

RELAZIONE PROGETTO PMCSN ANNO 2019/2020

---

# **Covid19 - Simulazione Gelateria**

---

Autori:

**Alessio Mazzola, Matricola: 0279323**

**Leonardo Petraglia Beniamino, Matricola: 0286559**

# Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
<b>2</b>	<b>Assunzioni Generali</b>	<b>5</b>
<b>3</b>	<b>Organizzazione della Simulazione</b>	<b>6</b>
<b>4</b>	<b>Batch-Means ed Intervalli di Confidenza</b>	<b>8</b>
4.1	Batch-Means . . . . .	8
4.2	Intervalli di Confidenza . . . . .	9
<b>5</b>	<b>Riproducibilità ed Indipendenza</b>	<b>10</b>
5.1	Riproducibilità . . . . .	10
5.2	Indipendenza . . . . .	10
<b>6</b>	<b>Service Server - Single Server</b>	<b>12</b>
6.1	Modello Concettuale . . . . .	12
6.2	Specification Model . . . . .	14
6.3	Risultati Analitici - Verifica . . . . .	15
6.4	Risultati Analitici - Sistema: . . . . .	15
6.5	Risultati Analitici - Job di Tipo 1: . . . . .	16
6.6	Risultati Analitici - Job di Tipo 2: . . . . .	17
6.7	Risultati Analitici - Job di Tipo 3: . . . . .	18
6.8	Confronto dei Risultati Analitici - Validazione . . . . .	20
6.9	Confronto dei Risultati del Sistema . . . . .	20
6.10	Confronto dei Risultati del Server Verifica: . . . . .	21
6.11	Confronto dei Risultati del Server Delay: . . . . .	21
6.12	Confronto dei Risultati del Server Service: . . . . .	22
6.13	Analisi Stazionaria . . . . .	23
6.14	Analisi Transiente . . . . .	25
<b>7</b>	<b>Service Server - Multi-Server</b>	<b>27</b>
7.1	Modello Concettuale . . . . .	27
7.2	Specification Model . . . . .	29
7.3	Risultati Analitici - Verifica . . . . .	30

7.4	Risultati Analitici - Sistema: . . . . .	30
7.5	Risultati Analitici - Job di Tipo 1: . . . . .	31
7.6	Risultati Analitici - Job di Tipo 2: . . . . .	32
7.7	Risultati Analitici - Job di Tipo 3: . . . . .	33
7.8	Confronto dei Risultati Analitici - Validazione . . . . .	35
7.9	Confronto dei Risultati del Sistema . . . . .	35
7.10	Confronto dei Risultati del Server Verifica: . . . . .	36
7.11	Confronto dei Risultati del Server Delay: . . . . .	36
7.12	Confronto dei Risultati del Server Service: . . . . .	37
7.13	Analisi Stazionaria . . . . .	38
7.14	Analisi Transiente . . . . .	40
<b>8</b>	<b>Confronto dei Sistemi</b>	<b>42</b>
<b>9</b>	<b>Conclusioni</b>	<b>44</b>

# Capitolo 1

## Introduzione

Il sistema in esame si propone di fornire supporto ad una Gelateria, in particolare questa viene rappresentata attraverso una rete formata dai seguenti server:

- Cassa+Verifica: tramite questo nodo i clienti (Job) entrano nel Sistema. E' necessario per pagare il gelato che viene scelto e per verificare se ci sono abbastanza palline di gelato rimanenti nel Service Server per completare il Job;
- Delay Server: questo nodo ha il compito di far aspettare i clienti fino a che non viene ripristinato il valore delle palline di gelato nel Service Server. Si ha inoltre una probabilità  $p$  che il cliente, una volta che ha aspettato, sia effettivamente servito, e quindi transiti nel Service Server, ed una probabilità  $(1-p)$  di uscire dal sistema senza aver consumato;
- Service Server: questo nodo è necessario all'effettivo soddisfacimento della domanda del cliente, fornendo dunque il gelato desiderato e permettendo al cliente di uscire dal sistema.

I clienti (Job) transiteranno nei server sopra citati per ricevere il servizio offerto dalla gelateria. I clienti che arrivano nel sistema possono essere di tre tipologie differenti:

- **Gelato con un gusto:**  $\lambda_1 = 3 \frac{task}{min}$  Arrivo Poissoniano  
Verifica  $\mu_1 = 13 \frac{task}{min}$  Distribuzione Esponenziale  
Delay  $\mu_1 = 2 \frac{task}{min}$  Distribuzione Esponenziale  
Service  $\mu_1 = 13 \frac{task}{min}$  Distribuzione Esponenziale

- **Gelato con due gusti:**  $\lambda_2 = 4 \frac{task}{min}$  Arrivo Poissoniano  
Verifica  $\mu_2 = 13 \frac{task}{min}$  Distribuzione Esponenziale  
Delay  $\mu_2 = 2 \frac{task}{min}$  Distribuzione Esponenziale  
Service  $\mu_2 = 13 \frac{task}{min}$  Distribuzione Esponenziale
- **Gelato con tre gusti:**  $\lambda_3 = 5 \frac{task}{min}$  Arrivo Poissoniano  
Verifica  $\mu_3 = 13 \frac{task}{min}$  Distribuzione Esponenziale  
Delay  $\mu_3 = 2 \frac{task}{min}$  Distribuzione Esponenziale  
Service  $\mu_3 = 13 \frac{task}{min}$  Distribuzione Esponenziale

Per simulare il sistema sono state sviluppate due soluzioni differenti. La prima soluzione è quella di andare a considerare il nodo Service Server come un Server Singolo, quindi un nodo M/M/1, che rappresenta un singolo commesso nella gelateria.

La seconda soluzione è quella di andare a considerare il nodo Service Server come un Multi-Server, quindi un nodo M/M/m, formato da m=2 Server distinti con la stessa capacità, quindi un aumento di personale della gelateria per servire i clienti.

Con riferimento all'algoritmo 1.1.1<sup>1</sup> del libro di testo andiamo di seguito a specificare quali sono i *Goal and Objectives* del Sistema in considerazione. In particolare si è voluto utilizzare come metrica il *Tempo di risposta* ed il *Throughput* del Sistema andando successivamente a migliorarlo passando da Server Singolo a Multi-Server, quindi andando ad aumentare il numero di commessi presenti nella Gelateria.

Per quanto riguarda la generazione di numeri casuali, questi sono stati ricavati tramite il **Generatore di Lehmer**. In particolare queste librerie sono state ricavate direttamente dal libro di testo, riferendoci dunque alla libreria **rngs.c**. Il generatore è stato sfruttato in particolare per l'interarrivo dei Job e per la probabilità p di essere rediretto verso il nodo Service se un Job è presente all'interno del nodo Delay.

---

<sup>1</sup> Algoritmo presente a pagina 4 del Libro

## Capitolo 2

# Assunzioni Generali

Gli arrivi delle classi sono stati assunti come casuali, modellati con distribuzione Poissoniana di parametri  $\lambda_1$ ,  $\lambda_2$  e  $\lambda_3$  rispetto al tipo di Job.

Il Server Verifica, inoltre, ha anche il compito di Controller in quanto va a verificare se all'arrivo di un Job la sua richiesta di Gelato può essere soddisfatta basandosi sul livello corrente di gelato disponibile. In particolare, se il livello di Gelato disponibile non dovesse consentire il soddisfacimento della domanda del Job, allora questo viene reindirizzato verso il Server Delay per rimanere in attesa, altrimenti transita verso il Service Server per essere servito ed uscire dal Sistema.

La presenza di questo Controller all'interno del Sistema non garantisce che gli inter-arrivi in Delay Server e Service Server siano esponenziali ma per semplicità sono stati assunti come tali.

Le code dei Server del Sistema, per semplicità, vengono assunte come infinite con una politica FIFO.

Gli arrivi sono stati generati tramite un generatore di numeri pseudo-casuali ricavati da una distribuzione esponenziale. Per tale generatore è stata utilizzata l'implementazione fornita dal libro di testo [1].

Per quanto riguarda il calcolo dei risultati analitici del Sistema, anche i tassi di servizio dei Server che compongono il Sistema stesso sono stati assunti come esponenziali.

## Capitolo 3

# Organizzazione della Simulazione

L'esecuzione della simulazione è stata organizzata e strutturata nelle seguenti fasi:

- **Fase di Inizializzazione:** in questa fase vengono definite le Macro e vengono inizializzate le variabili di stato necessarie alla simulazione stessa. Viene inoltre generato il primo evento del sistema. Terminata la fase si entrerà all'interno di un Ciclo While dove verranno eseguite le fasi di Gestione dell'evento Corrente e Generazione di un nuovo evento.
- **Gestione dell'evento Corrente:** In questa fase viene determinato qual è effettivamente l'evento corrente che dunque dovrà essere gestito. Come detto in precedenza se l'evento corrente è un arrivo all'interno del server Verifica, allora il controller andrà a determinare se il Job dovrà transitare verso il Server Delay o verso il Service Server.
- **Generazione di un nuovo evento:** In questa fase viene generato un nuovo evento in arrivo nel Sistema che dovrà essere poi successivamente gestito.
- **Fase di Terminazione:** Rappresenta l'ultima fase della simulazione e questa viene raggiunta una volta che le condizioni di fine simulazione vengono verificate contemporaneamente. In particolare queste condizioni sono le seguenti:
  - Il tempo corrente di simulazione ha ecceduto il tempo massimo di simulazione;
  - Non vi sono più Job all'interno del sistema

Di seguito andiamo a riportare il diagramma di flusso che va a rappresentare la struttura dei programmi simulativi realizzati per il progetto:

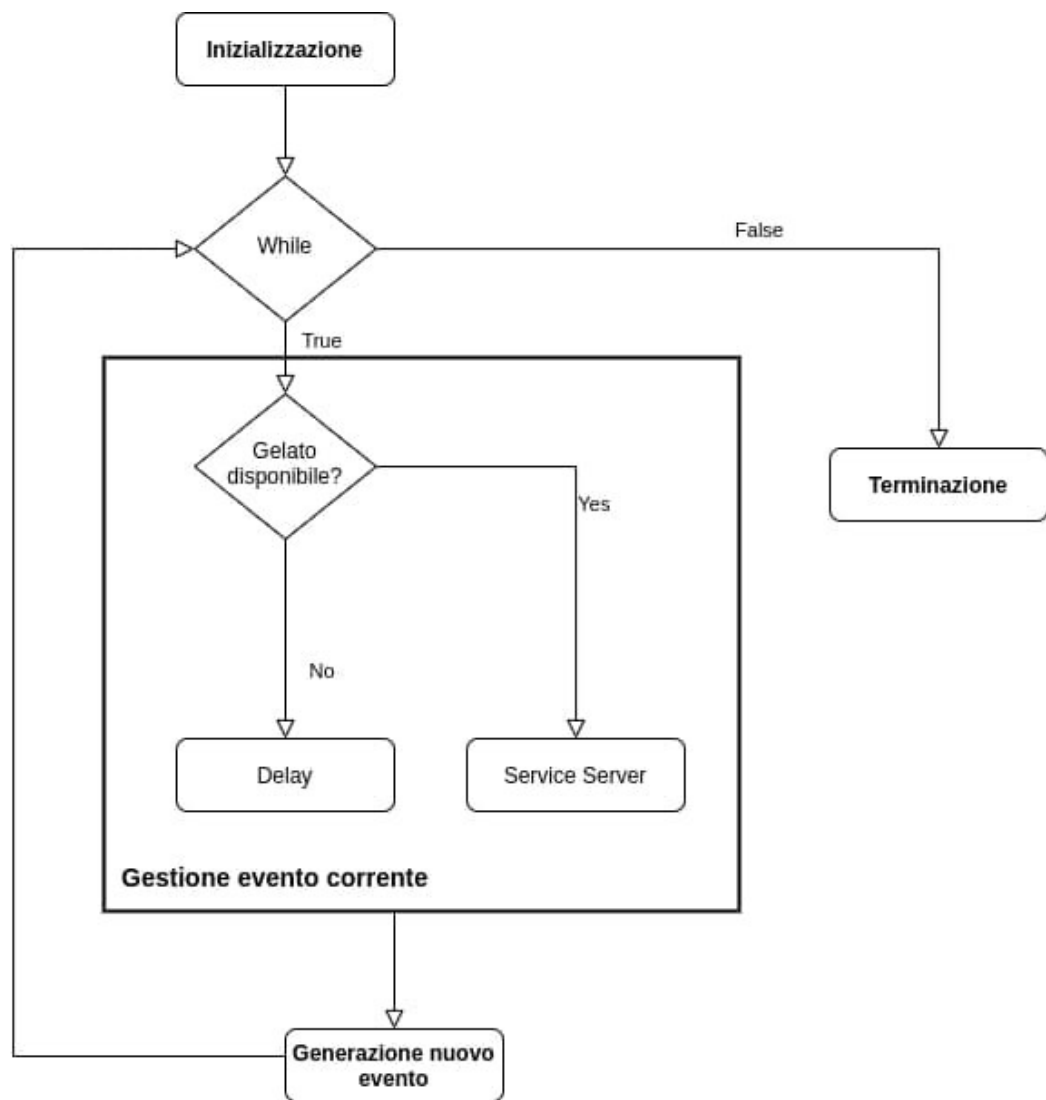


Figura 3.1: Diagramma di Flusso della Simulazione



## Capitolo 4

# Batch-Means ed Intervalli di Confidenza

### 4.1 Batch-Means

Generalmente quando bisogna trattare la simulazione a Stato Stazionario (Steady-State) bisogna andare a risolvere due problemi principali, cioè:

- **Individuazione del Periodo di Warm-UP:** Il periodo di Warm-UP identifica un intervallo di tempo costituito da un insieme di osservazioni che però non risultano essere significative in quanto queste sono legate alle condizioni iniziali del Sistema, questo significa effettivamente che il sistema stesso non ha ancora raggiunto il suo Stato Stazionario.
- **Determinazione della Lunghezza della Simulazione.**

Per le ragioni sopra citate le simulazioni sono state eseguite andando ad utilizzare la tecnica dei Batch-Means. In particolare questa tecnica permette di andare ad eliminare l'influenza dello stato iniziale del sistema andando a considerare un periodo di simulazione sufficientemente lungo ed unico che verrà successivamente partizionato in intervalli di egual misura aventi il nome di Batch.

L'eliminazione dell'influenza iniziale dello stato del Sistema, tramite le Batch-Means, si ha in quanto le statistiche calcolate per ogni Batch sono inizializzate allo stato del sistema nel momento in cui il periodo del Batch termina e ne comincia uno nuovo. Questo approccio fa sì che ogni Batch sia indipendente dagli altri.

Nella simulazione sono stati utilizzati 64 Batch con lunghezza individuata dal rapporto tra l'istante di tempo massimo entro cui possono arrivare i Job

(Valore di STOP) ed il numero di Batch stabilito. In questo modo, al presentarsi di un nuovo evento, le statistiche vengono calcolate e memorizzate solo nel caso in cui l'intervallo di tempo trascorso dall'ultima simulazione è maggiore del rapporto appena introdotto.

Una volta che il periodo di simulazione è terminato, vengono calcolate le seguenti stastiche:

- **Media Campionaria dei Batch i-esimi**
- **Media Campionaria tra le medie di tutti i Batch i-esimi**
- **Varianza Campionaria della media dei Batch i-esimi**

## 4.2 Intervalli di Confidenza

Gli intervalli di confidenza sono stati calcolati con la seguente formula:

$$\left[ \bar{x} - \frac{t \cdot s}{\sqrt{k-1}}, \bar{x} + \frac{t \cdot s}{\sqrt{k-1}} \right] \quad (4.1)$$

Dove:

- $\bar{x}$  rappresenta la media dei batch
- $t$  rappresenta il valore critico
- $s$  rappresenta la varianza campionaria delle medie dei batch
- $k$  rappresenta il numero di batch

Per ottenere un livello di confidenza del 95%,  $t$  è stato calcolato utilizzando la funzione **idfStudent**( $n - 1, 1 - \frac{\alpha}{2}$ ) con  $\alpha = 0.05$ .

## Capitolo 5

# Riproducibilità ed Indipendenza

### 5.1 Riproducibilità

I tempi di Arrivo e di Servizio sono stati simulati attraverso la libreria *rngs.c*, la quale fornisce un insieme di funzioni utili alla gestione e alla generazione di numeri pseudo-casuali andando a sfruttare il Generatore di Lehmer, come detto anche nell'introduzione.

Essendo numeri pseudo-casuali, essi saranno sempre gli stessi ad ogni run della simulazione e questo ci garantisce la riproducibilità dei dati che verranno riportati a patto di utilizzare il medesimo seme per ogni Run.

### 5.2 Indipendenza

Per quanto riguarda l'Indipendenza di ogni processo di generazione di numeri pseudo-casuali è stato utilizzato il meccanismo degli **Stream**.

Tramite l'ausilio di questa tecnica, infatti, semi diversi appartengono a Stream diverse, in modo tale che non ci sia una sovrapposizione dei Semi. Per questo motivo, nelle simulazioni realizzate, ciascun processo di generazione di numeri pseudo-casuali possiede il proprio Stream. In particolare questi vengono riportati di seguito:

- STREAM\_ARR1
- STREAM\_ARR2
- STREAM\_ARR3
- STREAM\_SERV\_VERIFY1

- STREAM\_SERV\_VERIFY2
- STREAM\_SERV\_VERIFY3
- STREAM\_SERV\_DELAY1
- STREAM\_SERV\_DELAY2
- STREAM\_SERV\_DELAY3
- STREAM\_SERV\_MULTISERVER1
- STREAM\_SERV\_MULTISERVER2
- STREAM\_SERV\_MULTISERVER3
- STREAM\_SETUP

## Capitolo 6

# Service Server - Single Server

Con riferimento all'algoritmo 1.1.1 del Libro di testo andiamo adesso ad elencare i punti dell'algoritmo, andando a giustificare le scelte che sono state fatte.

### 6.1 Modello Concettuale

Il sistema modellato è composto dai seguenti sottosistemi:

- **Cassa+Verifica:** Questo nodo della rete è costituito da un Server Singolo e permette l'entrata dei Job all'interno del Sistema indipendentemente dalla loro tipologia (Tipo1, Tipo2 o Tipo3). Viene inoltre eseguito il Controller presente che va a verificare se la richiesta del Job può essere soddisfatta. In particolare viene controllato se la tipologia di gelato richiesta dal Job in arrivo possa essere soddisfatta data la fornitura di gelato corrente presente nel Service Server. In caso affermativo, il Job verrà indirizzato verso il Service Server, altrimenti verrà indirizzato verso il Delay Server;
- **Delay Server:** Questo nodo della rete è costituito da un Server Singolo, esso fa in modo che i Job aspettino che la fornitura di gelato del Service Server venga ripristinata. All'interno del nodo si ha anche una probabilità  $p$  di rimanere in attesa per essere serviti ed una probabilità  $(1-p)$  di uscire dal sistema senza essere serviti. In questo ultimo caso il Job viene considerato come perso;
- **Service Server:** Questo nodo della rete è costituito da un Server Singolo e fa in modo che i Job vengano serviti e possano uscire dal Sistema.

Dati i sottosistemi sopracitati abbiamo individuato le seguenti variabili di stato:

- Numero Corrente e Totale di Job per ogni sottosistema:
  - actual\_system                      – total\_system
  - actual\_verify                      – total\_verify
  - actual\_delay                      – total\_delay
  - actual\_service                      – total\_service
- Numero Corrente e Totale di Job di tipo 1 per ogni sottosistema:
  - actual\_job1                      – total\_job1
  - actual\_job1\_verify                      – total\_job1\_verify
  - actual\_job1\_delay                      – total\_job1\_delay
  - actual\_job1\_service                      – total\_job1\_service
- Numero Corrente e Totale di Job di tipo 2 per ogni sottosistema:
  - actual\_job2                      – total\_job2
  - actual\_job2\_verify                      – total\_job2\_verify
  - actual\_job2\_delay                      – total\_job2\_delay
  - actual\_job2\_service                      – total\_job2\_service
- Numero Corrente e Totale di Job di tipo 3 per ogni sottosistema:
  - actual\_job3                      – total\_job3
  - actual\_job3\_verify                      – total\_job3\_verify
  - actual\_job3\_delay                      – total\_job3\_delay
  - actual\_job3\_service                      – total\_job3\_service

Le variabili *actual* vanno a contare i Job presenti nel sistema ad ogni istante di tempo, quindi subiranno incrementi e decrementi a seconda dell'evento corrente che viene processato dal Sistema. In particolare, gli eventi che possono avvenire all'interno del sistema sono i seguenti:

- Arrivo di un Job;
- Transito del Job da Verifica+Cassa a Service Server;
- Transito del Job da Verifica+Cassa a Delay Server;
- Transito del Job da Delay Server a Service Server;

- Uscita dal sistema del Job proveniente da Delay Server  $\rightarrow$  Job perso;
- Uscita dal sistema del Job proveniente da Service Server  $\rightarrow$  Job completato.

Le variabili *total*, invece, tengono conto di tutti i Job che sono transitati all'interno del Sistema e quindi andranno a subire solamente degli incrementi.

Sia durante l'esecuzione della Simulazione sia al termine della stessa, vengono effettuati dei controlli sulla consistenza delle variabili suddette.

## 6.2 Specification Model

Di seguito andiamo a presentare il Modello di Specifica che caratterizza il Sistema come specificato nel punto (3) dell'algoritmo 1.1.1.

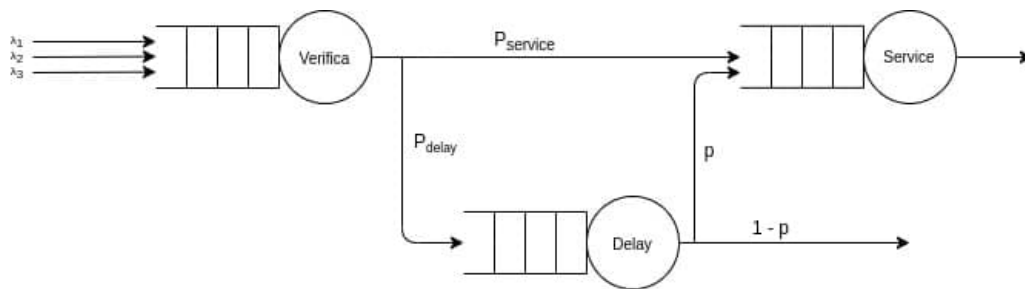


Figura 6.1: Modello della Rete del Sistema - Service Server M/M/1

Come possiamo notare, dal Server Cassa+Verifica entrano i Job e questi usciranno o dal Delay Server oppure dal Service Server.

Per andare a validare il Sistema abbiamo eseguito i calcoli sfruttando la rete di Jackson. Per quanto riguarda i valori delle probabilità si ha:

- $p = 0.8$
- $(1 - p) = 0.2$
- $p_{delay} = 0.063996$
- $p_{service} = 0.948820$

I valori di  $p_{delay}$  e  $p_{service}$  sono stati ottenuti andando ad effettuare il calcolo delle probabilità di terminazione della Vaschetta di gelato per ogni tipologia di Job in arrivo nel Sistema.

### 6.3 Risultati Analitici - Verifica

Andiamo adesso a presentare i risultati analitici che sono stati ricavati per il Sistema preso in considerazione. In particolare, per ottenere questi risultati è stata calcolata la rete di Jackson ottenuta dal modello di Specifica del Sistema. Questi risultati vengono sfruttati per andare a verificare il Sistema come specificato nel punto (4) dell'algoritmo 1.1.1.

### 6.4 Risultati Analitici - Sistema:

$$\begin{cases} \lambda_{verifica} = \lambda_1 + \lambda_2 + \lambda_3 \\ \lambda_{delay} = p_{delay}\lambda_{verifica} \\ \lambda_{service} = (1 - p_{delay})\lambda_{verifica} + p\lambda_{delay} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica} = 12[Job/min] \\ \lambda_{delay} = 0.767952[Job/min] \\ \lambda_{service} = 11.846410[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica} = \frac{\lambda_{verifica}}{\mu_{verifica}} = 0.923077 \\ \rho_{delay} = \frac{\lambda_{delay}}{\mu_{delay}} = 0.383976 \\ \rho_{service} = \frac{\lambda_{service}}{\mu_{service}} = 0.911262 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica} = \frac{\rho_{verifica}}{1 - \rho_{verifica}} = 12[Job] \\ E[N]_{delay} = \frac{\rho_{delay}}{1 - \rho_{delay}} = 0.623313[Job] \\ E[N]_{service} = \frac{\rho_{service}}{1 - \rho_{service}} = 10.269165[Job] \\ E[N] = E[N]_{verifica} + E[N]_{delay} + E[N]_{service} = 22.892478[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica} = \frac{\lambda_{verifica}}{\lambda_{tot}} = 1 \\ v_{delay} = \frac{\lambda_{delay}}{\lambda_{tot}} = 0.063996 \\ v_{service} = \frac{\lambda_{service}}{\lambda_{tot}} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica} = \frac{1}{\mu_{verifica} - \lambda_{verifica}} = 1[min] \\ E[Ts]_{delay} = \frac{1}{\mu_{delay} - \lambda_{delay}} = 0.811657[min] \\ E[Ts]_{service} = \frac{1}{\mu_{service} - \lambda_{service}} = 0.866859[min] \\ E[Ts] = v_{verifica}E[Ts]_{verifica} + v_{delay}E[Ts]_{delay} + v_{service}E[Ts]_{service} = 1.907706[min] \end{cases}$$



- **Throughput:**

$$\begin{cases} X_{verifica} = \frac{E[N]_{verifica}}{E[Ts]_{verifica}} = \rho_{verifica} \mu_{verifica} = 12 [Job/min] \\ X_{delay} = \frac{E[N]_{delay}}{E[Ts]_{delay}} = \rho_{delay} \mu_{delay} = 0.767952 [Job/min] \\ X_{service} = \frac{E[N]_{service}}{E[Ts]_{service}} = \rho_{service} \mu_{service} = 11.846410 [Job/min] \end{cases}$$

## 6.5 Risultati Analitici - Job di Tipo 1:

$$\begin{cases} \lambda_{verifica1} = \lambda_1 \\ \lambda_{delay1} = p_{delay} \lambda_{verifica1} \\ \lambda_{service1} = (1 - p_{delay}) \lambda_{verifica1} + p \lambda_{delay1} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica1} = 3 [Job/min] \\ \lambda_{delay1} = 0.191148 [Job/min] \\ \lambda_{service1} = 2.9617704 [Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica1} = \frac{\lambda_{verifica1}}{\mu_{verifica1}} = 0.230769 \\ \rho_{delay1} = \frac{\lambda_{delay1}}{\mu_{delay1}} = 0.095994 \\ \rho_{service1} = \frac{\lambda_{service1}}{\mu_{service1}} = 0.227816 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica1} = \frac{\rho_{verifica1}}{1-\rho} = 3 [Job] \\ E[N]_{delay1} = \frac{\rho_{delay1}}{1-\rho} = 0.155828 [Job] \\ E[N]_{service1} = \frac{\rho_{service1}}{1-\rho} = 2.567291 [Job] \\ E[N]^1 = E[N]_{verifica1} + E[N]_{delay1} + E[N]_{service1} = 5.723119 [Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica1} = \frac{\lambda_{verifica1}}{\lambda_1} = 1 \\ v_{delay1} = \frac{\lambda_{delay1}}{\lambda_1} = 0.063996 \\ v_{service1} = \frac{\lambda_{service1}}{\lambda_1} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica1} = \frac{1}{\mu_{verifica1} - \lambda_{verifica1}} = 1 [min] \\ E[Ts]_{delay1} = \frac{1}{\mu_{delay1} - \lambda_{delay1}} = 0.811657 [min] \\ E[Ts]_{service1} = \frac{1}{\mu_{service1} - \lambda_{service1}} = 0.866859 [min] \\ E[Ts]^1 = v_{verifica1} E[Ts]_{verifica1} + v_{delay1} E[Ts]_{delay1} + v_{service1} E[Ts]_{service1} = 1.907706 [min] \end{cases}$$

- **Throughput:**

$$\begin{cases} X_{verifica1} = \frac{E[N]_{verifica1}}{E[Ts]_{verifica1}} = \rho_{verifica1} \mu_{verifica1} = 3[Job/min] \\ X_{delay1} = \frac{E[N]_{delay1}}{E[Ts]_{delay1}} = \rho_{delay1} \mu_{delay1} = 0.191988[Job/min] \\ X_{service1} = \frac{E[N]_{service1}}{E[Ts]_{service1}} = \rho_{service1} \mu_{service1} = 2.961602[Job/min] \end{cases}$$

## 6.6 Risultati Analitici - Job di Tipo 2:

$$\begin{cases} \lambda_{verifica2} = \lambda_2 \\ \lambda_{delay2} = p_{delay} \lambda_{verifica2} \\ \lambda_{service2} = (1 - p_{delay}) \lambda_{verifica2} + p \lambda_{delay2} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica2} = 4[Job/min] \\ \lambda_{delay2} = 0.254864[Job/min] \\ \lambda_{service2} = 3.9490272[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica2} = \frac{\lambda_{verifica2}}{\mu_{verifica2}} = 0.307692 \\ \rho_{delay2} = \frac{\lambda_{delay2}}{\mu_{delay2}} = 0.127992 \\ \rho_{service2} = \frac{\lambda_{service2}}{\mu_{service2}} = 0.303754 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica2} = \frac{\rho_{verifica2}}{1-\rho} = 4[Job] \\ E[N]_{delay2} = \frac{\rho_{delay2}}{1-\rho} = 0.207771[Job] \\ E[N]_{service2} = \frac{\rho_{service2}}{1-\rho} = 3.423055[Job] \\ E[N]^2 = E[N]_{verifica2} + E[N]_{delay2} + E[N]_{service2} = 7.630826[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica2} = \frac{\lambda_{verifica2}}{\lambda_2} = 1 \\ v_{delay2} = \frac{\lambda_{delay2}}{\lambda_2} = 0.063996 \\ v_{service2} = \frac{\lambda_{service2}}{\lambda_2} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica2} = \frac{1}{\mu_{verifica2} - \lambda_{verifica2}} = 1[min] \\ E[Ts]_{delay2} = \frac{1}{\mu_{delay2} - \lambda_{delay2}} = 0.811657[min] \\ E[Ts]_{service2} = \frac{1}{\mu_{service2} - \lambda_{service2}} = 0.866859[min] \\ E[Ts]^2 = v_{verifica2} E[Ts]_{verifica2} + v_{delay2} E[Ts]_{delay2} + v_{service2} E[Ts]_{service2} = 1.907706[min] \end{cases}$$

- **Throughput:**

$$\begin{cases} X_{verifica2} = \frac{E[N]_{verifica2}}{E[Ts]_{verifica2}} = \rho_{verifica2} \mu_{verifica2} = 4[Job/min] \\ X_{delay2} = \frac{E[N]_{delay2}}{E[Ts]_{delay2}} = \rho_{delay2} \mu_{delay2} = 0.255984[Job/min] \\ X_{service2} = \frac{E[N]_{service2}}{E[Ts]_{service2}} = \rho_{service2} \mu_{service2} = 3.948803[Job/min] \end{cases}$$

## 6.7 Risultati Analitici - Job di Tipo 3:

$$\begin{cases} \lambda_{verifica3} = \lambda_3 \\ \lambda_{delay3} = p_{delay} \lambda_{verifica3} \\ \lambda_{service3} = (1 - p_{delay}) \lambda_{verifica3} + p \lambda_{delay3} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica3} = 5[Job/min] \\ \lambda_{delay3} = 0.31858[Job/min] \\ \lambda_{service3} = 4.936284[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica3} = \frac{\lambda_{verifica3}}{\mu_{verifica3}} = 0.384615 \\ \rho_{delay3} = \frac{\lambda_{delay3}}{\mu_{delay3}} = 0.159990 \\ \rho_{service3} = \frac{\lambda_{service3}}{\mu_{service3}} = 0.379693 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica3} = \frac{\rho_{verifica3}}{1-\rho} = 5[Job] \\ E[N]_{delay3} = \frac{\rho_{delay3}}{1-\rho} = 0.259714[Job] \\ E[N]_{service3} = \frac{\rho_{service3}}{1-\rho} = 4.278819[Job] \\ E[N] = E[N]_{verifica3} + E[N]_{delay3} + E[N]_{service3} = 9.538532[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica3} = \frac{\lambda_{verifica3}}{\lambda_3} = 1 \\ v_{delay3} = \frac{\lambda_{delay3}}{\lambda_3} = 0.063996 \\ v_{service3} = \frac{\lambda_{service3}}{\lambda_3} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica3} = \frac{1}{\mu_{verifica3} - \lambda_{verifica3}} = 1[min] \\ E[Ts]_{delay3} = \frac{1}{\mu_{delay3} - \lambda_{delay3}} = 0.811657[min] \\ E[Ts]_{service3} = \frac{1}{\mu_{service3} - \lambda_{service3}} = 0.866859[min] \\ E[Ts] = v_{verifica3} E[Ts]_{verifica3} + v_{delay3} E[Ts]_{delay3} + v_{service3} E[Ts]_{service3} = 1.907706[min] \end{cases}$$

- **Throughput:**

$$\left\{ \begin{array}{l} X_{verifica3} = \frac{E[N]_{verifica2}}{E[Ts]_{verifica2}} = \rho_{verifica2} \mu_{verifica2} = 5 [Job/min] \\ X_{delay3} = \frac{E[N]_{delay2}}{E[Ts]_{delay2}} = \rho_{delay2} \mu_{delay2} = 0.319980 [Job/min] \\ X_{service3} = \frac{E[N]_{service2}}{E[Ts]_{service2}} = \rho_{service2} \mu_{service2} = 4.936004 [Job/min] \end{array} \right.$$

## 6.8 Confronto dei Risultati Analitici - Validazione

E' stata verificata la correttezza dei dati ricavati dalla simulazione andandoli a confrontare con i risultati ottenuti dalla risoluzione del sistema modellato attraverso la rete di Jackson introdotta in precedenza. Questi risultati vengono sfruttati per andare ad effettuare la validazione del Sistema come specificato nel punto (6) dell'algoritmo 1.1.1.

Per poter effettuare un confronto coerente, la simulazione è stata eseguita sotto le seguenti condizioni:

- Tempi di servizio del Server Cassa+Verifica con distribuzione Esponenziale
- Tempi di servizio del Server Delay con distribuzione Esponenziale
- Tempi di servizio del Service Server con distribuzione Esponenziale
- Seme = 123456789
- Valore di STOP = 2'000'000 (Identifica l'istante di tempo massimo entro cui possono arrivare i Job nel Sistema)

In particolare, il valore di STOP è stato selezionato in base all'analisi della stazionarietà che verrà introdotta in seguito.

## 6.9 Confronto dei Risultati del Sistema

### Tempi di Risposta:

	Risultato Analitico	Risultato Simulato
$E[Ts]$	1.907706	1.882583 +/- 0.907789
$E[Ts]_1$	1.907706	1.841435 +/- 0.868513
$E[Ts]_2$	1.907706	1.845099 +/- 0.872478
$E[Ts]_3$	1.907706	1.849638 +/- 0.876535

### Throughput:

	Risultato Analitico	Risultato Simulato
X	11.846410	11.158054 +/- 0.848397
$X_1$	2.961602	2.877785 +/- 0.119141
$X_2$	3.948803	3.736229 +/- 0.573063
$X_3$	4.936004	4.543639 +/- 0.282255

**Numero medio di Job:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[N]$	22.892478	22.590997 +/- 0.721584
$E[N]_1$	5.723119	5.524305 +/- 0.816615
$E[N]_2$	7.630826	7.380396 +/- 0.959641
$E[N]_3$	9.538532	9.248190 +/- 0.913371

## 6.10 Confronto dei Risultati del Server Verifica:

**Tempi di Risposta:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[Ts]_{verifica}$	1	0.981175 +/- 0.247461
$E[Ts]_{verifica1}$	1	0.981839 +/- 0.247801
$E[Ts]_{verifica2}$	1	0.980790 +/- 0.247409
$E[Ts]_{verifica3}$	1	0.981095 +/- 0.247468

**Throughput:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$X_{verifica}$	12	11.813488 +/- 0.700090
$X_{verifica1}$	3	2.954025 +/- 0.232848
$X_{verifica2}$	4	3.938276 +/- 0.969051
$X_{verifica3}$	5	4.921187 +/- 0.196539

**Numero medio di Job:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[N]_{verifica}$	12	11.775119 +/- 0.657903
$E[N]_{verifica1}$	3	2.946410 +/- 0.231901
$E[N]_{verifica2}$	4	3.923919 +/- 0.960484
$E[N]_{verifica3}$	5	4.904790 +/- 0.187196

## 6.11 Confronto dei Risultati del Server Delay:

**Tempi di Risposta:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[Ts]_{delay}$	0.811657	0.844281 +/- 0.182412
$E[Ts]_{delay1}$	0.811657	0.847680 +/- 0.187017
$E[Ts]_{delay2}$	0.811657	0.844726 +/- 0.183714
$E[Ts]_{delay3}$	0.811657	0.843485 +/- 0.182627

### Throughput:

	Risultato Analitico	Risultato Simulato
$X_{delay}$	0.767952	0.819014 +/- 0.171596
$X_{delay1}$	0.191988	0.195279 +/- 0.002362
$X_{delay2}$	0.255984	0.251996 +/- 0.016352
$X_{delay3}$	0.319980	0.331739 +/- 0.057109

### Numero medio di Job:

	Risultato Analitico	Risultato Simulato
$E[N]_{delay}$	0.623313	0.702452 +/- 0.126238
$E[N]_{delay1}$	0.155828	0.182028 +/- 0.001727
$E[N]_{delay2}$	0.207771	0.216224 +/- 0.011961
$E[N]_{delay3}$	0.259714	0.304199 +/- 0.041795

## 6.12 Confronto dei Risultati del Server Service:

### Tempi di Risposta:

	Risultato Analitico	Risultato Simulato
$E[Ts]_{service}$	0.866859	0.892219 +/- 0.203636
$E[Ts]_{service1}$	0.866859	0.853742 +/- 0.186503
$E[Ts]_{service2}$	0.866859	0.853616 +/- 0.186463
$E[Ts]_{service3}$	0.866859	0.853429 +/- 0.186352

### Throughput:

	Risultato Analitico	Risultato Simulato
$X_{service}$	11.846410	11.158054 +/- 0.171596
$X_{service1}$	2.961602	2.877785 +/- 0.119139
$X_{service2}$	3.948803	3.736629 +/- 0.573065
$X_{service3}$	4.936004	4.543639 +/- 0.282255

### Numero medio di Job:

	Risultato Analitico	Risultato Simulato
$E[N]_{service}$	10.269165	10.113427 +/- 0.159853
$E[N]_{service1}$	2.567291	2.495867 +/- 0.593280
$E[N]_{service2}$	3.423055	3.240252 +/- 0.685742
$E[N]_{service3}$	4.278819	4.139200 +/- 0.968972

## 6.13 Analisi Stazionaria

Per effettuare l'analisi stazionaria sono state eseguite numerose simulazioni aventi in comune il medesimo seme, ma con istanti di STOP crescente. In particolare il valore di STOP va da 100'000 a 2'500'000, in modo tale da stabilire quando il Sistema inizia a perdere l'influenza dello stato iniziale.

Le simulazioni sono state effettuate ad intervalli del valore di STOP pari a 200'000 come è possibile notare dalla Tabella riportata di seguito.

I parametri della simulazione sono i seguenti:

- Seed = 123456789
- Tempi di Servizio di tutti i Server del Sistema con distribuzione Esponenziale.

STOP	Tempo Medio di Risposta	Throughput	Popolazione Media
100'000	1.834696	11.140612	22.016356
300'000	1.858349	11.150942	22.300193
500'000	1.869037	11.158280	22.428446
700'000	1.876262	11.161778	22.515143
900'000	1.879062	11.159785	22.548743
1'100'000	1.880458	11.159692	22.565500
1'300'000	1.881244	11.159551	22.574931
1'500'000	1.881633	11.158488	22.579597
1'700'000	1.881710	11.158317	22.580520
1'900'000	1.882360	11.158230	22.588320
2'100'000	1.882597	11.157488	22.591160
2'300'000	1.882932	11.158249	22.595189
2'500'000	1.883333	11.158713	22.599991



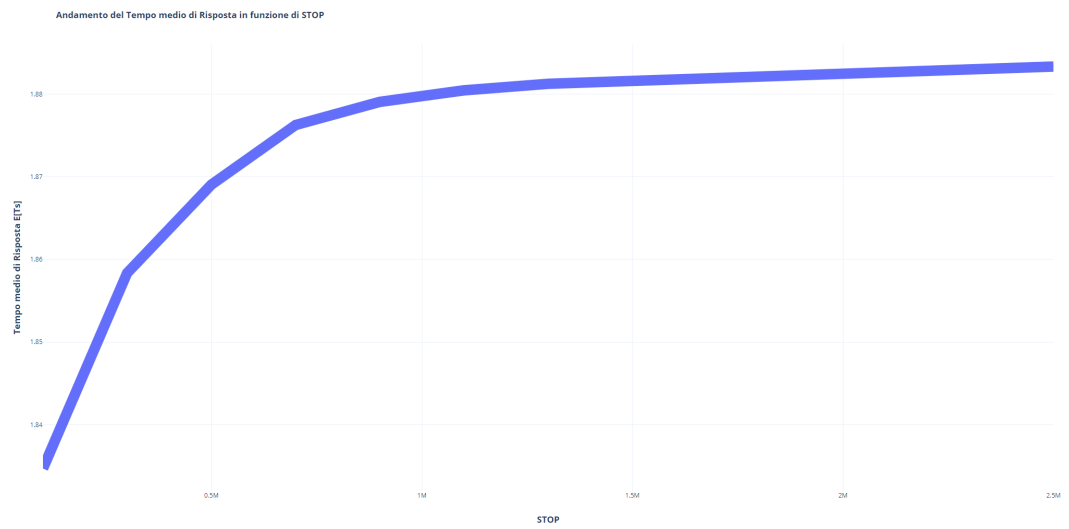


Figura 6.2: Andamento del Tempo medio di Risposta in funzione di STOP

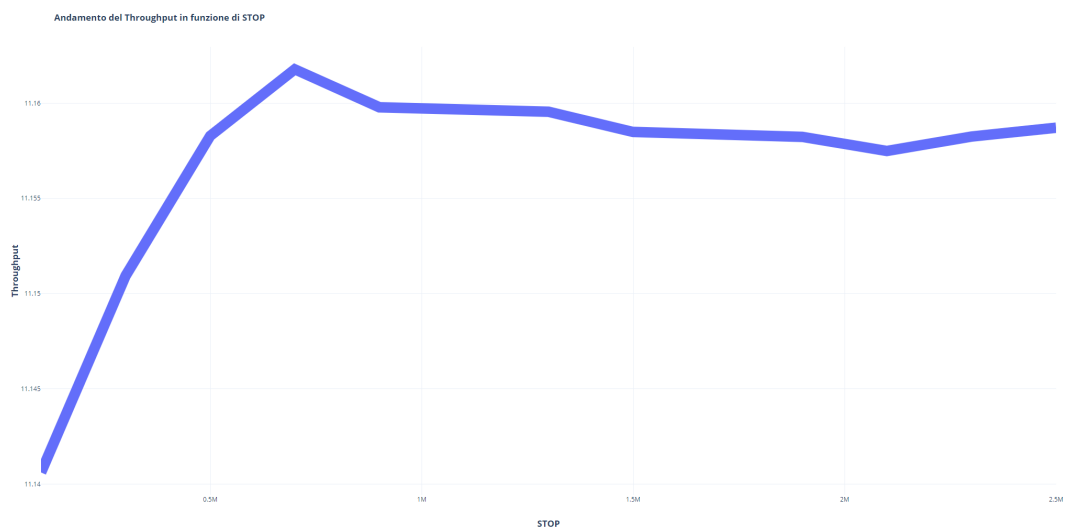


Figura 6.3: Andamento del Throughput in funzione di STOP

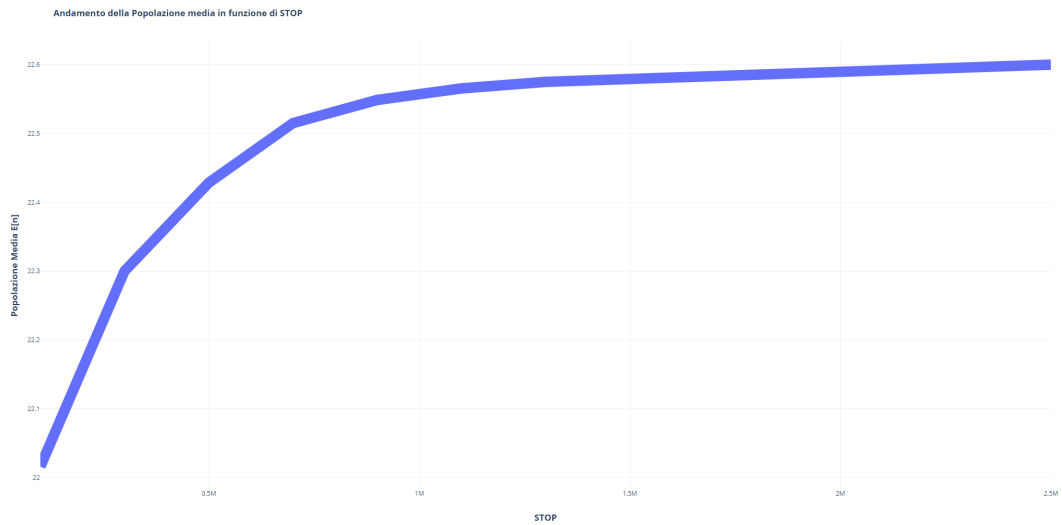


Figura 6.4: Andamento della Popolazione media in funzione di STOP

Come si può verificare dai risultati ottenuti, i valori del Tempo di Risposta medio, del Throughput e della Popolazione media, iniziano ad avvicinarsi sempre più tra loro con l'aumentare del valore di STOP.

In particolare possiamo notare che superato l'istante di STOP di 1'300'000, i valori del tempo di Risposta medio e del Throughput differiscono tra loro di un valore dell'ordine di  $10^{-2}$ . Possiamo quindi presupporre che proprio da questo momento in poi il Sistema inizi a comportarsi in maniera Stazionaria.

## 6.14 Analisi Transiente

Per effettuare l'analisi transiente sono state eseguite cinque simulazioni con semi diversi ed indipendenti tra loro.

I Semi risultano essere indipendenti poiché le simulazioni, effettuate in maniera sequenziale, utilizzano come nuovo seme l'ultimo valore restituito dal generatore di numeri pseudo-casuali durante la simulazione precedente.

Le simulazioni sono inoltre state eseguite con le seguenti condizioni:

- Seed Iniziale = 123456789
- Tempi di Servizio di tutti i Server del Sistema con distribuzione Esponenziale.
- Istante di STOP = 100'000

<b>Seed</b>	<b>Tempo Medio di Risposta</b>	<b>Throughput</b>	<b>Popolazione Media</b>
<b>123456789</b>	1.840401	11.162997	22.084811
<b>235351916</b>	1.834404	11.149903	22.012842
<b>511596416</b>	1.840221	11.137328	22.082652
<b>942677419</b>	1.925603	11.165578	23.107241
<b>989757435</b>	1.701814	11.067750	20.421763

Come si può vedere dai dati nella tabella, dipendentemente dal seme, i valori tendono ad essere molto diversi tra di loro, essendo la simulazione effettuata con uno STOP molto basso e quindi ancora dipendente dallo stato iniziale del sistema.

## Capitolo 7

# Service Server - Multi-Server

Andiamo adesso ad analizzare la variante del Sistema che è stata studiata. In particolare all'interno di questa variante il nodo del Sistema Service risulta essere un Multi-Server composto da  $m=2$  serventi con distribuzione Esponenziale. Sempre con riferimento all'algoritmo 1.1.1 del Libro di testo andiamo adesso ad elencare nuovamente i punti dell'algoritmo.

### 7.1 Modello Concettuale

Il sistema modellato è composto dai seguenti sottosistemi:

- **Cassa+Verifica:** Questo nodo della rete è costituito da un Server Singolo e permette l'entrata dei Job all'interno del Sistema indipendentemente dalla loro tipologia (Tipo1, Tipo2 o Tipo3). Viene inoltre eseguito il Controller presente che va a verificare se la richiesta del Job può essere soddisfatta. In particolare viene controllato se la tipologia di gelato richiesta dal Job in arrivo possa essere soddisfatta data la fornitura di gelato corrente presente nel Service Server. In caso affermativo, il Job verrà indirizzato verso il Service Server, altrimenti verrà indirizzato verso il Delay Server.
- **Delay Server:** Questo nodo della rete è costituito da un Server Singolo fa in modo che i Job aspettino che la fornitura di gelato del Service Server venga ripristinata. All'interno del nodo si ha anche una probabilità  $p$  di rimanere in attesa per essere serviti ed una probabilità  $(1-p)$  di uscire dal sistema senza essere serviti. In questo ultimo caso il Job viene considerato come perso.
- **Service Server:** Questo nodo della rete è costituito da un Multi-Server ( $m=2$ ) e fa in modo che i Job vengano serviti e possano uscire dal Sistema.

Dati i sottosistemi sopracitati abbiamo individuato le seguenti variabili di stato:

- Numero Corrente e Totale di Job per ogni sottosistema:
  - actual\_system                      – total\_system
  - actual\_verify                      – total\_verify
  - actual\_delay                      – total\_delay
  - actual\_service                      – total\_service
- Numero Corrente e Totale di Job di tipo 1 per ogni sottosistema:
  - actual\_job1                      – total\_job1
  - actual\_job1\_verify                      – total\_job1\_verify
  - actual\_job1\_delay                      – total\_job1\_delay
  - actual\_job1\_service                      – total\_job1\_service
- Numero Corrente e Totale di Job di tipo 2 per ogni sottosistema:
  - actual\_job2                      – total\_job2
  - actual\_job2\_verify                      – total\_job2\_verify
  - actual\_job2\_delay                      – total\_job2\_delay
  - actual\_job2\_service                      – total\_job2\_service
- Numero Corrente e Totale di Job di tipo 3 per ogni sottosistema:
  - actual\_job3                      – total\_job3
  - actual\_job3\_verify                      – total\_job3\_verify
  - actual\_job3\_delay                      – total\_job3\_delay
  - actual\_job3\_service                      – total\_job3\_service

Le variabili *actual* vanno a contare i Job presenti nel sistema ad ogni istante di tempo, quindi subiranno incrementi e decrementi a seconda dell'evento corrente che viene processato dal Sistema. In particolare, gli eventi che possono avvenire all'interno del sistema sono i seguenti:

- Arrivo di un Job;
- Transito del Job da Verifica+Cassa a Service Server;
- Transito del Job da Verifica+Cassa a Delay Server;
- Transito del Job da Delay Server a Service Server;

- Transito del Job da Delay Server uscendo dal Sistema → Job perso;
- Transito del Job da Service Server uscendo dal Sistema → Job completato.

Le variabili *total*, invece, tengono conto di tutti i Job che sono transitati all'interno del Sistema e quindi andranno a subire solamente degli incrementi. Sia durante l'esecuzione della Simulazione sia al termine della stessa, vengono effettuati dei controlli sulla consistenza delle variabili suddette.

## 7.2 Specification Model

Di seguito andiamo a presentare il Modello di Specifica che caratterizza il Sistema come specificato nel punto (3) dell'algoritmo 1.1.1.

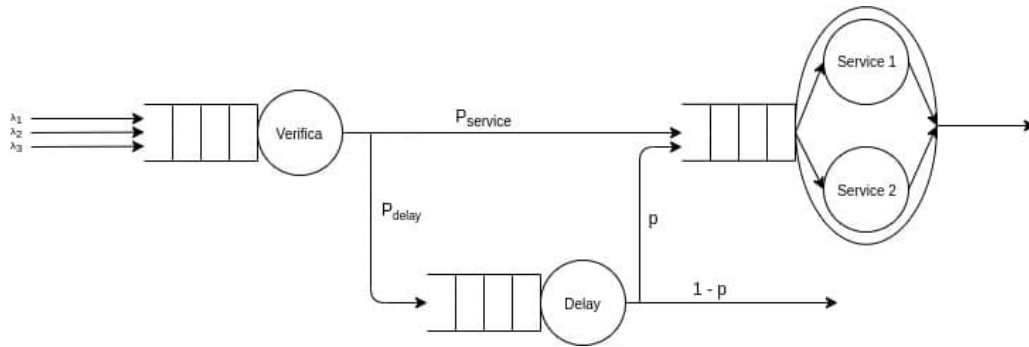


Figura 7.1: Modello della Rete del Sistema - Service Server M/M/m

Come possiamo notare, dal Server Cassa+Verifica entrano i Job e questi usciranno o dal Delay Server oppure dal Service Server. Per andare a validare il Sistema abbiamo eseguito i calcoli sfruttando la rete di Jackson. Per quanto riguarda i valori delle probabilità si ha:

- $p = 0.8$
- $(1 - p) = 0.2$
- $p_{delay} = 0.063996$
- $p_{service} = 0.948820$
- $m = 2$ ;

I valori di  $p_{delay}$  e  $p_{service}$  sono stati ottenuti andando ad effettuare il calcolo delle probabilità di terminazione della Vaschetta di gelato per ogni tipologia di Job in arrivo nel Sistema.

### 7.3 Risultati Analitici - Verifica

Andiamo adesso a presentare i risultati analitici che sono stati ricavati per il Sistema preso in considerazione. In particolare, per ottenere questi risultati è stata calcolata la rete di Jackson ottenuta dal modello di Specifica del Sistema. Questi risultati vengono sfruttati per andare a verificare il Sistema come specificato nel punto (4) dell'algoritmo 1.1.1.

### 7.4 Risultati Analitici - Sistema:

$$\begin{cases} \lambda_{verifica} = \lambda_1 + \lambda_2 + \lambda_3 \\ \lambda_{delay} = p_{delay}\lambda_{verifica} \\ \lambda_{service} = (1 - p_{delay})\lambda_{verifica} + p\lambda_{delay} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica} = 12[Job/min] \\ \lambda_{delay} = 0.767952[Job/min] \\ \lambda_{service} = 11.846410[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica} = \frac{\lambda_{verifica}}{\mu_{verifica}} = 0.923077 \\ \rho_{delay} = \frac{\lambda_{delay}}{\mu_{delay}} = 0.383976 \\ \rho_{service} = \frac{\lambda_{service}}{m\mu_{service}} = 0.455631 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica} = \frac{\rho_{verifica}}{1-\rho_{verifica}} = 12[Job] \\ E[N]_{delay} = \frac{\rho_{delay}}{1-\rho_{delay}} = 0.623313[Job] \\ E[N]_{service} = \frac{\rho_{service}}{1-\rho_{service}} + m\rho_{service} = 1.150002[Job] \\ E[N] = E[N]_{verifica} + E[N]_{delay} + E[N]_{service} = 13.773316[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica} = \frac{\lambda_{verifica}}{\lambda_{tot}} = 1 \\ v_{delay} = \frac{\lambda_{delay}}{\lambda_{tot}} = 0.063996 \\ v_{service} = \frac{\lambda_{service}}{\lambda_{tot}} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica} = 1[min] \\ E[Ts]_{delay} = 0.811657[min] \\ E[Ts]_{service} = 0.097076[min] \\ E[Ts] = 1.147776[min] \end{cases}$$

- **Throughput:**

$$\begin{cases} X_{verifica} = \frac{E[N]_{verifica}}{E[Ts]_{verifica}} = \rho_{verifica}\mu_{verifica} = 12[Job/min] \\ X_{delay} = \frac{E[N]_{delay}}{E[Ts]_{delay}} = \rho_{delay}\mu_{delay} = 0.767952[Job/min] \\ X_{service} = \frac{E[N]_{service}}{E[Ts]_{service}} = \rho_{service}\mu_{service} = 11.846410[Job/min] \end{cases}$$

## 7.5 Risultati Analitici - Job di Tipo 1:

$$\begin{cases} \lambda_{verifica1} = \lambda_1 \\ \lambda_{delay1} = p_{delay}\lambda_{verifica1} \\ \lambda_{service1} = (1 - p_{delay})\lambda_{verifica1} + p\lambda_{delay1} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica1} = 3[Job/min] \\ \lambda_{delay1} = 0.191148[Job/min] \\ \lambda_{service1} = 2.9617704[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica1} = \frac{\lambda_{verifica1}}{\mu_{verifica1}} = 0.230769 \\ \rho_{delay1} = \frac{\lambda_{delay1}}{\mu_{delay1}} = 0.095994 \\ \rho_{service1} = \frac{\lambda_{service1}}{m\mu_{service1}} = 0.113908 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica1} = \frac{\rho_{verifica1}}{1-\rho} = 3[Job] \\ E[N]_{delay1} = \frac{\rho_{delay1}}{1-\rho} = 0.155828[Job] \\ E[N]_{service1} = \frac{p_q\rho_{service1}}{1-\rho_{service}} + m\rho_{service1} = 0.287501[Job] \\ E[N]^1 = E[N]_{verifica1} + E[N]_{delay1} + E[N]_{service1} = 3.443329[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica1} = \frac{\lambda_{verifica1}}{\lambda_1} = 1 \\ v_{delay1} = \frac{\lambda_{delay1}}{\lambda_1} = 0.063996 \\ v_{service1} = \frac{\lambda_{service1}}{\lambda_1} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica1} = 1[min] \\ E[Ts]_{delay1} = 0.811657[min] \\ E[Ts]_{service1} = 0.097076[min] \\ E[Ts]^1 = 1.147776[min] \end{cases}$$



- **Throughput:**

$$\begin{cases} X_{verifica1} = \frac{E[N]_{verifica1}}{E[Ts]_{verifica1}} = \rho_{verifica1} \mu_{verifica1} = 3[Job/min] \\ X_{delay1} = \frac{E[N]_{delay1}}{E[Ts]_{delay1}} = \rho_{delay1} \mu_{delay1} = 0.191988[Job/min] \\ X_{service1} = \frac{E[N]_{service1}}{E[Ts]_{service1}} = \rho_{service1} \mu_{service1} = 2.961602[Job/min] \end{cases}$$

## 7.6 Risultati Analitici - Job di Tipo 2:

$$\begin{cases} \lambda_{verifica2} = \lambda_2 \\ \lambda_{delay2} = p_{delay} \lambda_{verifica2} \\ \lambda_{service2} = (1 - p_{delay}) \lambda_{verifica2} + p \lambda_{delay2} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica2} = 4[Job/min] \\ \lambda_{delay2} = 0.254864[Job/min] \\ \lambda_{service2} = 3.9490272[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica2} = \frac{\lambda_{verifica2}}{\mu_{verifica2}} = 0.307692 \\ \rho_{delay2} = \frac{\lambda_{delay2}}{\mu_{delay2}} = 0.127992 \\ \rho_{service2} = \frac{\lambda_{service2}}{m \mu_{service2}} = 0.151877 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica2} = \frac{\rho_{verifica2}}{1-\rho} = 4[Job] \\ E[N]_{delay2} = \frac{\rho_{delay2}}{1-\rho} = 0.207771[Job] \\ E[N]_{service2} = \frac{p_q \rho_{service2}}{1-\rho_{service}} + m \rho_{service2} = 0.383334[Job] \\ E[N]^2 = E[N]_{verifica2} + E[N]_{delay2} + E[N]_{service2} = 4.591105[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica2} = \frac{\lambda_{verifica2}}{\lambda_2} = 1 \\ v_{delay2} = \frac{\lambda_{delay2}}{\lambda_2} = 0.063996 \\ v_{service2} = \frac{\lambda_{service2}}{\lambda_2} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica2} = 1[min] \\ E[Ts]_{delay2} = 0.811657[min] \\ E[Ts]_{service2} = 0.097076[min] \\ E[Ts]^2 = 1.147776[min] \end{cases}$$

- **Throughput:**

$$\begin{cases} X_{verifica2} = \frac{E[N]_{verifica2}}{E[Ts]_{verifica2}} = \rho_{verifica2} \mu_{verifica2} = 4[Job/min] \\ X_{delay2} = \frac{E[N]_{delay2}}{E[Ts]_{delay2}} = \rho_{delay2} \mu_{delay2} = 0.255984[Job/min] \\ X_{service2} = \frac{E[N]_{service2}}{E[Ts]_{service2}} = \rho_{service2} \mu_{service2} = 3.948803[Job/min] \end{cases}$$

## 7.7 Risultati Analitici - Job di Tipo 3:

$$\begin{cases} \lambda_{verifica3} = \lambda_3 \\ \lambda_{delay3} = p_{delay} \lambda_{verifica3} \\ \lambda_{service3} = (1 - p_{delay}) \lambda_{verifica3} + p \lambda_{delay3} \end{cases}$$

In particolare, andando ad eseguire i calcoli sul sistema, sono stati ricavati:

- **Tassi di Arrivo:**

$$\begin{cases} \lambda_{verifica3} = 5[Job/min] \\ \lambda_{delay3} = 0.31858[Job/min] \\ \lambda_{service3} = 4.936284[Job/min] \end{cases}$$

- **Utilizzazione:**

$$\begin{cases} \rho_{verifica3} = \frac{\lambda_{verifica3}}{\mu_{verifica3}} = 0.384615 \\ \rho_{delay3} = \frac{\lambda_{delay3}}{\mu_{delay3}} = 0.159990 \\ \rho_{service3} = \frac{\lambda_{service3}}{m \mu_{service3}} = 0.189846 \end{cases}$$

- **Numero medio di Utenti:**

$$\begin{cases} E[N]_{verifica3} = \frac{\rho_{verifica3}}{1-\rho} = 5[Job] \\ E[N]_{delay3} = \frac{\rho_{delay3}}{1-\rho} = 0.259714[Job] \\ E[N]_{service3} = \frac{p_q \rho_{service3}}{1-\rho_{service}} + m \rho_{service3} = 0.479168[Job] \\ E[N] = E[N]_{verifica3} + E[N]_{delay3} + E[N]_{service3} = 5.738882[Job] \end{cases}$$

- **Tassi di Visita:**

$$\begin{cases} v_{verifica3} = \frac{\lambda_{verifica3}}{\lambda_3} = 1 \\ v_{delay3} = \frac{\lambda_{delay3}}{\lambda_3} = 0.063996 \\ v_{service3} = \frac{\lambda_{service3}}{\lambda_3} = 0.987201 \end{cases}$$

- **Tempo medio di Risposta:**

$$\begin{cases} E[Ts]_{verifica3} = 1[min] \\ E[Ts]_{delay3} = 0.811657[min] \\ E[Ts]_{service3} = 0.097076[min] \\ E[Ts] = 1.147776[min] \end{cases}$$

- **Throughput:**

$$\left\{ \begin{array}{l} X_{verifica3} = \frac{E[N]_{verifica2}}{E[Ts]_{verifica2}} = \rho_{verifica2} \mu_{verifica2} = 5 [Job/min] \\ X_{delay3} = \frac{E[N]_{delay2}}{E[Ts]_{delay2}} = \rho_{delay2} \mu_{delay2} = 0.319980 [Job/min] \\ X_{service3} = \frac{E[N]_{service2}}{E[Ts]_{service2}} = \rho_{service2} \mu_{service2} = 4.936004 [Job/min] \end{array} \right.$$

## 7.8 Confronto dei Risultati Analitici - Validazione

E' stata verificata la correttezza dei dati ricavati dalla simulazione andando a confrontare con i risultati ottenuti dalla risoluzione del sistema modellato attraverso la rete di Jackson introdotta in precedenza. Questi risultati vengono sfruttati per andare ad effettuare la validazione del Sistema come specificato nel punto (6) dell'algoritmo 1.1.1.

Per poter effettuare un confronto coerente, la simulazione è stata eseguita sotto le seguenti condizioni:

- Tempi di servizio del Server Cassa+Verifica con distribuzione Esponenziale
- Tempi di servizio del Server Delay con distribuzione Esponenziale
- Tempi di servizio del Service Server con distribuzione Esponenziale
- Seme = 123456789
- Valore di STOP = 2'000'000 (Identifica l'istante di tempo massimo entro cui possono arrivare i Job nel Sistema)

In particolare, il valore di STOP è stato selezionato in base all'analisi della stazionarietà che verrà introdotta in seguito.

## 7.9 Confronto dei Risultati del Sistema

### Tempi di Risposta:

	Risultato Analitico	Risultato Simulato
$E[Ts]$	1.147776	1.116024 +/- 0.319317
$E[Ts]_1$	1.147776	1.079696 +/- 0.298874
$E[Ts]_2$	1.147776	1.103393 +/- 0.312266
$E[Ts]_3$	1.147776	1.128292 +/- 0.326441

### Throughput:

	Risultato Analitico	Risultato Simulato
X	11.846410	11.158054 +/- 0.848397
$X_1$	2.961602	2.877785 +/- 0.119141
$X_2$	3.948803	3.736229 +/- 0.573063
$X_3$	4.936004	4.543639 +/- 0.282255

**Numero medio di Job:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[N]$	13.773316	13.392286 +/- 0.981663
$E[N]_1$	3.443329	3.239089 +/- 0.689865
$E[N]_2$	4.591105	4.413573 +/- 0.996257
$E[N]_3$	5.738882	5.641462 +/- 0.161017

## 7.10 Confronto dei Risultati del Server Verifica:

**Tempi di Risposta:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[Ts]_{verifica}$	1	0.981175 +/- 0.247461
$E[Ts]_{verifica1}$	1	0.981839 +/- 0.247801
$E[Ts]_{verifica2}$	1	0.980790 +/- 0.247409
$E[Ts]_{verifica3}$	1	0.981095 +/- 0.247468

**Throughput:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$X_{verifica}$	12	11.813488 +/- 0.700090
$X_{verifica1}$	3	2.954025 +/- 0.232848
$X_{verifica2}$	4	3.938276 +/- 0.969051
$X_{verifica3}$	5	4.921187 +/- 0.196539

**Numero medio di Job:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[N]_{verifica}$	12	11.775121 +/- 0.657912
$E[N]_{verifica1}$	3	2.946410 +/- 0.231901
$E[N]_{verifica2}$	4	3.923920 +/- 0.960485
$E[N]_{verifica3}$	5	4.904791 +/- 0.187198

## 7.11 Confronto dei Risultati del Server Delay:

**Tempi di Risposta:**

	<b>Risultato Analitico</b>	<b>Risultato Simulato</b>
$E[Ts]_{delay}$	0.811657	0.844281 +/- 0.182412
$E[Ts]_{delay1}$	0.811657	0.847680 +/- 0.187017
$E[Ts]_{delay2}$	0.811657	0.844726 +/- 0.183714
$E[Ts]_{delay3}$	0.811657	0.843485 +/- 0.182627

### Throughput:

	Risultato Analitico	Risultato Simulato
$X_{delay}$	0.767952	0.819014 +/- 0.171596
$X_{delay1}$	0.191988	0.195279 +/- 0.002362
$X_{delay2}$	0.255984	0.251996 +/- 0.016352
$X_{delay3}$	0.319980	0.331739 +/- 0.057109

### Numero medio di Job:

	Risultato Analitico	Risultato Simulato
$E[N]_{delay}$	0.623313	0.702452 +/- 0.126238
$E[N]_{delay1}$	0.155828	0.182028 +/- 0.001727
$E[N]_{delay2}$	0.207771	0.216224 +/- 0.011961
$E[N]_{delay3}$	0.259714	0.304199 +/- 0.041795

## 7.12 Confronto dei Risultati del Server Service:

### Tempi di Risposta:

	Risultato Analitico	Risultato Simulato
$E[Ts]_{service}$	0.097076	0.092698 +/- 0.001812
$E[Ts]_{service1}$	0.097076	0.092057 +/- 0.001445
$E[Ts]_{service2}$	0.097076	0.092034 +/- 0.001436
$E[Ts]_{service3}$	0.097076	0.092031 +/- 0.001439

### Throughput:

	Risultato Analitico	Risultato Simulato
$X_{service}$	11.846410	11.158054 +/- 0.171596
$X_{service1}$	2.961602	2.877785 +/- 0.119139
$X_{service2}$	3.948803	3.736629 +/- 0.573065
$X_{service3}$	4.936004	4.543639 +/- 0.282255

### Numero medio di Job:

	Risultato Analitico	Risultato Simulato
$E[N]_{service}$	1.150002	1.114714 +/- 0.232291
$E[N]_{service1}$	0.287501	0.270650 +/- 0.012313
$E[N]_{service2}$	0.383334	0.373428 +/- 0.020625
$E[N]_{service3}$	0.479168	0.462482 +/- 0.030614

### 7.13 Analisi Stazionaria

Per effettuare l'analisi stazionaria sono state eseguite numerose simulazioni aventi in comune il medesimo seme, ma con istanti di STOP crescente. In particolare il valore di STOP va da 100'000 a 2'500'000, in modo tale da stabilire quando il Sistema inizia a perdere l'influenza dello stato iniziale.

Le simulazioni sono state effettuate ad intervalli del valore di STOP pari a 200'000 come è possibile notare dalla Tabella riportata di seguito.

I parametri della simulazione sono i seguenti:

- Seed = 123456789
- Tempi di Servizio di tutti i Server del Sistema con distribuzione Esponenziale.

STOP	Tempo Medio di Risposta	Throughput	Popolazione Media
100'000	1.110464	11.167771	13.325571
300'000	1.116144	11.161915	13.393722
500'000	1.116942	11.161582	13.403305
700'000	1.118507	11.160242	13.422087
900'000	1.119413	11.161731	13.432955
1'100'000	1.119099	11.160725	13.429190
1'300'000	1.119391	11.160697	13.432696
1'500'000	1.119500	11.161197	13.434000
1'700'000	1.119655	11.161582	13.435856
1'900'000	1.120222	11.159987	13.440267
2'100'000	1.119604	11.160861	13.435243
2'300'000	1.119798	11.160656	13.437573
2'500'000	1.119322	11.160494	13.443860

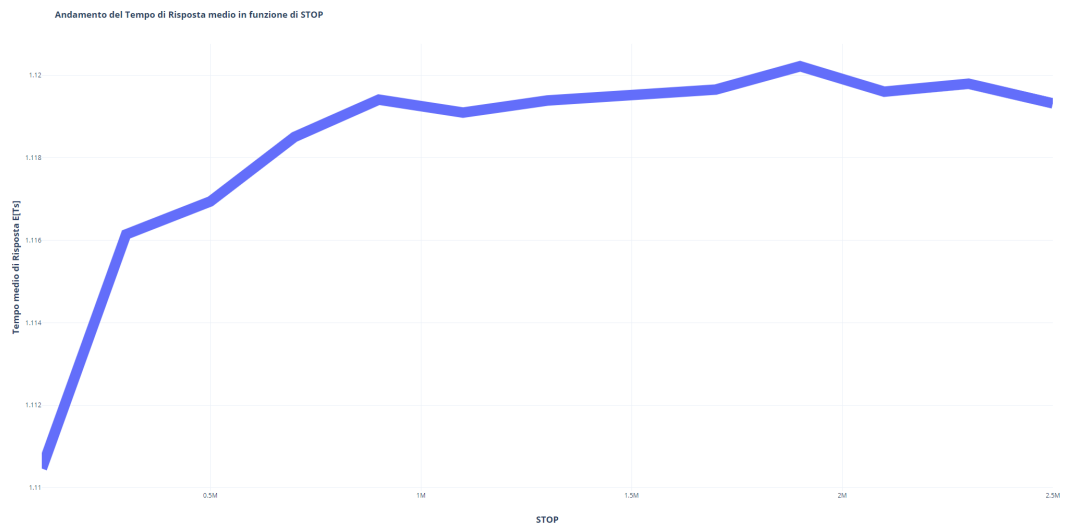


Figura 7.2: Andamento del Tempo medio di Risposta in funzione di STOP

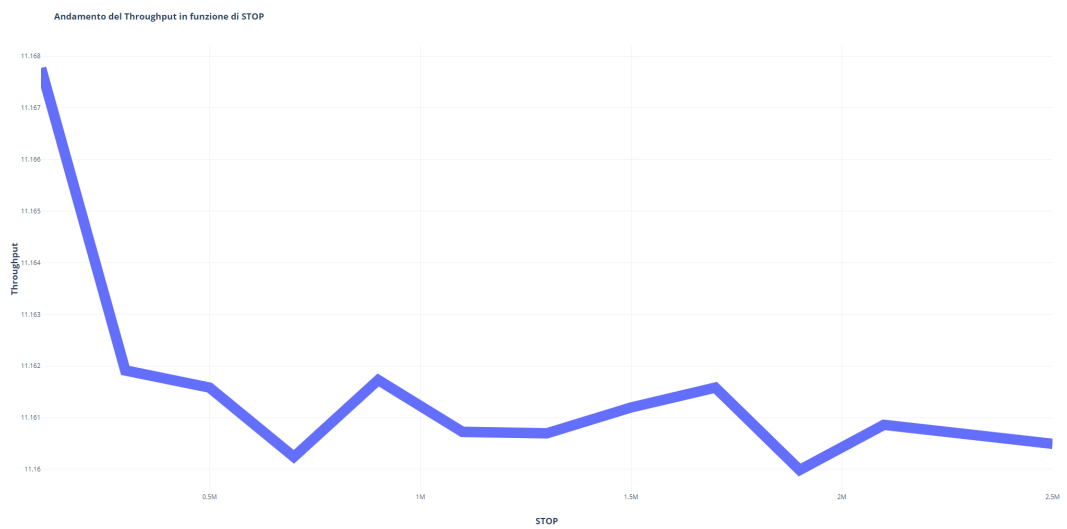


Figura 7.3: Andamento del Throughput in funzione di STOP



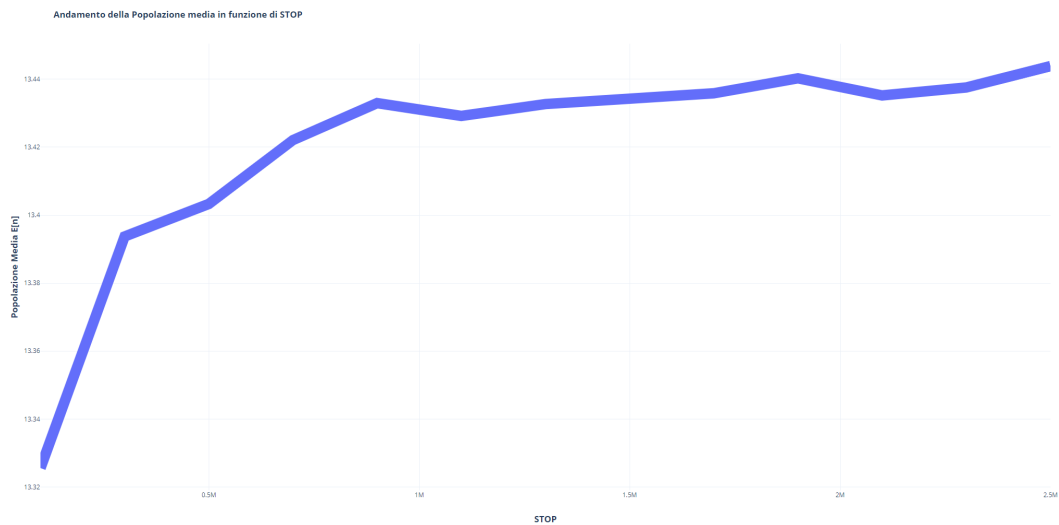


Figura 7.4: Andamento della Popolazione media in funzione di STOP

Come si può verificare dai risultati ottenuti, i valori del Tempo di Risposta medio, del Throughput e della Popolazione media, iniziano ad avvicinarsi sempre più tra loro con l'aumentare del valore di STOP.

In particolare possiamo notare che superato l'istante di STOP di 1'300'000, i valori del tempo di Risposta medio e del Throughput differiscono tra loro di un valore dell'ordine di  $10^{-2}$ . Possiamo quindi presupporre che proprio da questo momento in poi il Sistema inizi a comportarsi in maniera Stazionaria.

## 7.14 Analisi Transiente

Per effettuare l'analisi transiente sono state eseguite cinque simulazioni con semi diversi ed indipendenti tra loro.

I Semi risultano essere indipendenti poiché le simulazioni, effettuate in maniera sequenziale, utilizzano come nuovo seme l'ultimo valore restituito dal generatore di numeri pseudo-casuali durante la simulazione precedente.

Le simulazioni sono inoltre state eseguite con le seguenti condizioni:

- Seed Iniziale = 123456789
- Tempi di Servizio di tutti i Server del Sistema con distribuzione Esponenziale.
- Istante di STOP = 100'000

<b>Seed</b>	<b>Tempo Medio di Risposta</b>	<b>Throughput</b>	<b>Popolazione Media</b>
<b>123456789</b>	1.073840	11.162999	12.886082
<b>235351916</b>	1.067844	11.149903	12.814130
<b>511596416</b>	1.073657	11.137329	12.883879
<b>942677419</b>	1.159044	11.165578	12.908528
<b>989757435</b>	0.935254	11.067749	12.223049

Come si può vedere dai dati nella tabella, dipendentemente dal seme, i valori tendono ad essere molto diversi tra di loro, essendo la simulazione effettuata con uno STOP molto basso e quindi ancora dipendente dallo stato iniziale del sistema.

## Capitolo 8

# Confronto dei Sistemi

Per confrontare le prestazioni dei due sistemi le simulazioni sono state svolte con le seguenti caratteristiche:

- Tempo di Servizio dei nodi del sistema con distribuzione esponenziale
- Seed = 123456789
- STOP = 2'000'000

L'istante di STOP caratterizza l'istante di tempo in cui entrambi i sistemi si comportano in maniera stazionaria. Le differenze sono presenti all'interno del nodo service e di seguito riportiamo una tabella che mostra le tali differenze.

**Nodo Service**

	<b>Tempo Medio di Risposta</b>	<b>Throughput</b>	<b>Popolazione Media</b>
<b>Service Single Server</b>	0.892219	11.158054	10.113427
<b>Service Multi Server</b>	0.092698	11.158054	1.114714

Come possiamo notare dai valori riportati, l'aggiunta di un server all'interno del nodo service porta un notevole miglioramento in termini di tempo medio di risposta e di popolazione media lasciando il throughput invariato. Tutti gli altri nodi che compongono il sistema non vengono riportati in quanto rimangono invariati.

Di seguito vengono inoltre riportate le differenze introdotte al livello del Sistema andando a confrontare il nodo Service Single-Server e Multi-Server con  $m=2$ .

**Sistema**

	<b>Tempo Medio di Risposta</b>	<b>Throughput</b>	<b>Popolazione Media</b>
<b>Service Single Server</b>	1.882583	11.158054	22.590997
<b>Service Multi Server</b>	1.116024	11.158054	13.392286

## Capitolo 9

# Conclusioni

In conclusione possiamo affermare che, per ottenere una simulazione corretta, sono necessarie le seguenti operazioni:

- **Utilizzo dei batch means:** necessari per far sì che le variabili di stato siano identicamente distribuite ed indipendenti
- **Verifica della stazionarietà:** necessaria per l'ottenimento dei risultati nel momento in cui il sistema si comporta in maniera stazionaria
- **Attenzione sulla selezione dei seed:** necessaria in quanto i risultati della simulazione possono variare a causa della loro scelta
- **Utilizzo di poche e chiare variabili di stato:** permettono di rendere il sistema modulare e quindi facilmente modificabile andando a cambiare i parametri della simulazione

# Bibliografia

- [1] *Discrete Event Simulation*, Lawrence Leemis, 2004