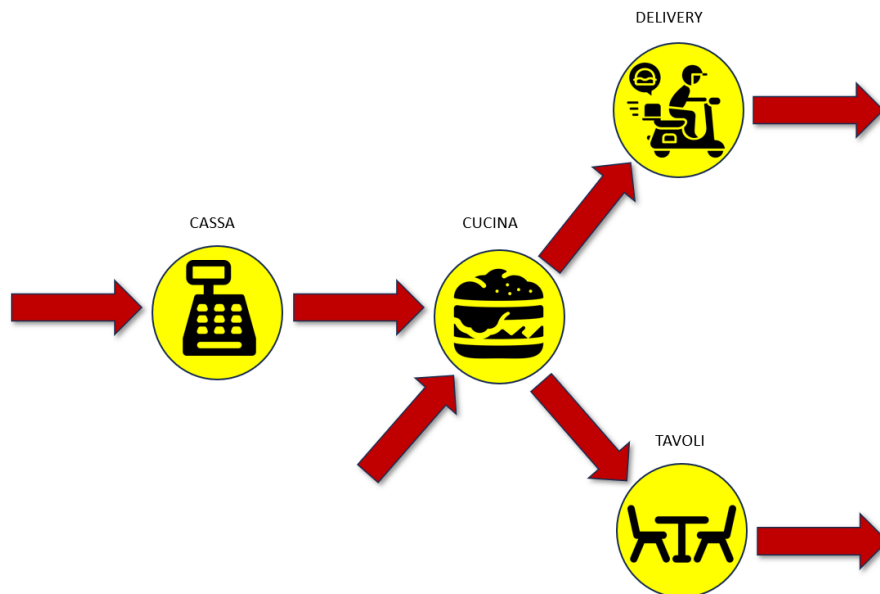


Figura 1. Sistema Completo

## 2 Problematiche del sistema

Essendo il McDonald's un ristorante molto frequentato, è possibile trovare un ingente numero di clienti al suo interno durante gli orari di punta. Per questo motivo, spesso l'attesa che il consumatore sperimenta prima di poter ordinare da mangiare è così elevata da spingerlo ad uscire dal ristorante. Non di rado capita, infatti, che persone prima desiderose di ordinare perdano il loro interesse a causa della fila non indifferente che si trovano ad affrontare davanti alle casse. Inoltre, una volta superate le casse va comunque preso in considerazione il problema del lungo ritardo alla preparazione del pasto e dell'attesa affinché si liberi un posto ai tavoli. Entrambi potrebbero portare al malcontento dei clienti. Gli aspetti critici, quindi, sono riportati qui di seguito:

- La lunga attesa alle casse che può comportare la rinuncia e quindi la perdita di un potenziale cliente.
- L'attesa alle cucine, centro in cui confluiscono i due flussi di utenti e che quindi è quello sottoposto ad un carico maggiore.
- L'attesa ai tavoli, per poter consumare un pasto comodamente seduti.
- L'attesa del cibo che il rider consegna direttamente a casa che porta ad una consumazione fredda.

### 3 Obiettivo

L'obiettivo dello studio è quello di minimizzare i costi e massimizzare i guadagni per la gestione del servizio di McDonald. Quindi in particolare si vuole individuare il numero ottimale di server ed operatori per ogni sottosistema nelle diverse fasce orarie.

Nello specifico si vogliono rispettare due QoS:

- Mantenere il tempo di risposta medio complessivo del sistema inferiore ai 900 secondi.
- Tenere la percentuale di utenti che, trovando fila alle casse, abbandonano il sistema sotto il 3%.

A tale scopo si effettua uno studio sia dello stato stazionario che del transiente, provando diverse configurazioni di server e cercando di individuare quale di queste rispetta i due vincoli mantenendo il minor costo totale.

### 4 Modello Concettuale

I jobs possono essere di due tipi: consumatori al tavolo ed utenti delivery. I primi sono coloro che entrano fisicamente nel McDonald, procedono con l'ordinazione e consumano ad uno dei tavoli della struttura prima di uscire dal sistema.

I secondi non hanno bisogno delle casse ed entrano direttamente nel nodo della cucina, dove la loro ordinazione viene preparata prima di essere affidata al rider che provvederà al trasporto.

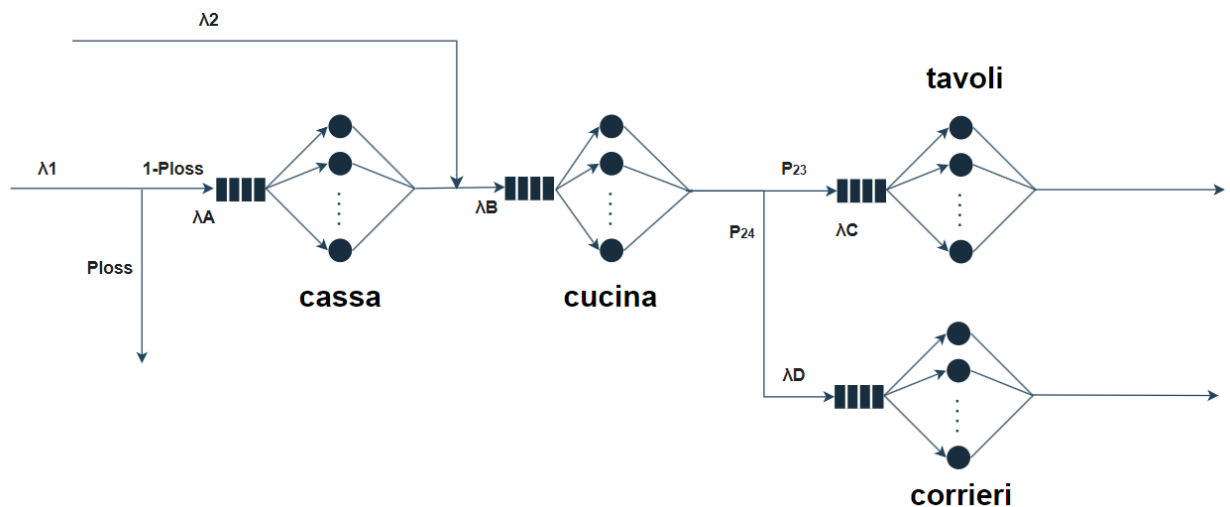
Si distinguono 4 sottosistemi :

- sottosistema 1: Acquisto Ordinazione; è un nodo multiserver a coda singola e scheduling FIFO ed è qui che avviene l'ingresso dei jobs di tipo "consumazione al tavolo". Una volta entrato il job controlla la lunghezza della coda e decide se entrare a farne parte o abbandonare il sistema. Una volta completato ed uscito dal server che l'ha processato si dirige al nodo "Preparazione Ordinazione"
- sottosistema 2: Preparazione Ordinazione; un nodo multiserver a coda singola e scheduling FIFO in cui entrano due flussi diversi di jobs, da una parte quelli uscenti dal nodo precedente, dall'altra i jobs "servizio delivery" per i quali questo nodo costituisce il punto di ingresso

nel sistema. Qui i jobs di entrambi i tipi entrano nell'unica coda presente, vengono elaborati senza alcuna distinzione dai server e, tenendo traccia del tipo di job elaborato, vengono instradati correttamente una volta usciti.

- sottosistema 3: Consumazione Ordinazione; un nodo multiserver a coda singola e scheduling FIFO in cui entrano i jobs "consumazione al tavolo" una volta usciti dal nodo precedente. Questo nodo costituisce una delle tre uscite dal sistema, in quanto una volta processati, i jobs del nodo 3 ne escono.
- sottosistema 4: Servizio Delivery; un nodo multiserver a coda singola e scheduling FIFO in cui entrano i jobs "servizio delivery" una volta usciti dal nodo "Preparazione Ordinazione". Questo nodo costituisce l'ultima delle tre uscite dal sistema, in quanto una volta processati, i jobs del nodo 4 ne escono.

Il McDonald è modellato secondo la rete riportata in figura.



**Figura 2. Sistema Completo**

Le code sono infinite in quanto un McDonald non limita il numero dei clienti che può servire. Quando c'è un arrivo in un sottosistema, si verifica sempre se c'è un servente libero; se ce n'è più di uno si sceglie sempre quello con minor valore di tempo di servizio, altrimenti si entra in coda.

Il McDonald preso in esame è aperto dalle 08:00 alle 05:00, ed in questo periodo il numero di ordinazioni non è costante e per questo sono state identificate 7 differenti fasce orarie (dati presi dall'affluenza stimata da Google):

- 08:00 – 10:00: traffico basso
- 10:00 – 15:00: traffico medio
- 15:00 – 17:00: traffico basso
- 17:00 – 18:00: traffico medio
- 18:00 – 22:00: traffico intenso
- 22:00 – 03:00: traffico medio
- 03:00 – 05:00: traffico basso

## 5 Definizione Dello Stato

Si definisce lo stato complessivo del sistema come la raccolta degli stati di ciascun sottosistema. In dettaglio, lo stato di ogni sottosistema comprende informazioni come il numero di jobs in attesa e il numero di jobs in fase di esecuzione, lo stato di ogni servente (se è libero o occupato), il numero di serventi disponibili per ogni sottosistema al cambiare della fascia, il numero di nuovi arrivi e il numero di jobs completati. Per quanto riguarda gli eventi, essi possono essere divisi in due categorie principali: l'arrivo di un nuovo cliente, che rappresenta l'ingresso di un job nel sistema, e il completamento del servizio da parte di un servente e il cambio fascia, che comporta il cambiamento delle combinazioni di serventi presenti. Quando un servente completa un job, esso passa ad un altro punto della rete, determinato in base alle probabilità di routing, oppure esce. Nel complesso, i jobs possono essere classificati in due categorie: consumazione al tavolo e servizio delivery. La gestione degli eventi segue il principio della simulazione "next-event": tutti gli eventi da processare vengono ordinati in una struttura dati specifica, e ciclicamente si estrae l'evento più prossimo nel tempo da elaborare. Naturalmente, l'elaborazione di un evento può generare ulteriori eventi, che vengono aggiunti alla struttura dati che tiene traccia di tutti gli eventi pendenti.



Gli eventi possibili sono:

- Arrivo di un job dall'esterno al sottosistema "Acquisto Ordinazione".
- Uscita di un job dal sistema in caso di coda eccessiva al sottosistema "Acquisto Ordinazione".
- Uscita di un job dal sottosistema "Acquisto Ordinazione" ed instradamento verso il sottosistema "Preparazione Ordinazione".
- Arrivo di un job dall'esterno al sottosistema "Preparazione ordinazione".
- Uscita di un job dal sottosistema "Preparazione ordinazione" ed instradamento verso il sottosistema "Consumazione al tavolo".
- Uscita di un job dal sottosistema "Preparazione ordinazione" ed instradamento verso il sottosistema "Servizio Delivery".
- Uscita di un job dal sottosistema "Consumazione al tavolo".
- Uscita di un job dal sottosistema "Servizio Delivery".
- Cambio di Fascia oraria; Il cambio di fascia oraria può causare l'apertura o la chiusura dei serventi attivi nella specifica fascia per mantenere il sistema stabile e far fronte alla variazione del flusso in ingresso.

## 6 Modello di Specifiche

### 6.1 Dati di input

Per la scelta dei parametri di input del sistema sono stati considerati i dati del McDonald di via Tuscolana 829, forniti dal sito ufficiale McDonald.[1] Da qui sono stati estrapolati i dati interessanti al fine dello studio e riorganizzati nella seguente tabella.

Valori estrapolati	
statistica	valore in euro
<i>cassiere</i>	864
<i>ristoratore</i>	750
<i>corriere</i>	700

I seguenti dati sono stati presi dal sito del McDonald di riferimento [2] :

- Periodo di osservazione: 21 ore
- Numero di Ordinazioni al Giorno: 1593
- Numero di Consumazioni al Tavolo al Giorno: 1443
- Numero di Ordinazioni Delivery al Giorno: 130

Per tutti gli altri dati, per i quali non è stato possibile trovare riscontri in rete, è stata fatta una stima basata su osservazioni fatte in loco in vari giorni e in differenti orari. Queste, assieme ad altre informazioni ottenute direttamente dai dipendenti di quella specifica sede, sono state utilizzate per il calcolo dei tempi di servizio e per un'aspettativa sui tempi di attesa da usare come riferimento.

Gli arrivi sono suddivisi in 3 differenti fasce:

- Fascia bassa (08:00 – 10:00, 15:00 – 17:00, 03:00 – 05:00):
  - 5% delle ordinazioni al tavolo, 72 ordinazioni totali
  - 3% delle ordinazioni delivery, 4 ordinazioni totali
  - \*  $\lambda_1 = 0.004067 \text{arrivi/sec}$
  - \*  $\lambda_2 = 0.000208 \text{arrivi/sec}$
- Fascia media (10:00 – 15:00, 17:00 – 19:00, 22:00 – 03:00):
  - 34.8% delle ordinazioni al tavolo, 502 ordinazioni totali
  - 17.6% delle ordinazioni delivery, 23 ordinazioni totali
  - \*  $\lambda_1 = 0.015373 \text{arrivi/sec}$
  - \*  $\lambda_2 = 0.000667 \text{arrivi/sec}$

- Fascia alta(19:00 – 22:00):
  - 60.2% delle ordinazioni al tavolo, 869 ordinazioni totali
  - 79.4% delle ordinazioni delivery, 103 ordinazioni totali
  - \*  $\lambda_1 = 0.073208 \text{arrivi/sec}$
  - \*  $\lambda_2 = 0.008271 \text{arrivi/sec}$

### 6.1.1 Distribuzioni

Nella simulazione condotta, sono state impiegate distribuzioni statistiche per modellare gli arrivi e i completamenti dei lavori nel sistema. Questo approccio è fondamentale per rappresentare con precisione il flusso di lavoro e l'efficienza del sistema in esame. Per quanto riguarda gli arrivi, si sono utilizzate distribuzioni esponenziali. Questo tipo di distribuzione è comunemente adottato per modellare gli intervalli tra gli arrivi dei jobs nel sistema. La distribuzione esponenziale è particolarmente idonea a tale scopo, in quanto tiene conto della natura casuale degli arrivi e del loro tasso medio. Per quanto concerne i completamenti dei jobs da parte dei serventi, si è fatto ricorso a distribuzioni normali troncate. Ciò significa che è stato rappresentato il tempo necessario per il completamento di un job attraverso una distribuzione normale, ma con limiti superiori e inferiori che rispecchiano le capacità dei serventi e le restrizioni del sistema e una verosomiglianza con la realtà. In conclusione, l'utilizzo di distribuzioni statistiche, come distribuzioni esponenziali per gli arrivi e distribuzioni normali troncate per i completamenti, ha consentito di creare una simulazione accurata e dettagliata del sistema.

### 6.1.2 Tempi medi dei sottosistemi

Di seguito vengono riportate le specifiche dei vari sottosistemi

#### **Tempo Medio di servizio del sottosistema Acquisto Ordinazione:**

- $\mu_2 = 45 \text{secondi}$

Esso è stato modellato secondo una normale troncata con media 45 secondi, standard deviation 4, e limiti a=10 e b=80

#### **Tempo Medio Arrivi Acquisto Ordinazione:**

Gli arrivi Markoviani con tasso lambda specifico a seconda della fascia oraria:

- $\lambda = 0.004067 \text{acquisti/secondo}$  per la fascia bassa
- $\lambda = 0.015373 \text{acquisti/secondo}$  per la fascia media
- $\lambda = 0.073208 \text{acquisti/secondo}$  per la fascia alta

**Tempo Medio di servizio del sottosistema Preparazione in Cucina:**

- $\mu_3 = 112 \text{secondi}$

Esso è stato modellato secondo una normale troncata con media 112 secondi, standard deviation 5, e limiti a=60 e b=164

**Tempo Medio Arrivi Preparazione in Cucina:**

Gli arrivi Markoviani con tasso lambda specifico a seconda della fascia oraria:

- $\lambda = 0.000208 \text{preparazioni/secondo}$  per la fascia bassa
- $\lambda = 0.000667 \text{preparazioni/secondo}$  per la fascia media
- $\lambda = 0.008271 \text{preparazioni/secondo}$  per la fascia alta

**Tempo Medio di servizio del sottosistema Consumazione al Tavolo:**

- $\mu_4 = 600 \text{secondi}$

Esso è stato modellato secondo una normale troncata con media 600 secondi, standard deviation 20, e limiti a=200 e b=1600

**Tempo Medio di servizio del sottosistema Consegna Delivery:**

- $\mu_5 = 1200 \text{secondi}$

Esso è stato modellato secondo una normale troncata con media 1200 secondi, standard deviation 40, e limiti a=800 e b=2000

### 6.1.3 Percentuali di utenti nel sistema

Le percentuali di utenti nel sistema sono diverse a seconda della fascia oraria considerata.

Per la fascia Low :

- Percentuale Clienti al Tavolo : 95.1185%
- Percentuale Clienti Delivery : 4.8815%

Per la fascia Mid :

- Percentuale Clienti al Tavolo : 95.8450%
- Percentuale Clienti Delivery : 4.1550%

Per la fascia High :

- Percentuale Clienti al Tavolo : 89.8474%
- Percentuale Clienti Delivery : 10.1526%

## 6.2 Analisi Spese e Profitti

In questa sezione vengono descritti i ricavi e le spese del modello che concorrono nel determinare il profitto del fastfood.

### 6.2.1 Costi di gestione

Mantenere un servente attivo implica avere un determinato costo, pertanto è necessario trovare il numero ottimo di serventi per ogni fascia oraria affinché le QoS siano rispettate ed il guadagno sia massimo. Si è assunto per ogni tipo di servente un costo di gestione giornaliero. Il costo viene calcolato su ogni servente per il tempo totale in cui questo è attivo, anche se poco utilizzato o non utilizzato.

Tali costi comprendono gli stipendi, presi ove possibili dalle fonti (vedi tabella precedente), e altre spese secondarie considerate nel totale.

- Costo mensile cassa: 980€
- Costo mensile ristorazione: 820€
- Costo mensile gestione tavolo: 50€
- Costo mensile servizio delivery: 790€

### 6.2.2 Ricavi

Per quanto riguarda il ricavo si è deciso di considerare l'ordinazione fatta da un utente medio che consuma uno dei menù offerti dal Mcdonal e considerati un pasto completo. Il costo medio di un menù è di circa 10 euro, pertanto è stato considerato questo il prezzo di ogni singola ordinazione.

## 6.3 Probabilità di Routing

A livello di modellazione si distribuisce il flusso totale in ingresso verso i vari blocchi tramite le probabilità di routing, definite utilizzando i dati percentuali ricavati in precedenza.

Probabilità di routing in uscita dal sottosistema Acquisto Ordinazione:

- Uscita dal sistema:  $p_1, exit = p$
- Acquisto Ordinazione:  $p_{12} = 1-p$

$p$  è una variabile che dipende dalla lunghezza della coda al momento dell'arrivo di un nuovo job e da un valore randomico la cui soglia varia a seconda della suddetta fila. Qualora l'utente dovesse trovare davanti a sé un numero di persone in coda superiore alla metà delle casse aperte può decidere di abbandonare la coda. Tale decisione dipende sia dalla lunghezza della coda (maggiore essa è e più è probabile che esca), sia da un numero randomicamente estratto a simulare la casualità della scelta dell'utente. Probabilità di routing in uscita dal sottosistema preparazione ordinazione: Attraverso questo sottosistema passano entrambi i tipi di jobs che successivamente verranno instradati gli uni ai tavoli e gli altri ai corrieri.

Le loro probabilità di routing rispettano i valori precedentemente indicati e sono quindi:

Per la fascia Low :

- $p_{23}$  : 95.1185%
- $p_{24}$  : 4.8815%

Per la fascia Mid :

- $p_{23}$  : 95.8450%
- $p_{24}$  : 4.1550%

Per la fascia High :

- $p_{23}$  : 89.8474%
- $p_{24}$  : 10.1526%

### 6.3.1 Tabella Di Routing

Di seguito la matrice di routing che rappresetnta tutte le probabilità di routing:

	Esterno	Sottosistema 1	Sottosistema 2	Sottosistema 3	Sottosistema 4
Esterno	0	$P_t$	$P_d$	0	0
Sottosistema 1	$P_{loss}$	0	$1 - P_{loss}$	0	0
Sottosistema 2	0	0	0	$p_{23}$	$p_{24}$
Sottosistema 3	1	0	0	0	0
Sottosistema 4	1	0	0	0	0

*eq:sistema generale*

$$\begin{cases} \lambda_A = \lambda_1 - P_{loss} \\ \lambda_B = \lambda_2 + \lambda_A \\ \lambda_C = \lambda_B \cdot p_{23} \\ \lambda_D = \lambda_B \cdot p_{24} \end{cases} \quad (1)$$

### 6.3.2 Flussi In Ingresso: Caso Low

Per la fascia Low i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.004067 - P_{loss}job/sec \\ \lambda_B = 0.004275 - P_{loss}job/sec \\ \lambda_C = 0.004066 - 0.951185 \cdot P_{loss}job/sec \\ \lambda_D = 0.0002087 - 0.048815 \cdot P_{loss}job/sec \end{cases} \quad (2)$$

### 6.3.3 Flussi In Ingresso: Caso Mid

Per la fascia Mid i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.015373 - P_{loss}job/sec \\ \lambda_B = 0.01604 - P_{loss}job/sec \\ \lambda_C = 0.015373 - 0.958450 \cdot P_{loss}job/sec \\ \lambda_D = 0.000666 - 0.041550 \cdot P_{loss}job/sec \end{cases} \quad (3)$$



#### 6.3.4 Flussi In Ingresso: Caso High

Per la fascia High i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.073208 - Plossjob/sec \\ \lambda_B = 0.081479 - Plossjob/sec \\ \lambda_C = 0.073207 - 0.898474 \cdot Plossjob/sec \\ \lambda_D = 0.0082722 - 0.101526 \cdot Plossjob/sec \end{cases} \quad (4)$$

## 7 Modello computazionale

Per la simulazione della rete è stato utilizzato l'approccio di tipo **Next-Event Simulation**, in cui l'avanzamento del tempo si effettua tramite processamento dell'evento successivo. Nell'implementazione del modello, è stato scelto di utilizzare il linguaggio di programmazione Python. La motivazione di questa scelta risiede principalmente nel fatto che Python adotta il paradigma di programmazione orientata agli oggetti. Questa caratteristica semplifica notevolmente il processo di implementazione e consente di sviluppare un software altamente versatile, adatto a una vasta gamma di modelli di reti di code. Il software è basato su classi di configurazione, attraverso le quali è possibile descrivere la topologia specifica della rete da simulare. In particolare, è richiesto di specificare il numero di centri presenti nella rete, le loro interconnessioni, il tasso di servizio medio per ciascun centro e la distribuzione del tempo di servizio.

### 7.1 Strutture Dati

Nel software utilizzato per la simulazione del modello sono state create ed utilizzate diverse strutture dati per poter salvare le informazioni necessarie all'analisi dell'esperimento ed altre necessarie allo svolgimento dello stesso.

- Struttura `TypeNode` che tiene le informazioni sul singolo sottosistema/nodo del sistema precisamente dei suoi:
  - serveri; il numero di centri presenti nel sottosistema a cui la struttura fa riferimento. Questi cambiano solo durante l'evento Cambio Fascia
  - popolazione; il numero di jobs presenti nel sottosistema
  - numero di completamenti; contatore dei jobs usciti dal sottosistema dopo essere stati correttamente elaborati.
- Una sottostruttura `Accsum` per ogni servere che tiene conto di:
  - numero job serviti; contatore dei jobs correttamente serviti dal singolo servere durante tutto il periodo in cui esso è stato attivo.
  - tempo di servizio; tempo totale di attività del servere durante lo stato busy
  - tempo di accensione; momento nell'arco della giornata in cui lo specifico servere è stato acceso

- tempo di spegnimento; momento nell’arco della giornata in cui lo specifico server è stato spento
- Struttura Event, ovvero una struttura ausiliaria che partecipa alla Next Event Simulation tenendo conto del tempo del prossimo evento, a quale server di quale sottosistema esso è riferito e a quale tipologia di job appartiene.
- Struttura Time che tiene conto dell’ avanzamento del clock che viene aggiornato tramite la sostituzione del valore del campo next al valore del campo current.

## 7.2 Generatori pseudocasuali

Nella simulazione condotta, è stato scelto di impiegare il generatore pseudocasuale di Lehmer per modellare gli arrivi dei jobs nel sistema. Questa scelta è stata determinata dalla necessità di generare sequenze di arrivi che, benché non presentino una casualità statistica completa, possano fornire un grado di casualità controllata sufficiente per gli scopi della simulazione. Il generatore pseudocasuale di Lehmer opera seguendo una relazione ricorsiva che coinvolge operazioni di moltiplicazione e divisione modulari. È stato selezionato un valore iniziale di seed per iniziare la sequenza di arrivi. È importante notare che, nonostante questa sequenza di arrivi sembri casuale, in realtà è deterministica e completamente basata sulla formula di generazione di Lehmer. Pertanto, il generatore, viene spesso utilizzato in scenari in cui è necessaria una casualità controllata, come ad esempio simulazioni o test di algoritmi, ma potrebbe non essere adatto per applicazioni che richiedono una casualità statistica più avanzata. In conclusione, il generatore pseudocasuale di Lehmer è stato impiegato per generare sequenze di arrivi di lavoro nella simulazione, consentendo di studiare e analizzare il comportamento del sistema in condizioni varie e controllate. Per gli arrivi esponenziali di entrambi i tipi di jobs e per i servizi di tutti i sottosistemi è stato ovviamente selezionato uno stream unico per ciascuno di essi in quanto gli eventi sovraelencati sono e devono essere tra loro indipendenti.

## 7.3 Gestione degli eventi

Gli eventi di completamento ed arrivo vengono gestiti in modo differente. Si avanza il clock a seconda del prossimo evento qualunque esso sia e grazie alle strutture sopracitate (in particolare la struttura data) è facilmente identificabile se si tratta di un arrivo o di un completamento (si identifica su quale nodo e server esso avviene).

### 7.3.1 Arrivo

Si genera il primo tempo di arrivo tramite `getArrival()` che utilizza la versione python della funzione `Exponential()` di `rvgs.c`.

Ad essa viene passato il parametro  $\frac{1}{\lambda}$  specifico e una volta completato l'arrivo ad un sottosistema viene immediatamente calcolato il prossimo. A questo punto il job appena arrivato se trova un servente libero entra in servizio e il suo tempo di servizio viene calcolato tramite la `GetService`, altrimenti viene messo in coda.

Il flusso degli arrivi termina quando il valore di ritorno della `GetService` supera il valore di `STOP` (75600 secondi) che rappresenta la fine di 21 ore di operatività del McDonald.

### 7.3.2 Completamento

Quando la Next-Event simulation porta a processare un completamento il servente che lo ha elaborato controlla se nella coda è presente un altro job e lo prende immediatamente in carico, altrimenti rimane vuoto in attesa di un arrivo. Il job completato passa al sottosistema successivo se elaborato dai nodi "Acquisizione Ordinazione" o "Preparazione Ordinazione" oppure esce dal sistema. Se un servente elabora un job durante un cambio fascia, esso rimane attivo fino al termine dell'operazione anche qualora esso dovesse essere spento a causa della riduzione dei centri. Nel processo si terrà conto dell'intera durata della sua operatività per il calcolo dei tempi e dei costi

### 7.3.3 Cambio di fascia oraria

Al cambio di fascia oraria varia il flusso di ingresso e viene quindi riconfigurata la rete per far fronte al nuovo traffico. A livello implementativo si utilizza una funzione `controlFunc()` che confronta la configurazione attuale con quella della nuova fascia, e attiva o disattiva per ogni blocco il numero di server necessari. Se un servente elabora un job durante un cambio fascia, esso rimane attivo fino al termine dell'operazione anche qualora esso dovesse essere spento a causa della riduzione dei centri. Nel processo si terrà conto dell'intera durata della sua operatività per il calcolo dei tempi e dei costi.

## 7.4 Simulazione a orizzonte infinito

Nella simulazione ad orizzonte infinito il sistema viene simulato per un tempo molto superiore al tempo reale. In questo modo si producono le statistiche a stato stazionario del sistema. In ogni simulazione si assume che il sistema sia statico, quindi per ogni fase il tasso di arrivo e il numero di serventi restano

costanti. Per ricavare la media campionaria del tempo di risposta si utilizza il metodo delle **Batch Means**, suddividendo la run di simulazione in  $k$  batches di dimensione  $b$ . Questi parametri sono stati opportunamente scelti tenendo conto del valore di autocorrelazione che non deve essere superiore allo 0.2. Si resettano, dopo averle opportunamente salvate, le statistiche del sistema ad ogni iterazione senza azzerarne lo stato, ovvero mantenendo tutti i job rimanenti dal precedente batch. In questo modo si genera un campione di  $k$  batches indipendenti, sul quale è possibile valutare la media campionaria. Per le medie, le deviazioni standard e gli intervalli di confidenza è stato utilizzato l'algoritmo di Welford che permette di svolgere i conti ed ottenere i risultati senza dover salvare tutti i dati di ogni batch nelle apposite strutture. Gli obiettivi della simulazione ad orizzonte infinito ai fini del nostro studio sono principalmente 2:

- Trovare lo stato stazionario modificando opportunamente il numero dei centri e tenendo conto dei tempi di risposta.
- Ricerca della configurazione ottima che permette per ogni fascia oraria di rispettare le QoS.

## 7.5 Simulazione ad orizzonte finito

Nella caso ad orizzonte finito viene effettuata una simulazione del sistema lungo le 21 ore lavorative; il sistema è assunto quindi dinamico, e viene considerata la variazione del flusso nelle diverse fasce orarie e la riorganizzazione del numero di server attivi. Il sistema è IDLE sia all'inizio che alla fine della simulazione. Un'analisi di questo tipo permette di ottenere le statistiche di sistema transienti. L'intera giornata viene ripetuta  $n$  volte e ogni replica viene utilizzata per misurare le stesse statistiche, e fornisce quindi un punto del nostro campione. La condizione di terminazione della simulazione è il raggiungimento dell'orario di chiusura (05:00), e quindi quando il clock di simulazione raggiunge il valore di 75600 secondi (=21 ore). Per analizzare le statistiche ottenute nel continuo, per ogni ripetizione, effettuiamo ogni 10 minuti una misurazione del tempo di risposta, che viene salvato in un array. Per ogni intervallo si avranno quindi 126 misurazioni del tempo di risposta. Calcolando la media di queste stime si ottiene quindi il valore del tempo di risposta considerato ai fini dell'analisi per quello specifico orario. La simulazione ad orizzonte finito ha come obiettivi principali:

- Analizzare il comportamento del sistema al cambio fascia, verificando che le configurazioni individuate ad orizzonte infinito siano valide effettivamente anche nel sistema reale.

- Analizzare i costi e i guadagni reali sull'arco dell'intera giornata.
- Testare diverse configurazioni per valutare più nel dettaglio il comportamento del sistema nel transiente ed in particolare al cambio di fascia oraria
- Rispettare entrambe le QoS e minimizzare i costi.

## 8 Verifica

Si Verifica che il modello computazionale è effettivamente conforme al modello di specifica, quindi che l'implementazione del modello computazionale sia effettivamente corretta. Per fare questo abbiamo verificato 4 condizioni:

A. Il tempo di risposta (wait) per un server è sempre uguale alla somma del tempo di attesa (delay) e del tempo di servizio (service).

B. La popolazione nel nodo deve essere uguale alla somma della popolazione in coda e della popolazione del server

C. La popolazione nel nodo "preparazione Ordinazione" viene smistata ai nodi "tavoli" e "corrieri", pertanto la somma delle popolazioni di questi ultimi 2 deve essere uguale alla prima.

D. Utilizzando le KP generali nei casi in cui in un sottosistema è presente un solo centro (ove possibile) verifichiamo che i risultati analitici siano uguali a quelli sperimentali.

E. Nel caso multiserver eseguendo una simulazione con un numero di centri abbastanza grande da far considerare trascurabile la popolazione in coda e quindi anche il tempo di attesa, si verifica che per valutare queste 3 condizioni sono state eseguite diverse run ad orizzonte sia finito che infinito.

Riportiamo qui i risultati di una run nell'arco delle 21 ore lavorative.

```
for 1442.000000 jobs the service node statistics are:
```

```
avg interarrivals .. = 52.9812435
avg wait ..... = 47.4313261
avg # in node ..... = 0.8952475
avg delay ..... = 2.3170612
avg # in queue ..... = 0.0437336
avg service time.....= 45.1140767
```

**Figura 3.** risultati nodo acquisto ordinazione

for 1559.000000 jobs the service node statistics are:

```
avg interarrivals .. = 49.0051014
avg wait ..... = 115.9815608
avg # in node ..... = 2.3667242
avg delay ..... = 4.1436867
avg # in queue ..... = 0.0845562
avg service time.....= 111.8381402
```

**Figura 4. risultati preparazione ordinazione**

for 1442.000000 jobs the service node statistics are:

```
avg interarrivals .. = 52.9812435
avg wait ..... = 601.5260913
avg # in node ..... = 11.3535669
avg delay ..... = 2.5656638
avg # in queue ..... = 0.0484259
avg service time.....= 598.9764881
```

**Figura 5. risultati nodo consumazione ordinazione**

for 117.000000 jobs the service node statistics are:

```
avg interarrivals .. = 652.9825053
avg wait ..... = 1357.0124832
avg # in node ..... = 2.0781759
avg delay ..... = 157.3069849
avg # in queue ..... = 0.2409054
avg service time.....= 1199.6839055
```

**Figura 6. risultati nodo servizio delivery**

A:

nodo 1:  $47.4313261 \approx 2.3170612 + 45.1140767$

*nodo2* :  $115.9815608 \approx 4.1436867 + 111.8381402$

*nodo3* :  $601.5260913 \approx 2.5656638 + 598.9764881$

*nodo4* :  $1357.0124832 \approx 157.3069849 + 1199.6839055$

B:

nodo 1:  $0.8952475 \approx 0.0437336 + 0.1703666 * 5$

*nodo2* :  $2.3667242 \approx 0.0845562 + 0.2073513 * 11$

*nodo3* :  $11.3535669 \approx 0.0484259 + 0.2265537 * 50$



nodo4 :  $2.0781759 \approx 0.2409054 + 0.1537886 * 12$

C:

jobs nodo Preparazione Ordinazione = jobs nodo "tavoli" + jobs "corrieri"  
1559=1442+117

D:

L'unico caso in cui la configurazione single server è idonea è nella fascia Low, nello specifico per i nodi "Acquisizione Ordinazione", "Preparazione Ordinazione" e "Servizio Delivery".

$$E(T_Q) = \frac{pE(S)}{2(1-p)} \cdot (1 + C^2)$$

$$C^2 = \frac{\text{var}}{E(S)^2}$$

Fascia Low		
statistica	risultato analitico	risultato simulazione $\alpha = 0.05$
$E(T_{S1})$	50.06120455	49.9756036 +/- 1.8416161
$E(T_{S2})$	163.7595123	157.7344152 +/- 3.2717741
$E(T_{S4})$	1387.929979	1362.5799459 +/- 9.6161466

E:

Utilizzando una configurazione di centri sufficientemente grande, si è verificato che PQ (ovvero la probabilità che tutti i server siano occupati) così da poter considerare  $E(N_Q)=0$  e di conseguenza  $E(T_Q)=0$ .

$$P_Q = P_0 \cdot \frac{(mp)^m}{m!(1-p)}$$

$$P_0 = \frac{1}{\sum_{i=0}^{m-1} \frac{(mp)^i}{i!} + \frac{(mp)^m}{m!(1-p)}}$$

$$E(N_S) = E(N_Q) + mp$$

Pertanto  $E(N_S)=mp$ .

Si verifica tramite Little METTERE FORMULA che  $E(T_S)$  calcolata analiticamente sia uguale a quella ottenuta dai valori della simulazione.

Fascia Low		
statistica	risultato analitico	risultato simulazione $\alpha = 0.05$
$E(T_{S1})$	45.17703467	44.9772051 +/- 0.0617883
$E(T_{S2})$	112.38017	112.0483288 +/- 0.0924193
$E(T_{S3})$	602.5260698	599.9383699 +/- 0.7353477
$E(T_{S4})$	1177.220891	1200.4883297 +/- 5.9099119

Fascia Mid		
statistica	risultato analitico	risultato simulazione $\alpha = 0.05$
$E(T_{S1})$	45.20379887	44.9837288 +/- 0.061189
$E(T_{S2})$	112.4067955	112.0553249 +/- 0.1459916
$E(T_{S3})$	602.6624732	599.9468306 +/- 1.9940851
$E(T_{S4})$	1170.993994	1199.9618354 +/- 12.1383543

Fascia High		
statistica	risultato analitico	risultato simulazione $\alpha = 0.05$
$E(T_{S1})$	45.03944651	44.9711830 +/- 0.096872
$E(T_{S2})$	111.926478	112.0850403 +/- 0.3072413
$E(T_{S3})$	600.622211	600.6567294 +/- 3.8929259
$E(T_{S4})$	1179.946254	1209.4524092 +/- 29.0687722

## 9 Validazione

Nella fase di validazione vogliamo verificare che il modello computazionale sia coerente con il sistema reale che abbiamo analizzato. Non avendo un dataset con dati reali a disposizione, si opera una validazione che mostri, come avverrebbe nella realtà, l'aumentare dei tempi di attesa e di risposta in concomitanza dell'aumentare dei flussi di jobs in ingresso. In particolare si mostra come l'aumento del 10% di entrambi i flussi di jobs in entrata cambi i valori dei tempi di una giornata lavorativa.

Come si evince dalle immagini l'incremento del tempo di attesa nei nodi è evidente e in alcuni casi anche notevole.

Un'ulteriore dimostrazione è stata fatta nel caso stazionario attraverso l'ausilio del seguente grafico di confronto che mostra come l'incremento del 10% dei lambda negli arrivi comporta un'evidente differenza del tempo di risposta totale del sistema.

In rosso la configurazione incrementata il cui tempo di risposta supera i 1000 secondi.

for 1442.000000 jobs the service node statistics are:for 1603.000000 jobs the service node statistics are:

avg interarrivals .. = 52.9812435	avg interarrivals .. = 47.7235582
avg wait ..... = 47.4313261	avg wait ..... = 49.2976248
avg # in node ..... = 0.8952475	avg # in node ..... = 1.0329830
avg delay ..... = 2.3170612	avg delay ..... = 4.2131812
avg # in queue ..... = 0.0437336	avg # in queue ..... = 0.0882830
avg service time.....= 45.1140767	avg service time.....= 45.0841737

for 1559.000000 jobs the service node statistics are:for 1733.000000 jobs the service node statistics are:

avg interarrivals .. = 49.0051014	avg interarrivals .. = 44.1436028
avg wait ..... = 115.9815608	avg wait ..... = 120.1490564
avg # in node ..... = 2.3667242	avg # in node ..... = 2.7217773
avg delay ..... = 4.1436867	avg delay ..... = 8.3241547
avg # in queue ..... = 0.0845562	avg # in queue ..... = 0.1885699
avg service time.....= 111.8381402	avg service time.....= 111.8253600

for 1442.000000 jobs the service node statistics are:for 1603.000000 jobs the service node statistics are:

avg interarrivals .. = 52.9812435	avg interarrivals .. = 47.7235582
avg wait ..... = 601.5260913	avg wait ..... = 609.5815782
avg # in node ..... = 11.3535669	avg # in node ..... = 12.7731796
avg delay ..... = 2.5656638	avg delay ..... = 10.4140779
avg # in queue ..... = 0.0484259	avg # in queue ..... = 0.2182167
avg service time.....= 598.9764881	avg service time.....= 599.1821819

for 117.000000 jobs the service node statistics are: for 130.000000 jobs the service node statistics are:

avg interarrivals .. = 652.9825053	avg interarrivals .. = 588.4681825
avg wait ..... = 1357.0124832	avg wait ..... = 1405.2300031
avg # in node ..... = 2.0781759	avg # in node ..... = 2.3879456
avg delay ..... = 157.3069849	avg delay ..... = 207.1801066
avg # in queue ..... = 0.2409054	avg # in queue ..... = 0.3520668
avg service time.....= 1199.6839055	avg service time.....= 1198.0734212

(a) Valori Nodi Caso Base

(b) Valori Nodi Caso Base aumentati 10%

Figura 7: Confronto Valori

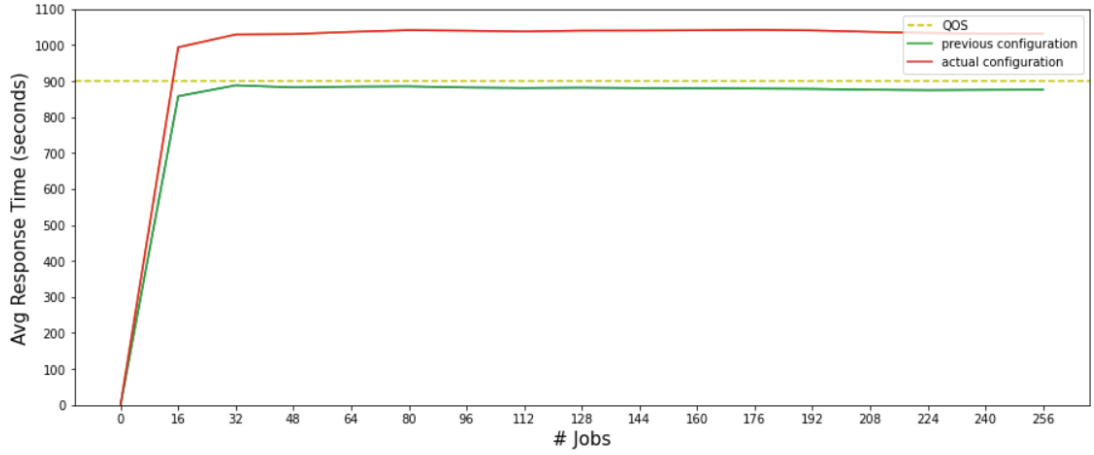


Figura 8. Grafico di Confronto

## 10 Analisi dei risultati

### 10.1 Analisi Collo di Bottiglia

L'analisi del collo di bottiglia è un processo in cui si utilizza l'analisi operativa per calcolare la domanda media per ciascun centro di un sistema. L'obiettivo è identificare i centri con la domanda più alta, che vengono considerati i "colli di bottiglia". Identificarli è essenziale per migliorare l'efficienza del sistema, in quanto consente di concentrare gli sforzi sulle aree critiche per ottimizzare le prestazioni globali.

$$\begin{cases} V_1 = \frac{\lambda_A}{\gamma} \\ V_2 = \frac{\lambda_B}{\gamma} \\ V_3 = \frac{\lambda_C}{\gamma} \\ V_4 = \frac{\lambda_D}{\gamma} \\ \gamma = \lambda_1 + \lambda_2 \end{cases} \quad (5)$$

Per la scelta di Ploss si è deciso di considerarla trascurabile sia per il caso Low che Mid poichè la coda nel nodo della cassa ha sempre un valore molto basso, mentre le è stata assegnata il caso peggiore (3% nel caso High in quanto fascia con elevata affluenza) , poichè l'analisi del collo di bottiglia ha lo scopo di evidenziare il caso peggiore.

<b>Visite/ Fascia Oraria</b>	Low	Mid	High
<b>Cassa</b>	0.95134503	0.958541147	0.87153451
<b>Cucina</b>	1	1	0.97304532
<b>Tavoli</b>	0.951111	0.95841646	0.87425883
<b>Delivery</b>	0.0488187	0.041521197	0.09878889

Si procede con il calcolo delle domande medie per verificare quale nodo sia il collo di bottiglia, ovvero quello con la domanda maggiore.

$$D^i = V^i S^i$$

<b>Domande/ Fascia Oraria</b>	Low	Mid	High
<b>Cassa</b>	42.81052635	43.13435162	39.219050
<b>Cucina</b>	112	112	108.9810758
<b>Tavoli</b>	570.666666	575.049876	524.555298
<b>Delivery</b>	58.58244	49.8254364	118.5466685

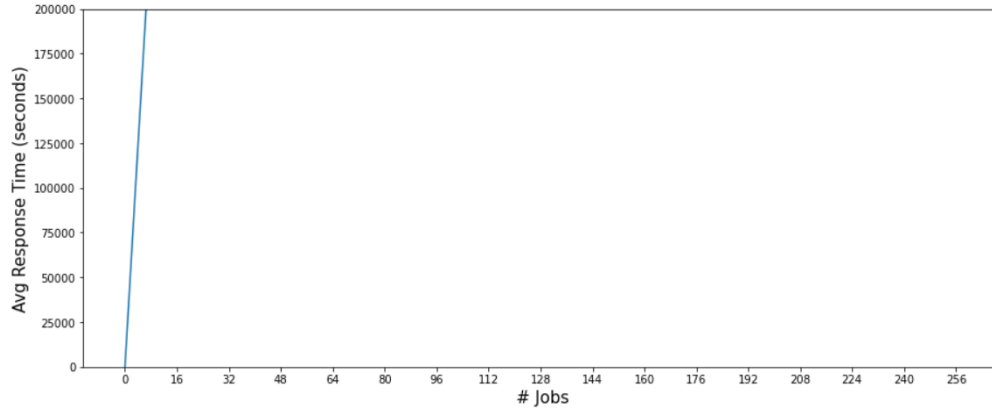
Come si evince dalla tabella, per ogni fascia oraria la domanda massima è quella relativa al nodo della consumazione al tavolo, pertanto esso costituisce il collo di bottiglia del sistema.

## 10.2 Analisi ad Orizzonte Infinito

Nella simulazione ad orizzonte infinito si procede con l'analisi dello stato stazionario del sistema, raggiungibile mediante una riproduzione del funzionamento del McDonald per un tempo sufficientemente lungo da portarlo alla stazionarietà. Per fare ciò, anziché fare diverse repliche indipendenti, si procede con un'unica simulazione che viene poi suddivisa in un numero K di batch ciascuno da B jobs. I valori di K e B sono opportunamente selezionati affinché il valore di autocorrelazione dei batch non superi il valore soglia di 0,2. Nel sistema considerato tali valori sono K=64 e B=256. Le statistiche rilevanti di ogni batch vengono estrapolate e consegnate all'algoritmo di Welford per ottenere medie, varianze, deviazioni standard e intervalli di confidenza riferiti all'intera simulazione ad orizzonte infinito. Grazie allo studio di questi ultimi è stato possibile fare confronti tra configurazioni differenti di server per trovare la condizione ottimale che soddisfi le QoS. All'interno di una singola simulazione il numero dei centri così come il flusso di jobs in ingresso rimane costante, pertanto tale analisi è stata suddivisa in tre diversi casi, ciascuno riferito ad una specifica tipologia di fascia oraria.

### 10.2.1 Analisi ad Orizzonte Infinito: Caso Low

Si decide di partire dalla configurazione scelta nel caso base. Essa viene inserita nella simulazione con la speranza che soddisfi la stazionarietà ricercata.



**Figura 9. Configurazione1,1,2,1**

for this configuration lambda is Low and centers are 1 1 2 1

the actual node statistics are:

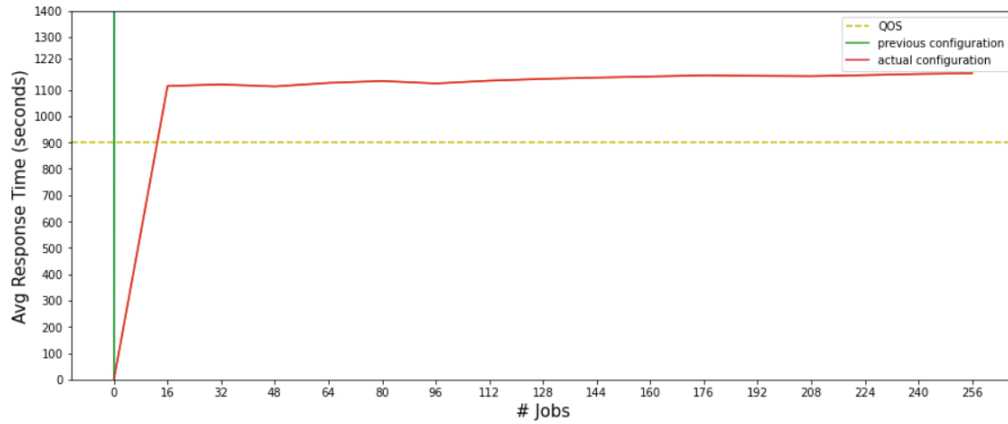
node	mean response time	std response time	w response time	utilization
1	50.1088430	7.0787600	1.8440694	0.1832506
2	158.8513377	12.6036240	3.2833374	0.4795424
3	517375.8608075	719.2884406	187.3799681	0.9999751
4	1374.5637112	37.0751091	9.6583406	0.2479732

the system statistics are:

mean response time	std response time	w response time	income
492393.7716072	701.7077537	182.8000689	1520.1880653 \$

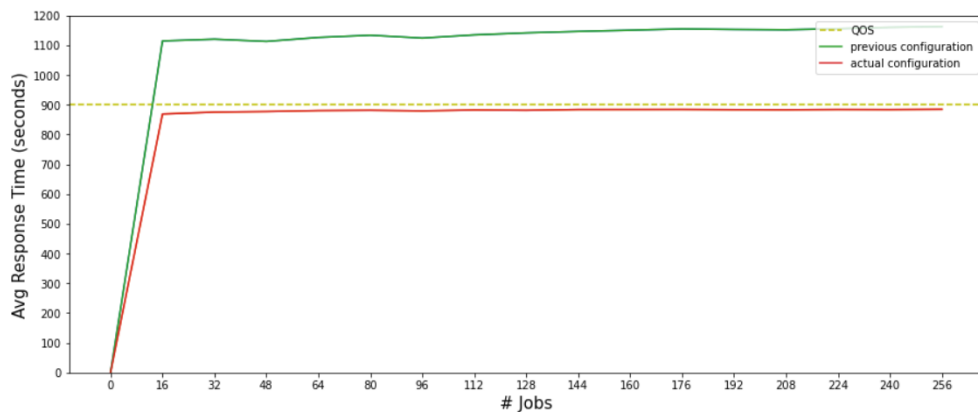
**Figura 10. Statistiche Configurazione1,1,2,1**

é verificato facilmente, però, che con questa configurazione non si raggiunge mai lo stato stazionario; il sistema non è quindi stabile ed aumenta all'infinito il tempo di risposta. Vi è infatti una utilizzazione tendente a 1 per il sottosistema "consumazione ordinazione", e quindi tutti i job in ingresso non vengono smaltiti con la stessa velocità con cui arrivano, causando accodamenti.



**Figura 11. Configurazione 1,1,2,1 vs Configurazione 1,1,3,1**

Si migliora quindi la configurazione aggiungendo un servente nel nodo "consumazione ordinazione", raggiungendo così la stazionarietà. Ancora, però, non rispetta la QoS. Il tempo medio di risposta è infatti 1162.6058251 secondi, con un intervallo di confidenza  $\pm 41.0363145$  secondi. Sono circa tre minuti più di quanto si è prefissato. Come viene mostrato nelle immagini seguenti, la configurazione ottimale del caso Low è [1,1,4,1]. Con essa la QoS sul tempo di risposta viene rispettata. Il tempo di risposta medio raggiunto è di 884.5236338  $\pm 7.7477358$  secondi e il guadagno medio stimato è di 1623.22 € circa, il quale ci mostra, al di là della cifra in sé, che questa configurazione comporta un ricavo maggiore della spesa.



**Figura 12. Configurazione 1,1,4,1**

the actual node statistics are:

node	mean response time	std response time	w response time	utilization
1	50.1059161	7.0785532	1.8440156	0.1837367
2	158.8344690	12.6029548	3.2831631	0.4804103
3	642.6495333	25.3505332	6.6040017	0.6125000
4	1367.4192895	36.9786329	9.6332078	0.2457535

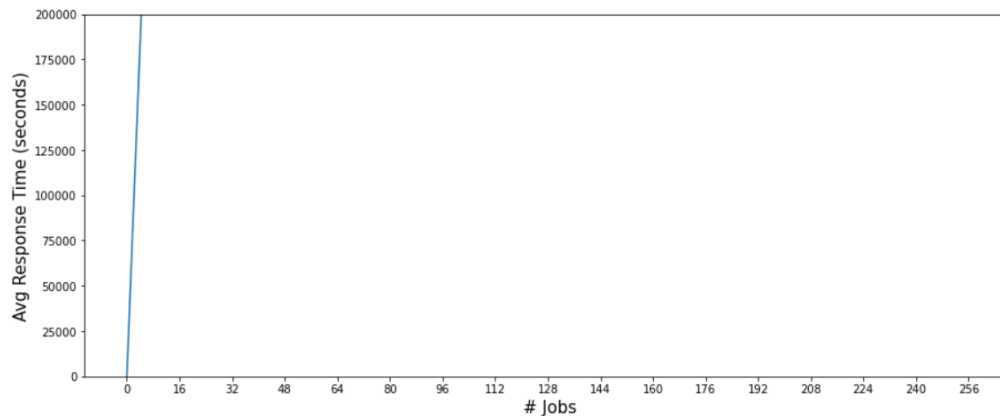
the system statistics are:

mean response time	std response time	w response time	income
884.5236338	29.7409420	7.7477358	1623.2163610 \$

**Figura 13. Configurazione 1,1,4,1**

### 10.2.2 Analisi ad Orizzonte Infinito: Caso Mid

Anche in questo caso, partendo dalla configurazione scelta nel caso base, non si raggiunge la stazionarietà. Il collo di bottiglia è, come ci si sarebbe aspettato, sempre il sottosistema "consumazione ordinazione". La sua utilizzazione è infatti nuovamente tendente ad 1.



**Figura 14. Configurazione1,2,5,1**



for this configuration lambda is Mid and centers are 1 2 5 1

the actual node statistics are:

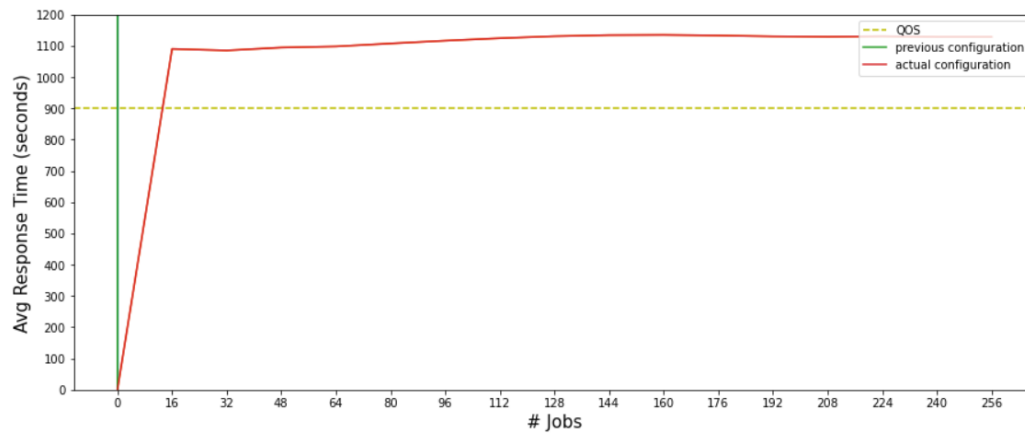
node	mean response time	std response time	w response time	utilization
1	95.0020275	9.7468984	2.5391392	0.6917704
2	308.5487557	17.5655559	4.5759575	0.8974325
3	771439.6837014	878.3163916	228.8079276	0.9999884
4	2676.7472326	51.7372905	13.4779475	0.7777852

the system statistics are:

mean response time	std response time	w response time	income
739897.1871401	860.1727659	224.0813787	1886.6233285 \$

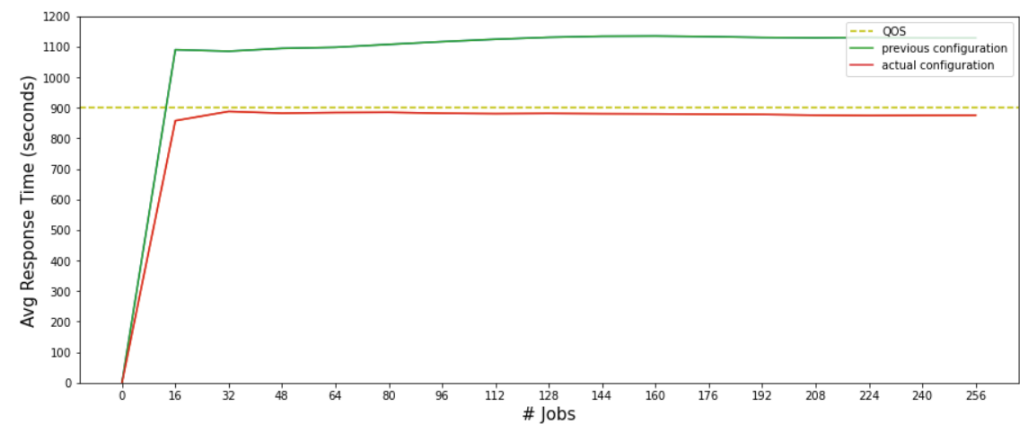
**Figura 15. Statistiche Configurazione1,2,5,1**

Si opta quindi per un raddoppiamento del numero dei centri del sottosistema sotto sforzo. La nuova configurazione scelta porta alla stazionarietà ma, come in precedenza nel caso Low, anche qui siamo decisamente oltre la QoS richiesta. Il tempo di risposta è 1128.291831 secondi con un intervallo di confidenza pari a  $\pm 61.3575386$  secondi.



**Figura 16. Configurazione 1,2,5,1 vs Configurazione 1,2,10,2**

La configurazione finale, ottenuta dopo un'analisi delle varie possibilità e considerando, tra i criteri di scelta, anche una massimizzazione del guadagno, è la seguente [2,3,11,2]. Con essa il tempo di risposta medio risulta essere di  $875.499667 \pm 7.708113$  secondi e il guadagno pari a 1913.09 €.



**Figura 17. Configurazione 2,3,11,2**

for new configuration lambda is Mid and centers are 2 3 11 2

the actual node statistics are:

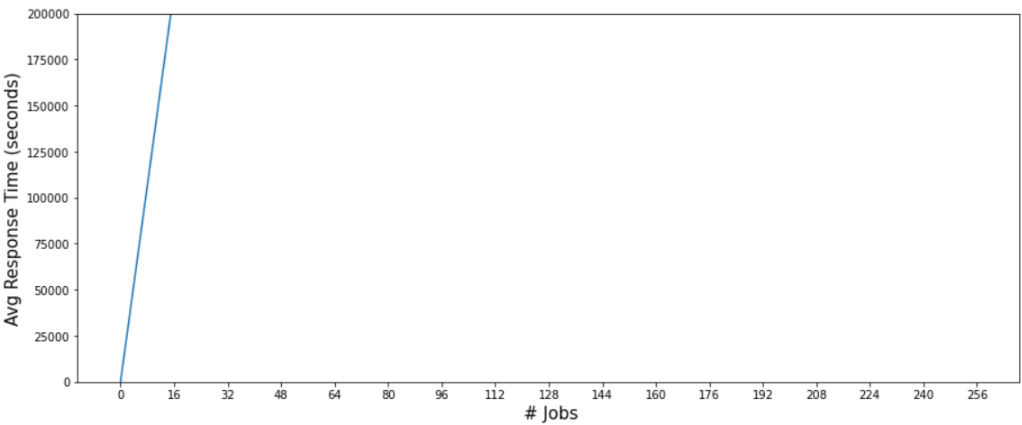
node	mean response time	std response time	w response time	utilization
1	48.4033249	6.9572498	1.8124151	0.3473894
2	127.7278527	11.3016748	2.9441700	0.6009385
3	675.7410129	25.9950190	6.7718951	0.8422270
4	1292.7959929	35.9554724	9.3666670	0.3895117

the system statistics are:

mean response time	std response time	w response time	income
875.4996668	29.5888436	7.7081130	1913.0912302 \$

**Figura 18. Configurazione 2,3,11,2**

### 10.2.3 Analisi ad Orizzonte Infinito: Caso high



**Figura 19. Configurazione 4,7,20,5**

for this configuration lambda is High and centers are 4 7 20 5

the actual node statistics are:

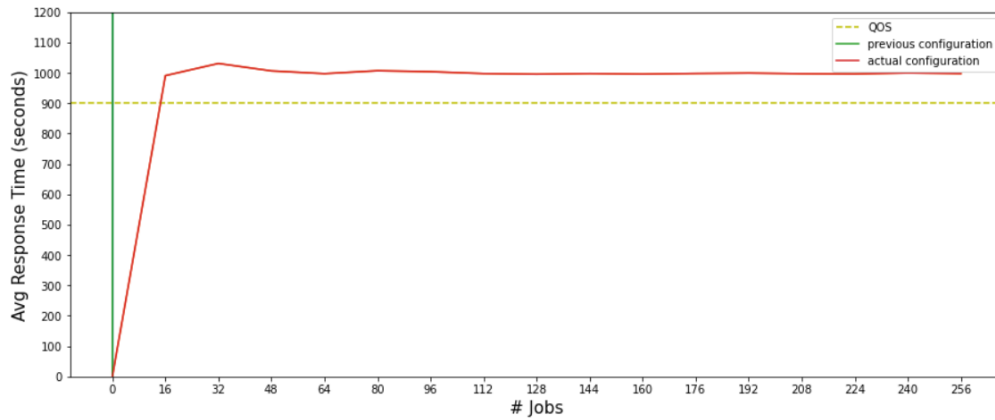
node	mean response time	std response time	w response time	utilization
1	66.0142037	8.1249125	2.1165999	0.8241482
2	66515.5350424	257.9060586	67.1864391	0.9999048
3	151150.8739309	388.7812675	101.2804007	0.9999518
4	108369.7809595	329.1956576	85.7579079	0.9997477

the system statistics are:

mean response time	std response time	w response time	income
213382.3277739	461.9332503	120.3370342	1962.2844588 \$

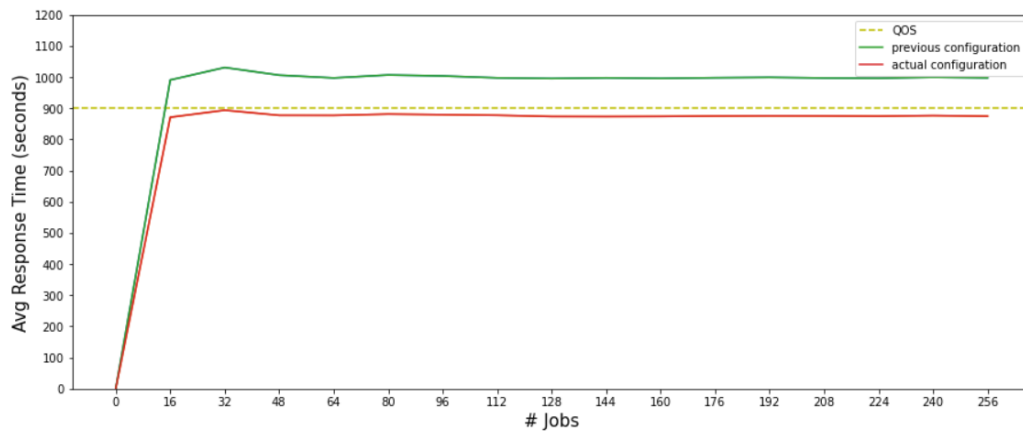
**Figura 20. Statistiche Configurazione 4,7,20,5**

Si nota che con questa configurazione non si raggiunge mai lo stato stazionario (in particolare sono 3 i nodi che hanno un roh tendente a 1); il sistema non è quindi stabile ed aumenta all'infinito il tempo di risposta.



**Figura 21. Configurazione 4,7,20,5 vs Configurazione 4,12,45,11**

Si migliora quindi la configurazione aggiungendo server nei vari nodi in cui  $\rho$  tende a 1 (nello specifico il nodo "preparazione ordinazione", il nodo "consumazione tavolo" e quello "servizio delivery"). Ora, con la nuova configurazione 4,12,45,11 il sistema è stabile, ma, con un valore di  $997.2959517 \pm 35.3318567$  secondi, ancora non rispetta la QoS del tempo di risposta totale. Ottimizziamo quindi con la seguente configurazione 4,10,48,13



**Figura 22. Configurazione 4,10,48,13**

for new configuration lambda is High and centers are 4 10 48 13

the actual node statistics are:

node	mean response time	std response time	w response time	utilization
1	65.2200082	8.0758906	2.1038293	0.8234211
2	139.4532060	11.8090307	3.0763400	0.9108794
3	611.7192012	24.7329578	6.4431188	0.9147277
4	1248.6462666	35.3361892	9.2053391	0.7431568

the system statistics are:

mean response time	std response time	w response time	income
874.4355462	29.5708564	7.7034272	2148.6703827 \$

**Figura 23. Configurazione 4,10,48,13**

Questa configurazione rispetta finalmente il limite imposto di 900 secondi. Infatti il valore riscontrato è di 874.435546 +/- 7.703427 secondi e l'income medio è anch'esso positivo, con un valore di 2148.67 €

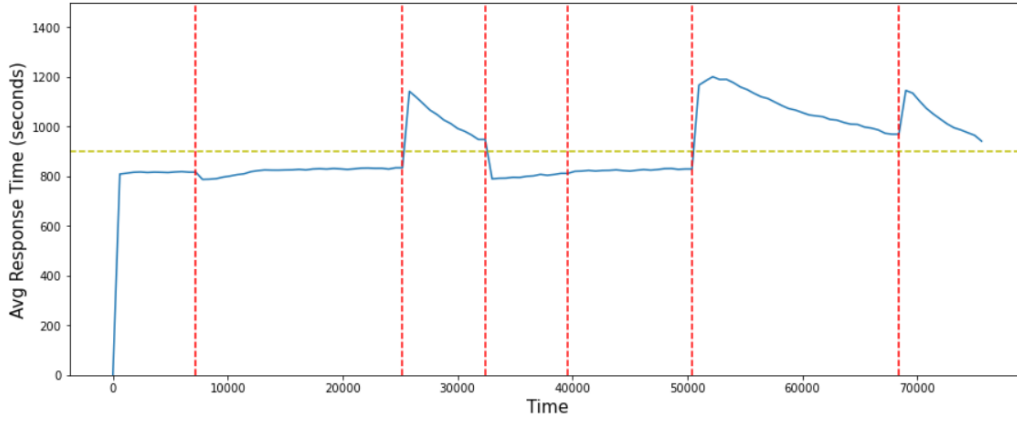
### 10.3 Analisi ad Orizzonte Finito

Con l'analisi ad orizzonte infinito delle singole fasce abbiamo individuato la configurazione ottima, che permette di minimizzare il costo rispettando i QoS durante l'intera giornata lavorativa.

**{ 1, 1, 4, 1}, {2, 3, 11, 2}, {4, 10, 48, 13}**

Con l'analisi ad orizzonte finito andiamo a verificare che lo scenario ottimo minimizzi effettivamente i costi, analizzandolo nell'arco di un'intera giornata lavorativa. Il cambio di fascia oraria, che avviene diverse volte nell'arco della giornata, è il momento in cui varia il tasso di ingresso e vengono di conseguenza accesi o spenti un certo numero di server. Ciò comporta delle problematiche nello smaltimento dei job presenti nelle code durante il cambio, in particolare quando si passa ad una configurazione con meno server.

### 10.3.1 Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11



**Figura 24. Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11**

L'attuale configurazione, estratta dall'analisi del caso stazionario, non rispetta la QoS richiesta. In particolare il cambio fascia da un lambda maggiore ad uno minore comporta un rapido picco nei tempi di risposta. Inoltre la Ploss supera il 3% arrivando ad un valore di 4.2649%. Osservando i roh di tale configurazione, sono stati opportunamente modificati i vari centri per le fasce interessate. Una maggiorazione è stata fatta anche nei centri delle fasce precedenti a quelle presentanti il picco in ingresso, così da ridurre il numero di jobs ancora presenti nelle code al momento del cambio fascia.

### 10.3.2 Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11

INVALID CONFIGURATION

the average taking of a day at McDonald in Via Tuscolana is 12868.876721904699

the average loss percentage of a day at McDonald in Via Tuscolana is 4.264973042321775 %

the average utilization of the servers present in the nodes during the time slot 08:00 - 10:00 is:

cash register	kitchen	table	delivery
0.1670704	0.4358320	0.5527007	0.2367753

the average utilization of the servers present in the nodes during the time slot 10:00 - 15:00 is:

cash register	kitchen	table	delivery
0.3335475	0.5737991	0.8203443	0.3322802

the average utilization of the servers present in the nodes during the time slot 15:00 - 17:00 is:

cash register	kitchen	table	delivery
0.1773589	0.4900844	0.7727321	0.5152469

the average utilization of the servers present in the nodes during the time slot 17:00 - 19:00 is:

cash register	kitchen	table	delivery
0.3304127	0.5615163	0.8414347	0.2573024

the average utilization of the servers present in the nodes during the time slot 19:00 - 22:00 is:

cash register	kitchen	table	delivery
0.7662701	0.8390054	0.8776987	0.6507012

the average utilization of the servers present in the nodes during the time slot 22:00 - 03:00 is:

cash register	kitchen	table	delivery
0.3446925	0.6104275	0.9201637	0.5890397

the average utilization of the servers present in the nodes during the time slot 03:00 - 05:00 is:

cash register	kitchen	table	delivery
0.1843493	0.5110642	0.7985983	0.5402829

Figura 25. Scenario 1 - 1,1,4,1, 2,3,11,2, 4,12,45,11

### 10.3.3 Scenario 2 - 2,2,5,2, 2,5,20,5, 4,12,50,11

In questa configurazione però, nonostante un considerevole incremento nel numero dei centri, si riscontrano ancora picchi che superano la QoS di 900 secondi sul tempo di risposta medio.

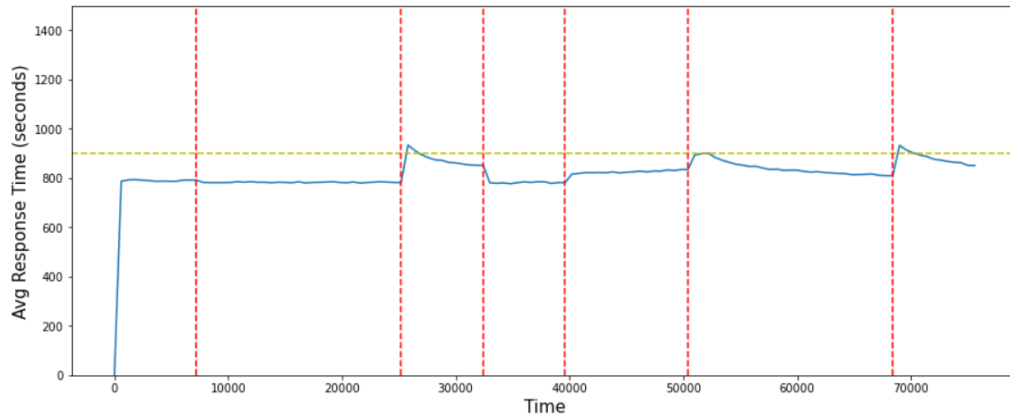


Figura 26. Scenario 2 - 2,2,5,2, 2,5,20,5, 4,12,50,11



```

INVALID CONFIGURATION
the average taking of a day at McDonald in Via Tuscolana is 12160.316661531871

the average loss percentage of a day at McDonald in Via Tuscolana is 3.725592045492975 %

the average utilization of the servers present in the nodes during the time slot 08:00 - 10:00 is:

cash register      kitchen      table      delivery
0.0897197         0.2335633   0.4771529   0.1216968

the average utilization of the servers present in the nodes during the time slot 10:00 - 15:00 is:

cash register      kitchen      table      delivery
0.3333961         0.3455099   0.4538645   0.1479841

the average utilization of the servers present in the nodes during the time slot 15:00 - 17:00 is:

cash register      kitchen      table      delivery
0.1000855         0.2733815   0.6598561   0.2440394

the average utilization of the servers present in the nodes during the time slot 17:00 - 19:00 is:

cash register      kitchen      table      delivery
0.3344804         0.3450536   0.4762738   0.1228437

the average utilization of the servers present in the nodes during the time slot 19:00 - 22:00 is:

cash register      kitchen      table      delivery
0.7690035         0.7035561   0.8471179   0.7672202

the average utilization of the servers present in the nodes during the time slot 22:00 - 03:00 is:

cash register      kitchen      table      delivery
0.3443287         0.3679865   0.5281055   0.3234202

the average utilization of the servers present in the nodes during the time slot 03:00 - 05:00 is:

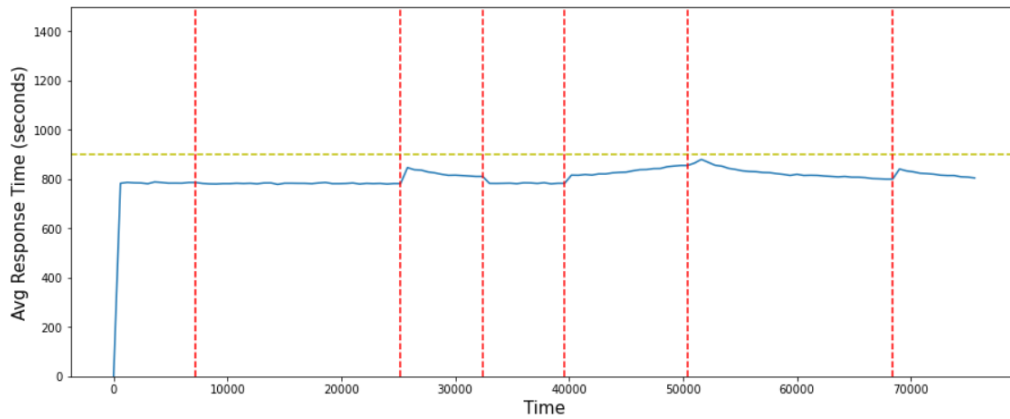
cash register      kitchen      table      delivery
0.0992029         0.2693436   0.6399501   0.3045860

```

**Figura 27. Scenario 2 - 2,2,5,2, 2,5,20,5, 4,12,50,11**

#### 10.3.4 Scenario ottimo - 2,3,6,3, 2,6,23,9, 5,11,50,10

Infine si analizza il comportamento nel transiente della configurazione ottima (ottenuta incrementando oculatamente i centri con roh più elevati nelle fasce che lo richiedevano ed eliminando ove possibile i centri superflui al raggiungimento degli obiettivi prefissati), verificando che effettivamente il costo sulle 21 ore sia minimo, e che vengano rispettate entrambe le QoS.



**Figura 28.** Scenario ottimo - 2,3,6,3, 2,6,23,9, 5,11,50,10

Il guadagno giornaliero medio è di 12055.21€ ed è il massimo ottenibile mantenendo un tempo di risposta sotto i 900 secondi. Per quanto riguarda la probabilità di perdita essa è pari all'1.6625% e quindi inferiore alla soglia imposta. Come si evince dal grafico, infine, in tutte le fasce orarie non viene superata la soglia di 900 secondi per il tempo di risposta, confermando quindi l'ottimalità della configurazione. Le osservazioni sono state eseguite con intervalli di 10 minuti l'una dall'altra ed ognuna di esse non ha raggiunto il limite. Il maggior tempo di risposta osservato è il picco di 880.0630789 secondi riscontrato all'inizio della penultima fascia, la Mid che segue l'unica fascia high.

the average taking of a day at McDonald in Via Tuscolana is 12052.755916700837

the average loss percentage of a day at McDonald in Via Tuscolana is 1.6625229713520069 %

the average utilization of the servers present in the nodes during the time slot 08:00 - 10:00 is:

cash register	kitchen	table	delivery
0.0897916	0.1564708	0.4021889	0.0810144

the average utilization of the servers present in the nodes during the time slot 10:00 - 15:00 is:

cash register	kitchen	table	delivery
0.3370280	0.2908451	0.3991141	0.0820672

the average utilization of the servers present in the nodes during the time slot 15:00 - 17:00 is:

cash register	kitchen	table	delivery
0.0976100	0.1777096	0.5364277	0.1620344

the average utilization of the servers present in the nodes during the time slot 17:00 - 19:00 is:

cash register	kitchen	table	delivery
0.3325142	0.2845049	0.4129019	0.0554611

the average utilization of the servers present in the nodes during the time slot 19:00 - 22:00 is:

cash register	kitchen	table	delivery
0.6443863	0.7989910	0.8860473	0.7912179

the average utilization of the servers present in the nodes during the time slot 22:00 - 03:00 is:

cash register	kitchen	table	delivery
0.3443499	0.3091551	0.4684390	0.2379328

the average utilization of the servers present in the nodes during the time slot 03:00 - 05:00 is:

cash register	kitchen	table	delivery
0.1003125	0.1829111	0.5308945	0.2073243

**Figura 29. ottimo - 2,3,6,3, 2,6,23,9, 5,11,50,10**

## 11 Algoritmo Migliorativo

Come algoritmo migliorativo si vuole ridurre la percentuale di perdite che il McDonald riscontra nell'arco della giornata, dovute alla presenza di una eccessiva coda alle casse che porta gli utenti a desistere dal loro intento originale. Per fare ciò si è considerata la creazione di un'applicazione che consente all'utente di pagare direttamente tramite essa inviando così l'ordinazione direttamente alle cucine.

Per sopperire al possibile malcontento dovuto alle lunghe attese del cibo (in particolare per gli utenti delivery) si è deciso di introdurre la priorità alle cucine. Le ordinazioni di tipo delivery, date le lunghe attese, ottengono il primo livello di priorità, seguite dalle ordinazioni fatte tramite applicazione che precedono le ordinazioni fatte alle casse.

L'introduzione della priorità è stata affiancata a quella di un supplemento nei prezzi dell'ordinazione per i jobs di tipo delivery e di tipo applicazione, in quanto un nuovo servizio offerto dovrebbe essere correttamente ricompensato. Viene così massimizzato, sia tramite il supplemento sia tramite la riduzione della probabilità di perdita, il guadagno giornaliero del McDonald.

### 11.1 Obiettivo

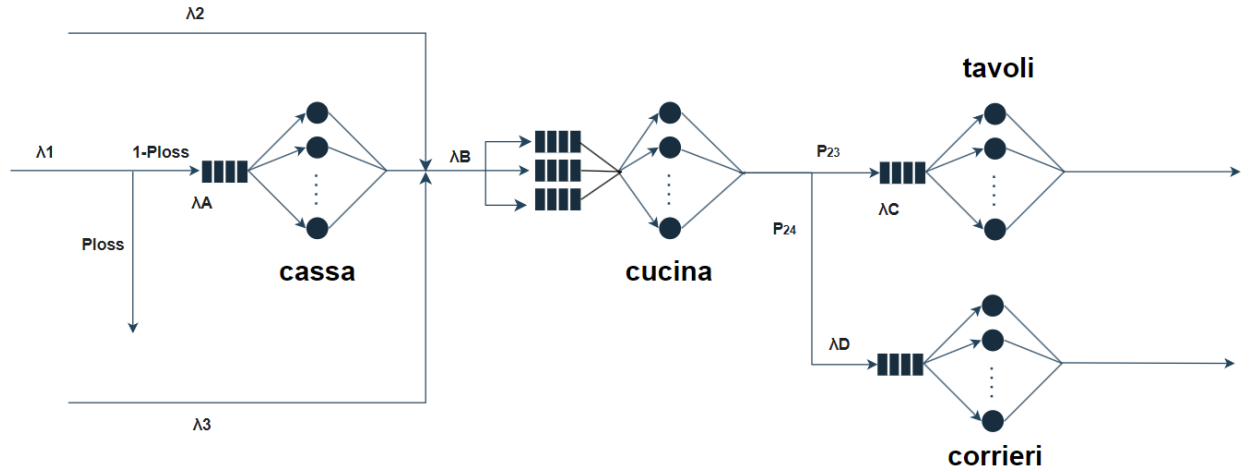
Oltre mantenere la QoS precedente sul tempo di risposta, che deve rimanere inferiore ai 900 secondi, l'introduzione dell'applicazione mira a ridurre la percentuale di perdita degli utenti dalla coda del sottosistema "acquisizione ordinazione". La seconda QoS viene quindi abbassata dal 3% del caso base all'1% qui nel caso migliorato. Oltre a ciò si dovrà anche registrare un aumento del guadagno giornaliero dovuto alle nuove politiche di priorità introdotte nel sottosistema "preparazione ordinazione".

### 11.2 Modello Concettuale

Rispetto al modello concettuale originale, si aggiunge una nuova tipologia di utente, "utenti applicazione". Vi sarà quindi un nuovo flusso in ingresso, anch'esso entrante nel sottosistema cucina, il quale terrà conto di questo nuovo tipo di job per smistarlo correttamente nei nodi successivi.

L'arrivo di un job di questa terza categoria costituisce quindi un'evento aggiuntivo rispetto a tutti quelli precedentemente elencati. Il nodo "preparazione ordinazione" presenta ora una coda con tre differenti livelli di priorità, in ordine dal più importante, quella per i job di tipo delivery, quella per i job applicazione e infine quella per i job che passano per la Cassa. Dal nodo con

priorità i tre tipi di job vengono poi smistati nei due sottosistemi successivi. Le ordinazioni tramite applicazione sono indirizzate al nodo dei tavoli.



**Figura 30. Modello Concettuale Avanzato**

### 11.3 Modello Specifiche

Nell'algoritmo migliorativo la presenza di un nuovo tipo di job che, nella totalità delle ordinazioni giornaliere (le quali restano invariate) va a sostituire alcune delle ordinazioni passanti per il primo sottosistema. Un nuovo lambda viene aggiunto ai precedenti che diventano:

- Fascia bassa (08:00 – 10:00, 15:00 – 17:00, 03:00 – 05:00):
  - $\lambda_1 = 0.0036603 \text{ arrivi/sec}$
  - $\lambda_2 = 0.000208 \text{ arrivi/sec}$
  - $\lambda_3 = 0.0004067 \text{ arrivi/sec}$
- Fascia media (10:00 – 15:00, 17:00 – 19:00, 22:00 – 03:00):
  - $\lambda_1 = 0.0122984 \text{ arrivi/sec}$
  - $\lambda_2 = 0.000667 \text{ arrivi/sec}$
  - $\lambda_3 = 0.0030746 \text{ arrivi/sec}$

- Fascia alta(19:00 – 22:00):
  - $\lambda_1 = 0.0512456 \text{arrivi/sec}$
  - $\lambda_2 = 0.008271 \text{arrivi/sec}$
  - $\lambda_3 = 0.0219624 \text{arrivi/sec}$

Le percentuali di utenti nel sistema sono diverse a seconda della fascia oraria considerata.

Per la fascia Low :

- Percentuale Clienti al Tavolo : 85.60665%
- Percentuale Clienti Delivery : 4.8815%
- Percentuale Clienti Applicazione : 9.51185%

Per la fascia Mid :

- Percentuale Clienti al Tavolo : 76.6760%
- Percentuale Clienti Delivery : 4.1550%
- Percentuale Clienti Applicazione : 19.1690%

Per la fascia High :

- Percentuale Clienti al Tavolo : 62.89318%
- Percentuale Clienti Delivery : 10.1526%
- Percentuale Clienti Applicazione : 26.95422%

Per quanto riguarda spese e ricavi l'unica differenza è l'aggiunta di un supplemento al prezzo del menù dovuto all'inserimento delle priorità. Tale aumento consiste in 2 euro per gli utenti delivery che godono della priorità massima mentre gli utenti che pagano sull'applicazione e che usufruiscono del secondo livello di priorità avranno un incremento della spesa di solo 1 euro.

## 11.4 Probabilità di Routing

Le probabilità di routing non variano rispetto al caso base con l'aggiunta del nuovo tipo di job in quanto il flusso resta lo stesso. Ciò che varia è il flusso in ingresso nei vari sottosistemi.

### 11.4.1 Tabella Di Routing

Di seguito la matrice di routing che rappresenta tutte le probabilità di routing:

	Esterno	Sottosistema 1	Sottosistema 2	Sottosistema 3	Sottosistema 4
Esterno	0	$P_t$	$P_d + P_a$	0	0
Sottosistema 1	$P_{loss}$	0	$1 - P_{loss}$	0	0
Sottosistema 2	0	0	0	$p_{23}$	$p_{24}$
Sottosistema 3	1	0	0	0	0
Sottosistema 4	1	0	0	0	0

*eq:sistema generale*

$$\begin{cases} \lambda_A = \lambda_1 - P_{loss} \\ \lambda_B = \lambda_2 + \lambda_3 + \lambda_A \\ \lambda_C = \lambda_B \cdot p_{23} \\ \lambda_D = \lambda_B \cdot p_{24} \end{cases} \quad (6)$$

### 11.4.2 Flussi In Ingresso: Caso Low

Per la fascia Low i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.003660 - P_{loss}job/sec \\ \lambda_B = 0.004275 - P_{loss}job/sec \\ \lambda_C = 0.004066 - 0.951185 \cdot P_{loss}job/sec \\ \lambda_D = 0.000208 - 0.048815 \cdot P_{loss}job/sec \end{cases} \quad (7)$$

### 11.4.3 Flussi In Ingresso: Caso Mid

Per la fascia Mid i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.012298 - P_{loss}job/sec \\ \lambda_B = 0.016040 - P_{loss}job/sec \\ \lambda_C = 0.015373 - 0.958450 \cdot P_{loss}job/sec \\ \lambda_D = 0.000666 - 0.041550 \cdot P_{loss}job/sec \end{cases} \quad (8)$$

#### 11.4.4 Flussi In Ingresso: Caso High

Per la fascia High i valori dei  $\lambda$  sono i seguenti:

$$\begin{cases} \lambda_A = 0.051246 - Plossjob/sec \\ \lambda_B = 0.081479 - Plossjob/sec \\ \lambda_C = 0.073207 - 0.898474 \cdot Plossjob/sec \\ \lambda_D = 0.008272 - 0.101526 \cdot Plossjob/sec \end{cases} \quad (9)$$

### 11.5 Modello Computazionale

Anche il modello Computazionale resta pressocchè invariato rispetto al caso base. Unica eccezione, nell'ambito delle strutture dati, la fa l'aggiunta di una struttura chiamata Priority. Essa è utilizzata per tener conto della popolazione totale, dei completamenti e del numero di jobs in coda per i jobs prioritari all'interno del sottosistema "preparazione ordinazione" dove tutte e tre le tipologie di utenti confluiscono.

### 11.6 Verifica

Poichè la maggior parte del sistema resta invariato, per la verifica è stato preso in considerazione soltanto il nodo relativo a "Preparazione Ordinazione" in cui è stata aggiunta la coda prioritaria. Per effettuare la verifica sono stati utilizzati gli stessi vincoli del caso base.

A. Il tempo di risposta (wait) per un server è sempre uguale alla somma del tempo di attesa (delay) e del tempo di servizio (service).

B. La popolazione nel nodo deve essere uguale alla somma della popolazione in coda e della popolazione del servente

C. La popolazione nel nodo "preparazione Ordinazione" viene smistata ai nodi "tavoli" e "corrieri", tenendo conto anche del nuovo tipo di jobs. Pertanto la somma delle popolazioni di questi ultimi 2 deve essere uguale alla prima.

D. Utilizzando le KP generali nei casi in cui in un sottosistema è presente un solo centro (ove possibile) verifichiamo che i risultati analitici siano uguali a quelli sperimentali.



for 137.000000 priority jobs the service node statistics are:

```

avg interarrivals .. = 553.7062207
avg wait ..... = 117.8704240
avg # in node ..... = 0.2128754
avg delay ..... = 6.1046500
avg # in queue ..... = 0.0110251
avg service time.....= 111.6105430

```

for 555.000000 priority jobs the service node statistics are:

```

avg interarrivals .. = 136.6806347
avg wait ..... = 118.6703988
avg # in node ..... = 0.8682313
avg delay ..... = 7.0063867
avg # in queue ..... = 0.0512610
avg service time.....= 111.6390024

```

**Figura 31. Priorità VS Delivery**

A:

$$117.9704240 \approx 6.1046500 + 111.6105430 + 118.6703988 \approx 7.0063867 + 111.6390024$$

B:

$$0.2128754 \approx 0.0110251 + 0.2022020 + 0.8682313 \approx 0.0512610 + 0.819140$$

C:

$$555 + 137 + 1106 = 1798$$

D:

$$E(T_Q) = \frac{pE(S)}{2(1 - \sum_{i=1}^{k-1} p^i)(1 - \sum_{i=1}^k p^i)} \cdot (1 + C^2)$$

Fascia Low		
statistica	risultato analitico	risultato simulazione $\alpha = 0.05$
$E(T_{Qdv})$	27.6334415	27.1215348 +/- 2.5815468
$E(T_{Qapp})$	29.807028	33.052831 +/- 2.597653
$E(T_{Qnp})$	55.878421	49.971293 +/- 2.956930

## 11.7 Validazione

```
for 137.000000 priority jobs the service node statistics are:  for 217.000000 priority jobs the service node statistics are:

avg interarrivals .. = 553.7062207          avg interarrivals .. = 350.2710511
avg wait ..... = 117.8704240          avg wait ..... = 120.1989459
avg # in node ..... = 0.2128754          avg # in node ..... = 0.3431598
avg delay ..... = 6.1046500          avg delay ..... = 7.7634652
avg # in queue ..... = 0.0110251          avg # in queue ..... = 0.0221642
avg service time.....= 111.6105430          avg service time.....= 112.3584497

for 555.000000 priority jobs the service node statistics are:  for 745.000000 priority jobs the service node statistics are:

avg interarrivals .. = 136.6806347          avg interarrivals .. = 102.0252592
avg wait ..... = 118.6703988          avg wait ..... = 122.7330242
avg # in node ..... = 0.8682313          avg # in node ..... = 1.2029670
avg delay ..... = 7.0063867          avg delay ..... = 11.2496970
avg # in queue ..... = 0.0512610          avg # in queue ..... = 0.1102638
avg service time.....= 111.6390024          avg service time.....= 111.4643141
```

(a) Valori Caso Migliorato

(b) Valori Caso Migliorato aumento 10%

Figura 32: Confronto Valori

Nella fase di validazione vogliamo verificare che il modello computazionale sia coerente con il sistema reale che abbiamo analizzato. Non avendo un dataset con dati reali a disposizione, si opera una validazione che mostri, come avverrebbe nella realtà, l'aumentare dei tempi di attesa e di risposta in concomitanza dell'aumentare dei flussi di jobs in ingresso. In particolare si mostra come l'aumento del 10% di entrambi i flussi di jobs in entrata cambi i valori dei tempi di una giornata lavorativa. Come si evince dalle immagini l'incremento del tempo di attesa sia per i jobs delivery che per i jobs applicazione che godono della priorità è evidente, maggiormente per i secondi come ci si aspetterebbe dal tipo di priorità scelto.

## 12 Analisi dei risultati

Si procede ora con l'analisi dei risultati degli esperimenti svolti in questo nuovo algoritmo migliorativo, per verificare le condizioni ottimali e soddisfare le QoS prefissate.

### 12.1 Analisi Collo di Bottiglia

Nell'analisi del collo di bottiglia del modello migliorato si usano le stesse tecniche e le stesse formule del caso base, occorre però tenero conto del flusso  $\lambda_3$  in ingresso.

$$\begin{cases} V_1 = \frac{\lambda_A}{\gamma} \\ V_2 = \frac{\lambda_B}{\gamma} \\ V_3 = \frac{\lambda_C}{\gamma} \\ V_4 = \frac{\lambda_D}{\gamma} \\ \gamma = \lambda_1 + \lambda_2 + \lambda_3 \end{cases} \quad (10)$$

Per la scelta di Ploss si è deciso di considerarla trascurabile sia per il caso Low che Mid poichè la coda nel nodo della cassa ha sempre un valore molto basso, mentre le è stata assegnata il caso peggiore (3% nel caso High in quanto fascia con elevata affluenza) , poichè l'analisi del collo di bottiglia ha lo scopo di evidenziare il caso peggiore.

<b>Visite/ Fascia Oraria</b>	Low	Mid	High
<b>Cassa</b>	0.856211	0.766733	0.628942
<b>Cucina</b>	1	1	0.973045
<b>Tavoli</b>	0.951111	0.958416	0.874259
<b>Delivery</b>	0.048819	0.041521	0.098789

Si procede con il calcolo delle domande medie per verificare quale nodo sia il collo di bottiglia, ovvero quello con la domanda maggiore.

$$D^i = V^i S^i$$

<b>Domande/ Fascia Oraria</b>	Low	Mid	High
<b>Cassa</b>	38.529495	34.502985	28.302390
<b>Cucina</b>	112	112	108.9810758
<b>Tavoli</b>	570.666666	575.049876	524.555298
<b>Delivery</b>	58.58244	49.8254364	118.5466685

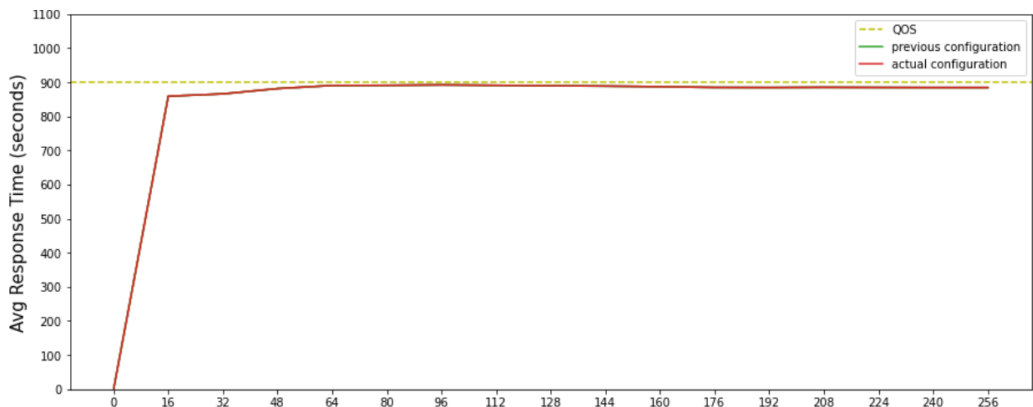
Come si evince dalla tabella, per ogni fascia oraria la domanda massima è quella relativa al nodo della consumazione al tavolo, pertanto esso costituisce il collo di bottiglia del sistema.

Come da aspettarsi la situazione è identica al caso base in quanto non sono state apportate modifiche al sottosistema dei tavoli.

L'obiettivo era quello di ridurre l'affluenza al nodo "Acquisizione Ordine" e come si vede dalla domanda è stato raggiunto.

### 12.1.1 Analisi ad Orizzonte Infinito: Caso Low

Adottando la stessa configurazione del caso base per la fascia Low la QoS è stata ovviamente rispettata e inoltre seppur modesto è stata riscontrata una riduzione del tempo di risposta medio sia del sistema che del sottosistema "Acquisizione Ordinazione" dovuto appunto all'aggiunta dell'applicazione.



**Figura 33. Configurazione1,1,4,1**

the actual node statistics are:

node	mean response time	std response time	w response time	utilizzazione
1	49.4818908	7.0343366	1.8324968	0.1649969
2	162.0188238	12.7286615	3.3159106	0.4805371
3	644.1730981	25.3805654	6.6118253	0.6127236
4	1372.8267324	37.0516765	9.6522363	0.2456566

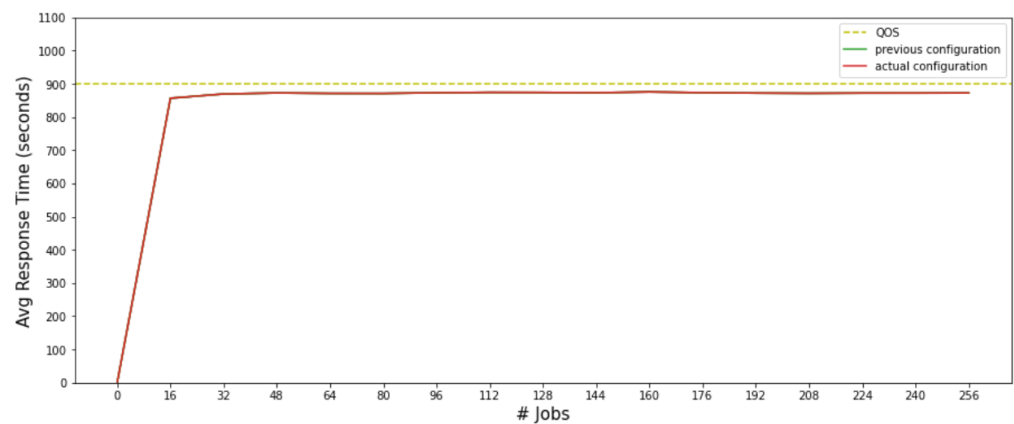
the system statistics are:

mean response time	std response time	w response time	income
884.1209381	29.7341712	7.7459719	1672.9491084 \$

**Figura 34. Configurazione1,1,4,1 Statistics**

### 12.1.2 Analisi ad Orizzonte Infinito: Caso Mid

Nel caso mid è stato possibile, grazie all'algoritmo migliorativo togliere uno dei due centri del primo sottosistema che erano invece presenti nel caso non migliorato. Un lieve aumento del tempo di risposta del primo sottosistema è ampiamente ripagato dalla riduzione del tempo medio del sistema e di quella delle spese per i centri attivi.



**Figura 35. Configurazione1,3,11,2**

the actual node statistics are:

node	mean response time	std response time	w response time	utilization
1	73.1769754	8.5543542	2.2284726	0.5565361
2	120.7318601	10.9878051	2.8624046	0.6029045
3	669.0661733	25.8663135	6.7383663	0.8448142
4	1298.2297808	36.0309559	9.3863310	0.3885789

the system statistics are:

mean response time	std response time	w response time	income
872.0489589	29.5304751	7.6929076	2071.7318922 \$

**Figura 36. Configurazione1,3,11,2 Statistics**

12.1.3    Analisi ad Orizzonte Infinito: Caso High

Anche in questa situazione la stessa configurazione del caso non migliorato è stata utilizzata e si è riscontrato un buon miglioramento del tempo di risposta totale e in particolare anche quello del sottosistema "Acquisizione Ordinazione".

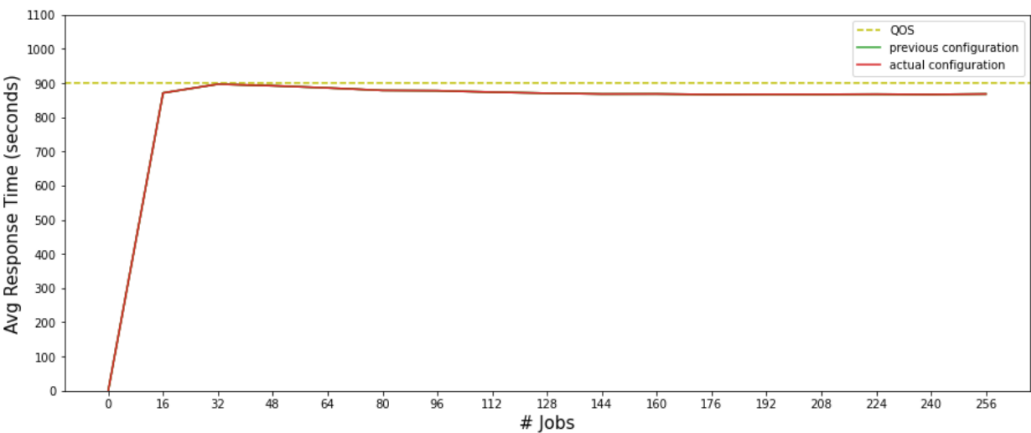


Figura 37. Configurazione4,10,48,13

the actual node statistics are:

node	mean response time	std response time	w response time	utilizzazione
1	48.6637335	6.9759396	1.8172840	0.5768962
2	148.3431039	12.1796184	3.1728808	0.9121483
3	625.9516789	25.0190263	6.5176417	0.9361791
4	1245.4924917	35.2915357	9.1937065	0.7412369

the system statistics are:

mean response time	std response time	w response time	income
867.8004529	29.4584530	7.6741453	2271.9891469 \$

Figura 38. Configurazione4,10,48,13 Statistics

## 12.2 Analisi ad Orizzonte Finito

L'analisi ad Orizzonte finito del caso migliorato evidenzia come tutti i nuovi obiettivi siano stati soddisfatti. Il tempo di risposta medio infatti è stato mantenuto sotto la soglia di 900 secondi nell'arco di tutta la giornata. La QoS sulla percentuale di utenti perduti è stata abbondantemente rispettata. Essa infatti presenta in media un valore di 0.4232%. Inoltre il guadagno medio giornaliero, grazie al sovrapprezzo sui servizi offerti, è stato incrementato, passando da un valore di 1252.75 euro a quello di 15028.22 euro.

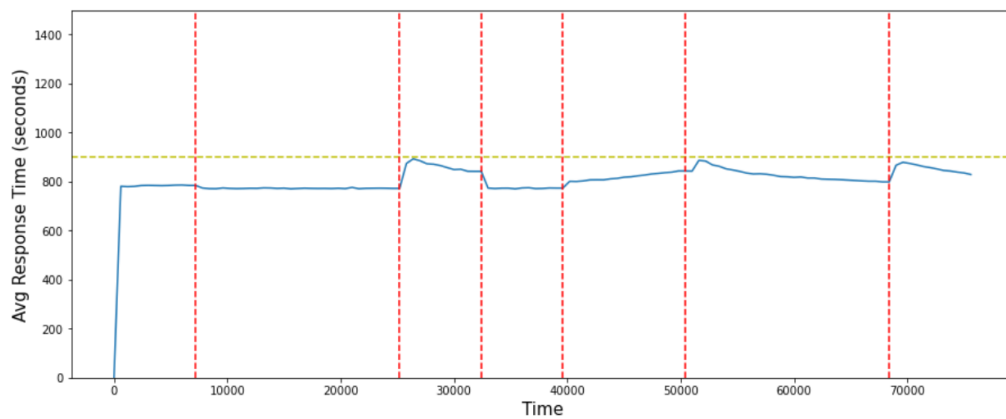


Figura 39. Scenario Ottimo - 2,3,5,2, 2,6,22,8, 5,11,50,10



the average taking of a day at McDonald in Via Tuscolana is 15028.226409894802

the average loss percentage of a day at McDonald in Via Tuscolana is 0.4232758047164095 %

the average utilization of the servers present in the nodes during the time slot 08:00 - 10:00 is:

cash register	kitchen	table	delivery
0.0825263	0.1587013	0.4870532	0.1204271

the average utilization of the servers present in the nodes during the time slot 10:00 - 15:00 is:

cash register	kitchen	table	delivery
0.2701220	0.2805944	0.4009451	0.0937620

the average utilization of the servers present in the nodes during the time slot 15:00 - 17:00 is:

cash register	kitchen	table	delivery
0.0878822	0.1808721	0.6388023	0.2535163

the average utilization of the servers present in the nodes during the time slot 17:00 - 19:00 is:

cash register	kitchen	table	delivery
0.2678676	0.2626313	0.3963048	0.0717006

the average utilization of the servers present in the nodes during the time slot 19:00 - 22:00 is:

cash register	kitchen	table	delivery
0.4595279	0.7653592	0.8461130	0.7956549

the average utilization of the servers present in the nodes during the time slot 22:00 - 03:00 is:

cash register	kitchen	table	delivery
0.2740122	0.3084054	0.4867733	0.2477447

the average utilization of the servers present in the nodes during the time slot 03:00 - 05:00 is:

cash register	kitchen	table	delivery
0.0900585	0.1857257	0.6359944	0.2862797

**Figura 40. Scenario Ottimo - 2,3,5,2, 2,6,22,8, 5,11,50,10**

Infine di seguito è riportata la tabella che evidenzia il guadagno sui tempi di risposta per i job di tipo delivery e per quelli di tipo applicazione, confermando così che il sovrapprezzo sia una spesa giustificata.

Figura 41: **Tabella Priorità**

	LOW	MID	HIGH
DELIVERY	27.121534764220932	5.590437743815321	6.94933342309698
APPLICATION	33.05283131066122	7.2250752534099405	9.43074563792791
NO-PRIORITY	49.971293418910165	8.66490836514196	36.26469654006546

## 12.3 Conclusioni

Durante lo svolgimento di questa simulazione si è passati dal caso stazionario al caso transiente osservando che il numero di centri richiesto per mantenere le QoS è via via crescente in quanto la stazionarietà non tiene conto dei jobs rimasti in coda nelle fasce precedenti. Le configurazioni trovate trovate nel transiente, atto a simulare per 64 volte l'arco di una giornata lavorativa, sono le configurazioni ottime per far sì che il tempo medio di attesa di un utente non superi il minuto permettendo comunque al fast food un guadagno molto simile a quello riscontrato sul sito ufficiale. Nel caso avanzato si è tentato di rendere più bilanciato il carico sulle casse e di ridurre i tempi di risposta del delivery (che hanno già di per se una lunga attesa per la consegna dell'ordinazione). Non vi sono state significative migliorie rispetto al caso base se non per questi obiettivi sovraccitati.

## 12.4 Bibliografia

[1]:McDonal via Tuscolana