

Progetto PMCSN  
Analisi dei tempi di risposta di un centro vaccinale per  
l'inoculazione del vaccino antinfluenzale  
Open Day 17 dicembre 2022

Giulia Menichini 0298906

February 19, 2023

# Contents

<b>1</b>	<b>Caso di Studio</b>	<b>2</b>
1.1	Obiettivo dello studio . . . . .	2
<b>2</b>	<b>Modello base</b>	<b>3</b>
2.1	Modello concettuale . . . . .	3
2.1.1	Stato del sistema . . . . .	4
2.1.2	Eventi . . . . .	4
2.2	Modello delle specifiche . . . . .	4
2.2.1	Dati di input . . . . .	4
2.3	Modello Computazionale . . . . .	6
2.3.1	Clock di simulazione . . . . .	6
2.3.2	Scheduler . . . . .	7
2.3.3	Eventi . . . . .	7
2.3.4	Generazione degli eventi . . . . .	7
2.3.5	Campionamento delle statistiche . . . . .	8
2.4	Verifica . . . . .	9
2.4.1	Controlli di consistenza sulle statistiche di output . . . . .	10
2.4.2	Confronto statistiche di simulazione con valori teorici . . . . .	12
2.5	Validazione . . . . .	14
2.5.1	Test Effettuati . . . . .	14
2.6	Simulazione ad orizzonte infinito . . . . .	16
2.6.1	Esperimenti . . . . .	19
2.7	Simulazione orizzonte finito . . . . .	22
2.7.1	Esperimenti . . . . .	22
<b>3</b>	<b>Sistema Migliorato</b>	<b>24</b>
3.1	Modifiche apportate . . . . .	24
3.1.1	Obiettivi . . . . .	24
3.2	Modello Concettuale . . . . .	25
3.3	Verifica Modello Migliorativo . . . . .	25
3.3.1	Controllo di consistenza per le statistiche di output . . . . .	26
3.3.2	Confronto statistiche di simulazione con valori teorici . . . . .	27
3.4	Simulazione ad orizzonte infinito - Configurazione Ottima . . . . .	28
3.4.1	Simulazione ad orizzonte finito . . . . .	28
3.5	Confronti . . . . .	30
3.5.1	Confronto con precedente modello - Simulazione ad orizzonte infinito . . . . .	30
3.5.2	Confronto con precedente modello - Simulazione ad orizzonte finito . . . . .	31
3.6	Atro modello migliorativo . . . . .	31
.1	Appendice . . . . .	32

# Chapter 1

## Caso di Studio

Il progetto ha come obiettivo la realizzazione di una simulazione di un centro vaccinale in particolare fa riferimento al giorno 17 dicembre 2022 data in cui la regione lazio in collaborazione con la provincia di Frosinone ha organizzato una giornata di openDay in cui ogni cittadino, se voleva, poteva recarsi liberamente nel centro e ricevere gratuitamente la somministrazione del vaccino antinfluenzale per garantire la copertura anticorpale prima delle vacanze natalizie. I dati utilizzati (le probabilità, il numero di pazienti che hanno fatto accesso nel centro e la configurazione utilizzata) per la definizione del sistema sono dati reali gentilmente forniti dalla direttrice dell'unità operativa per il Coordinamento dell'attività vaccinale.

Il centro vaccinale garantiva la vaccinazione dalle ore 8.30 alle ore 14.30 senza preventiva prenotazione.

Sono state inoltre messe a disposizione due tipologie di vaccino, il vaccino antinfluenzale con iniezione (vaccino non vivo) e il vaccino antinfluenzale a virus attenuato somministrato tramite un semplice spray nasale. Questa tipologia di vaccino può essere somministrata a bambini ed adolescenti di età compresa tra i 2 e i 18 anni.

### 1.1 Obiettivo dello studio

L'obiettivo principale prevede l'analisi dei tempi di risposta del sistema, rispettando il seguente QoS:

- Il tempo di risposta medio del sistema non deve superare i 25 min, i pazienti devono trascorrere all'interno del centro vaccinale non più di 25 minuti da quando entrano a quando escono da esso dopo aver ricevuto la somministrazione del vaccino.

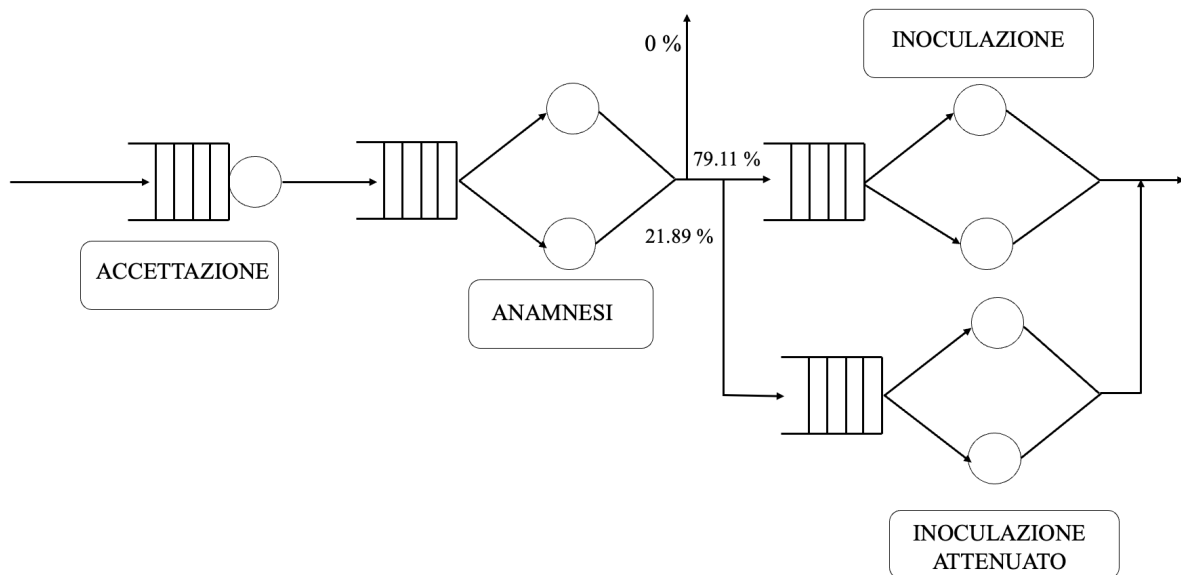
# Chapter 2

## Modello base

### 2.1 Modello concettuale

Sono stati modellati i seguenti centri:

- Accettazione
- Anamnesi
- Inoculazione
- Inoculazione Attenuato



Il centro **accettazione** rispettando la realtà viene modellato con coda singola a servente singolo **M/M/1**.

I centri **anamnesi**, **inoculazione** e **inoculazione attenuato** rispettando anche in questo caso la realtà vengono modellati con code singole a servente multiplo **M/M/2**.

### 2.1.1 Stato del sistema

Gli stati del sistema sono una sua caratterizzazione completa ad ogni istante di tempo in particolare il sistema viene descritto dalle seguenti variabili di stato:

- Numero di utenti per centro
- Numero di utenti in servizio
- Numero di utenti in coda
- Stato di ogni servente che può essere *idle* o *busy*
- Stato di ogni coda che può essere *empty* o *not\_empty*

### 2.1.2 Eventi

Gli eventi causano un cambiamento dello stato del sistema e questo si verifica per gli eventi di completamento del servizio e gli arrivi dei pazienti all'interno del centro.

## 2.2 Modello delle specifiche

### 2.2.1 Dati di input

I dati di input forniti al modello sono: il tasso di arrivo definito come  $LAMBDA$ , il tempo medio di servizio per i diversi centri per ogni singolo servente e i valori delle probabilità *prob\_vaccino* e *prob\_vaccino\_attenuato*.

In particolare:

<b>PAZIENTI</b>	164 in 6h
<b>LAMBDA</b>	0.456 job/min

Table 2.1: Numero di pazienti in ingresso al centro e tasso medio di arrivo

<b>MU_ACCETTAZIONE</b>	0.515 job/min
<b>MU_ANAMNESI</b>	0.3 job/min
<b>MU_INOCULAZIONE</b>	0.249 job/min
<b>MU_INOCULAZIONE_ATTENUATO</b>	0.36 job/min

Table 2.2: Tabella valori dei  $\mu$  dei singoli centri

<b>prob_vaccino</b>	100 %
<b>prob_vaccino_attenuato</b>	21.897819 %

Table 2.3: Probabilità di routing

In particolare prob\_vaccino rappresenta la probabilità che un paziente, sottoposto al centro di anamnesi, abbia un esito positivo mentre prob\_vaccino\_attenuato indica la probabilità che un paziente, dopo aver ricevuto l'idoneità al vaccino, faccia il vaccino vivo attenuato.

Nel caso specifico della giornata del 17 dicembre nessun paziente è risultato non idoneo al vaccino. Ogni paziente che si è recato nel centro vaccinale ha quindi ricevuto la somministrazione.

Le assunzioni che vengono fatte sono le seguenti:

- I pazienti hanno un comportamento *one-step* vi può essere infatti lo spostamento di un solo job o paziente alla volta
- Sistema non-preemptive, non è infatti previsto un meccanismo di priorità che causa l'interruzione di job già in servizio per servire job con priorità più alta.
- I pazienti, non potendo effettuare una prenotazione, arrivano al centro vaccinale ad istanti di tempo casuali questo implica che si ha una distribuzione poissoniana degli arrivi e tempi di interarrivo che seguono una distribuzione esponenziale.
- Tempi di servizio distribuiti esponenzialmente: La distribuzione esponenziale è una distribuzione "senza memoria", il tempo trascorso non influenza l'evento che si verifica in futuro. In questo caso si è quindi supposto che il tempo di servizio dipenda principalmente dal tempo necessario per eseguire il "servizio". Tuttavia tale tempo nella realtà può essere influenzato da altri fattori ad esempio il numero di persone in attesa, la complessità dell'attività da eseguire, la disponibilità di risorse e del personale sotto questa assunzione sarebbe stato opportuno utilizzare una distribuzione più complessa come la distribuzione Normale (o Gaussiana) la quale tiene conto in modo più esplicito dei fattori che influenzando il tempo medio di servizio rispetto alla distribuzione esponenziale.<sup>1</sup>

---

<sup>1</sup>In questo paper [www.medrxiv.org/content/10.1101/2021.04.07.21255067v1.full](https://www.medrxiv.org/content/10.1101/2021.04.07.21255067v1.full) vengono fatte le stesse assunzioni e viene utilizzata una distribuzione esponenziale : la maggior parte dei pazienti impiegano per un servizio un tempo relativamente breve ma con una coda lunga per alcuni pazienti che chiedono più tempo e altri che impiegano molto più tempo

In questo paper <https://www.medrxiv.org/content/10.1101/2021.03.24.21253517v1.full-text> vengono mostrate diverse assunzioni (la stagione invernale richiede un maggior numero di indumenti, la possibile inesperienza del personale) e per ogni distribuzione testata vengono forniti dei "punteggi" con il criterio AKAIKE. Sotto quelle assunzioni la distribuzione esponenziale, come si immaginava è quella meno performante

- Tutte le code presenti all'interno del sistema (coda per l'accettazione, coda per l'anamnesi, coda per l'inoculazione e coda per l'inoculazione attenuato) seguono la politica di scheduling **FIFO**. E' ragionevole infatti pensare che il primo job o paziente avrà precedenza sui job che lo susseguono in base all'istante di arrivo.

## 2.3 Modello Computazionale

La simulazione è stata interamente progettata in linguaggio C fatta eccezione per la sezione di visualizzazione con grafici definita con le librerie matplotlib e seaborn fornite dal linguaggio di programmazione Python. L'approccio utilizzato per la realizzazione della simulazione è quello della next-event simulation vengono infatti definiti degli eventi che si susseguono. Di seguito vengono presentate le fasi attraversate:

### 2.3.1 Clock di simulazione

L'orologio di sistema viene posto all'interno della struct "t" . Tale struct memorizza le informazioni relative al tempo di simulazione. In particolare:

```
struct t{  
    double current;  
    double next;  
    double last;  
}t
```

- **double next**: Istante di occorrenza del successivo evento nel sistema.
- **double last**: Ultimo istante di arrivo del job.
- **double current**: Istante corrente, l'orologio di sistema.

Di seguito sono riportate alcune osservazioni relative al tempo di simulazione:

- Il tempo viene misurato in minuti.
- L'orologio di sistema viene inizializzato a START.
- Il tempo di simulazione è pari a STOP (per la simulazione ad orizzonte finito).

<b>START</b>	0.0
<b>STOP</b>	360.0 min

Table 2.4: Tabella dati sul tempo di simulazione

### 2.3.2 Scheduler

Il sistema di gestione del tempo, che assicura la sequenza ordinata degli eventi temporali, in ogni iterazione del ciclo di simulazione:

- Calcola la prossima occorrenza dell'evento  $t.next = next\_event(..)$
- Aggiorna l'orologio di sistema  $t.current = t.next$  finché il tempo di occorrenza non supera STOP nel caso di simulazione ad orizzonte finito.

### 2.3.3 Eventi

Gli eventi che si susseguono nella simulazione vengono gestiti dalla struttura dati "event\_list".

```
struct event_list{
    double current;
    double x;
}event_list
```

- **double** current: Prossimo istante dell'evento.
- **double** x: Stato dell'evento  $\in \{0,1\}$ .

In particolare gli eventi sono: l'arrivo dall'esterno e il completamento dei centri. Il completamento del centro di accettazione è l'arrivo per il centro di anamnesi. Il completamento del centro di anamnesi è l'arrivo del centro di inoculazione o inoculazione attenuato considerando per ogni centro le probabilità di routing.

### 2.3.4 Generazione degli eventi

La generazione degli eventi avviene grazie alla libreria "rngs.c". In particolare viene invocata la funzione **PlantSeeds()** una sola volta per l'inizializzazione degli Stream utilizzati. L'arrivo viene gestito con la funzione **GetArrival()**, si seleziona lo stream mediante la funzione **SelectStream()** e viene generata una variata pseudocasuale di tipo esponenziale:  $Exponential(1/LAMBDA)$ .

Per la definizione del tempo medio di servizio viene utilizzata la funzione **GetService()**, si



seleziona nuovamente lo stream con la funzione **SelectStream()** e vengono generate variate esponenziali con media dipendente dallo specifico centro.

- Exponential ( $1/MU\_ACCETTAZIONE$ )
- Exponential ( $1/MU\_ANAMNESI$ )
- Exponential ( $1/MU\_INOCULAZIONE$ )
- Exponential ( $1/MU\_INOCULAZIONE\_ATTENUATO$ )

I valori dei tassi di servizio sono fissati e presenti nel paragrafo 2.2.1.

### 2.3.5 Campionamento delle statistiche

Avviata la simulazione, le statistiche ottenute vengono mantenute nella struct "dati". Tale struct è stata utilizzata per la stampa delle statistiche sia per la simulazione ad orizzonte finito che per la simulazione ad orizzonte infinito.

```
struct dati{
    double lambda;
    double Ts;
    double Tq;
    double Es;
    double N;
    double Nq;
    double ro;
    char *name;
    double job;
}dati
```

In particolare contiene:

- **Lambda** indica il tasso medio di arrivo, quanti job al minuto fanno ingresso nel sistema sopra descritto
- **Ts** indica il tempio medio di risposta ovvero il tempo totale che il job ha passato all'interno del singolo centro:

$$\bar{w}_c = \frac{1}{n_c} \int_0^{t_{end}} l_c(t) dt$$

- **Tq** indica il tempo medio in coda nel singolo centro:

$$\bar{d}_c = \frac{1}{n_c} \int_0^{t_{end}} q_c(t) dt$$

- **Es** indica il tempo medio di servizio nel centro:

$$\bar{s}_c = \frac{1}{n_c} \int_0^{t_{end}} s_c(t) dt$$

- **N** è il numero medio di job che hanno completato l'operazione nel centro:

$$\bar{l}_c = \frac{1}{t_{end}} \int_0^{t_{end}} l_c(t) dt$$

- **Nq** è il numero medio di job che attendono in coda nel centro:

$$\bar{q}_c = \frac{1}{t_{end}} \int_0^{t_{end}} q_c(t) dt$$

- **ro** indica l'utilizzazione del centro:

$$\bar{x}_c = \frac{1}{t_{end} * m_c} \int_0^{t_{end}} s_c(t) dt$$

con  $m_c$  numero di server (per il servente MULTI) e  $t_{end}$  istante di fine simulazione.

- **name** indica il nome del centro, utile per la stampa degli output.
- **job** indica il numero totale di job all'interno del centro considerato.

## 2.4 Verifica

Nella fase di verifica è stata valutata la correttezza del simulatore.

Nello specifico si è andati a valutare la consistenza delle statistiche di output, in particolare verificando specifiche uguaglianze sulle statistiche job-averaged e time-averaged. Sono stati inoltre confrontati i risultati ottenuti dall'analisi teorica con i risultati ottenuti lanciando la simulazione.

### 2.4.1 Controlli di consistenza sulle statistiche di output

Viene verificata la seguente uguaglianza:

$$\bar{d} = \bar{w} - \bar{s}$$

Nel centro **Accettazione** si ha:

$$\bar{w} = 17.136825 \text{ min}$$

$$\bar{d} = 15.193973 \text{ min}$$

$$\bar{s} = 1.942852 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $15.193973 = 17.136825 - 1.942852$

Nel centro **Anamnesi**, centro con coda singola e servente multiplo (2 serventi per questa configurazione) si ha:

$$\bar{w} = 7.826133 \text{ min}$$

$$\bar{d} = 4.502273 \text{ min}$$

$$\bar{s} = 1.661930 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $4.502273 = 7.826133 - (1.661930 * 2)$

Nel centro **Inoculazione**, centro con coda singola e servente multiplo (2 serventi per questa configurazione) si ha:

$$\bar{w} = 8.160273 \text{ min}$$

$$\bar{d} = 4.140499 \text{ min}$$

$$\bar{s} = 2.009887 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $4.140499 = 8.160273 - (2.009887 * 2)$

Nel centro **Inoculazione attenuato**, centro con coda singola e servente multiplo (2 serventi per questa configurazione) si ha:

$$\bar{w} = 2.829814 \text{ min}$$

$$\bar{d} = 0.051900 \text{ min}$$

$$\bar{s} = 1.388957 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $0.051900 = 2.829814 - (1.388957 * 2)$

Il successivo controllo effettuato riguarda il numero di job in coda e all'interno del sistema. In particolare si vuole verificare la seguente uguaglianza:

$$\boxed{\bar{q} = \bar{l} - \bar{y}}$$

Nel centro **Accettazione** si ha:

$$\bar{q} = 6.957809 \text{ job}$$

$$\bar{l} = 7.844163 \text{ job}$$

$$\bar{y} = 0.886354 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $6.957809 = 7.844163 - 0.886354$

Nel centro **Anamnesi** si ha:

$$\bar{q} = 2.057999 \text{ job}$$

$$\bar{l} = 3.574441 \text{ job}$$

$$\bar{y} = 0.758221 * 2 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $2.057999 = 3.574441 - (0.758221 * 2)$

Nel centro **Inoculazione** si ha:

$$\bar{q} = 1.477867 \text{ job}$$

$$\bar{l} = 2.910998 \text{ job}$$

$$\bar{y} = 0.716565 * 2 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $1.477867 = 2.910998 - (0.716565 * 2)$

Nel centro **Inoculazione attenuato** si ha:

$$\bar{q} = 0.005176 \text{ job}$$

$$\bar{l} = 0.282058 \text{ job}$$

$$\bar{y} = 0.138441 * 2 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $0.005176 = 0.282058 - (0.138441 * 2)$

#### 2.4.2 Confronto statistiche di simulazione con valori teorici

Accettazione	Analisi	Simulazione
$\lambda$	0.456000	$0.456417 \pm 0.001755$
$E(S)$	1.941748	$1.942852 \pm 0.007626$
$\rho$	0.885437	$0.886354 \pm 0.004775$
$E(T_Q)$	15.007405	$15.193973 \pm 1.036312$
$E(T_S)$	16.949153	$17.136825 \pm 1.040625$
$E(N)$	7.728813	$7.844163 \pm 0.492143$
$E(N_Q)$	6.843376	$6.957809 \pm 0.488581$

Table 2.5: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Accettazione

Anamnesi	Analisi	Simulazione
$\lambda$	0.456000	$0.456410 \pm 0.001755$
$E(S)$	1.666667	$1.661930 \pm 0.006707$
$\rho$	0.760000	$0.758221 \pm 0.004228$
$E(T_Q)$	4.558081	$4.502273 \pm 0.170769$
$E(T_S)$	7.891414	$7.826133 \pm 0.179463$
$E(N)$	3.598484	$3.574441 \pm 0.088296$
$E(N_Q)$	2.078484	$2.057999 \pm 0.081670$

Table 2.6: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Anamnesi

Inoculazione	Analisi	Simulazione
$\lambda$	0.356146	$0.356657 \pm 0.001258$
$E(S)$	2.008032	$2.009887 \pm 0.007261$
$\rho$	0.715153	$0.716565 \pm 0.003388$
$E(T_Q)$	4.204199	$4.140499 \pm 0.122924$
$E(T_S)$	8.220263	$8.160273 \pm 0.131915$
$E(N)$	2.927613	$2.910998 \pm 0.050786$
$E(N_Q)$	1.497308	$1.477867 \pm 0.045625$

Table 2.7: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Inoculazione

Inoculazione Attenuato	Analisi	Simulazione
$\lambda$	0.099854	$0.099686 \pm 0.000408$
$E(S)$	1.388889	$1.387980 \pm 0.004598$
$\rho$	0.138686	$0.138441 \pm 0.000697$
$E(T_Q)$	0.054475	$0.051900 \pm 0.001882$
$E(T_S)$	2.832253	$2.829814 \pm 0.010098$
$E(N)$	0.282811	$0.282058 \pm 0.001476$
$E(N_Q)$	0.00543954	$0.005176 \pm 0.000192$

Table 2.8: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Inoculazione Attenuato

## 2.5 Validazione

La validazione viene utilizzata per dimostrare la correttezza del simulatore in relazione al caso reale. In particolare dimostrare che il sistema si comporta in maniera coerente rispetto al caso di studio preso in esame.

### 2.5.1 Test Effettuati

Per poter confermare la coerenza del sistema rispetto al caso reale e non avendo dati reali sui tempi medi di risposta, di servizio ed attesa del sistema reale sono stati eseguiti due test. Il primo test prevede l'aumento del Lambda e la verifica che all'aumentare del tasso di arrivo e quindi del numero di pazienti che visitano il centro i tempi di risposta aumentino così come l'utilizzazione dei diversi centri. Il tasso di arrivo iniziale è uguale a  $0.456 \text{ job/min}$  ovvero nel centro nelle 6 ore di apertura entrano 164 persone. Volendo aumentare tale numero e far entrare nel centro 174 persone il tasso di arrivo è pari a  $0.483 \text{ job/min}$ .

I barplot seguenti mostrano la variazione del tempo medio di risposta:

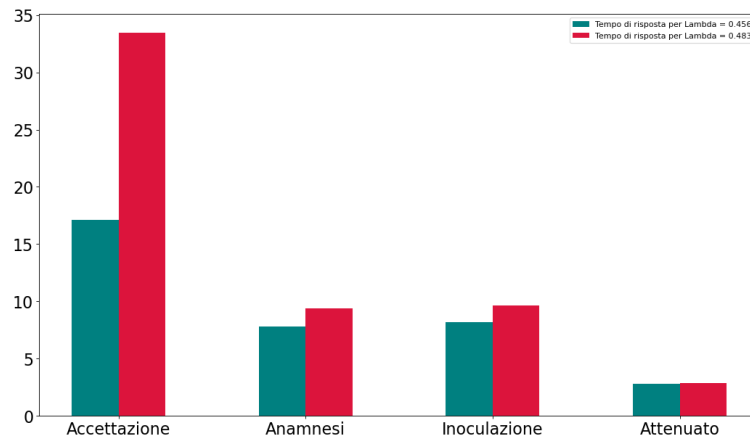


Figure 2.1: Variazione del tempo medio di risposta al variare del tasso di arrivo per tutti i centri del sistema

Per quanto riguarda l'utilizzazione l'aumento di quest'ultima all'aumentare del tasso di ingresso viene mostrato dai barplot che seguono:

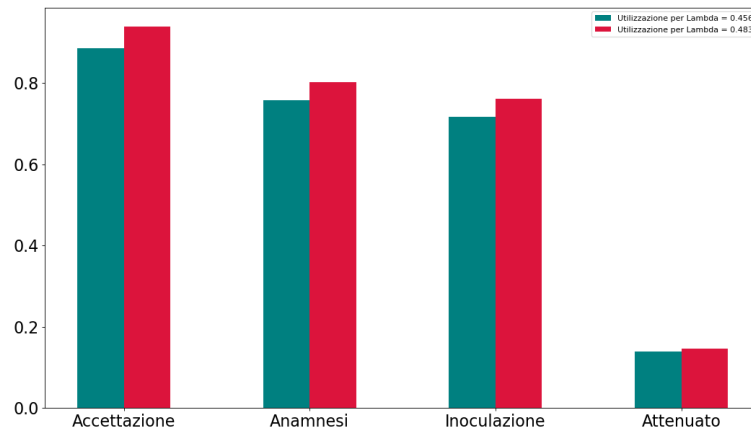


Figure 2.2: Variazione dell'utilizzazione al variare del tasso di arrivo per tutti i centri del sistema

Il test ha confermato l'ipotesi iniziale: nei due grafici presentati in figura 2.1 e 2.2 per i barplot di destra che indicano la statistica, rispettivamente tempo medio di risposta ed utilizzazione avendo applicato un  $\lambda$  pari a 0.483 job/min, l'altezza è maggiore indice dell'aumento del valore della statistica all'aumentare del tasso di ingresso del sistema.

Un ulteriore test viene effettuato sul numero di server scelto per i centri multiservente. Quello che si aspetta è la diminuzione del tempo medio di risposta del centro all'aumentare del numero di server. Il punto di partenza è stata la configurazione non ottimale di server così definita  $\{1,2,2,2\}$  quindi un server per il centro di accettazione, 2 per i rimanenti centri. La modifica è stata effettuata sul centro di anamnesi in cui è stato maggiorato il numero di server di una unità ottenendo quindi la configurazione  $\{1,3,2,2\}$ .

Il grafico mostra i risultati ottenuti:

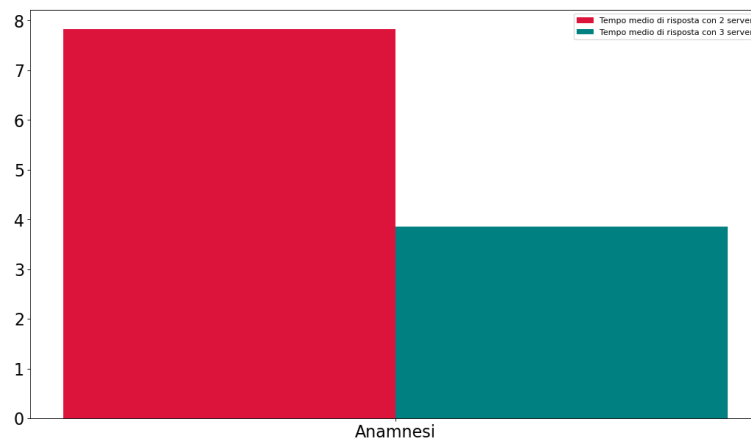


Figure 2.3: Valutazione della coerenza del centro di anamnesi aumentando il numero di server



Anche questo test può quindi confermare l'ipotesi poiché il tempo medio di risposta applicando al sistema la configurazione  $\{1,3,2,2\}$  risulta inferiore al tempo medio di risposta che si ottiene con la configurazione  $\{1,2,2,2\}$ .

## 2.6 Simulazione ad orizzonte infinito

La simulazione ad orizzonte infinito è stata utilizzata per verificare il comportamento del modello in regime stazionario.

E' stato utilizzato il metodo delle batch means con il quale si procede suddividendo i dati di simulazione in batch o gruppi, e calcolando le statistiche per ogni batch per ottenere una stima a lungo termine del sistema.

In particolare per utilizzare tale metodo è necessario scegliere i seguenti parametri:

- Numero di batch : il numero di batch in cui i dati di simulazione devono essere suddivisi. Per questa simulazione è stato scelto  $K = 128$ .
- Dimensione del batch: La dimensione di ogni batch, ovvero il numero di dati che compongono ogni batch. Tale dimensione può influire sul numero di medie calcolate e, di conseguenza, sulla precisione della stima del comportamento a lungo termine del sistema. Per questa simulazione è stato scelto  $B = 2048$ .

I valori di  $K$  e  $B$  sono stati definiti andando a verificare come al variare della grandezza del batch e della numerosità dei batch la statistica sul tempo medio di risposta del sistema convergesse al tempo medio di risposta del sistema ottenuto dall'analisi teorica. I valori scelti sono stati quelli che hanno garantito una convergenza migliore senza peggiorare troppo l'onere computazionale in termini di tempi di esecuzione. E' stato inoltre verificata l'autocorrelazione per i valori di  $K$  e  $B$  scelti in particolare è stato verificato che questa fosse inferiore a 0.2 per  $LAG = 1$ . A parità di convergenza è stata scelta la dimensione che ha garantito una autocorrelazione più bassa. Graficamente:

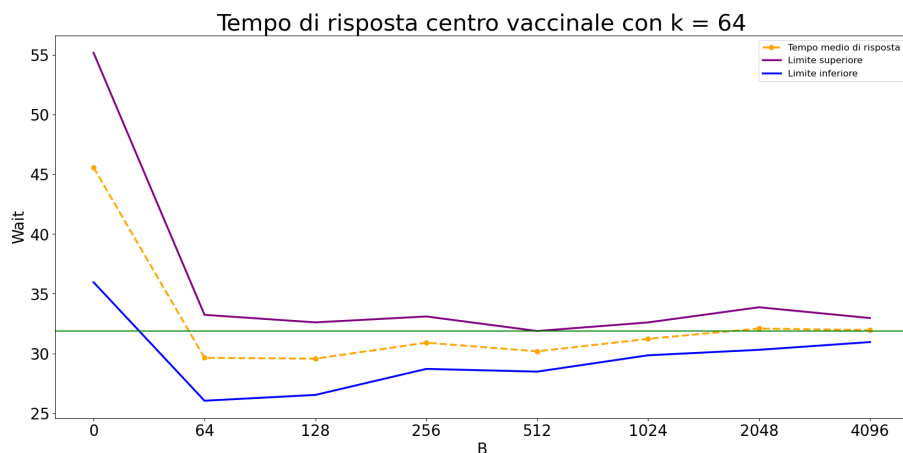


Figure 2.4: Andamento del tempo medio di risposta al variare della grandezza del batch

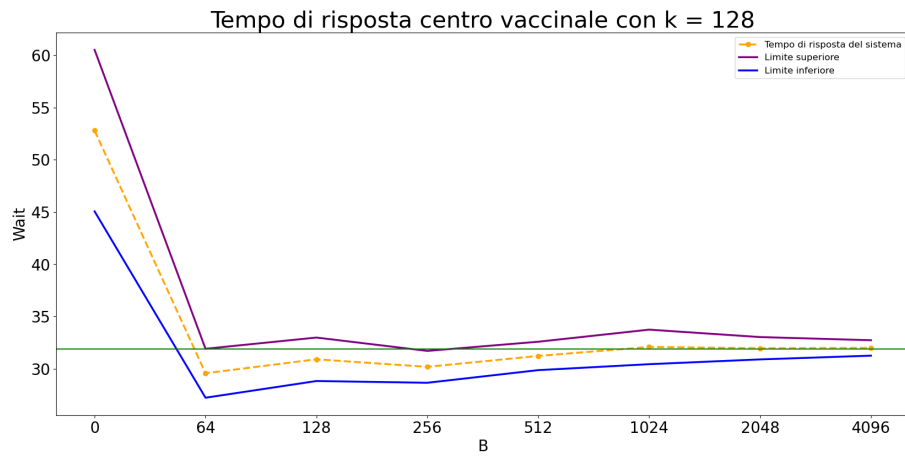


Figure 2.5: Andamento del tempo medio di risposta al variare della grandezza del batch

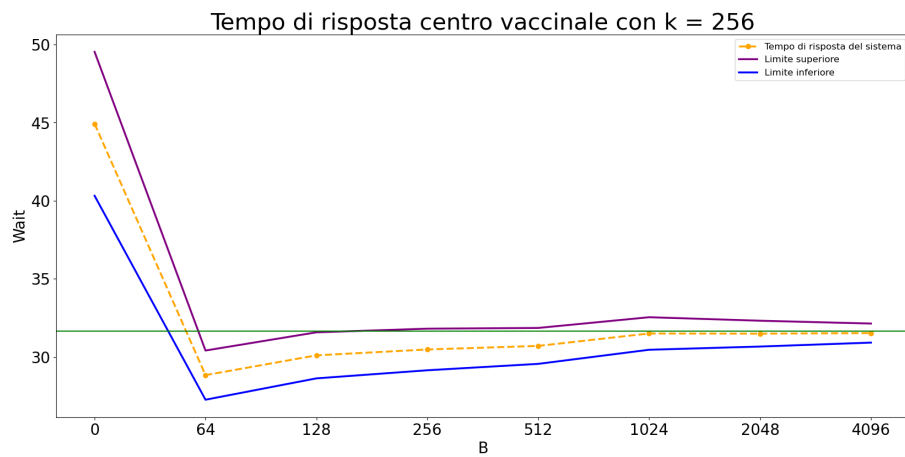


Figure 2.6: Andamento del tempo medio di risposta al variare della grandezza del batch

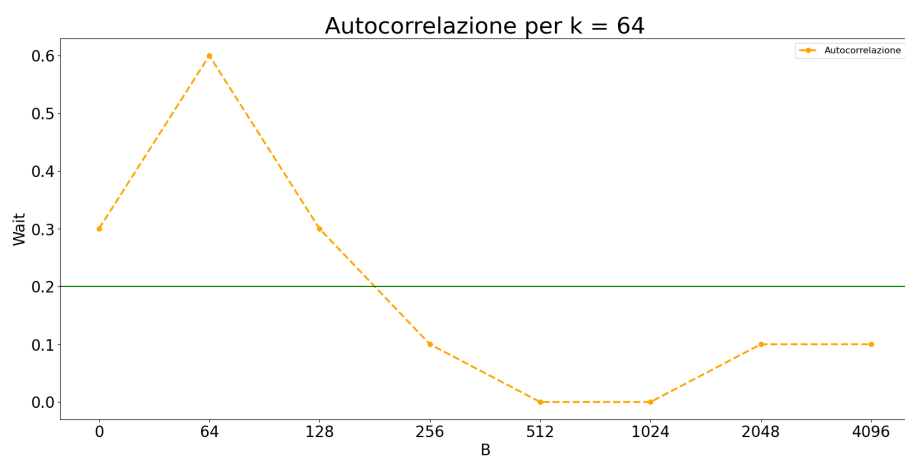


Figure 2.7: Autocorrelazione al variare della dimensionalità del batch

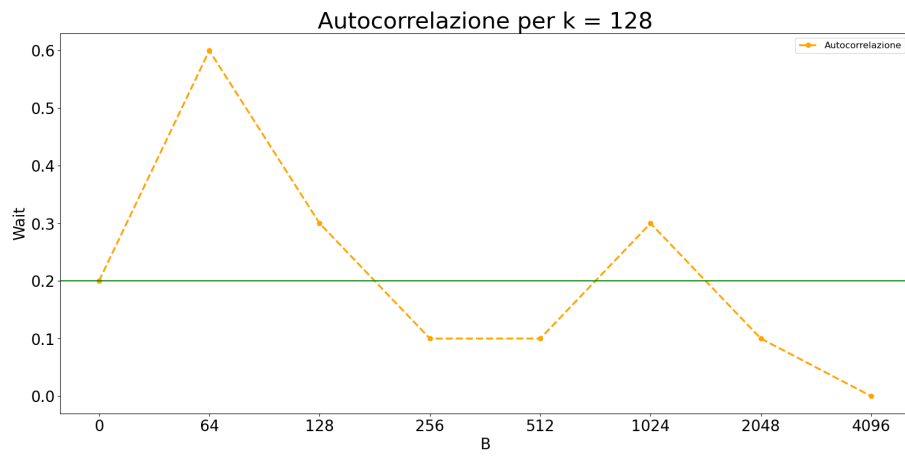


Figure 2.8: Autocorrelazione al variare della dimensionalità del batch

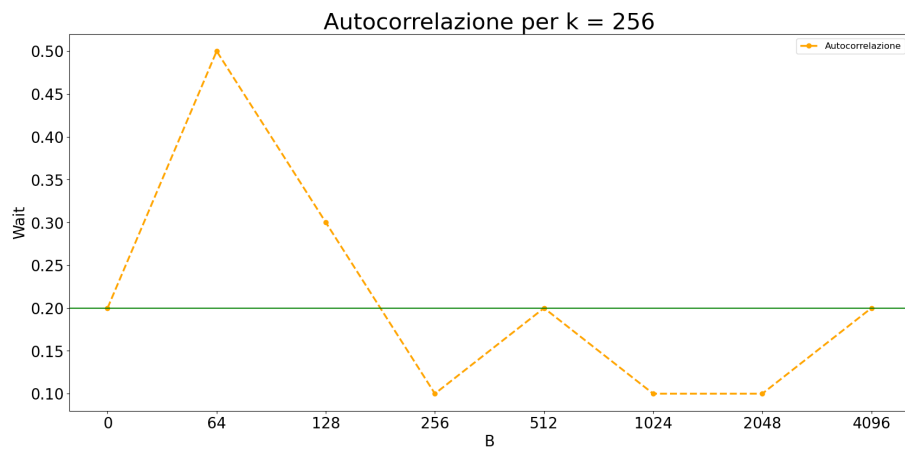


Figure 2.9: Autocorrelazione al variare della dimensionalità del batch

Inoltre al termine di ogni replica per la simulazione ad orizzonte finito e al termine del batch per la simulazione ad orizzonte infinito, vengono stimate le statistiche per ogni centro con la funzione "estimate".

### 2.6.1 Esperimenti

La simulazione ad orizzonte infinito è stata utilizzata per determinare la configurazione ottima ovvero la configurazione che garantisce il soddisfacimento del QoS. Una volta ottenuta tale configurazione si è poi andati a verificare che tale configurazione fosse ottima anche per la simulazione ad orizzonte finito.

La configurazione  $\{1,2,2,2\}$  utilizzata fino a questo punto risulta essere la configurazione realmente utilizzata tuttavia per il rispetto della stazionarietà del sistema si dimostra in Appendice che la configurazione minima risulta essere  $\{1,2,2,1\}$ . Per la valutazione del QoS viene utilizzata la configurazione di partenza  $\{1,2,2,2\}$ .

Questa configurazione tuttavia non garantisce il rispetto del QoS infatti fa ottenere un tempo di risposta medio del sistema pari a  $31.880974 \text{ min}$  con un limite massimo imposto dal QoS di 25 minuti.

Dal grafico che segue si può notare il mancato rispetto del vincolo e la convergenza dei valori della simulazione ai valori teorici.

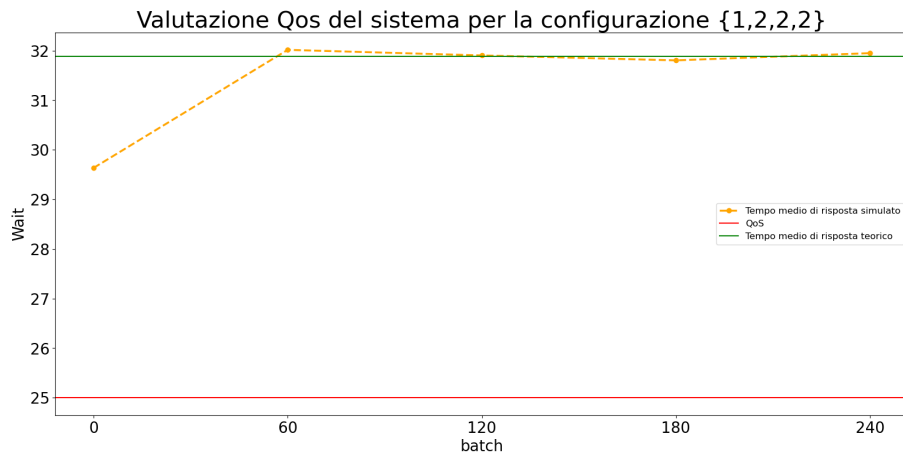


Figure 2.10: Grafico del QoS non rispettato

E' necessario quindi comprendere quale modifica sulla configurazione è bene effettuare, quindi di quanto aumentare i server nei centri multiservente.

La prima configurazione che viene testata partendo da quella iniziale  $\{1,2,2,2\}$  è la seguente  $\{1,3,2,2\}$  dal momento che il centro di Anamnesi risulta essere il centro con utilizzazione maggiore dopo il centro di Accettazione che essendo centro coda singola servente singolo non può al momento subire modifiche.

L'aumento del numero di serventi di una unità nel centro di anamnesi permette di ottenere il seguente tempo medio di risposta del sistema:

$$E(T_s) = 27.552782 \text{ min}$$

inferiore rispetto al tempo medio di risposta che si otteneva con la configurazione  $\{1,2,2,2\}$  ma non ancora sufficientemente inferiore per il rispetto del QoS.

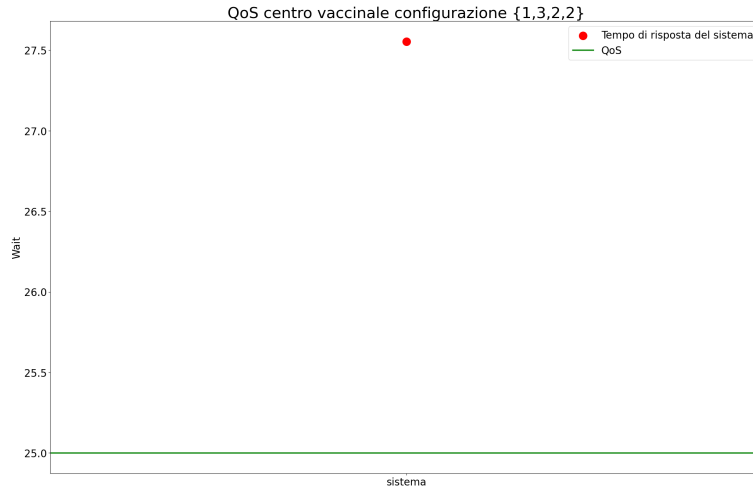


Figure 2.11: Grafico del QoS non rispettato

La seconda configurazione testata è la seguente  $\{1,3,3,2\}$ , viene aumentato quindi di una unità il numero di server del centro di inoculazione poiché risulta essere il centro con l'utilizzazione più alta (0.717447).

Effettuando questa modifica si ottiene il seguente tempo di risposta globale medio del sistema:

$$E(T_s) = 24.975333min$$

La configurazione  $\{1,3,3,2\}$  risulta quindi ottima.

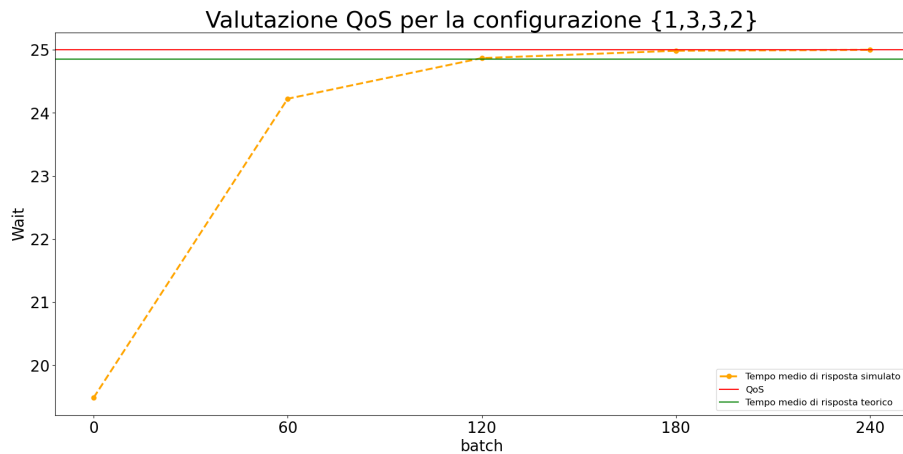


Figure 2.12: Grafico del QoS rispettato

Il grafico in figura 2.12 mostra il rispetto del QoS imposto.

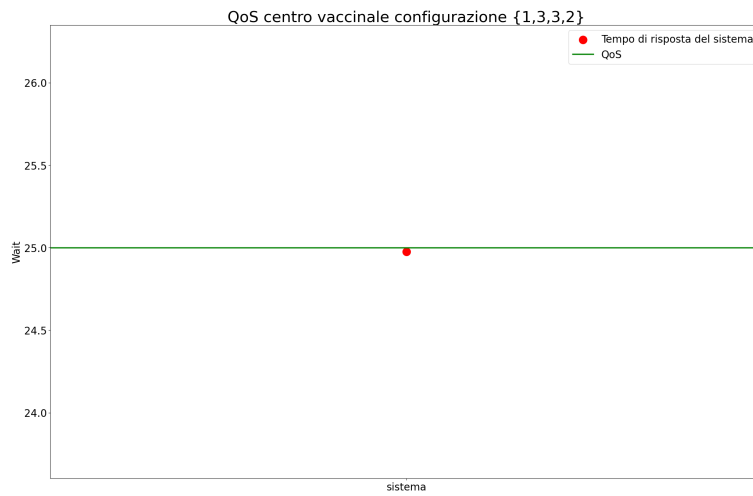


Figure 2.13: Grafico del QoS rispettato

Ottenuta la configurazione si è andati a valutare anche la statistica sul numero medio di job nel sistema, nel grafico che segue per tutti i centri i valori della simulazione convergono ai valori teorici.

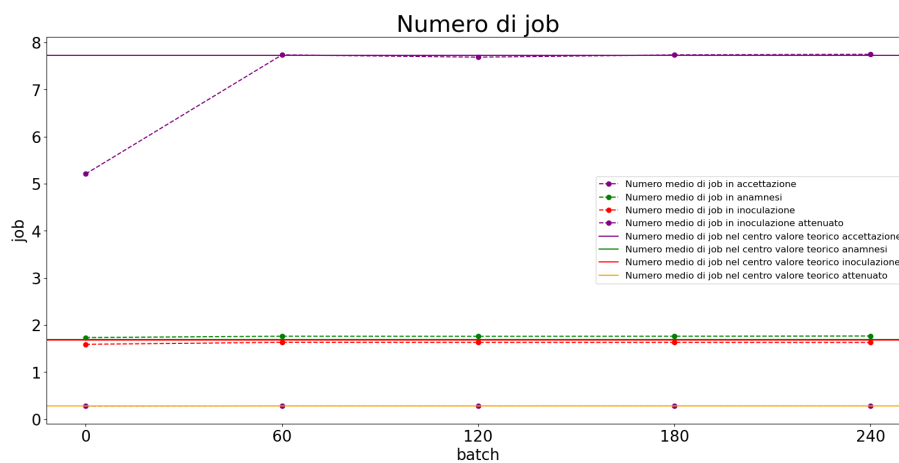


Figure 2.14: Grafico del numero medio di job nel sistema

## 2.7 Simulazione orizzonte finito

Per l'analisi dello stato transiente è stata adottata la tecnica *replication*, dove la run è stata eseguita 128 volte in modo tale da poterne derivare le statistiche mediate sulle 128 repliche. Ogni replica misura in maniera indipendente le stesse metriche, tale indipendenza viene garantita evitando la sovrapposizione tra le repliche effettuando la chiamata alla funzione PlantSeeds() una sola volta all'inizio della simulazione. Il sistema risulta inoltre idle all'inizio ed alla fine della simulazione ed è stato simulato per un tempo pari a 360 minuti ovvero il tempo di reale apertura del centro vaccinale.

### 2.7.1 Esperimenti

Una volta ottenuta dalla simulazione ad orizzonte infinito la configurazione ottima viene verificato il rispetto del vincolo sul tempo imposto anche nella simulazione ad orizzonte finito.

Il tempo medio di risposta globale del sistema che si ottiene è il seguente:

$$E(T_s) = 21.640407 \pm 1.344313min$$

Il tempo di risposta medio resta quindi al di sotto del limite imposto dal QoS. Graficamente:

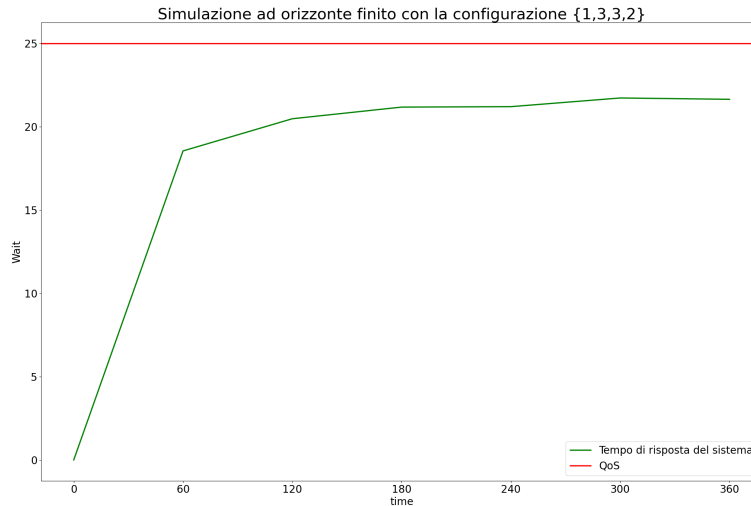


Figure 2.15: Grafico del QoS rispettato nella simulazione ad orizzonte finito

Il QoS viene quindi rispettato anche nella simulazione ad orizzonte finito si vuole inoltre comprendere quale centro causa un aumento dei tempi di risposta del sistema, quale dei quattro centri quindi contribuisce maggiormente all'aumento dei tempi di risposta del sistema. Graficamente:

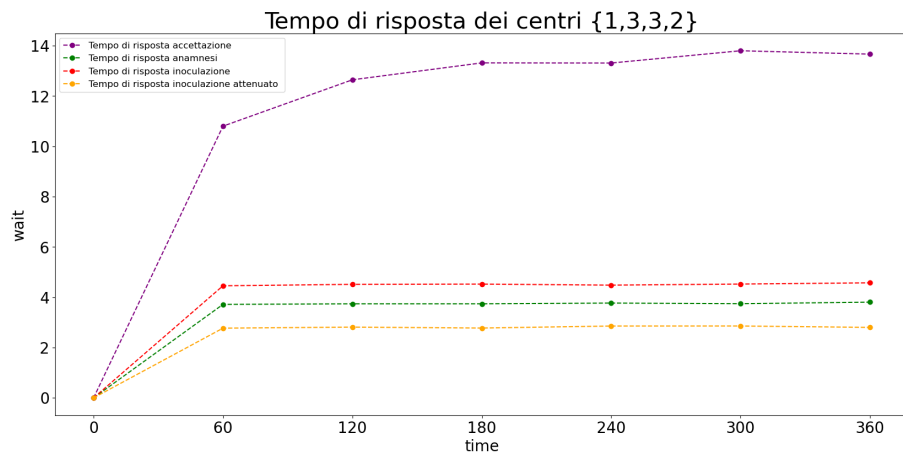


Figure 2.16: Tempi di risposta dei centri nella simulazione ad orizzonte finito

Il centro che "soffre" maggiormente e che registra tempi di risposta maggiori è il centro accettazione. Questo dato non stupisce poiché il centro accettazione riceve la totalità del flusso di ingresso (0.456 job/min) ma ha una singola coda ed un singolo servente. Nonostante abbia il tasso di servizio più alto tra tutti i centri (0.515 job/min) non riesce a smaltire velocemente le code che vengono a formarsi. Per questa ragione nel modello migliorativo la modifica ha interessato proprio questo centro.



# Chapter 3

## Sistema Migliorato

### 3.1 Modifiche apportate

Per il modello migliorativo visto il grande afflusso di pazienti in accettazione e il poco afflusso di pazienti nel centro di inoculazione attenuato sono state effettuate le seguenti modifiche:

- Il centro di accettazione diventa un centro con coda singola a servente multiplo in questo modo si riducono i tempi di attesa in coda e di risposta del centro e di conseguenza i tempi di attesa in coda e risposta globali del sistema.
- Visto il basso afflusso di pazienti nel centro di inoculazione attenuato non risulta necessario avere 2 serventi per la somministrazione del vaccino ma è sufficiente un solo servente per gestire le poche richieste durante l'orario di apertura del centro. Viene reso quindi centro con coda singola e servente singolo.

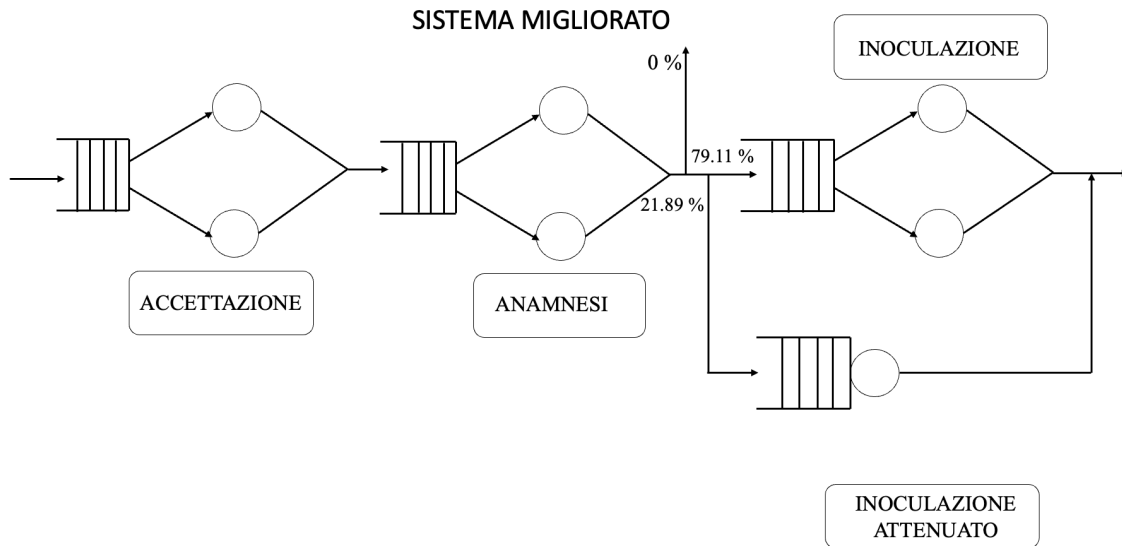
Queste modifiche non aumentano il numero di serventi utilizzati poiché nell'ipotesi che il personale infermieristico possa, per la sola giornata dell'open day, svolgere anche servizi di accettazione, il servente non utilizzato nel centro di inoculazione attenuato viene impegnato nel centro di accettazione mantenendo quindi invariato il numero totale di operatori e riducendo al minimo il numero di serventi che risultano essere in numero inferiore rispetto alla configurazione ottima del sistema non migliorativo.

#### 3.1.1 Obiettivi

- Migliorare il tempo medio di risposta rendendolo inferiore a quello ottenuto con la configurazione ottima nel modello base.
- Minimizzare il numero di operatori

## 3.2 Modello Concettuale

I settori vengono quindi così modellati:



## 3.3 Verifica Modello Migliorativo

Come per il modello precedentemente descritto, per ogni statistica ottenuta dal simulatore si è proceduto ad effettuare i controlli di consistenza per le statistiche di output e i confronti tra i valori simulati ed i valori teorici. Per la fase di verifica è stata utilizzata la configurazione {2,2,2,1} configurazione ottima ai fini del QoS come viene mostrato nel paragrafo successivo.

### 3.3.1 Controllo di consistenza per le statistiche di output

Prima uguaglianza verificata:

$$\bar{d} = \bar{w} - \bar{s}$$

Nel centro **Accettazione** si ha:

$$\bar{w} = 2.415474 \text{ min}$$

$$\bar{d} = 0.475132 \text{ min}$$

$$\bar{s} = 0.970171 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $0.475132 = 2.415474 - (0.970171 * 2) \text{ min}$

Nel centro **Inoculazione attenuato**, centro con coda singola e servente multiplo (2 serventi per questa configurazione) si ha:

$$\bar{w} = 3.861350 \text{ min}$$

$$\bar{d} = 1.079800 \text{ min}$$

$$\bar{s} = 2.781550 \text{ min}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $1.079800 = 3.861350 - 2.781550 \text{ min}$

Il successivo controllo effettuato riguarda il numero di job in coda e all'interno del sistema. In particolare si vuole verificare la seguente uguaglianza:

$$\bar{q} = \bar{l} - \bar{y}$$

Nel centro **Accettazione** si ha:

$$\bar{q} = 0.212619 \text{ job}$$

$$\bar{l} = 1.103348 \text{ job}$$

$$\bar{y} = 0.443135 * 2 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $0.217077 = 1.103348 - (0.443135 * 2) \text{ job}$

Nel centro **Inoculazione attenuato** si ha:

$$\bar{q} = 0.107705 \text{ job}$$

$$\bar{l} = 0.384923 \text{ job}$$

$$\bar{y} = 0.277218 \text{ job}$$

Risulta quindi verificata l'uguaglianza precedentemente descritta poiché :  $0.107705 = 0.384923 - 0.277218 \text{ job}$

### 3.3.2 Confronto statistiche di simulazione con valori teorici

Accettazione	Analisi	Simulazione
$\Lambda$	0.456000	$0.456856 \pm 0.002341$
$E(S)$	0.970874	$0.970171 \pm 0.005285$
$\rho$	0.442718	$0.443135 \pm 0.003552$
$E(T_Q)$	0.473360	$0.475132 \pm 0.017607$
$E(T_S)$	2.415108	$2.415474 \pm 0.024745$
$E(N)$	1.101289	$1.103348 \pm 0.013357$
$E(N_Q)$	0.215852	$0.217077 \pm 0.008242$

Table 3.1: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Accettazione

Inoculazione Attenuato	Analisi	Simulazione
$\Lambda$	0.099854	$0.099666 \pm 0.000579$
$E(S)$	2.777777	$2.781550 \pm 0.014181$
$\rho$	0.277372	$0.277218 \pm 0.002356$
$E(T_Q)$	1.066217	$1.079800 \pm 0.028255$
$E(T_S)$	3.843995	$3.861350 \pm 0.037978$
$E(N)$	0.383838	$0.384923 \pm 0.005052$
$E(N_Q)$	0.106466	$0.107705 \pm 0.003109$

Table 3.2: Confronto statistiche tra l'analisi teorica e il simulatore per il centro Inoculazione Attenuato

### 3.4 Simulazione ad orizzonte infinito - Configurazione Ottima

La configurazione  $\{2,2,2,1\}$  utilizzata per la fase di verifica risulta essere anche la configurazione ottima poiché permette di ottenere tempi medi globali di risposta nel limite di 25 min concesso dal QoS. Con questa configurazione e con le variazioni applicate al sistema è possibile infatti ottenere un tempo di risposta medio pari a  $17.420526 \text{ min}$  a fronte dei  $24.975333$  ottenuti con il modello base avendo applicato ad esso la configurazione ottima. Il miglioramento risulta essere quindi evidente. Graficamente:

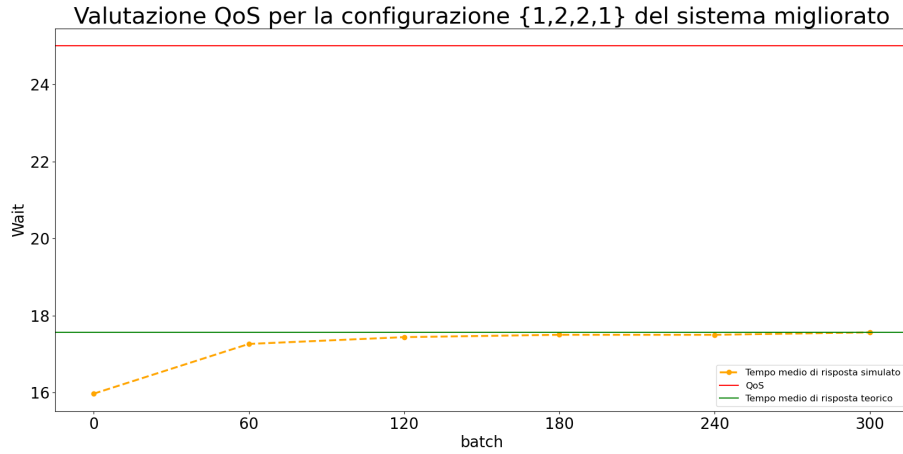


Figure 3.1: QoS rispettato nel modello migliorativo con la configurazione  $\{2,2,2,1\}$

#### 3.4.1 Simulazione ad orizzonte finito

Una volta ottenuta dalla simulazione ad orizzonte infinito la configurazione ottima viene verificato il rispetto del vincolo sul tempo imposto anche nella simulazione ad orizzonte finito in cui la finestra di simulazione è pari a 360 minuti ovvero il tempo di apertura del centro vaccinale nel giorno 17 dicembre.

$$E(T_s) = 16.512028 \pm 0.853413 \text{ min}$$

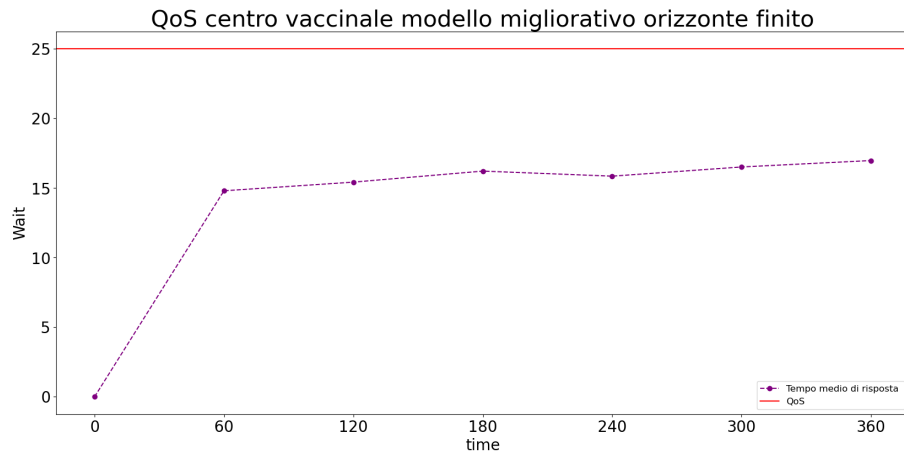


Figure 3.2: QoS rispettato nel modello migliorativo con la configurazione  $\{2,2,2,1\}$  per la simulazione ad orizzonte finito

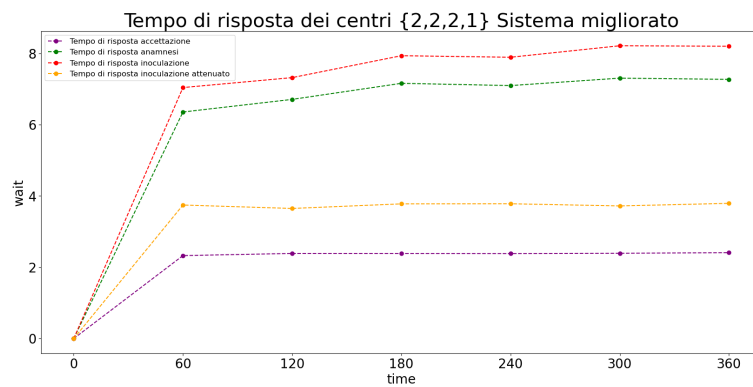


Figure 3.3: Tempi di risposta dei centri all'avanzare della finestra temporale

## 3.5 Confronti

### 3.5.1 Confronto con precedente modello - Simulazione ad orizzonte infinito

Il modello appena descritto con le modifiche effettuate garantisce un miglioramento sulle prestazioni. Tale miglioramento viene per una maggior facilità di lettura rappresentato dal seguente barplot:

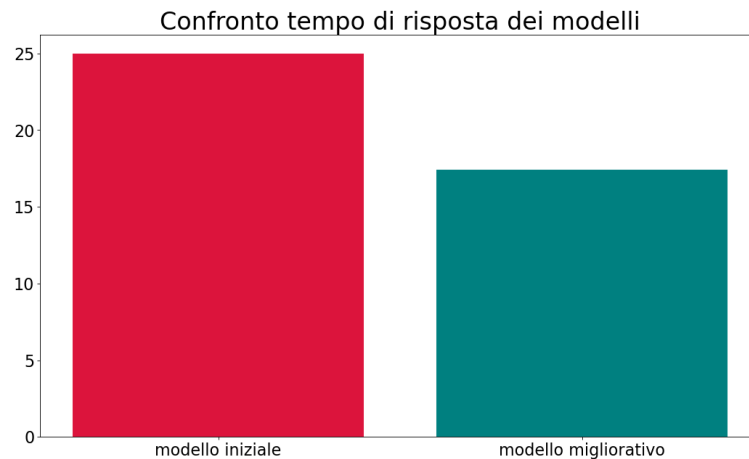


Figure 3.4: Confronto delle statistiche sul tempo medio di risposta tra il modello iniziale e il modello migliorativo

Come è possibile osservare nel modello iniziale il tempo medio di risposta, per la simulazione ad orizzonte infinito, è 24.975333 minuti mentre nel modello migliorativo è 17.420526 minuti. L'aumento del numero di serventi nel solo centro di accettazione senza quindi aumentare il numero di serventi nel centro di anamnesi ed inoculazione come accadeva nella configurazione ottimale del modello precedente, ha permesso una diminuzione del tempo medio di risposta di 7.554807 minuti.

### 3.5.2 Confronto con precedente modello - Simulazione ad orizzonte finito

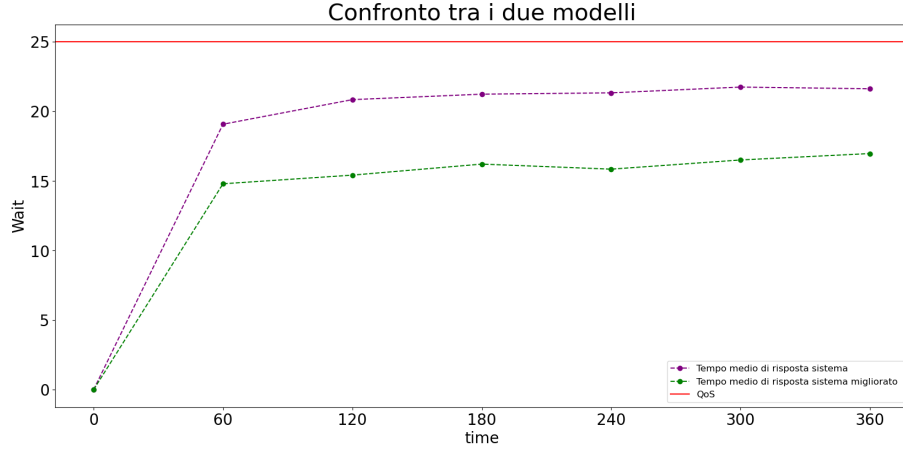


Figure 3.5: Confronto delle statistiche sul tempo medio di risposta tra il modello iniziale e il modello migliorativo per la simulazione ad orizzonte finito

## 3.6 Atro modello migliorativo

Un ulteriore modello migliorativo poteva prevedere la modifica effettuata sul centro di accettazione quindi renderla un centro con coda singola e servente multiplo come è avvenuto nel modello precedentemente descritto ma lasciare invariato il centro di inoculazione attenuato. Visto il poco afflusso di pazienti nel centro di inoculazione attenuato il vantaggio sul tempo medio di risposta del sistema è pressoché irrisorio dal momento che il QoS viene più che garantito anche dal modello sopra descritto. Vengono riportati i dati ottenuti mantenendo invariato rispetto al caso iniziale il centro di inoculazione attenuato.

$$E(T_S)_{globale} = 17.297140 \pm 0.206816min$$

La configurazione precedente garantiva un tempo di risposta medio pari a:

$$E(T_S)_{globale} = 17.420526 \pm 0.209155min$$

Il guadagno quindi in termini di tempo è di  $0.123386 min$ , tuttavia nella configurazione descritta nel modello migliorativo viene utilizzato un servente in meno e nell'ottica in cui si vuole minimizzare il numero di operatori per la giornata del 17 dicembre a parità di tempi medi di servizio risulta essere la scelta migliore.



## .1 Appendice

Affinché un sistema sia stazionario l'utilizzazione dei suoi centri (e quindi del sistema stesso poiché se un solo centro non rispetta la stazionarietà l'intero sistema a sua volta non risulterà stazionario) deve essere inferiore ad 1. L'utilizzazione viene calcolata con la seguente formula:

$$\rho = \lambda * E(s)$$

Il  $\mu$  definito nei paragrafi precedente è il  $\mu$  del singolo server (nel caso di centro multiservente) per poter ottenere quindi il valore di  $E(S)$  viene utilizzata questa formula:

$$E(S) = \frac{1}{m * \mu}$$

con  $m$  = numero di server del centro.

Il centro **Accettazione** è stato modellato con coda singola e servente singolo avendo un tasso di arrivo pari a 0.456 *job/min* e un  $\mu = 0.515$  *job/min* l'utilizzazione risulta essere  $\rho = 0.885436 < 1$

Il centro **Anamnesi** è stato modellato con una singola a servente multiplo è importante quindi comprendere quale è il numero minimo di server per permette di garantire la stazionarietà:

$$\begin{aligned}\lambda &= 0.456 \text{ job/min} \\ \mu &= 0.3 \text{ job/min} \\ \rho &= \frac{0.456}{m * 0.3} < 1 \rightarrow m > 1.52 \rightarrow m = 2\end{aligned}$$

Il centro **Inoculazione** è stato modellato con una singola a servente multiplo è importante quindi comprendere quale è il numero minimo di server per permette di garantire la stazionarietà:

$$\begin{aligned}\lambda &= 0.456 \text{ job/min} \\ \mu &= 0.249 \text{ job/min} \\ \rho &= \frac{0.456}{m * 0.249} < 1 \rightarrow m > 1.83 \rightarrow m = 2\end{aligned}$$

Il centro **Inoculazione Attenuato** è stato modellato con una singola a servente multiplo è importante quindi comprendere quale è il numero minimo di server per permette di garantire la stazionarietà:

$$\begin{aligned}\lambda &= 0.104880 \text{ job/min} \\ \mu &= 0.36 \text{ job/min}\end{aligned}$$

$$\rho = \frac{0.104880}{m * 0.36} < 1 \rightarrow m > 0.291333$$