Performance Modeling Of Computer Systems And Networks

Progetto: Modellazione Ufficio Postale

Alessia Cinelli - Andrea Tozzi

Anno Accademico 2023/2024

Presentazione del Sistema

- È stato modellato un ufficio di Poste Italiane:
  - La filiale Frosinone 3.
- Dopo l’orario di chiusura (4 ore lavorative):
  - Se ci sono, vengono serviti i clienti che sono già in attesa.
- L'ufficio dispone di 6 sportelli interni e di un Bancomat (ATM) esterno.

Raccolta Dati

- Sono stati raccolti dati e informazioni attraverso:
  1. Osservazioni sul campo
     - Numero di sportelli per tipo di operazione
     - Affluenza clienti
     - Tempi di servizio
  2. Richiesta ufficiale alla sede Postale Centrale di Roma.
     - I dati non sono ancora stati ricevuti.

Obiettivi e Scopi

- **Obiettivo 1**: Ridurre il tempo medio di coda a meno di 15 minuti per ciascuna classe d’utenza.
  - QoS aziendale comunicatoci dalla direttrice della filiale.

- **Obiettivo 2**: Ridurre il sovraccarico di lavoro dei dipendenti a meno di 30 minuti giornalieri.
  - Meno del 15% di lavoro giornaliero.

Modello Concettuale (CM)

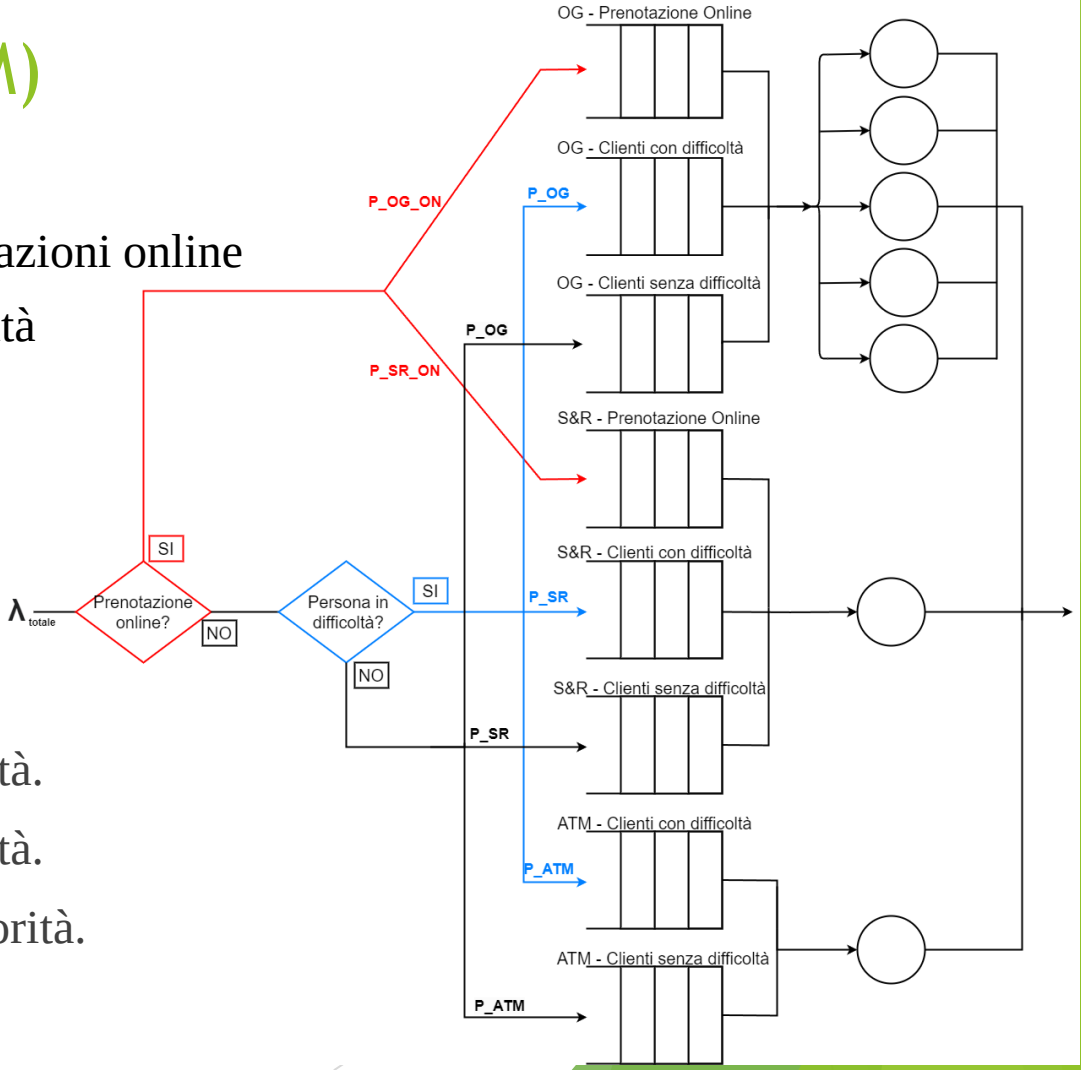
- Distinguiamo i 3 tipi di clienti, con priorità decrescente:
  1) Clienti con Prenotazione Online
  2) Clienti con Difficoltà
  3) Clienti senza Difficoltà

- Il sistema prevede tre tipologie di operazioni diverse:
  1) Operazioni Generiche
  2) Spedizioni e Ritiri
  3) Versamenti e Prelievi

Modello Concettuale (CM)

Sono evidenziati:

- In Rosso il flusso dei clienti con operazioni online
- In Blu il flusso dei clienti con difficoltà



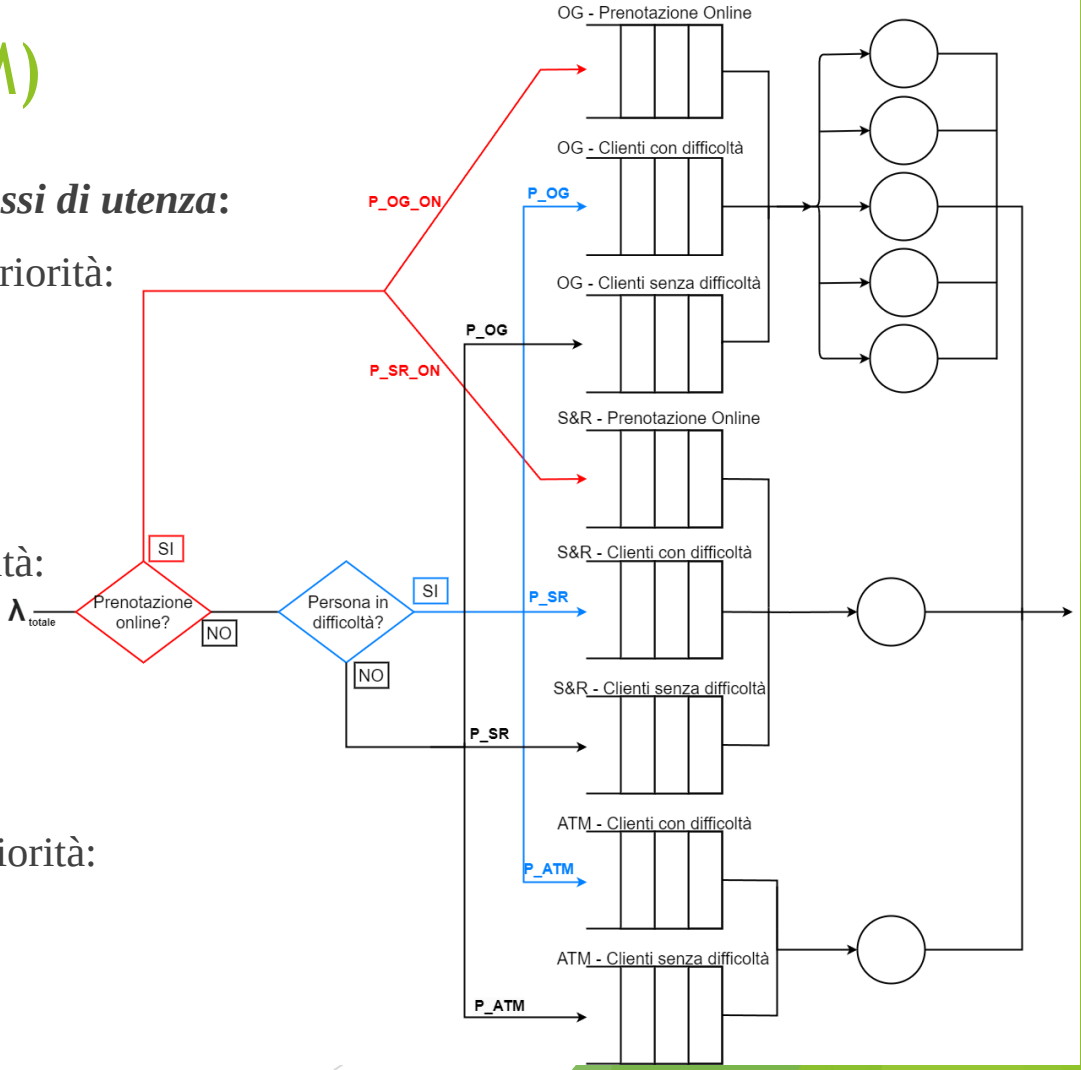
- I centri sono:

1. OG - 5 sportelli con 3 code di priorità.
2. SR - 1 sportello con 3 code di priorità.
3. ATM - 1 sportello con 2 code di priorità.

Modello Concettuale (CM)

Consideriamo **tante code** quante sono le ***classi di utenza***:

- Per Operazione Generiche, in ordine di priorità:
  1. Clienti con prenotazione Online
  2. Clienti con difficoltà
  3. Clienti senza difficoltà
- Per Spedizioni e Ritiri, in ordine di priorità:
  4. Clienti con prenotazione Online
  5. Clienti con difficoltà
  6. Clienti senza difficoltà
- Per Versamenti e Prelievi, in ordine di priorità:
  7. Clienti con difficoltà
  8. Clienti senza difficoltà



Gestione delle Code

- Code gestite con metodologia FIFO, senza prelazione.
- Le code sono assunte come infinite.
- Vale la proprietà del work-conserving:
  - ***Se c'è almeno uno sportello libero (IDLE), allora non devono esserci clienti nelle code da esso servite***.

Stato del Sistema

- Ad ogni istante di tempo lo stato del sistema è determinato da:

1. **Stato degli sportelli:**
   - **IDLE:** Lo sportello è libero e può accogliere un nuovo cliente.
   - **BUSY:** Lo sportello è occupato nel servire un cliente.
2. **Numero di clienti in servizio**, *per ciascuna classe di utenza*.
3. **Numero di clienti in coda**, *per ciascuna classe di utenza*.

Modello delle Specifiche (SM)

- Ad ogni istante di tempo *"t"* lo stato del sistema è univocamente determinato dalle seguenti variabili:

- **Stato del k-esimo sportello**: $Sportello_k(t) \in \{0,1\}$

- **Numero di clienti in servizio** per ogni classe di utenza "c":

  per c ∈ Operazioni Generiche: $InServizio_c(t) \in \{0, 1, \dots, 5\}$

  per c ∈ Spedizioni e Ritiri, e ATM: $InServizio_c(t) \in \{0, 1\}$

- **Numero di clienti in coda** per ogni classe di utenza "c": $Queue_c(t) \in \{0, 1, \dots\}$

Relazioni:

- Dalla somma tra il numero totale di clienti presenti in servizio e il numero di clienti in coda otteniamo il numero di clienti nel sistema:

$$InSistema_c(t) = Queue_c(t) + InServizio_c(t)$$

Assunzioni del modello

- Gli arrivi dei clienti nell'ufficio postale avvengono in modo casuale.
  - **Tempi di inter-arrivo**: *Distribuzione Esponenziale.*

- I tempi di servizio di una operazione si concentrano attorno a un valore medio, con variazioni simmetriche.
  - **Tempi di servizio**: *Distribuzione Normale Troncata.*
  - Il troncamento inferiore è stato effettuato ad un valore di 10^{-6}, necessario per evitare tempi di servizio negativi.

Valori utilizzati

Tasso in ingresso:

- $\lambda_{totale} = \lambda + \lambda_{online} = 0{,}766666\ job/min$
- $\lambda = \frac{1}{1.5}$ job/min
- $\lambda_{online} = 0{,}1\ job/min$

Da qui, ricaviamo le probabilità:

- Probabilità Prenotazione Online: $P_{ON} = \frac{\lambda_{\text{online}}}{\lambda_{\text{totale}}} \cong 0.13$
- Probabilità di non avere prenotazione Online: $P_{N_ON} = \frac{\lambda}{\lambda_{totale}} = 1 - P_{ON} \cong 0.87$

Clienti senza prenotazione
$P_{OG} = 0{,}45$
$P_{SR} = 0{,}20$
$P_{ATM} = 0{,}35$

Clienti con prenotazione
$P_{OG_ON} = 0{,}60$
$P_{SR_ON} = 0{,}40$

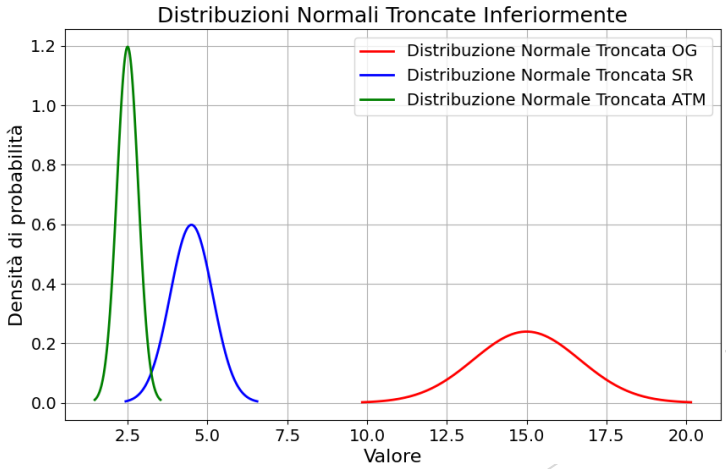
Valori utilizzati

Tassi di servizio:

- Sono differenziati per i tre tipi di operazioni, mantenendoli tali a prescindere se l'operazione è stata prenotata online o meno.

- Circa il 99.7% dei dati è contenuto entro $\mu \pm 3\sigma$.

- $\mu_{OG} = \frac{1}{15}\frac{job}{min}; \quad \sigma = \frac{5}{3}$
- $\mu_{SR} = \frac{1}{4{,}5}\frac{job}{min}; \quad \sigma = \frac{2}{3}$
- $\mu_{ATM} = \frac{1}{2{,}5}\frac{job}{min}; \quad \sigma = \frac{1}{3}$



Modello Computazionale (CPTM)

- Linguaggio: Python 3.12

- Approccio: Next Event Simulation
  - Lo **stato del sistema può cambiare** solo al verificarsi di un **evento.**

- Eventi del sistema:
  - **Arrivo** di un cliente della classe di utenza “c”.
  - **Completamento** di un servizio da parte di uno sportello.

Stato del Sistema (CPTM)

Ad ogni istante di tempo lo ***stato del sitema*** *è definito dalle seguenti strutture dati:*

- $\boldsymbol{server_state}[\boldsymbol{SERVER_NUM}]$: Array binario per tracciare lo stato degli sportelli.

- $\boldsymbol{num_client_in_service}[\boldsymbol{QUEUES_NUM}]$: Array di valori interi per tracciare il numero di clienti in servizio per ciascuna classe di utenza.

- $\boldsymbol{num_client_in_queue}[\boldsymbol{QUEUES_NUM}]$: Array di valori interi per tracciare il numero di clienti in ciascuna coda.

Gestione evento di arrivo

Arrivo di un utente della classe “c”:

- Caso 1: C’è almeno un server libero (IDLE), viene scelto il server che dovrà servirlo seguendo la **politica equity.**
  - $\boldsymbol{num_client_in_service[c]\ \mathrel{+}= 1}$
  - $\boldsymbol{server_state[idle_{id}] = 1}$

- Caso 2: Tutti i server sono BUSY, il cliente viene accodato:
  - $\boldsymbol{num_client_in_queue[c]}\ \mathrel{+}= 1$

**politica equity: seleziona il server libero da più tempo*

Gestione evento di completamento

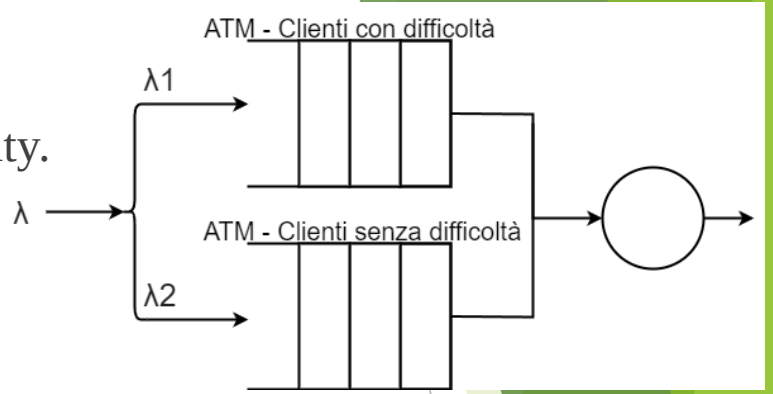
Completamento di un utente della classe “c”:

- Caso 1: Non ci sono clienti da servire (code vuote):
  - $\boldsymbol{num_client_in_service[c]} \ -= 1$
  - $\boldsymbol{server_state[server_{id}] = 0}$

- Caso 2: Ci sono clienti in coda da servire. Viene immediatamente servito il prossimo cliente secondo le regole di scheduling:
  - $\boldsymbol{num_client_in_queue[c]} \ -= 1$

Verifica – Centro ATM

- Modellato come un centro M/M/1 con abstract priority.

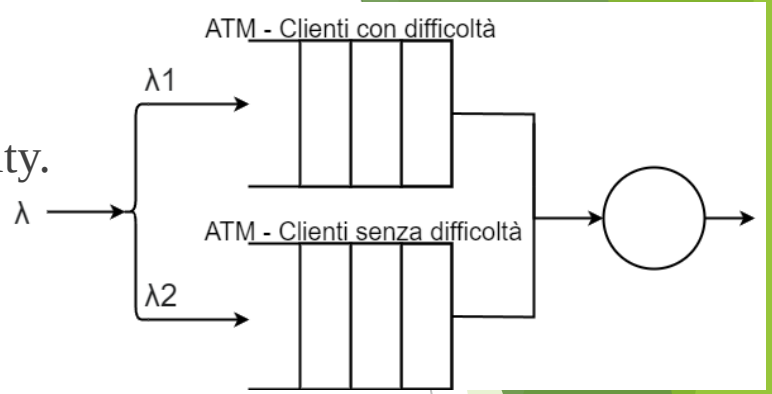


- $E(Tq_k) = \frac{\lambda/_2 E(S^2)}{(1-\sum_{i=0}^{k}\rho_i)(1-\sum_{i=0}^{k-1}\rho_i)}$

- $\lambda_1 = \lambda_{totale} * (1 - P_{ON}) * (P_{diff}) * P_{ATM} = 0{,}034999\ job/min$
- $\lambda_2 = \lambda_{totale} * (1 - P_{ON}) * (1 - P_{diff}) * P_{ATM} = 0{,}198333\, job/min$
- $\lambda = \lambda_1 + \lambda_2 = 0{,}233333\ job/min$

- $\mu = {}^1/_{2,5} = 0{,}4\, job/min$
- $E(S^2) = \frac{2}{\mu^2} = 12{,}499999$
- $\rho_1 = \lambda_1 * E(S) = 0{,}087499$
- $\rho_2 = \lambda_2 * E(S) = 0{,}495832$

Verifica – Centro ATM



- Modellato come un centro M/M/1 con abstract priority.

- $E(Tq_1) = 1{,}5981873\ minuti$

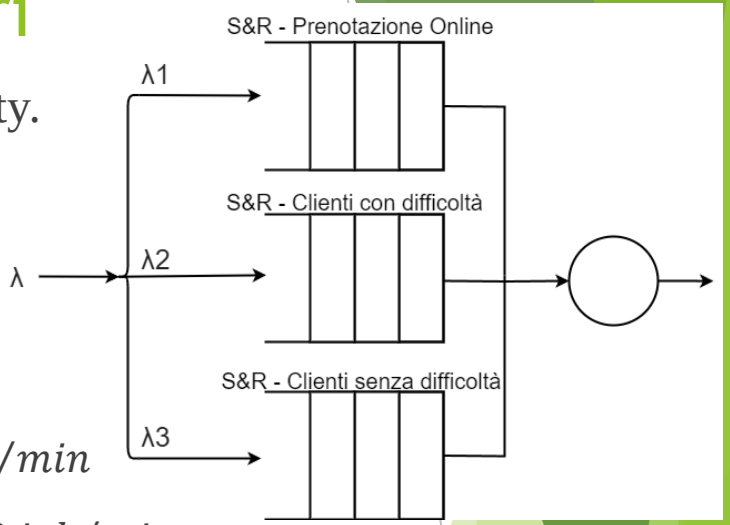
```
based upon 1024 data points and with 95 confidence
the expected value is in the interval       1.59 +/-   0.04
```

- $E(Tq_2) = 3{,}835616\ minuti$

```
based upon 1024 data points and with 95 confidence
the expected value is in the interval       3.76 +/-   0.11
```

Tempo medio in coda	Risultato Analitico	Risultato Simulazione
$E(Tq_1)$	$1{,}5981873\ minuti$	$1{,}59 \pm 0{,}04$ minuti
$E(Tq_2)$	$3{,}835616\ minuti$	$3{,}76 \pm 0{,}11$ minuti

Verifica – Centro Spedizioni e Ritiri



- Modellato come un centro M/M/1 con abstract priority.
- $E(Tq_k) = \frac{\lambda/_2 E(S^2)}{(1-\sum_{i=0}^{k}\rho_i)(1-\sum_{i=0}^{k-1}\rho_i)}$

- $\lambda_1 = \lambda_{totale} * P_{ON} * P_{SR_ON} = 0{,}04\ job/min$
- $\lambda_2 = \lambda_{totale} * (1 - P_{ON}) * (P_{diff}) * P_{SR} = 0{,}019999\, job/min$
- $\lambda_3 = \lambda_{totale} * (1 - P_{ON}) * (1 - P_{diff}) * P_{SR} = 0{,}113333\, job/min$

- $\lambda = \lambda_1 + \lambda_2 + \lambda_3 = 0{,}173333\ job/min$

- $\mu = {}^1/_3 = 0{,}333333\, job/min$
- $E(S^2) = \frac{2}{\mu^2} = 18{,}000036$

- $\rho_1 = \lambda_1 * E(S) = 0{,}12$
- $\rho_2 = \lambda_2 * E(S) = 0{,}0599997$
- $\rho_3 = \lambda_3 * E(S) = 0{,}339999$

Verifica – Centro Spedizioni e Ritiri

- Modellato come un centro M/M/1 con abstract priority
- $E(Tq_1) = 1{,}772727\ minuti$

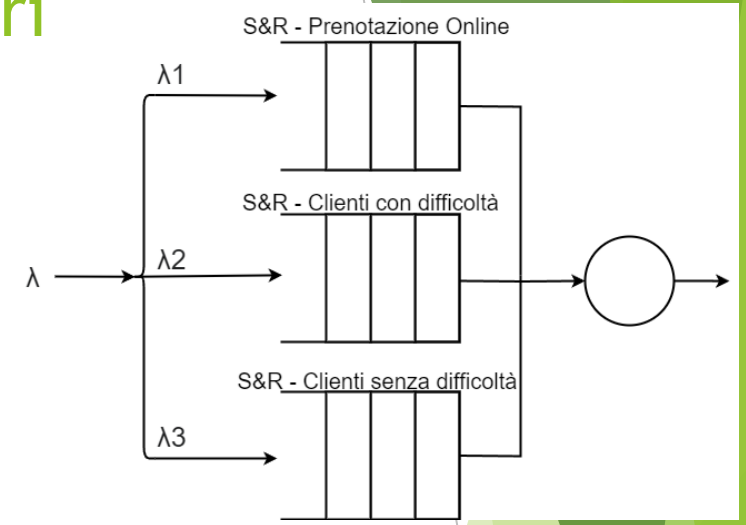
```
based upon 1024 data points and with 95 confidence
the expected value is in the interval       1.79 +/-   0.05
```

- $E(Tq_2) = 2{,}161862\ minuti$

```
based upon 1024 data points and with 95 confidence
the expected value is in the interval       2.17 +/-   0.08
```

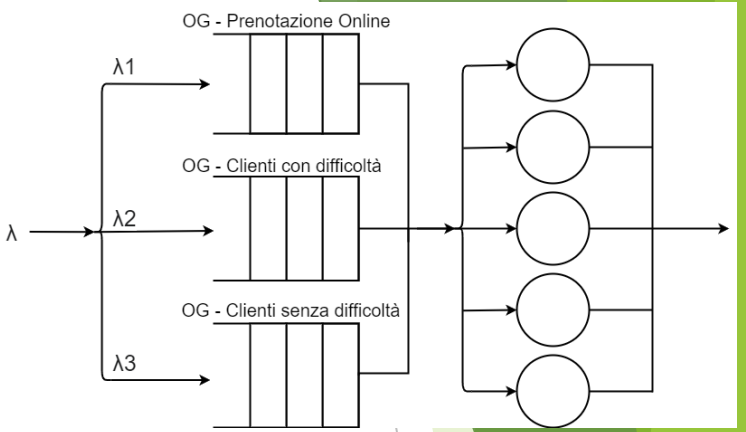
- $E(Tq_3) = 3{,}963414\ minuti$

```
based upon 1024 data points and with 95 confidence
the expected value is in the interval       3.90 +/-   0.13
```



Tempo medio in coda	Risultato Analitico	Risultato Simulazione
$E(Tq_1)$	$1{,}772727\ minuti$	$1{,}79 \pm 0{,}05$ minuti
$E(Tq_2)$	$2{,}161862\ minuti$	$2{,}17 \pm 0{,}08$ minuti
$E(Tq_3)$	$3{,}963414\ minuti$	$3{,}90 \pm 0{,}13$ minuti

Verifica – Centro Multi server



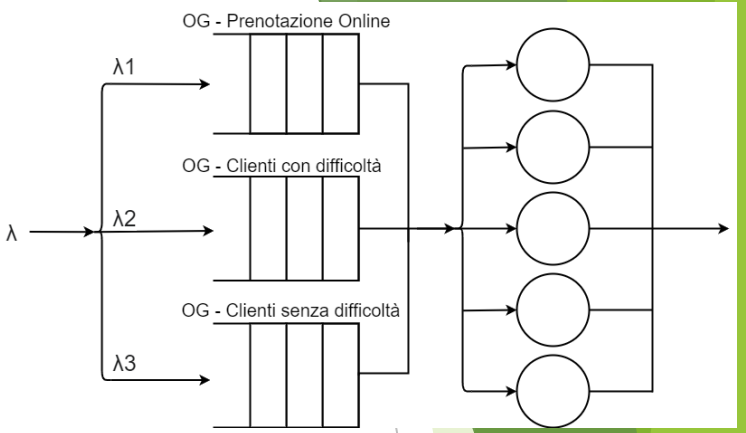
- Modellato come un centro M/M/5 con abstract priority
- Avendo calcolato ciascuna statistica in maniera indipendente dalle altre, possiamo notare:
- $\bar{w} = \bar{d} + \bar{s}\,, \qquad \bar{l} = \bar{q} + \bar{x}$ *sono rispettate*

queue	avg wait t	avg delay t	avg service t	avg # in node	avg # in queue	avg # in service
1	17.48882967	2.92454821	14.56428146	0.51235028	0.08567715	0.42667313
2	17.55592272	2.51365164	15.04227108	0.83576322	0.11966432	0.71609890
3	36.84031140	22.41511116	14.42520024	8.90396558	5.41752690	3.48643867
4	4.86612915	0.46538188	4.40074727	0.08909839	0.00852110	0.08057729
5	5.02660170	0.72095263	4.30564907	0.09203663	0.01320058	0.07883605
6	8.21018426	3.71836718	4.49181707	0.87190094	0.39488125	0.47701969
7	3.09811666	0.86869051	2.22942615	0.11345248	0.03181129	0.08164119
8	3.59748242	1.40050713	2.19697530	0.57965235	0.22565982	0.35399253

```
Simulation complete. Clients served: [8, 13, 66, 5, 5, 29, 10, 44]
last completion:  273.0761402290368
Tempo Totale 04:33:08
```

Verifica – Centro Multi server

- Modellato come un centro M/M/5 con abstract priority
- Estrapoliamo i seguenti dati dalla coda 3:
  - $n = 66$
  - $c_n = 273.07614\ min$
  - $\overline{w} = 36.84031140\ min$
  - $\bar{d} = 22.415111\ min$
  - $\bar{s} = 14.4252\ min$
- *Dal teorema di Little:*
  - $\bar{l} = \frac{n}{c_n} * \overline{w} = 8.90396558$
  - $\bar{q} = \frac{n}{c_n} * \bar{d} = 5.417526904$
  - $\bar{x} = \frac{n}{c_n} * \bar{s} = 3.486438673$



queue	avg wait t	avg delay t	avg service t	avg # in node	avg # in queue	avg # in service
1	17.48882967	2.92454821	14.56428146	0.51235028	0.08567715	0.42667313
2	17.55592272	2.51365164	15.04227108	0.83576322	0.11966432	0.71609890
3	36.84031140	22.41511116	14.42520024	8.90396558	5.41752690	3.48643867
4	4.86612915	0.46538188	4.40074727	0.08909839	0.00852110	0.08057729
5	5.02660170	0.72095263	4.30564907	0.09203663	0.01320058	0.07883605
6	8.21018426	3.71836718	4.49181707	0.87190094	0.39488125	0.47701969
7	3.09811666	0.86869051	2.22942615	0.11345248	0.03181129	0.08164119
8	3.59748242	1.40050713	2.19697530	0.57965235	0.22565982	0.35399253

```
Simulation complete. Clients served: [8, 13, 66, 5, 5, 29, 10, 44]
last completion: 273.0761402290368
Tempo Totale 04:33:08
```

Validazione

- Nella fase di validazione, ci si domanda se il modello computazionale realizzato sia consistente con il sistema che desideravamo rappresentare.
- Variamo il tasso di inter-arrivo dei <u>clienti senza prenotazione</u>.
- Parametro base: $\lambda = \frac{1}{1.5}$ Parametro ridotto: $\lambda = \frac{1}{3}$ Parametro aumentato: $\lambda = 1$

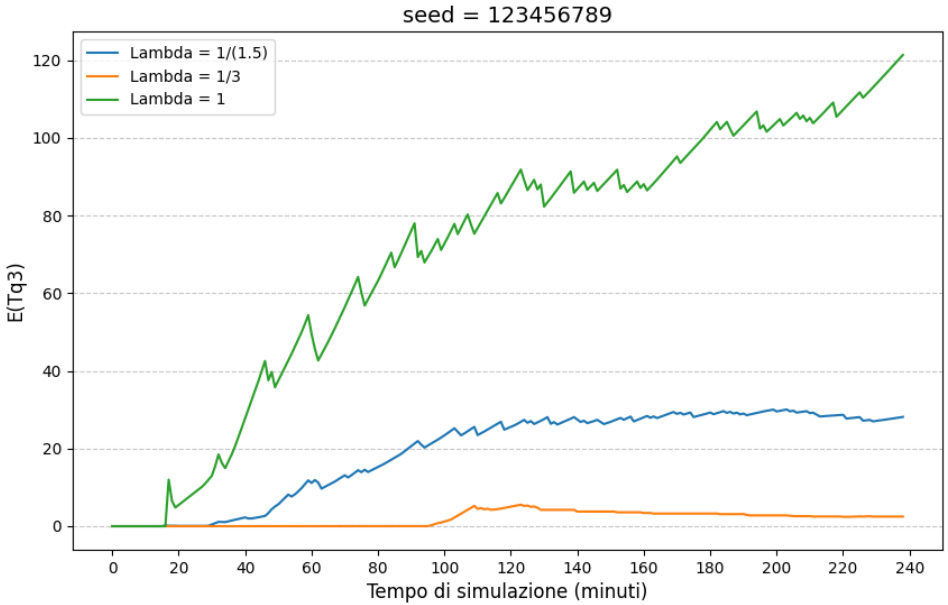


Grafico rappresentato al variare del lambda in ingresso.
Una replica, orizzonte finito

Validazione

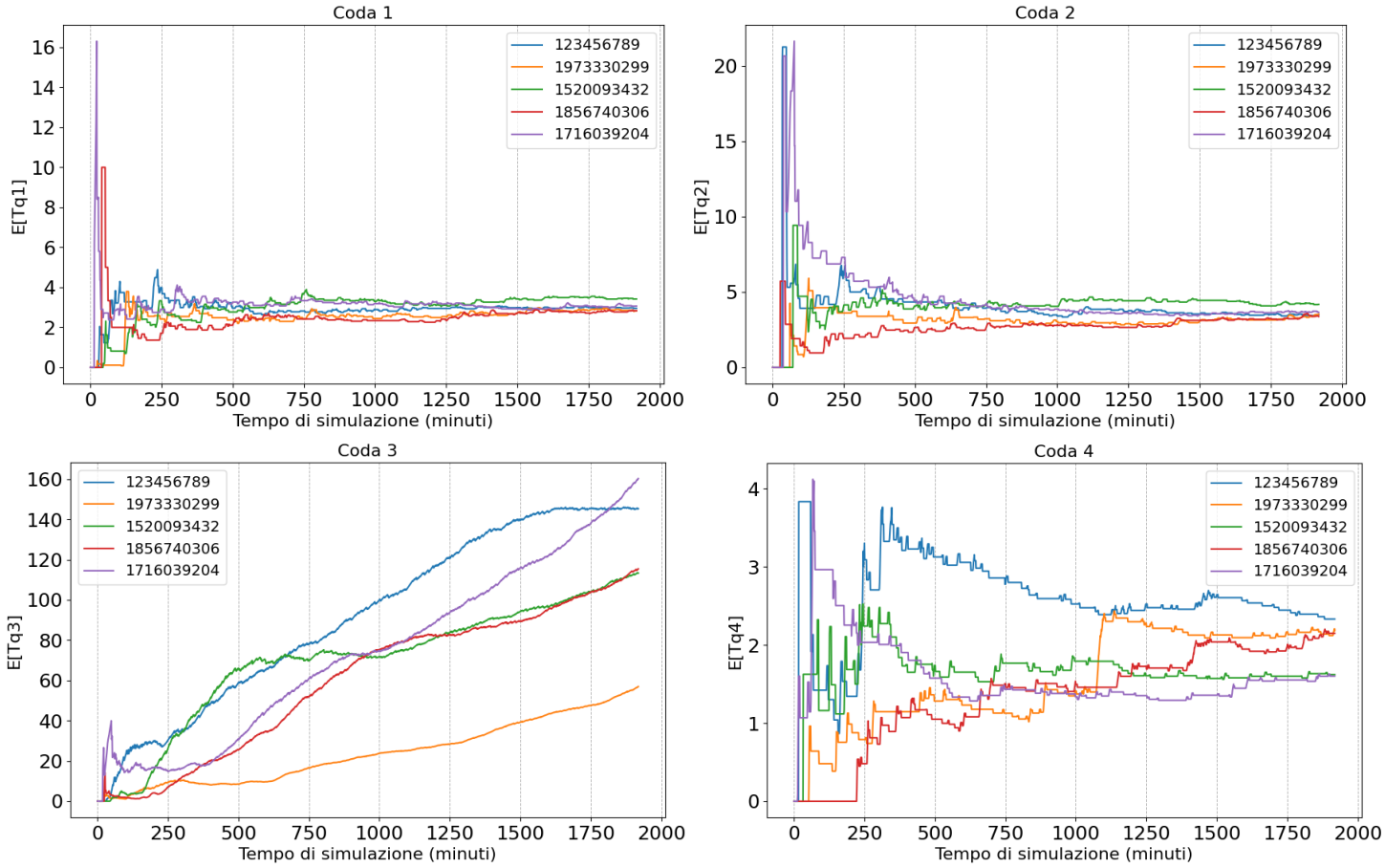
- Variamo il tasso di inter-arrivo (1000 repliche, orizzonte finito):

λ	*E[Tq3]*
Base $= 1/1{,}5\ job/min$	$23{,}081 \pm 0{,}898$ minuti
Ridotto $= 1/3\ job/min$	$1{,}363 \pm 0{,}105$ minuti
Aumentato $= 1\ job/min$	$83{,}853 \pm 1{,}366$ minuti

- Variamo il tasso di inter-arrivo (1000 repliche, orizzonte finito), calcolandola all'ultimo istante di completamento:

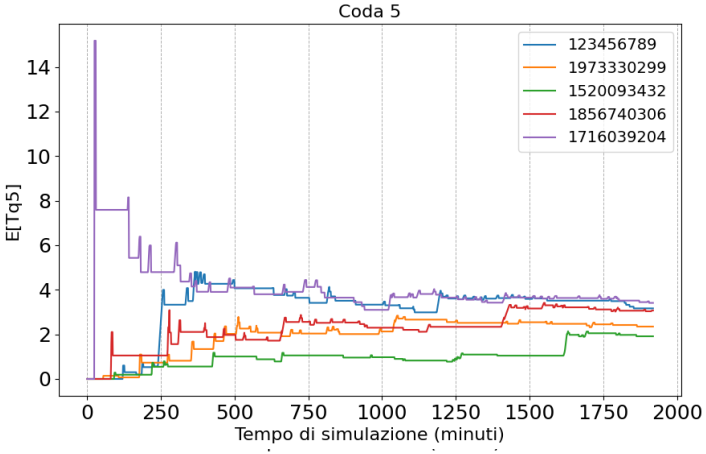
λ	ρ *(multi server)*
Base $= 1/1{,}5\ job/min$	0,844
Ridotto $= 1/3\ job/min$	0,567
Aumentato $= 1\ job/min$	0,925

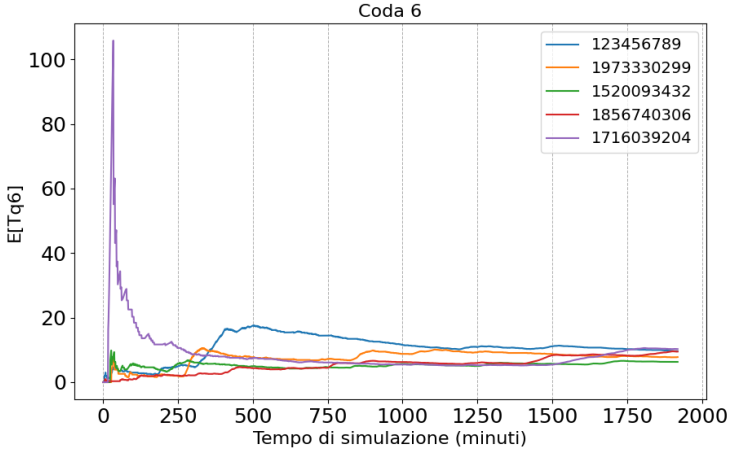
Analisi dello stato transitorio – E[Tq]

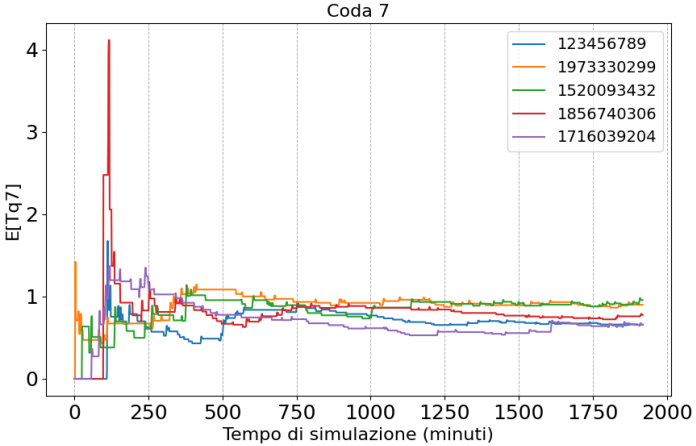


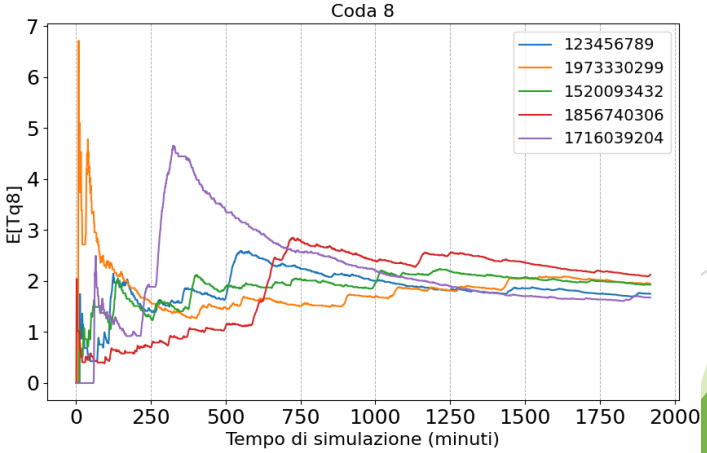
Valutato con 5 repliche, impostando un tempo limite di 1920 minuti (32 ore)

Analisi dello stato transitorio – E[Tq]



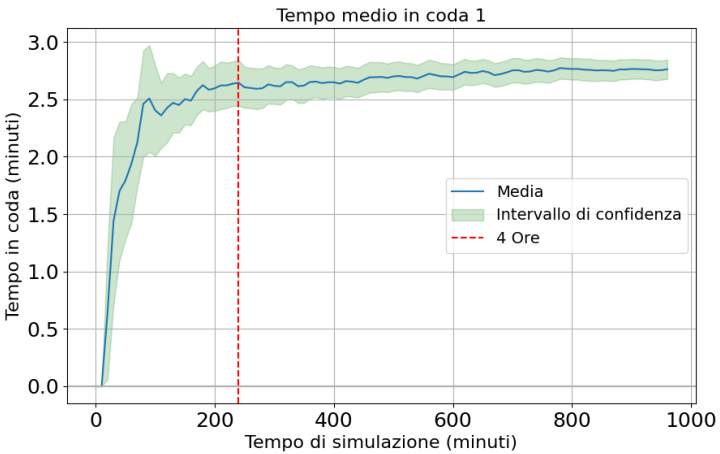


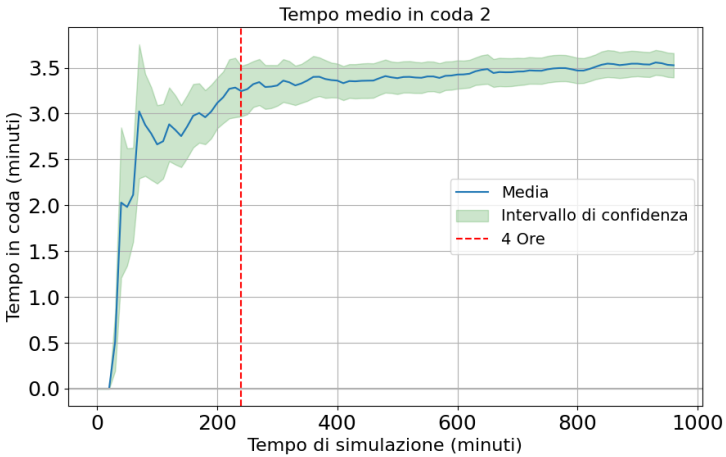


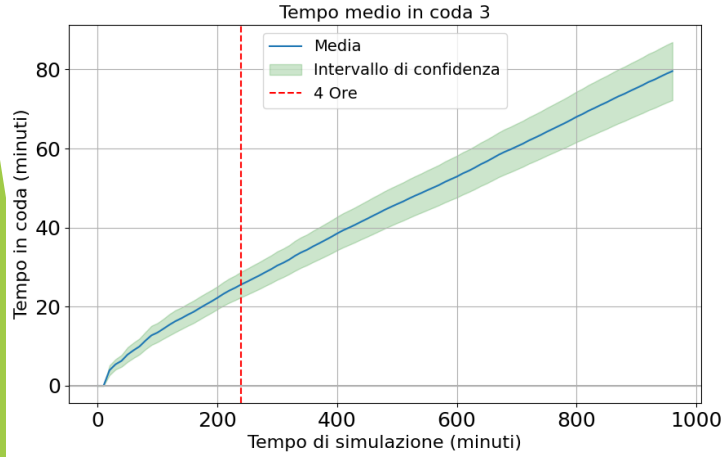


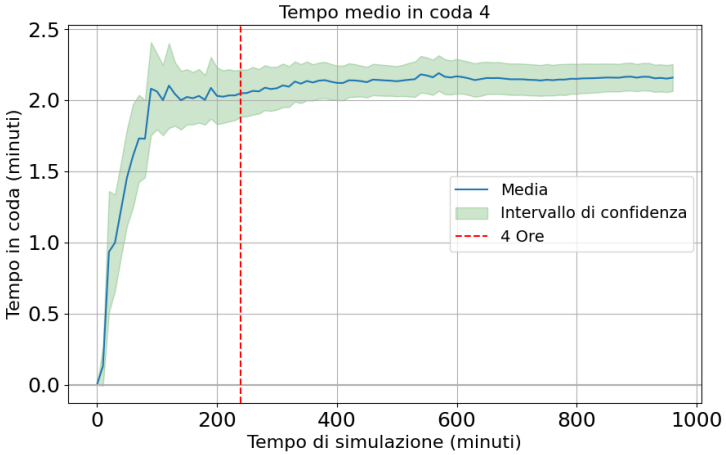
Valutato con 5 repliche, impostando un tempo limite di 1920 minuti (32 ore)

Analisi dello stato transitorio – E[Tq]



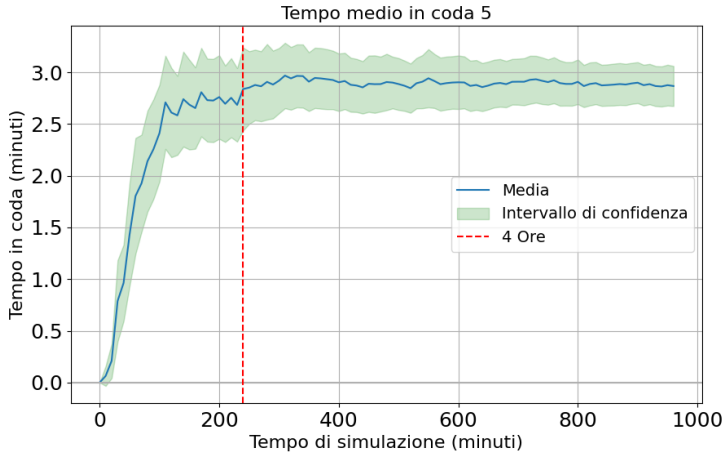


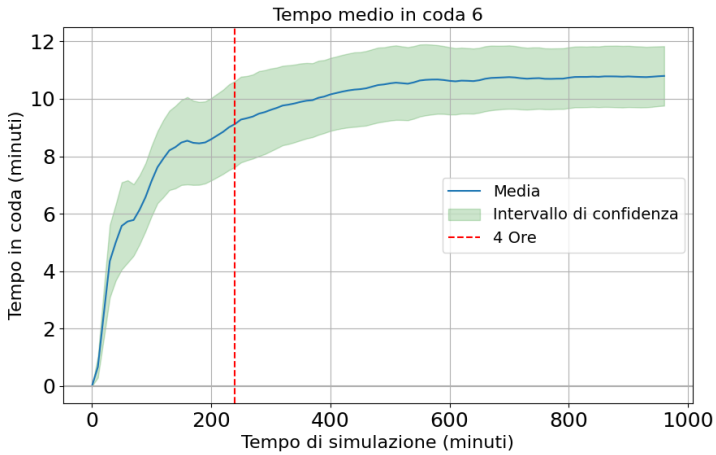


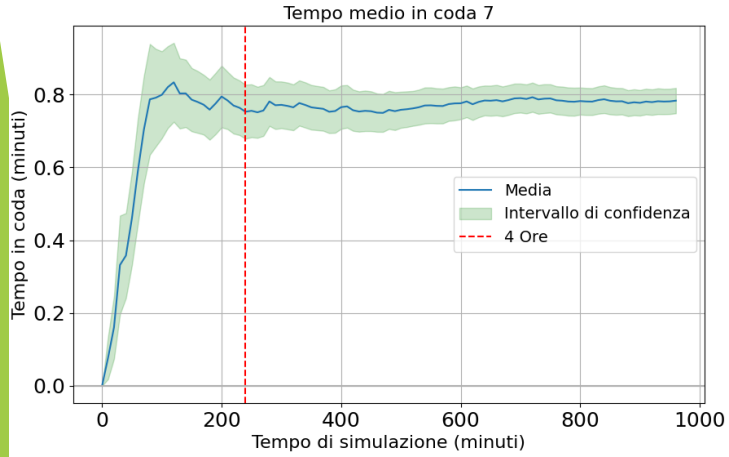


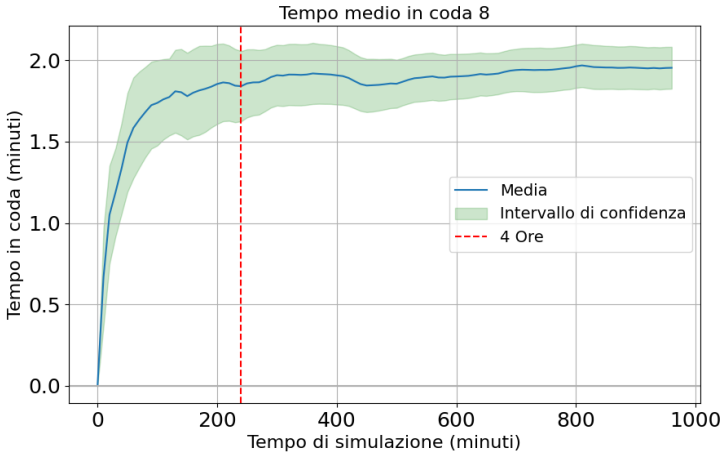
Valutato con 100 repliche, impostando un tempo limite di 960 minuti (16 ore)

Analisi dello stato transitorio – E[Tq]



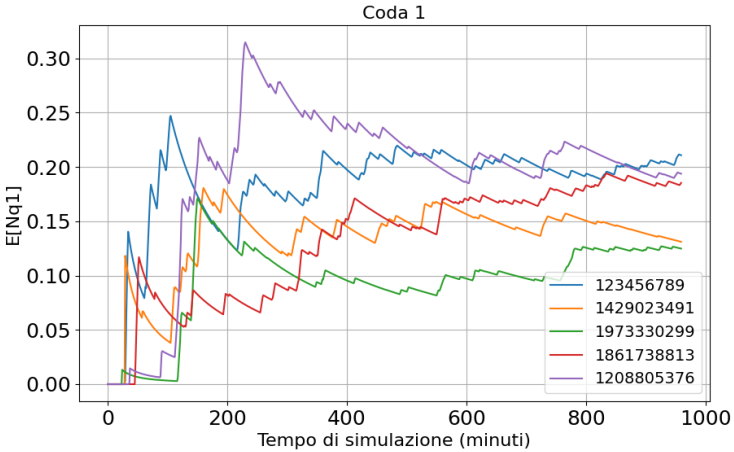


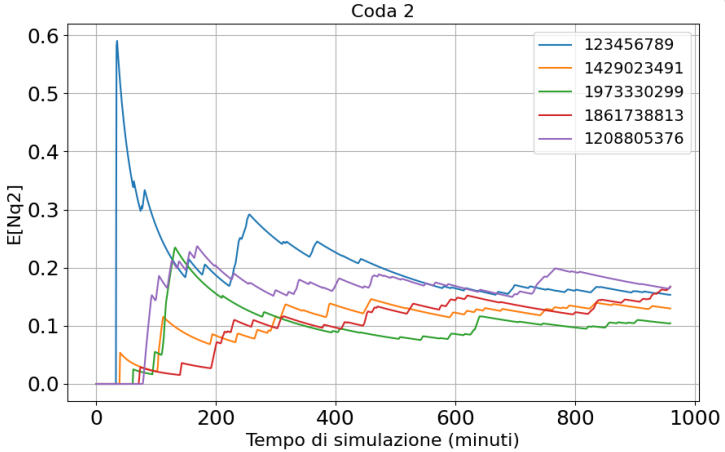


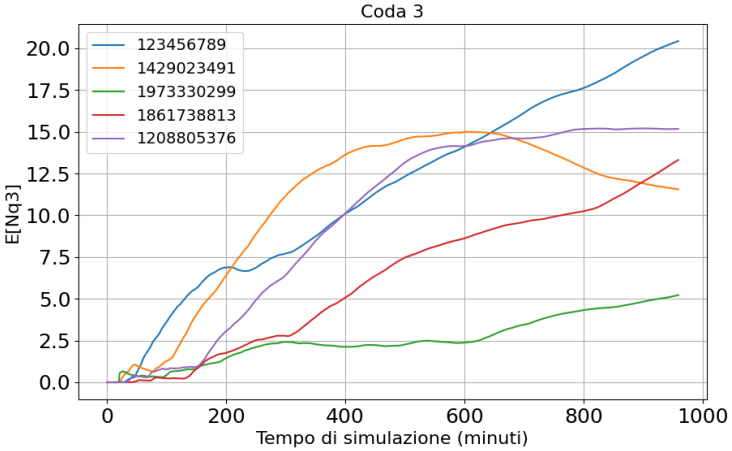


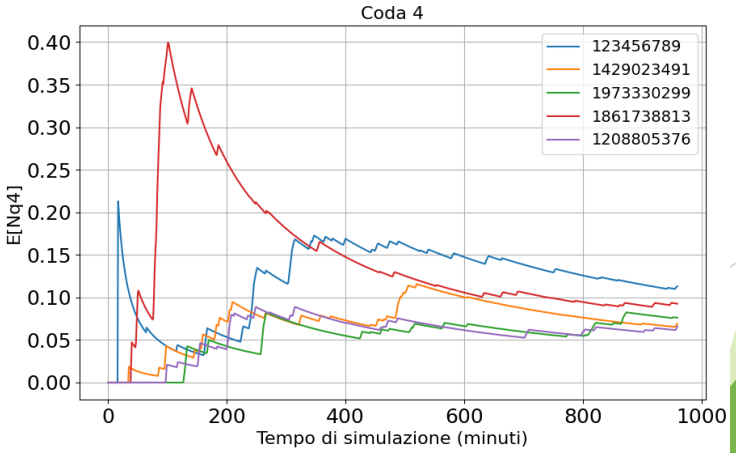
Valutato con 100 repliche, impostando un tempo limite di 960 minuti (16 ore)

Analisi dello stato transitorio – E[Nq]



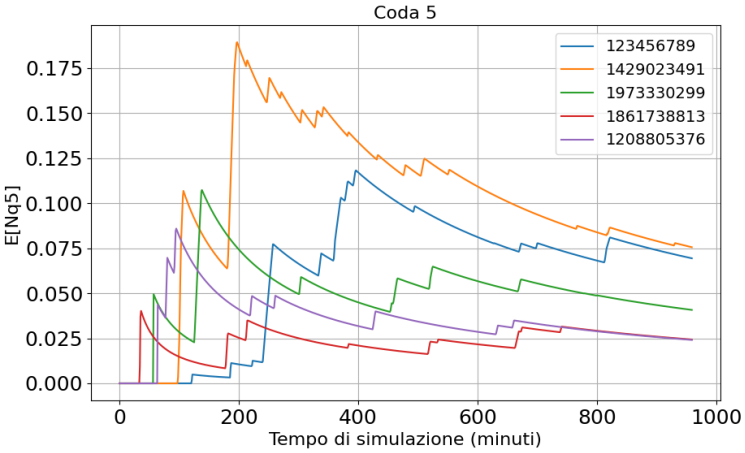


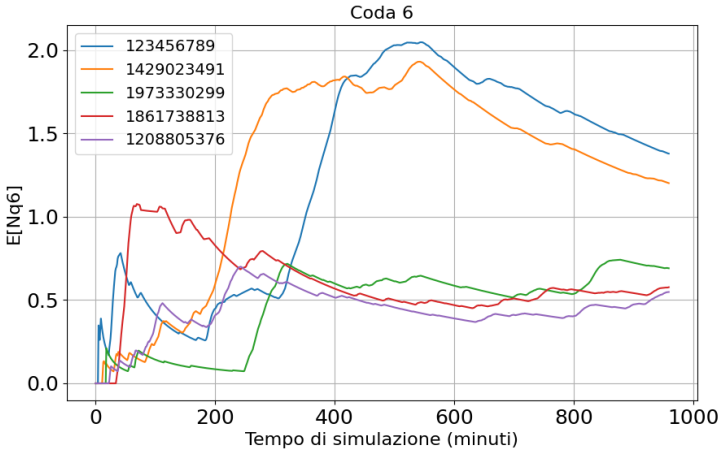


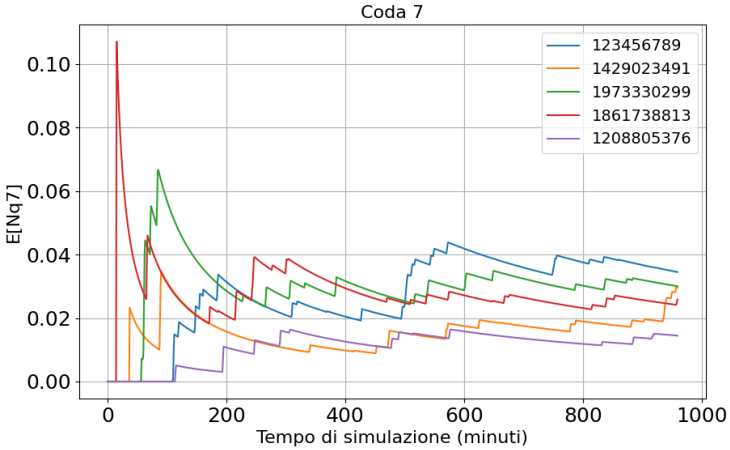


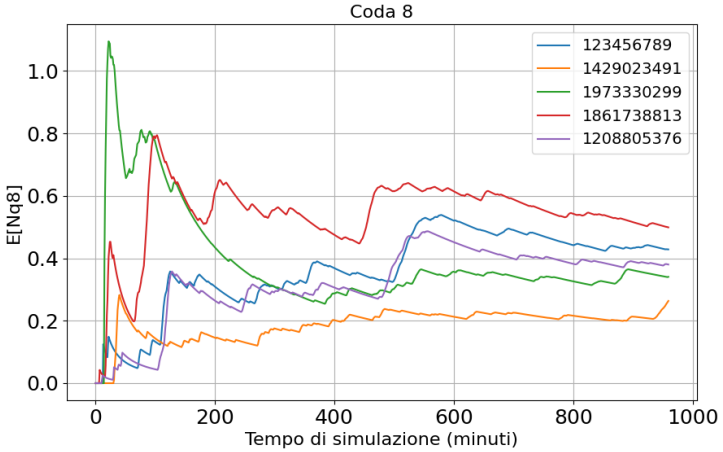
Valutato con 5 repliche, impostando un tempo limite di 960 minuti (16 ore)

Analisi dello stato transitorio – E[Nq]



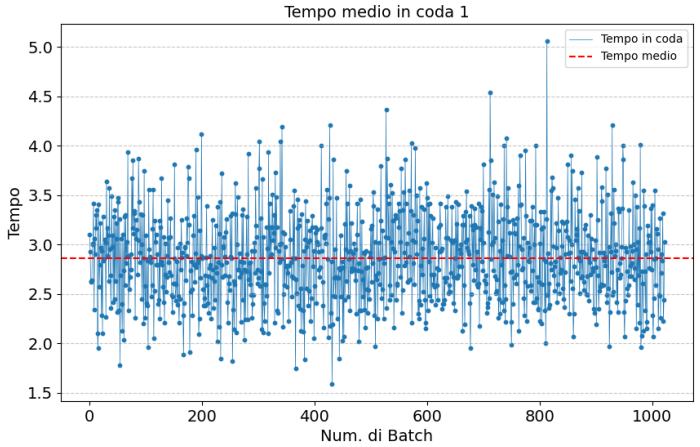


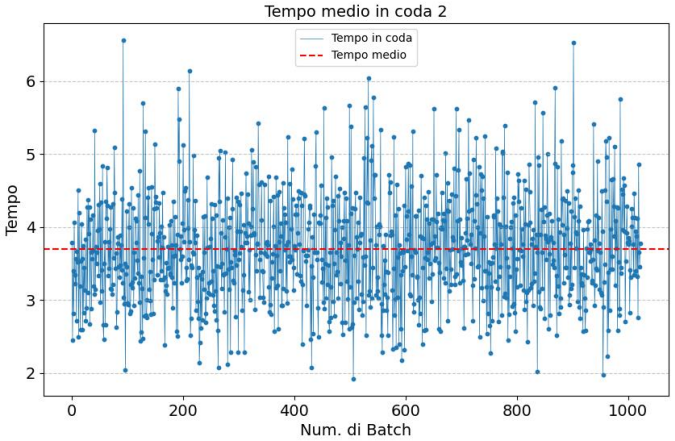


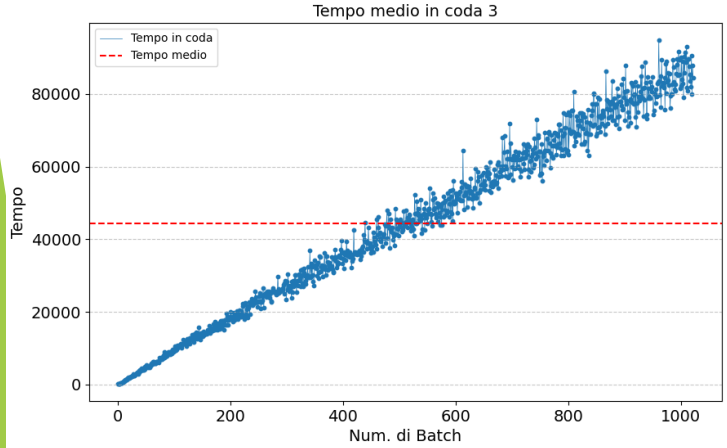


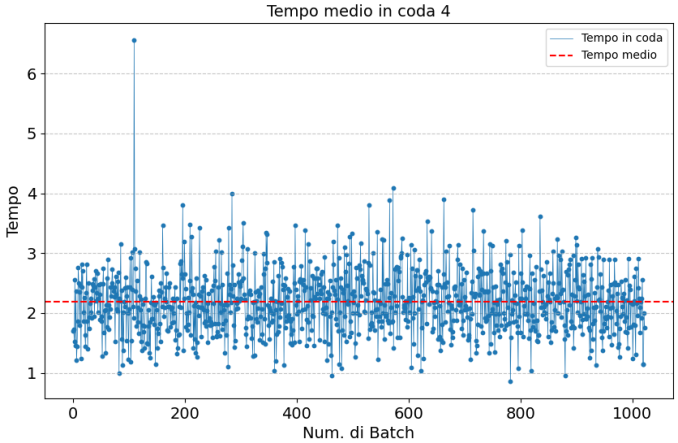
Valutato con 5 repliche, impostando un tempo limite di 960 minuti (16 ore)

Analisi dello stato stazionario



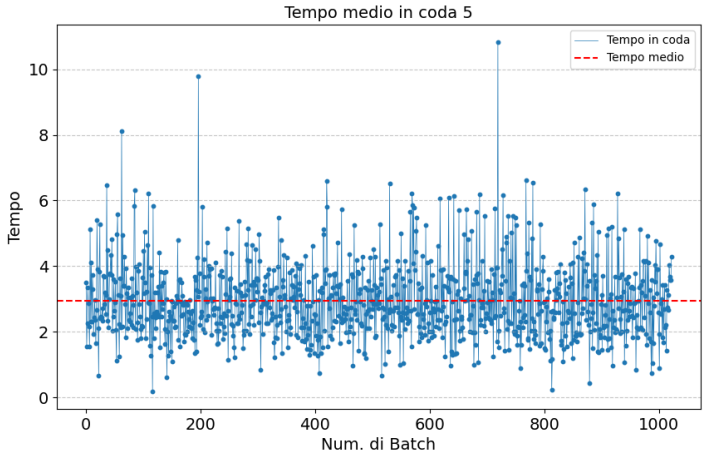


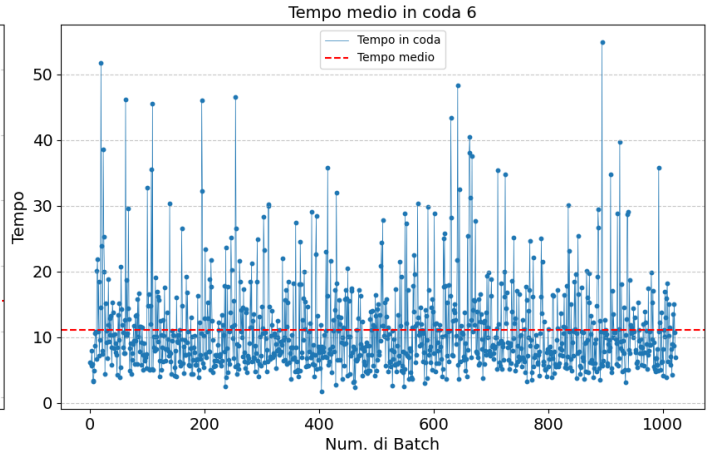


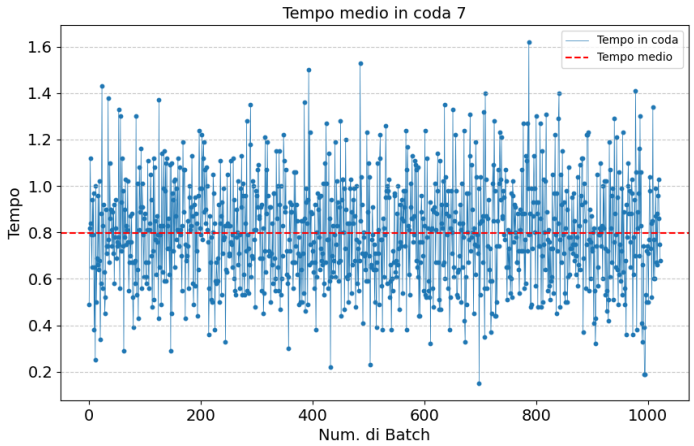


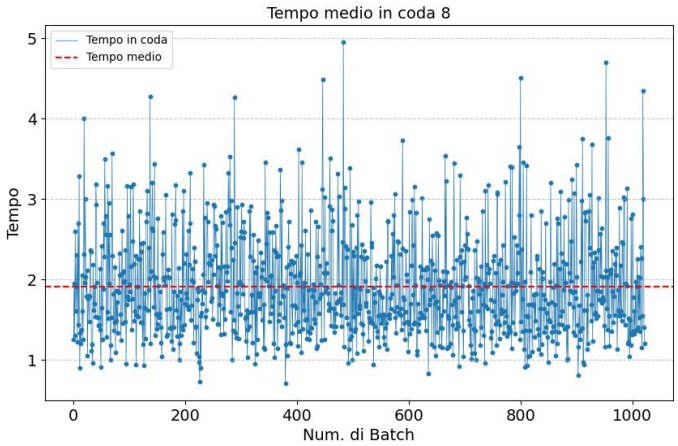
Valutato con la tecnica delle Batch Means:
b = 512
k = 1024

Analisi dello stato stazionario









Valutato con la tecnica delle Batch Means:
b = 512
k = 1024

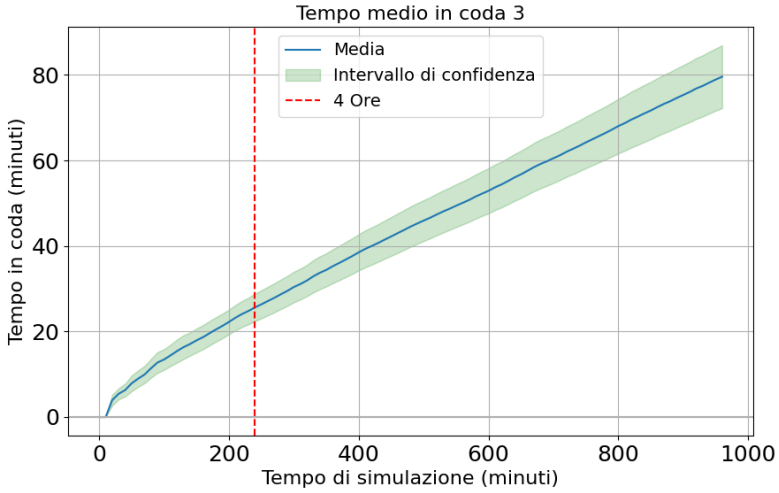
Utilizzazione del sistema

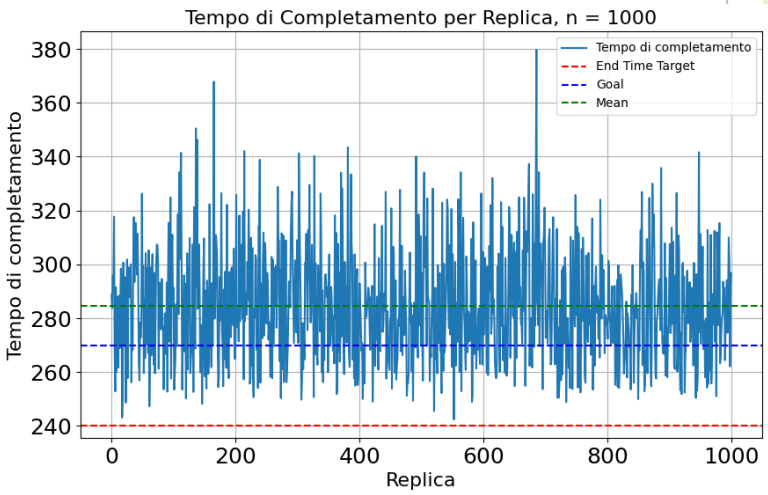
- È interessante valutare l'utilizzazione di ciascun centro.
- Il valore è stato calcolato mediando ciascuna utilizzazione delle mille repliche.
- Viene valutata al tempo di chiusura della filiale, 240 minuti e non al tempo dell'ultimo completamento, poiché sarebbe influenzata della chiusura degli arrivi: Ciascun centro può solamente svuotarsi, e/o possono esserci serventi liberi.

Multi server	0.89851858 ± 0.00296195
Spedizioni e Ritiri	0.74158964 ± 0.00682837
ATM	0.57433404 ± 0.00468570

Obiettivi soddisfatti?

- Obiettivo 1: Non viene soddisfatto per la classe d'utenza 3, che ha un tempo medio in coda di circa 23 minuti.
- Obiettivo 2: Si ha una media di 45 minuti di lavoro dopo l'orario di chiusura.





Modello Migliorativo

- Due strategie migliorative proposte:
  1. Rendere lo sportello Spedizioni e Ritiri più dinamico:
     - Se non ci sono clienti nelle code da lui servite, può ora servire Operazioni Generiche.
  2. Installazione di un Locker per gestire le operazioni di Spedizioni e Ritiri:
     - Tramite una spesa iniziale, come cambiano le performance?

Obiettivi e Scopi

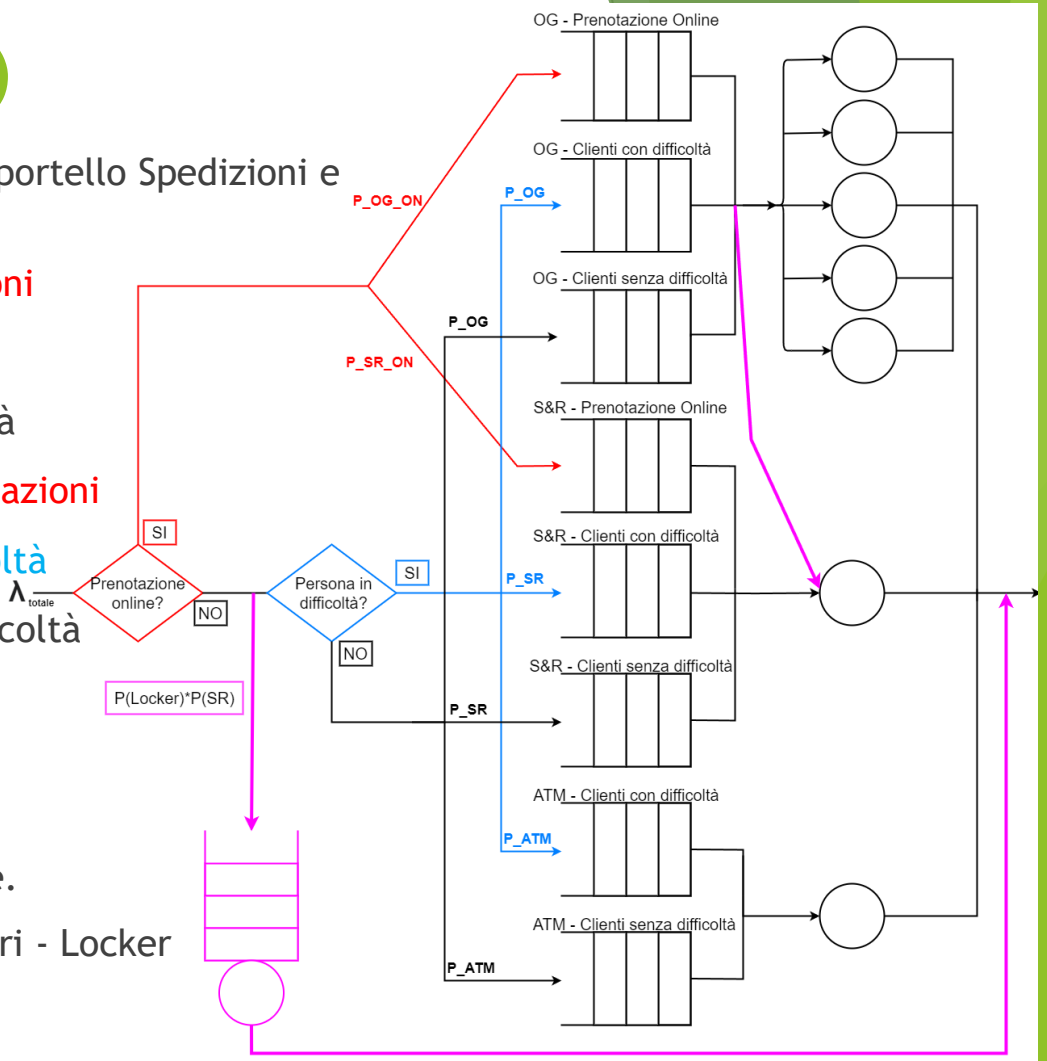
Invariati rispetto il modello base

- Obiettivo 1: Ridurre il tempo medio di coda a meno di 15 minuti per ciascuna classe d'utenza.
  - QoS aziendale comunicatoci dalla direttrice della filiale.

- Obiettivo 2: Ridurre il sovraccarico di lavoro dei dipendenti a meno di 30 minuti giornalieri.
  - Meno del 15% di sovraccarico.

Modello Concettuale (CM)

- 1° miglioramento: Le code di priorità per lo sportello Spedizioni e Ritiri sono ora così definite:
  1. Spedizioni e Ritiri – Utenti con Prenotazioni
  2. Spedizioni e Ritiri – Utenti con Difficoltà
  3. Spedizioni e Ritiri – Utenti senza Difficoltà
  4. Operazioni Generiche – Utenti con Prenotazioni
  5. Operazioni Generiche – Utenti con Difficoltà
  6. Operazioni Generiche – Utenti senza Difficoltà



- 2° miglioramento:
  - Aggiunta di una coda con relativo servente.
  - Aggiunta classe d'utenza: Spedizioni e Ritiri - Locker

Modello delle Specifiche (SM)

- 1° miglioramento:
  - In questo caso, il modello delle specifiche rimane invariato rispetto al modello base.

- 2° miglioramento:
  - La probabilità che un cliente che intende effettuare operazioni di Spedizione e Ritiri, vada in un Locker è P_{Locker} = 0,5.
  - Il tasso di ingresso al Locker è: $\lambda_{totale} * (1 - P_{ON}) * P_{SR} * P_{Locker}$
  - È stata adottata una distribuzione Normale troncata su valori inferiori a 10^{-6}, con $\mu_{Locker} = 1$ job/min e σ = 0.5/3

Modello Computazionale (CPTM)

- **Prima Ottimizzazione**: In questo caso è stato modificato unicamente l'array relativo alle code associate al servente Spedizioni e Ritiri, garantendo una corretta gestione della priorità.

- **Seconda Ottimizzazione**: La modifica del codice ha influenzato più funzioni, poiché c'è stata l'aggiunta e la gestione di una nuova coda e un nuovo servente, mantenendo lo stesso funzionamento di base dell'algoritmo Next event Simulation.

Verifica

- Verifichiamo il corretto funzionamento del centro Locker, modellato come un centro M/M/1 solo per questa fase.

- $\lambda_{locker} = \lambda_{totale} * (1 - P_{ON}) * P_{SR} * P_{Locker} = 0{,}066666\ j/min$
- $E(s) = 1/\mu_{locker} = 1\ min$
- $E(Tq) = \frac{\rho E(s)}{1-\rho} = 0{,}071428\ min$

- Calcolando un intervallo di confidenza al 95% tramite estimate.py, sui 1024 valori raccolti all'interno di ogni batch, riscontriamo che la media teorica ricade al suo interno:

```
based upon 1024 data points and with 95 confidence
the expected value is in the interval 0.06841337 +/- 0.00330980
```

Validazione

- Per semplicità, in questa fase, consideriamo $P_{SR_ON} = 0$
- Per validare il sistema, è stato deviato il flusso delle operazioni Spedizioni e Ritiri unicamente verso il Locker, con $P_{Locker} = 1$
- Di conseguenza, il servente dedicato a Spedizioni e Ritiri, gestisce esclusivamente Operazioni Generiche.

queue	avg wait t	avg delay t	avg service t	avg # in node	avg # in queue	avg # in service
1	16.71489349	1.04883224	15.66606124	0.80924619	0.05077888	0.75846732
2	18.36154514	3.11831832	15.24322683	0.88896830	0.15097238	0.73799593
3	26.48585167	11.80707512	14.67877655	6.51015873	2.90215071	3.60800802
5	8.09123043	2.24057339	5.85065704	0.03013341	0.00834436	0.02178905
6	12.57658165	8.04111020	4.53547146	0.70256672	0.44920127	0.25336545
7	3.17318010	0.62932746	2.54385264	0.11817576	0.02343745	0.09473831
8	3.87305838	1.41236939	2.46068899	0.63465897	0.23143800	0.40322096
9	0.99629030	0.00000000	0.99629030	0.06307664	0.00000000	0.06307664

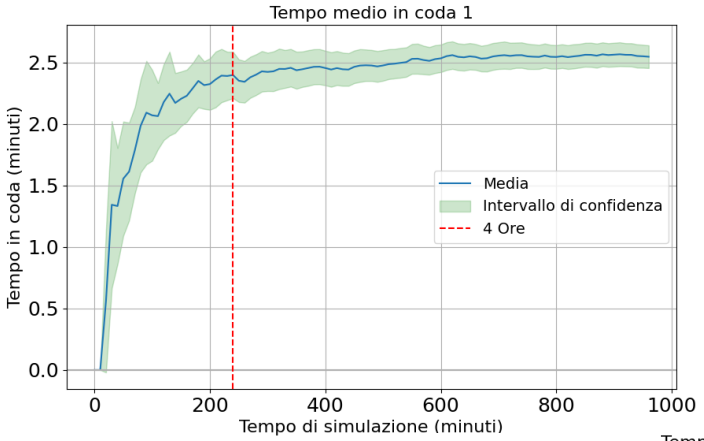
Simulation complete. Clients served: [13, 13, 66, 0, 1, 15, 10, 44, 17]

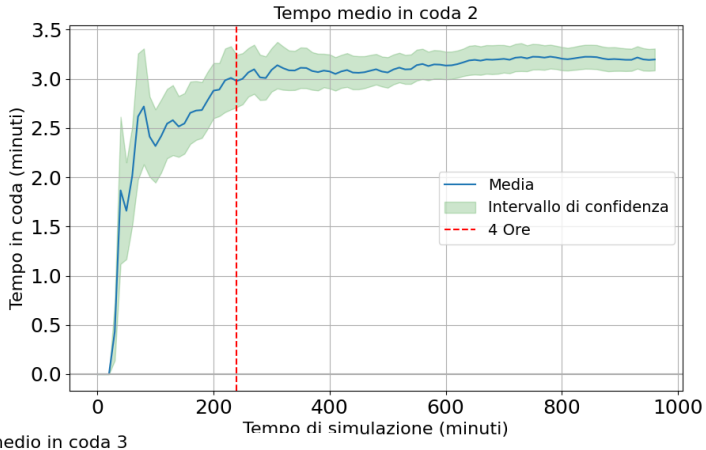
queue	avg wait t	avg delay t	avg service t	avg # in node	avg # in queue	avg # in service
1	16.63364134	0.87358073	15.76006061	0.80531240	0.04229413	0.76301827
2	18.77696677	3.16934254	15.60762423	0.90908081	0.15344270	0.75563811
3	21.00356303	6.41507676	14.58848627	5.16262535	1.57681046	3.58581489
7	3.17318010	0.62932746	2.54385264	0.11817576	0.02343745	0.09473831
8	3.87305838	1.41236939	2.46068899	0.63465897	0.23143800	0.40322096
9	1.02972822	0.02367963	1.00604859	0.13422220	0.00308657	0.13113563

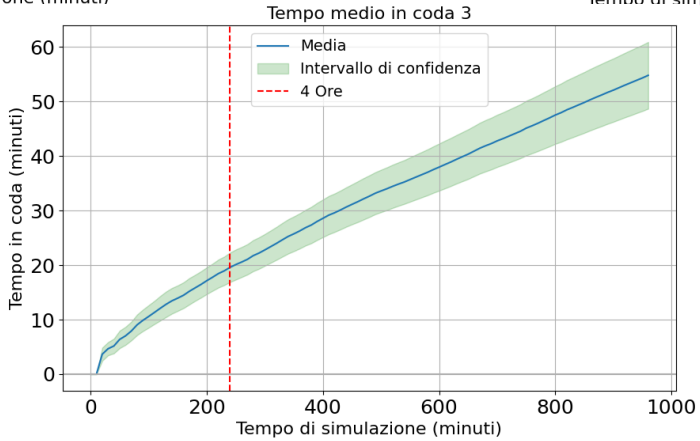
Simulation complete. Clients served: [13, 13, 66, 0, 0, 0, 10, 44, 35]

Analisi dello stato transitorio – 1° Miglioramento

- Analizziamo le classi d'utenza relative alle **Operazioni Generiche**.

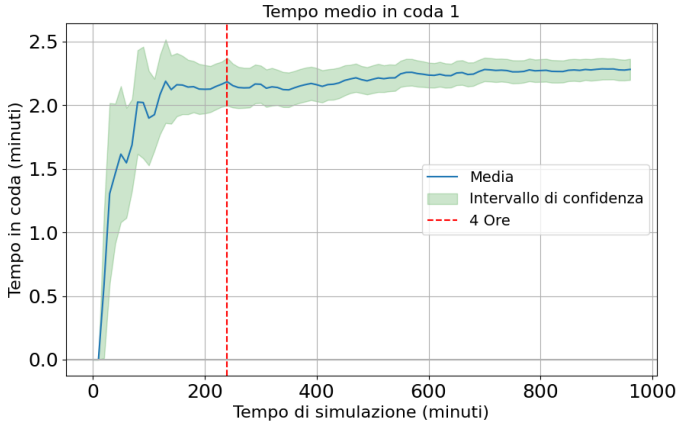


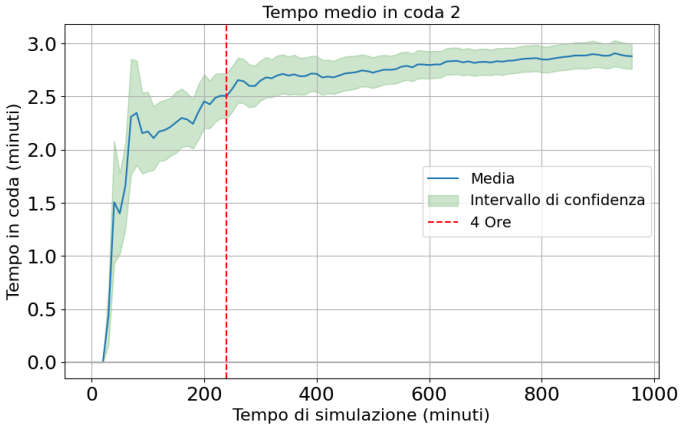


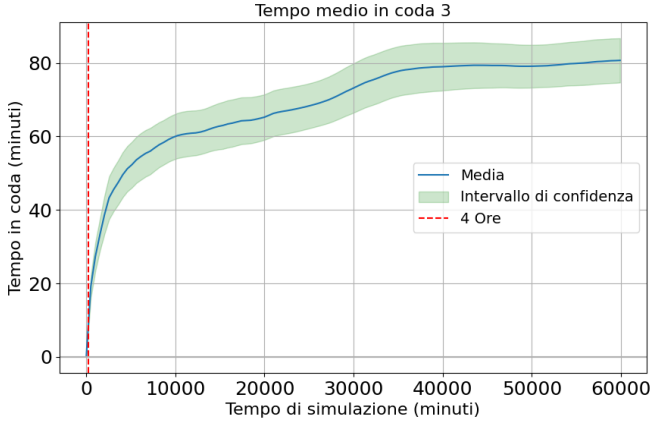


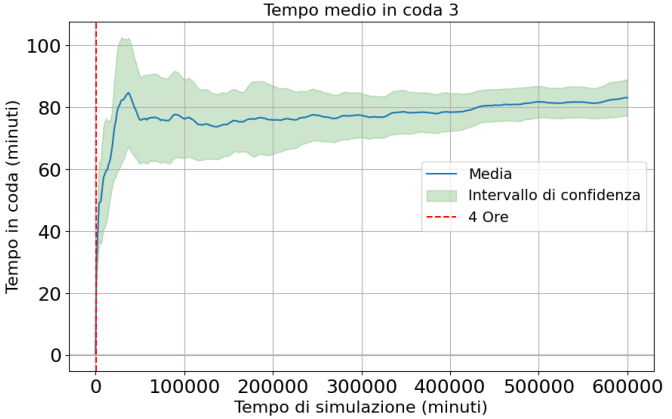
Analisi dello stato transitorio – 2° Miglioramento

- Analizziamo le classi d'utenza relative alle **Operazioni Generiche.**



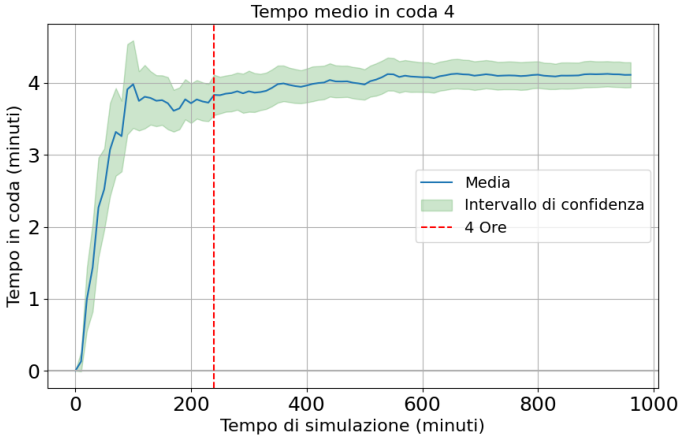


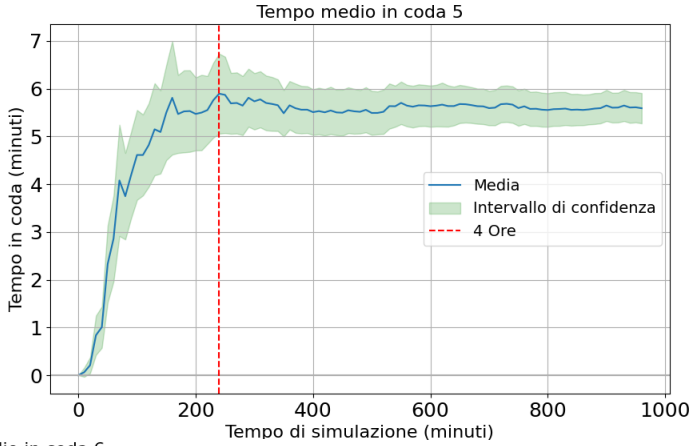


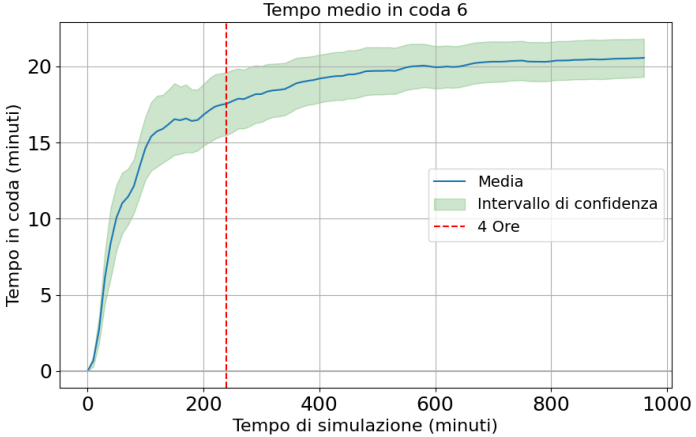


Analisi dello stato transitorio – 1° Miglioramento

- Analizziamo le classi d'utenza relative a **Spedizioni e Ritiri**.

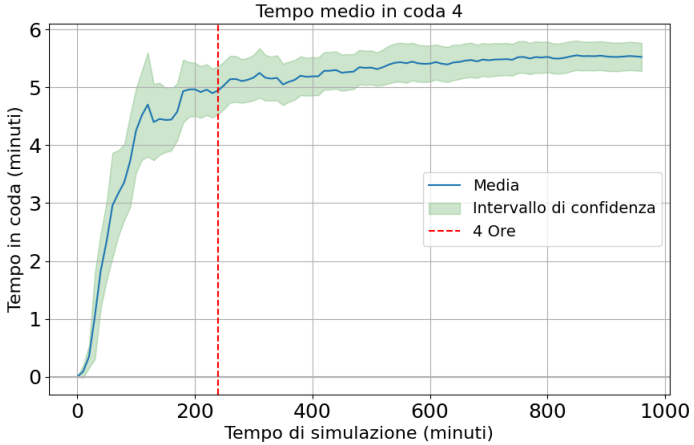


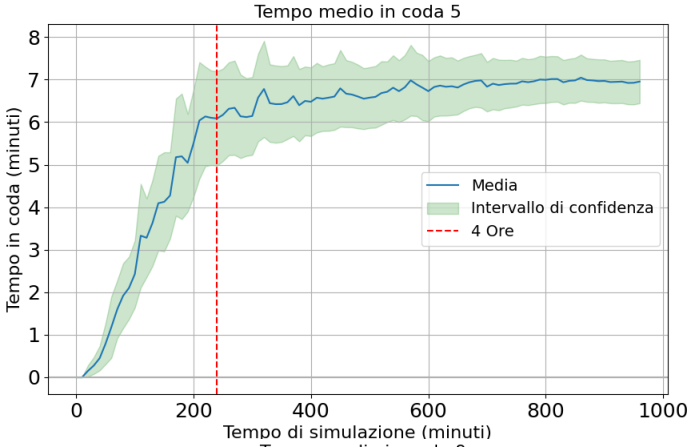


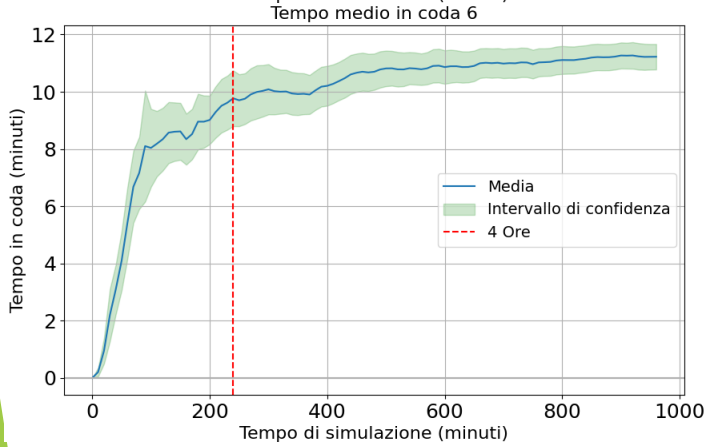


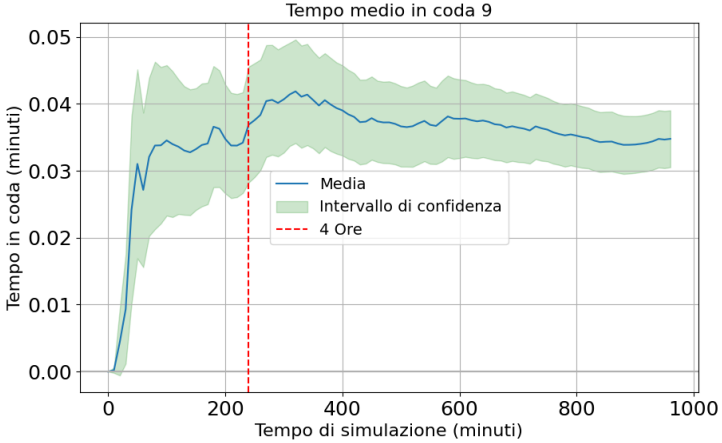
Analisi dello stato transitorio – 2° Miglioramento

- Analizziamo le classi d'utenza relative a **Spedizioni e Ritiri.**





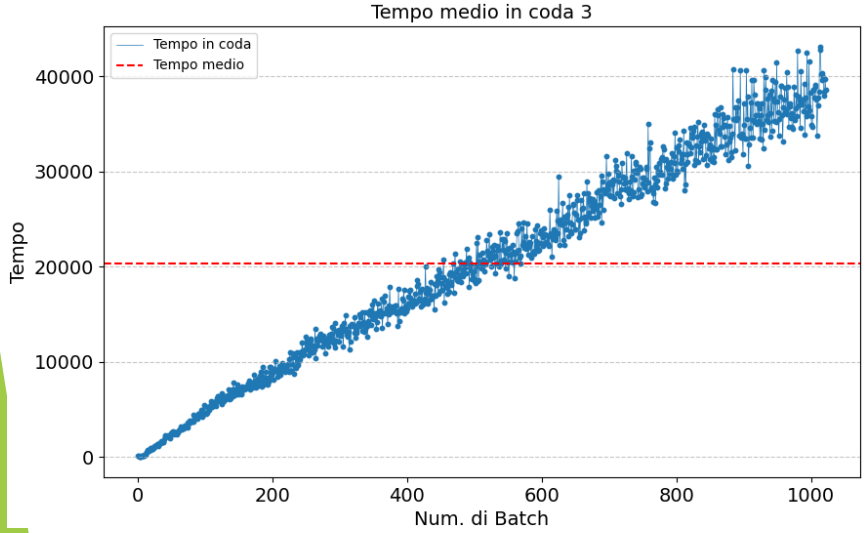




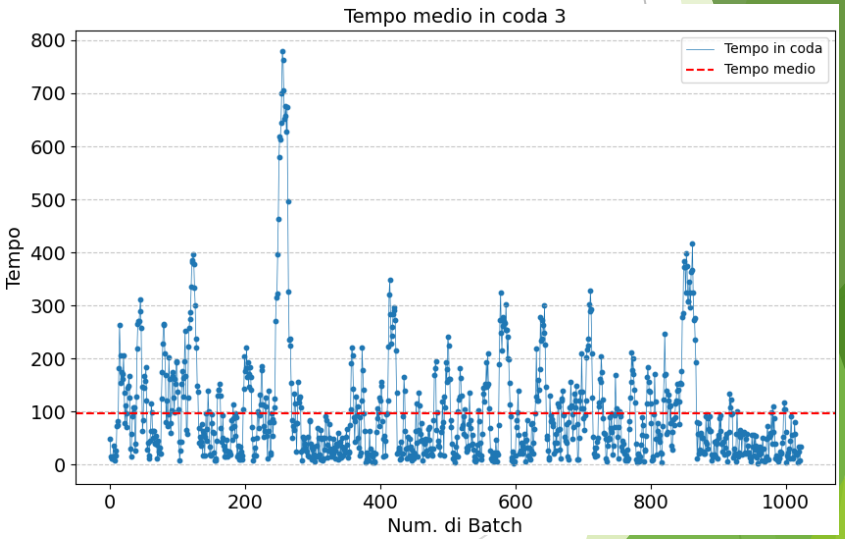
Analisi dello stato stazionario

- Analizziamo la sola classe d'utenza 3, poiché ha subito il cambiamento più significativo.

Primo miglioramento

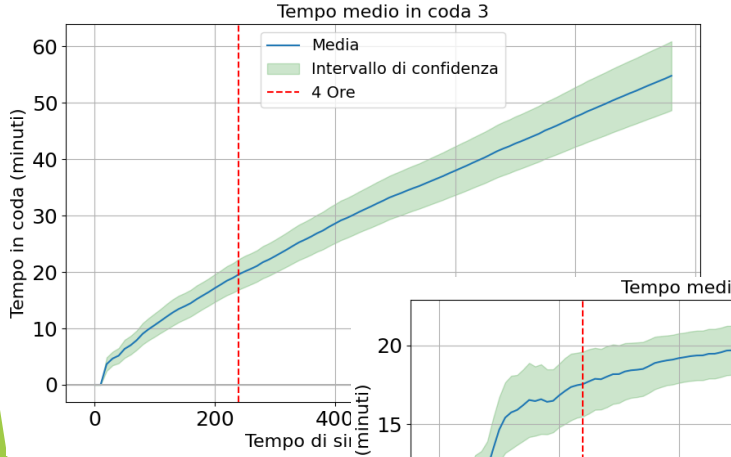


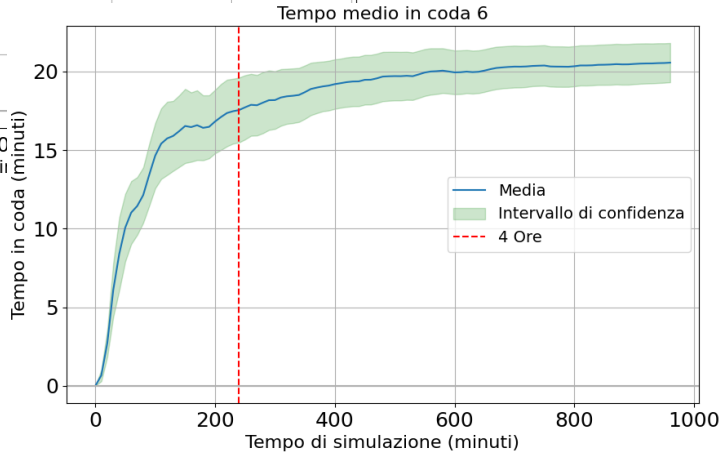
Secondo miglioramento

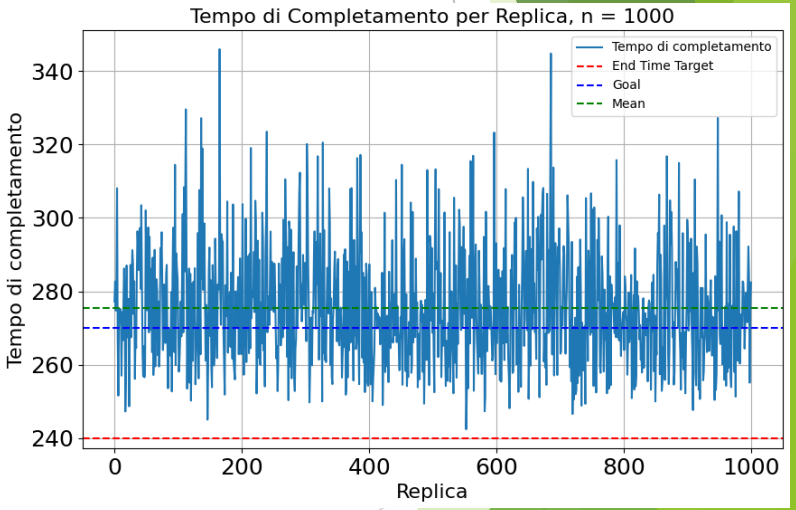


Obiettivi soddisfatti? – 1° Miglioramento

- Obiettivo 1: Non viene soddisfatto ne dalla coda 3, ne dalla coda 6, avendo un tempo medio in entrambi di 17 minuti circa.
- Obiettivo 2: Si ha una media di 35 minuti di lavoro dopo l'orario di chiusura.

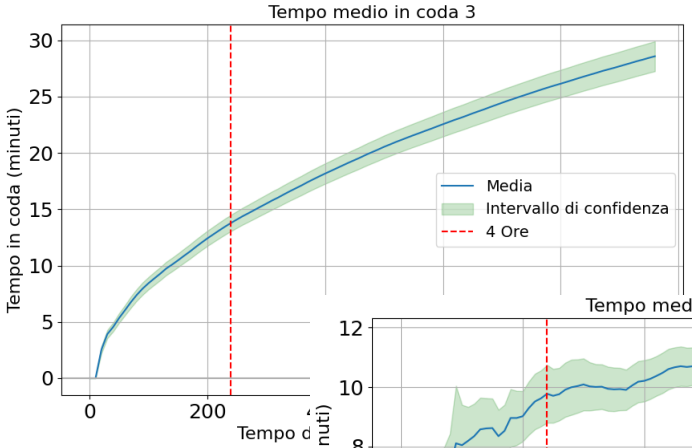


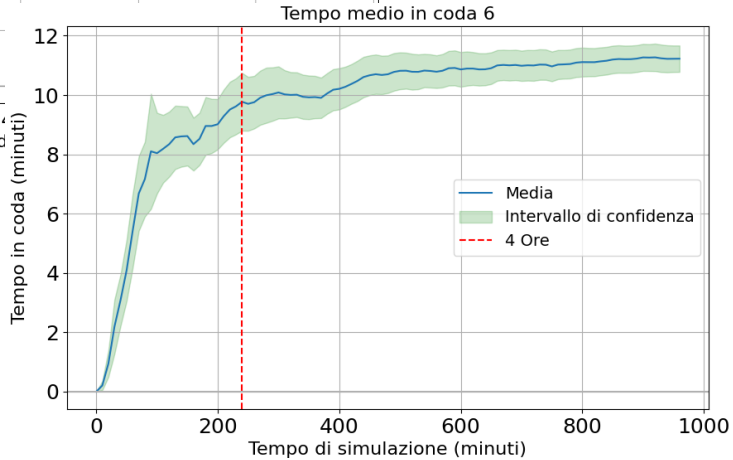


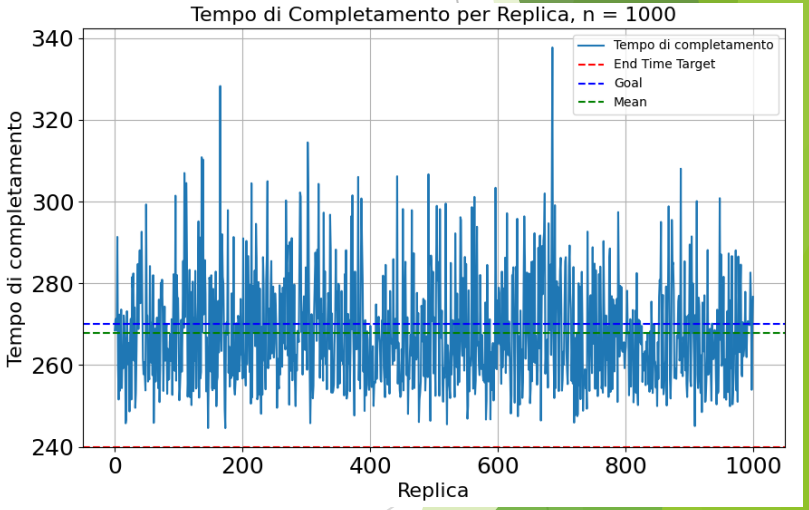


Obiettivi soddisfatti? – 2° Miglioramento

- Obiettivo 1: Viene soddisfatto sia dalla coda 3, sia dalla coda 6.
- Obiettivo 2: Si ha una media di 28 minuti di lavoro dopo l'orario di chiusura.







Risultati e Conclusioni

- Primo miglioramento:
  - Permette di ridurre i tempi medi in coda per i clienti che intendono effettuare Operazioni Generiche senza priorità.
  - Comporta un ovvio peggioramento per i clienti di Spedizioni e Ritiri.
  - In generale permette di bilanciare i tempi in coda tra le due classi di utenza, ma non rispettando gli obiettivi stabiliti.

- Secondo miglioramento:
  - Il Locker riduce il carico sullo sportello dedicato a Spedizioni e Ritiri.
  - Permette una riduzione significativa del tempo medio in coda per le classi senza priorità di Operazioni Generiche e Spedizioni e Ritiri.
  - Riduce il sovraccarico di lavoro sotto i 30 minuti.
  - Entrambi gli obiettivi sono soddisfatti.

Risultati e Conclusioni

Valore Osservato	Modello Base	1° Modello Migliorativo	2° Modello Migliorativo
$E(Tq_3)$	$\cong$ 23 min	$\cong$ 17 min	$\cong$ 13 min
$E(Tq_6)$	$\cong$ 9 min	$\cong$ 17 min	$\cong$ 10 min
Tempo di sovraccarico medio	$\cong$ 45 min	$\cong$ 35 min	$\cong$ 28 min