

Progetto *Performance Modeling of Computer
Systems and Networks*
a.a: 2022/2023

Luca Mastrobattista
0292461

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Si prende in esame il contesto di un locale che offre servizi di bar e pizzeria in un piccolo paese.

- ▶ si analizza la gestione delle richieste in un'intera giornata lavorativa, sia nei giorni settimanali che finesettimanali
- ▶ il locale è aperto dalle 7.00 alle 15.00 e dalle 18.00 alle 2.00

- ▶ Rispettare dei servizi di qualità:
 - Limitare i tempi di risposta per le richieste al bar a 3 minuti
 - Limitare i tempi di risposta per le richieste alla pizzeria a 10 minuti
- ▶ Massimizzare i guadagni cercando il numero di serventi ottimali

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

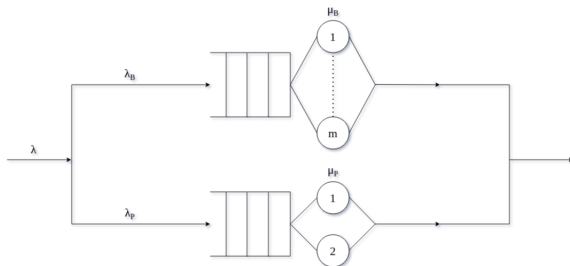
Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Visualizzazione grafica



- La frequenza di arrivo λ si compone della frequenza di arrivo λ_B e λ_P
- Ogni servente di tipo B rappresenta un barista assunto, che lavora con una frequenza μ_B
- Ogni servente di tipo P, invece, rappresenta una delle due richieste che il pizzaiolo è in grado di gestire contemporaneamente e lavora con frequenza μ_P

► Eventi

- Arrivi dall'esterno
- Completamenti

► Completamenti

- Numero di richieste di tipo B al centro
- Numero di richieste di tipo P al centro
- Stato del servente, per ogni servente di tipo B e P

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Distribuzione degli arrivi

- Distribuzione base: Gli arrivi sono assunti esponenziali, per offrire una flessibilità dei tempi di interarrivo che riflette accuratamente la casualità degli arrivi in un bar.
- Ritardo gaussiano: È stato poi introdotto un ritardo gaussiano per rendere il modello più realistico permettendo di tener conto di effetti come le ore punta.

Fascia oraria	$\lambda_{B,W}$	$\lambda_{P,W}$	$\lambda_{B,WE}$	$\lambda_{P,WE}$	μ_B	σ_B	μ_P	σ_P
07:00 → 11:00	30 j/h	✗	30 j/h	✗	8	1.2	✗	✗
11:00 → 15:00	12.5 j/h	✗	20 j/h	✗	13.5	2	✗	✗
18:00 → 19:00	25 j/h	✗	45 j/h	✗	18.5	0.4	✗	✗
19:00 → 23:00	12.5 j/h	10 j/h	22.5 j/h	30 j/h	22.5	2	20.5	1
23:00 → 02:00	10 j/h	✗	20 j/h	✗	24	0.9	✗	✗

I tempi di servizio si assumono esponenziali per entrambi i tipi di serventi.

- ▶ Servente di tipo B: Si assume che ogni servente sia in grado di completare una richiesta con il tempo medio di 2 minuti, durante i quali si dedica esclusivamente a quella richiesta.
- ▶ Servente di tipo P: Si assume che ogni pizza possa essere completamente preparata con un tempo medio di 3 minuti.

► Guadagni

- Per ogni richiesta di tipo B si assume un guadagno medio di 5,00 €
- Per ogni richiesta di tipo P si assume un guadagno medio di 10,00 €

► Costi

- Stipendio medio di un barista per 8 ore: 40,00 €
- Stipendio del pizzaiolo per giorno: 50,00 €
- Costo medio delle bollette: 2.750,00 € al mese
- Costo medio dell'affitto: 1.500,00 € al mese
- Costo medio per il rifornimento: 2.000,00 € al mese

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Descrizione generale del programma

Il simulatore, implementato in *Python*, segue l'approccio della *next-event simulation* ed è altamente configurabile in base alle esigenze, specificando opportuni flag a riga comando:

```
~/Scrivania/PMCSN/PMCSN-simulation (main*) » python simulation.py -h
usage: simulation.py [-h] [-cf FILEPATH] [-scf FILEPATH] [-fh] [-lh] [-cc OPTION VALUE] [-fb THRESHOLD] [-s SEED] [-ngf]
                    [-of FILE] [-wed] [-ns]

PMCSN project command line interface

options:
  -h, --help            show this help message and exit
  -cf FILEPATH, --configFile FILEPATH
                        specify a configuration file to load
  -scf FILEPATH, --storeConfigFile FILEPATH
                        specify an output file where to store config
  -fh, --finite_horizont
                        simulate a finite horizon case
  -lh, --infinite_horizont
                        simulate an infinite horizon case. Using this, gaussian factor is automatically disabled
  -cc OPTION VALUE, --change_config OPTION VALUE
                        specify configuration to change
  -fb THRESHOLD, --find_b_value THRESHOLD
                        find the value of b such that autocorrelation lag j=1 is <= THRESHOLD
  -s SEED, --seed SEED  use the given SEED as random seed. If SEED = 0 then the initial seed is to be supplied
                        interactively; if SEED < 0 then the initial seed is obtained from the system clock; if SEED > 0 >
                        0 then it is the initial seed (unless too large). default value is 0
  -ngf, --no_gaussian_factor
                        don't use the gaussian probability value to weight interarrival times
  -of FILE, --output_file FILE
                        save output in ./output/OUTPUTFILE.csv. The file format .csv is added if not already present
  -wed, --weekend_day   simulate a weekend day, with the proper system variables. Default this option is disabled, meaning
                        that a week day is simulate
  -ns, --no_split       don't split the analysis using 2 lists for first half and second half of the day
```

Politica di scelta del servente

La scelta del servente per gestire una nuova richiesta è diversa per i due tipi B e P:

- ▶ Richieste al bar: La scelta del prossimo servente segue la politica di *equity*
- ▶ Richieste alla pizzeria: Il prossimo servente selezionato è il primo trovato libero scandendo la lista in ordine crescente

Evento di campionamento

Il tempo del prossimo evento di campionamento viene impostato in modo da evitare che due eventi successivi siano entrambi eventi di campionamento:

simulation.py

```
242 times = []
243 for index, ev in enumerate(stats.events):
244     if index != e and ev.x == 1:
245         times.append(ev.t)
246 stats.events[e].t = min(times) +
        samplingInterarrivalTime
```

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Differenza tra risultati teorici e sperimentali

Ogni *run* di simulazione con le configurazioni di default produce statistiche sperimentali che sono diverse rispetto a quelle teoriche. Il motivo di tale differenza è che il numero di job processati è piccolo, e perciò le statistiche sperimentali sono poco rappresentative.

<i>B type</i>	Week		Weekend	
Statistica	Risultato teorico	Risultato sperimentale	Risultato teorico	Risultato sperimentale
avgInterarrivals	3.471 min	2.984 ± 0.076 min	2.413 min	2.494 ± 0.025 min
avgWaits	2.251 min	2.284 ± 0.047 min	2.524 min	2.266 ± 0.036 min
avgNumNodes	0.682 j	0.788 ± 0.029 j	1.102 j	0.801 ± 0.021 j
avgDelays	0.251 min	0.268 ± 0.014 min	0.524 min	0.286 ± 0.010 min
avgNumQueues	0.106 j	0.091 ± 0.008 j	0.273 j	0.103 ± 0.005 j
avgService (1)	2 min	1.724 ± 0.047 min	2 min	1.772 ± 0.035 min
avgService (2)	2 min	2.367 ± 0.039 min	2 min	2.216 ± 0.036 min

<i>P type</i>	Week		Weekend	
Statistica	Risultato teorico	Risultato sperimentale	Risultato teorico	Risultato sperimentale
avgInterarrivals	5.882 min	4.408 ± 0.160 min	2.000 min	1.846 ± 0.036 min
avgWaits	3.209 min	3.465 ± 0.116 min	6.857 min	5.667 ± 0.151 min
avgNumNodes	0.545 j	0.761 ± 0.042 j	3.429 j	2.962 ± 0.046 j
avgDelays	0.209 min	0.050 ± 0.008 min	3.857 min	2.514 ± 0.083 min
avgNumQueues	0.035 j	0.013 ± 0.003 j	1.929 j	1.316 ± 0.037 j
avgService (1)	3 min	2.918 ± 0.136 min	3 min	2.947 ± 0.116
avgService (2)	3 min	4.722 ± 0.201 min	3 min	3.361 ± 0.069 min

Nuova configurazione: verify1.py

Si definisce il nuovo file di configurazione per cercare di aumentare la rappresentatività delle statistiche:

configurations/verify1.py

```
# per esprimere i tempi in secondi:
SLOTSTIME = [ (i * 3600) for i in [7, 11, 15, 18, 19,
    23] ]
STOP_B = 26 * 3600
# per rendere il sistema stabile anche dal punto di
    vista analitico
MEAN_SERVICE_TIME_B = 0.5
MEAN_SERVICE_TIME_P = 0.5
# i lambda:
WEEK_LAMBDA_B = [2, 2, 0, 3, 3, 3]
WEEK_LAMBDA_P = 1.7
```

verify1.py - Risultati

Eseguendo il comando `python simulation.py -s 123 -ngf -cf configurations/verify1.py`, si ottengono i seguenti risultati:

<i>B type</i> Statistica	Week	
	Risultato teorico	Risultato sperimentale
avgInterarrivals	0.317 s	0.466 ± 0.002 s
avgWaits	0.415 s	0.337 ± 0.000 s
avgNumNodes	1.394 j	0.730 ± 0.001 j
avgDelays	0.115 s	0.037 ± 0.000 s
avgNumQueues	0.449 j	0.083 ± 0.000 j
avgService (1)	0.3 s	0.300 ± 0.000 s
avgService (2)	0.3 s	0.300 ± 0.000 s

<i>P type</i> Statistica	Week	
	Risultato teorico	Risultato sperimentale
avgInterarrivals	0.588 s	0.593 ± 0.000 s
avgWaits	0.610 s	0.603 ± 0.000 s
avgNumNodes	1.037 j	1.016 ± 0.000 j
avgDelays	0.110 s	0.105 ± 0.000 s
avgNumQueues	0.187 j	0.176 ± 0.000 j
avgService (1)	0.5 s	0.496 ± 0.000 s
avgService (2)	0.5 s	0.501 ± 0.000 s

Si può osservare come i valori teorici per le richieste di tipo "P", che sono state in tutto 24407, cominciano a convergere ai valori teorici. Per le richieste di tipo "B", nonostante siano state 181785, i valori ottenuti sono ancora lontani da quelli teorici.

Possibile spiegazione e mitigazione

- ▶ Si può notare che la frequenza di interarrivo media nella prima metà della giornata è più bassa rispetto alla seconda metà e perciò il numero di job processati e i campioni raccolti nelle prime 8 ore saranno inferiori rispetto alle ultime 8 ore.
- ▶ Per mantenere un'analisi media equilibrata, si è introdotta un'opzione predefinita che suddivide l'analisi in due fasce orarie separate. Si raccolgono e si calcolano i dati statistici separatamente per entrambe le metà della giornata.
- ▶ Le medie e varianze globali si otterranno mediando i risultati ponderati per il numero di job processati.

$$\mu_{glob} = \frac{n_1\mu_1 + n_2\mu_2}{n_1 + n_2}$$

$$\sigma_{glob}^2 = \frac{n_1\sigma_1^2 + n_2\sigma_2^2}{n_1 + n_2}$$

Nuova opzione - Risultati

Con la nuova opzione attiva, lanciando il comando

```
python simulation.py -s 123 -ngf -cf configurations/  
verify1.py
```

si ottengono i seguenti risultati:

<i>B type</i> Statistica	Week	
	Risultato teorico	Risultato sperimentale
avgInterarrivals	0.400 s	0.399 ± 0.000 s
avgWaits	0.353 s	0.355 ± 0.000 s
avgNumNodes	0.894 j	0.938 ± 0.000 j
avgDelays	0.053 s	0.055 ± 0.000 s
avgNumQueues	0.144 j	0.156 ± 0.000 j
avgService (1)	0.3 s	0.300 ± 0.000 s
avgService (2)	0.3 s	0.299 ± 0.000 s

Anche per le richieste di tipo "B", adesso i risultati convergono a quelli teorici.

Verifica della politica di selezione del server

Per valutare la politica di selezione è stato eseguito il seguente comando:

```
python simulation.py -s 123 -ngf -cf configurations/  
verify1.py -cc servers_b 10 -cc servers_p 10
```

I risultati ottenuti da questa configurazione sono riportati di seguito:

```
for 144427 jobs in the Bar and 28075 sample:  
  statistic      mean      variance      conf int  
  avg interarrivals .. : 0.399      0.000      0.000  
  avg wait ..... : 0.300      0.000      0.000  
  avg # in node ..... : 0.782      0.000      0.000  
  avg delay ..... : 0.000      0.000      0.000  
  avg # in queue ..... : 0.000      0.000      0.000  
  
the server statistics are:  
  server      utilization      avg service +/- w      share  
  1      0.078      0.298 +/- 0.000      0.100  
  2      0.078      0.298 +/- 0.000      0.100  
  3      0.079      0.304 +/- 0.000      0.100  
  4      0.078      0.300 +/- 0.000      0.100  
  5      0.079      0.302 +/- 0.000      0.100  
  6      0.078      0.298 +/- 0.000      0.100  
  7      0.077      0.295 +/- 0.000      0.100  
  8      0.078      0.300 +/- 0.000      0.100  
  9      0.080      0.305 +/- 0.000      0.100  
  10     0.078      0.298 +/- 0.000      0.100
```

```
for 24407 jobs in the Pizzeria and 7495 sample:  
  statistic      mean      variance      conf int  
  avg interarrivals .. : 0.593      0.000      0.000  
  avg wait ..... : 0.498      0.000      0.000  
  avg # in node ..... : 0.840      0.000      0.000  
  avg delay ..... : -0.000      0.000      0.000  
  avg # in queue ..... : -0.000      0.000      0.000  
  
the server statistics are:  
  server      utilization      avg service +/- w      share  
  12     0.458      0.502 +/- 0.000      0.542  
  13     0.252      0.501 +/- 0.000      0.299  
  14     0.097      0.489 +/- 0.000      0.117  
  15     0.026      0.463 +/- 0.001      0.033  
  16     0.006      0.476 +/- 0.001      0.008  
  17     0.001      0.382 +/- 0.003      0.001  
  18     0.000      0.283 +/- 0.006      0.000  
  19     0.000      0.152 +/- 0.004      0.000  
  20     0.000      0.163 +/- 0.007      0.000  
  21     0.000      0.000 +/- 0.000      0.000
```

Tutti i server di tipo "B" presentano valori simili per utilizzazione, tempo di servizio e share, mentre per i server di tipo "P" c'è un decremento dei valori all'aumentare dell'indice del server.

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

Problematiche riscontrate

- ▶ Durante il processo di validazione si sono riscontrate delle sfide significative nell'utilizzo dell'analisi a orizzonte finito, principalmente dovute al numero limitato di job processati. Questa limitazione ha comportato un campione statistico ridotto e, di conseguenza, una maggiore variabilità nei risultati osservati.
- ▶ Per validare il modello, quindi, si è deciso di adottare un'analisi a orizzonte infinito in modo da ottenere risultati più stabili. In questa fase verifichiamo se, per ciascuna fascia oraria, le statistiche generate dal nostro simulatore convergono ai valori reali.

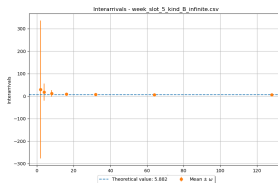
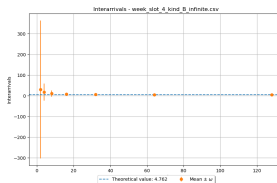
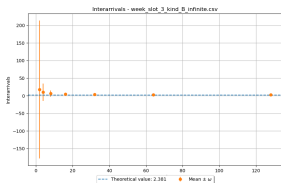
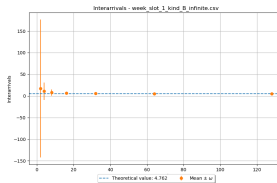
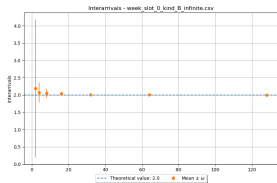
Analisi a orizzonte infinito

- ▶ Utilizzando $k = 128$ batches, ciascuno con $b = 1024$ campioni, l'autocorrelazione per lag $j = 1$ è inferiore a 0.2, contribuendo così alla stabilità e all'affidabilità dei risultati.
- ▶ L'analisi viene condotta per ciascun tipo di richiesta e per ciascuna statistica di interesse, coprendo tutte le fasce orarie sia nei giorni lavorativi che nei giorni del fine settimana.

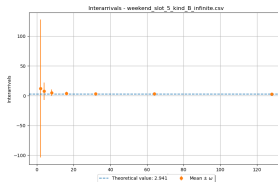
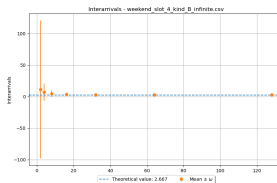
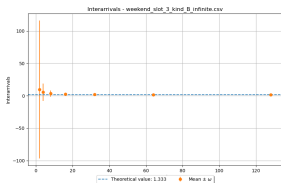
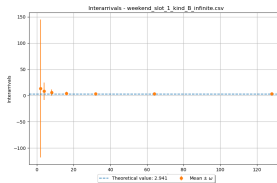
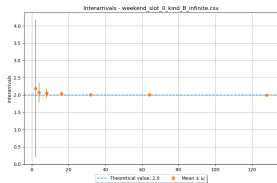
Interarrivi - bar

<i>B type - Interarrivals</i>					
	Slot	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	0	2.000 min	2.000 ± 0.014 min	✓	
	1	4.762 min	4.950 ± 0.389 min	✓	
	3	2.381 min	2.614 ± 0.477 min	✓	
	4	4.762 min	5.154 ± 0.811 min	✓	
	5	5.882 min	6.217 ± 0.074 min	✓	
Weekend	0	2.000 min	2.000 ± 0.014 min	✓	
	1	2.941 min	3.091 ± 0.322 min	✓	
	3	1.333 min	1.460 ± 0.259 min	✓	
	4	2.667 min	2.794 ± 0.268 min	✓	
	5	2.941 min	3.065 ± 0.285 min	✓	

Interarrivi al bar - Immagini week



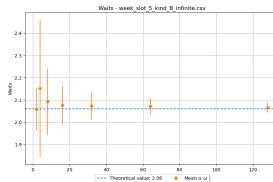
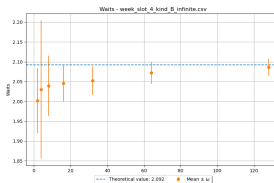
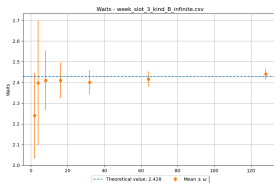
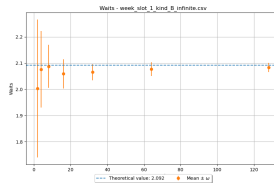
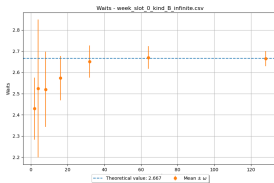
Interarrivi al bar - Immagini weekend



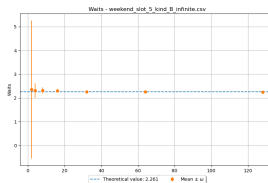
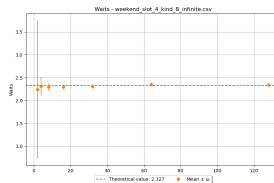
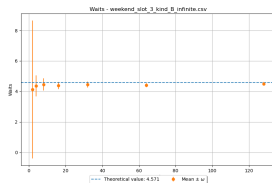
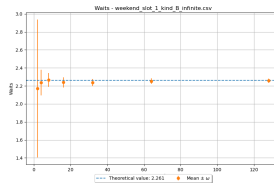
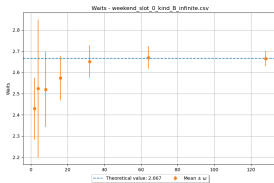
Tempi di risposta - bar

<i>B type - Waits</i>					
	Slot	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	0	2.667 min	2.665 ± 0.035 min	✓	
	1	2.092 min	2.083 ± 0.018 min	✓	
	3	2.428 min	2.440 ± 0.027 min	✓	
	4	2.092 min	2.086 ± 0.021 min	✓	
	5	2.060 min	2.067 ± 0.023 min	✓	
Weekend	0	2.667 min	2.665 ± 0.035 min	✓	
	1	2.261 min	2.257 ± 0.024 min	✓	
	3	4.571 min	4.499 ± 0.115 min	✓	
	4	2.327 min	2.339 ± 0.031 min	✓	
	5	2.261 min	2.246 ± 0.025 min	✓	

Tempi di risposta al bar - Immagini week



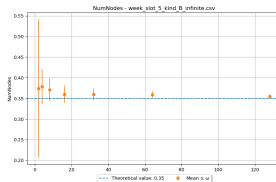
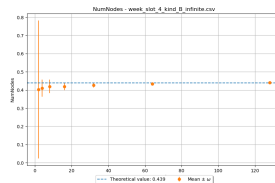
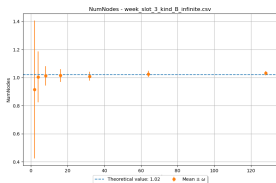
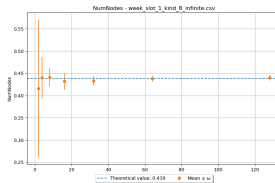
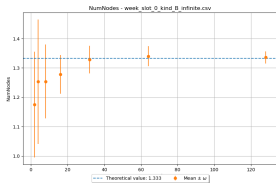
Tempi di risposta al bar - Immagini weekend



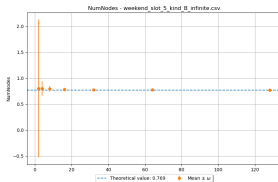
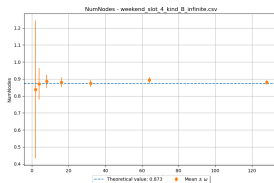
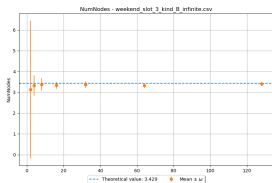
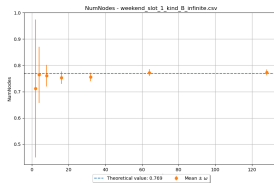
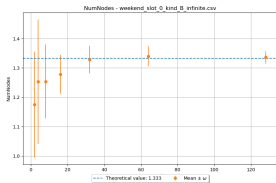
Numero di richieste nel centro - bar

<i>B type - Num. in the nodes</i>					
	Slot	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	0	1.333 min	1.336 ± 0.021 min	✓	
	1	0.439 min	0.440 ± 0.006 min	✓	
	3	1.020 min	1.031 ± 0.016 min	✓	
	4	0.439 min	0.440 ± 0.007 min	✓	
	5	0.350 min	0.355 ± 0.005 min	✓	
Weekend	0	1.333 min	1.336 ± 0.021 min	✓	
	1	0.769 min	0.773 ± 0.011 min	✓	
	3	3.429 min	3.399 ± 0.095 min	✓	
	4	0.873 min	0.881 ± 0.014 min	✓	
	5	0.769 min	0.769 ± 0.011 min	✓	

Numero di richieste nel centro al bar - Immagini week



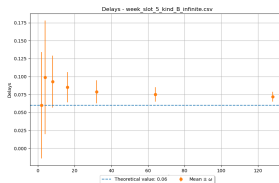
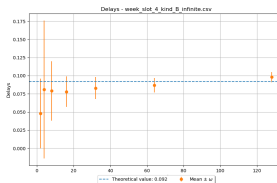
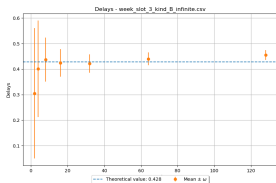
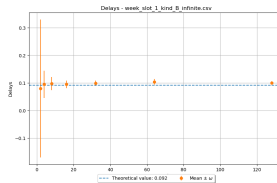
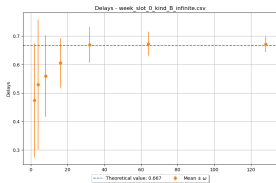
Numero di richieste nel centro al bar - Immagini weekend



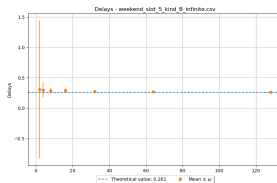
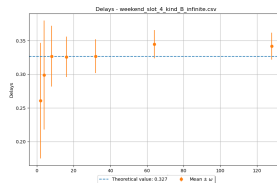
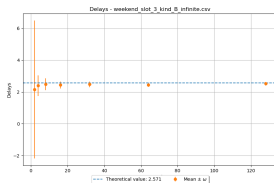
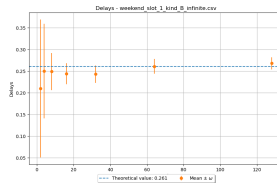
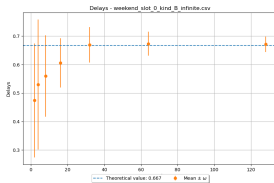
Tempi in coda - bar

<i>B type - Delays</i>					
	Slot	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	0	0.667 min	0.671 ± 0.027 min	✓	
	1	0.092 min	0.100 ± 0.006 min	✗	0.002
	3	0.428 min	0.455 ± 0.019 min	✗	0.008
	4	0.092 min	0.098 ± 0.007 min	✓	
	5	0.060 min	0.072 ± 0.007 min	✗	0.005
Weekend	0	0.667 min	0.671 ± 0.027 min	✓	
	1	0.261 min	0.268 ± 0.014 min	✓	
	3	2.571 min	2.519 ± 0.108 min	✓	
	4	0.327 min	0.342 ± 0.020 min	✓	
	5	0.261 min	0.265 ± 0.014 min	✓	

Tempi in coda al bar - Immagini week



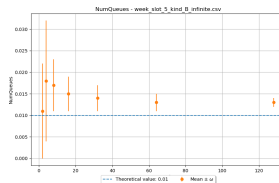
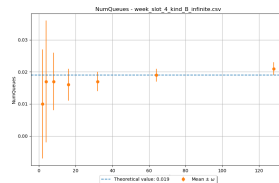
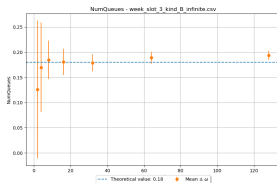
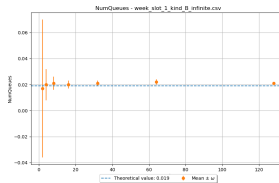
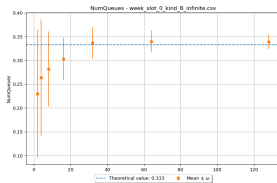
Tempi in coda al bar - Immagini weekend



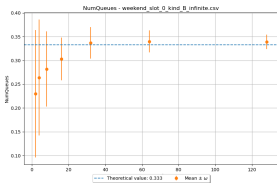
Numero di richieste in coda - bar

<i>B type - Num. in the queue</i>					
	Slot	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	0	0.333 min	0.339 ± 0.015 min	✓	
	1	0.019 min	0.021 ± 0.001 min	✗	0.001
	3	0.180 min	0.194 ± 0.005 min	✗	0.009
	4	0.019 min	0.021 ± 0.001 min	✗	0.001
	5	0.010 min	0.013 ± 0.001 min	✗	0.002
Weekend	0	0.333 min	0.339 ± 0.015 min	✓	
	1	0.089 min	0.093 ± 0.005 min	✓	
	3	1.929 min	1.910 ± 0.087 min	✓	
	4	0.123 min	0.130 ± 0.008 min	✓	
	5	0.089 min	0.092 ± 0.005 min	✓	

Numero di richieste in coda - Immagini week



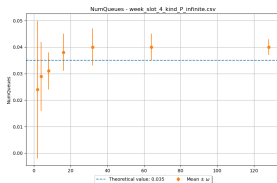
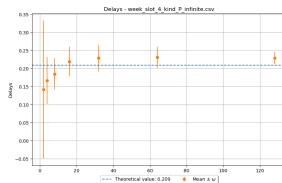
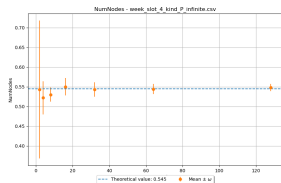
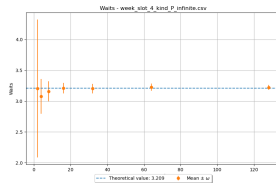
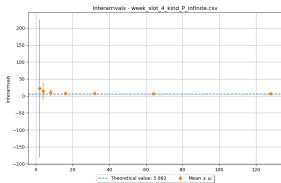
Numero di richieste in coda al bar - Immagini weekend



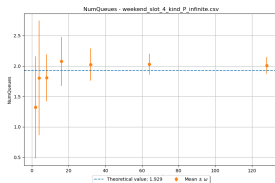
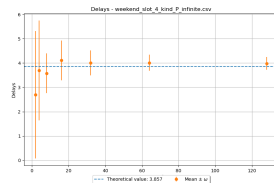
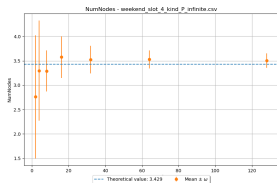
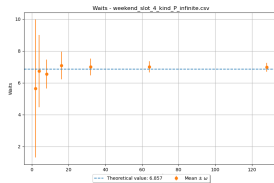
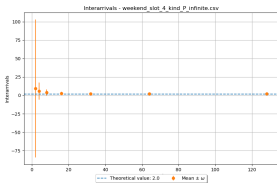
Statistiche - pizzeria

<i>P type - all statistics in the slot</i>					
Statistica		Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore
Week	Interarrivo	5.882 min	6.145 ± 0.499 min	✓	
	Attesa	3.209 min	3.223 ± 0.038 min	✓	
	Num. nel nodo	0.545 min	0.548 ± 0.009 min	✓	
	Ritardo	0.209 min	0.229 ± 0.017 min	✗	0.003
	Num. in coda	0.035 min	0.040 ± 0.003 min	✗	0.002
Weekend	Interarrivo	2.000 min	2.114 ± 0.228 min	✓	
	Attesa	6.867 min	6.979 ± 0.281 min	✓	
	Num. nel nodo	3.429 min	3.505 ± 0.150 min	✓	
	Ritardo	3.857 min	3.985 ± 0.267 min	✓	
	Num. in coda	1.929 min	2.010 ± 0.139 min	✓	

Statistiche pizzeria - Immagini week



Statistiche pizzeria - Immagini weekend



Conclusioni

- ▶ L'analisi a orizzonte infinito ha dimostrato che la maggior parte dei risultati converge in modo affidabile ai valori teorici previsti per il caso di studio in esame, confermando così l'accuratezza del nostro simulatore nel rappresentare il comportamento del sistema in uno stato stazionario.
- ▶ È interessante notare anche che nei casi in cui la frequenza di arrivo è leggermente più alta, come nel fine settimana, questi errori risultano azzerati anche con lo stesso seme 123. Ciò evidenzia come la bassa frequenza di arrivo possa influenzare la precisione delle stime di queste grandezze, persino nell'analisi a orizzonte infinito.

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

Verifica

Validazione

Analisi dei costi e dei guadagni

Altre statistiche transienti

- ▶ Motivazione: Permette effettivamente di valutare le entrate e le uscite per un sistema con queste specifiche su un periodo di 16 ore lavorative effettive.
- ▶ Robustezza: La simulazione di una giornata lavorativa è stata ripetuta 1024 volte. In ciascuna di esse sono state ripristinate tutte le statistiche mentre lo stato del generatore di numeri casuali è stato mantenuto inalterato, e calcolando un intervallo di confidenza al 95%.

$$m_B = 2$$

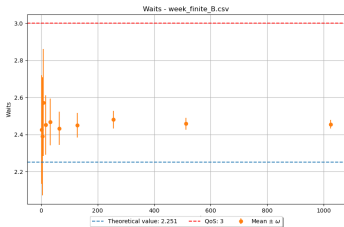
- Il valore teorico raramente rientra nell'intervallo di confidenza della statistica sperimentale.
- Tutti i requisiti di Qualità del Servizio (QoS) sono costantemente rispettati.

Analisi senza fattore gaussiano						
	Statistica	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore	Rispetta QoS
Week	Attesa di tipo B	2.251 min	2.455 ± 0.023 min	✗	0.181	✓
	Attesa di tipo P	3.209 min	3.212 ± 0.049 min	✓		✓
Weekend	Attesa di tipo B	2.524 min	2.761 ± 0.032 min	✗	0.205	✓
	Attesa di tipo P	6.857 min	5.813 ± 0.150 min	✗	0.894	✓

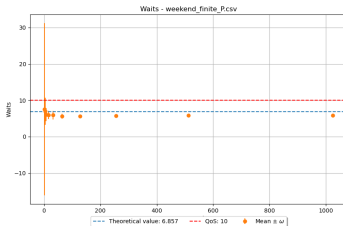
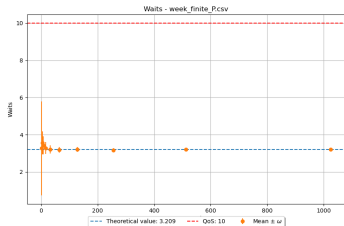
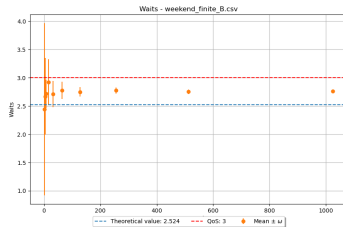
Analisi con fattore gaussiano						
	Statistica	Risultato teorico	Risultato sperimentale	Media nell'intervallo	Errore	Rispetta QoS
Week	Attesa di tipo B	2.251 min	2.184 ± 0.027 min	✗	0.040	✓
	Attesa di tipo P	3.209 min	3.082 ± 0.078 min	✗	0.049	✓
Weekend	Attesa di tipo B	2.524 min	2.987 ± 0.075 min	✗	0.388	✓
	Attesa di tipo P	6.857 min	3.209 ± 0.056 min	✗	3.592	✓

Senza fattore gaussiano - immagini

Week

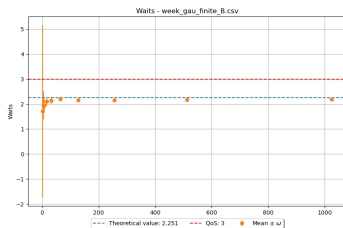


Weekend

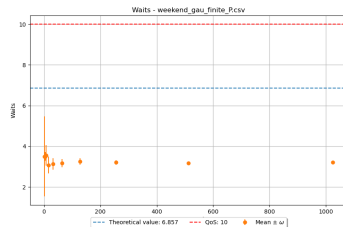
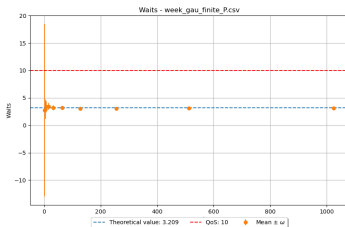
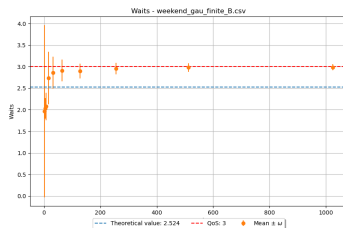


Con fattore gaussiano - immagini

Week



Weekend



Costi e guadagni

Spesa	Valore	Contributo mensile
Baristi	40,00€ al giorno per barista	2.240,00 € al mese
Pizzaiolo	40,00€ al giorno	1.400,00 € al mese
Bollette	2.750,00€ al mese	2.750,00€ al mese
Affitto	1.500,00€ al mese	1.500,00€ al mese
Fornitori	2.000,00€ al mese	2.000,00€ al mese
Totale		12.970,00€

	Tipo richiesta	Richieste week	Richieste weekend	Guadagno	Contributo mensile	Con IVA al 10%
No gauss.	Tipo B	275	395	5,00 € a richiesta	43.300,00 € al mese	38.970,00 €
	Tipo P	41	116	10 € a richiesta	17.480,00 € al mese	15.732,00 €
	Totale					54.702,00 €
Gauss.	Tipo B	77	120	5,00 € a richiesta	12.500,00€ al mese	11.250,00 €
	Tipo P	10	31	10,00 € a richiesta	4.480,00€ al mese	4.032,00 €
	Totale					15.282,00€

Il guadagno netto mensile risulta essere 41.732,00€ nel caso non si usi il ritardo gaussiano e 2.312,00€ nel caso in cui lo si utilizzi.

Introduzione

Modello concettuale

Modello delle specifiche

Modello computazionale

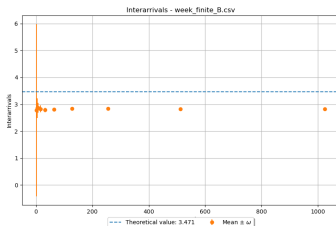
Verifica

Validazione

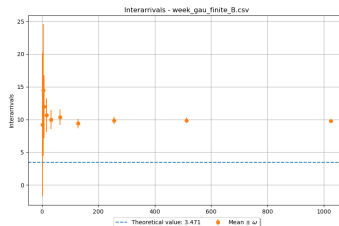
Analisi dei costi e dei guadagni

Altre statistiche transienti

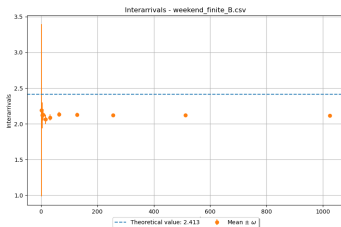
Interarrivi di tipo B



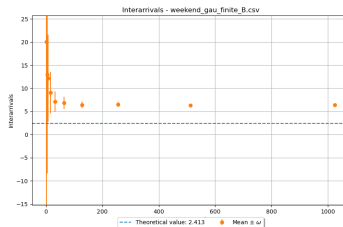
week - no gaussian factor



week - gaussian factor

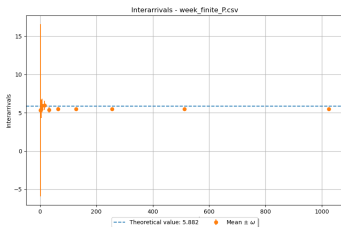


weekend - no gaussian factor

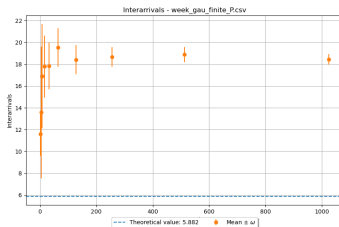


weekend - gaussian factor

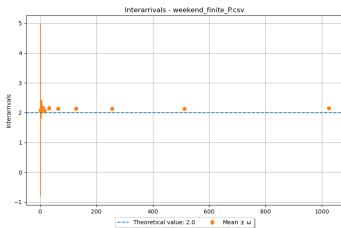
Interarrivi di tipo P



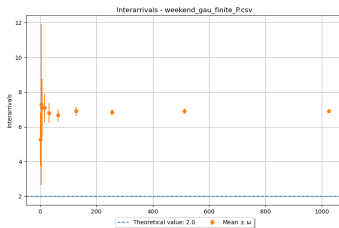
week - no gaussian factor



week - gaussian factor

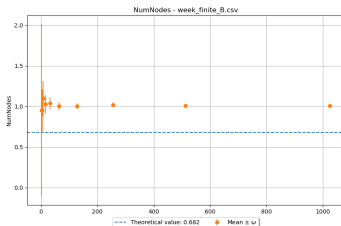


weekend - no gaussian factor

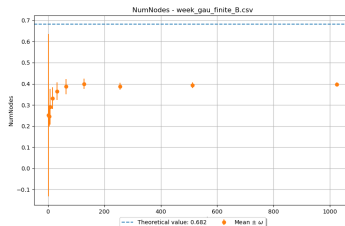


weekend - gaussian factor

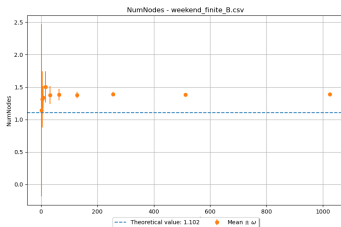
Popolazioni di tipo B nel centro



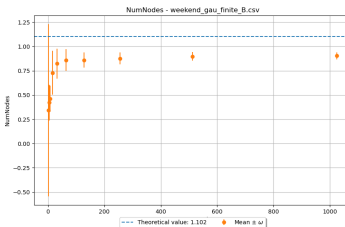
week - no gaussian factor



week - gaussian factor

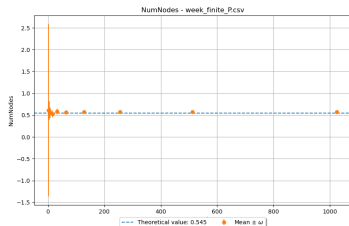


weekend - no gaussian factor

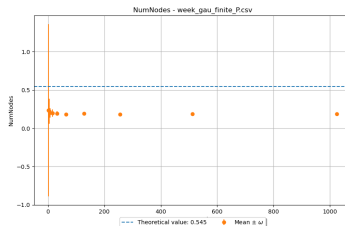


weekend - gaussian factor

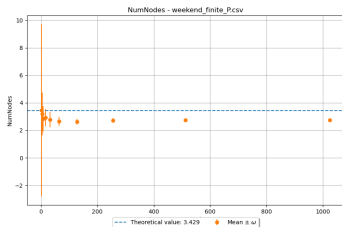
Popolazioni di tipo P nel centro



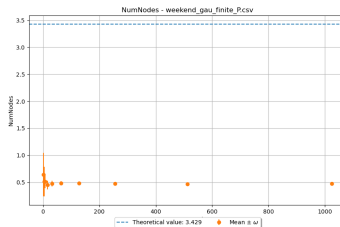
week - no gaussian factor



week - gaussian factor

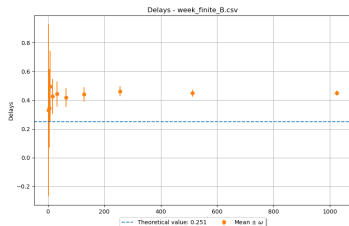


weekend - no gaussian factor

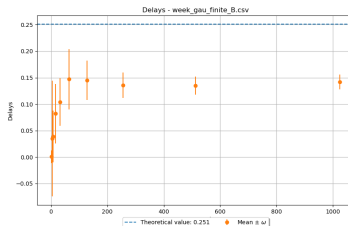


weekend - gaussian factor

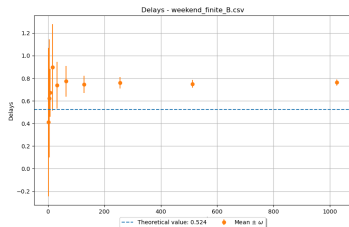
Ritardo di tipo B



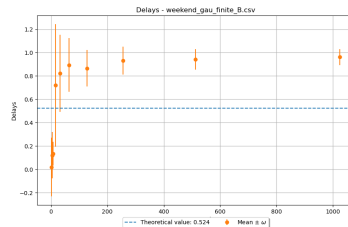
week - no gaussian factor



week - gaussian factor

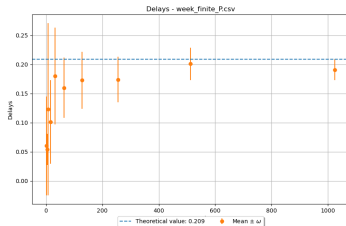


weekend - no gaussian factor

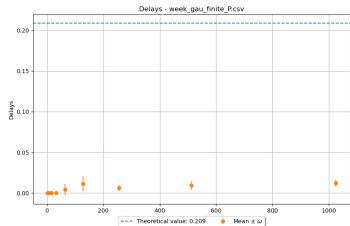


weekend - gaussian factor

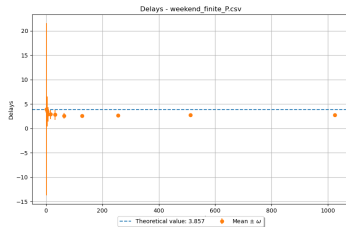
Ritardo di tipo P



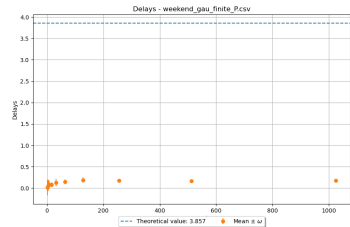
week - no gaussian factor



week - gaussian factor

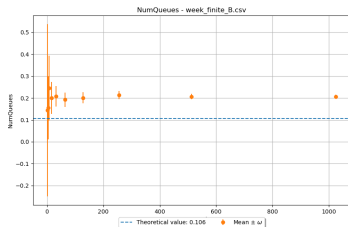


weekend - no gaussian factor

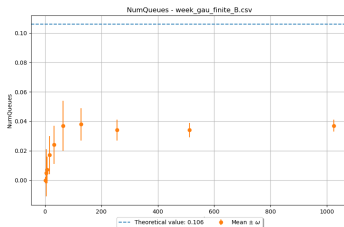


weekend - gaussian factor

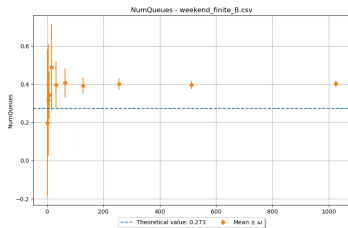
Popolazioni di tipo B in coda



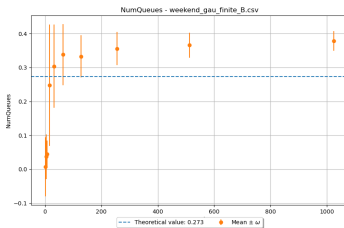
week - no gaussian factor



week - gaussian factor

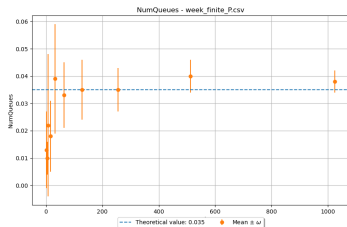


weekend - no gaussian factor

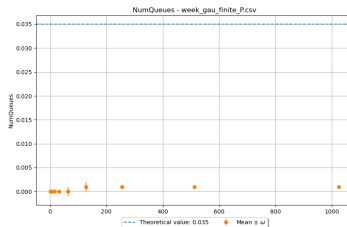


weekend - gaussian factor

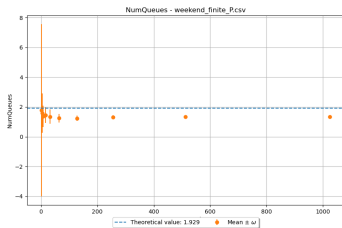
Popolazioni di tipo P in coda



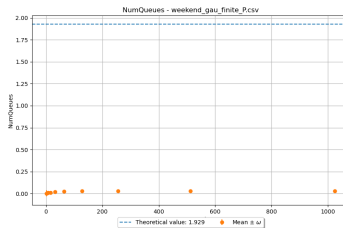
week - no gaussian factor



week - gaussian factor



weekend - no gaussian factor



weekend - gaussian factor

Conclusioni

- ▶ L'utilizzo di 3 serventi di tipo B porta a un costo mensile maggiorato di 2.240,00 €, rendendo il guadagno netto inferiore: la scelta migliore rimane $m_B = 2$.
- ▶ Il guadagno ottenuto introducendo il ritardo gaussiano sembra essere più realistico e più aderente alla realtà rispetto alla prima, poiché sono i risultati di uno scenario che tiene conto di condizioni più realistiche.