

Esame Software Engineering (AA 2024/25)

12 Giugno 2025 Lab. Colossus - Via salaria 113

Enrico Tronci

Computer Science Department, Sapienza University of Rome
Via Salaria 113 - 00198 Roma - Italy

tronci@di.uniroma1.it

<https://raise.uniroma1.it>

Esercizio 3 (25 punti)

L'unità di tempo per il presente esercizio è il secondo e $T = 1$ (secondi) è il time step per tutti i sistemi nel presente esercizio.

Tutti i sistemi hanno, per ciascun canale di input, una coda FIFO infinita. Ciascuna coda può essere realizzata con una FIFO molto grande, con una resize in caso di overflow ovvero, più semplicemente, usando il tipo `queue` del C++.

Un *customers* per un sito di *e-commerce* può essere modellato come una catena di Markov con un solo stato (0). Il customer resta nello stato 0 per un tempo τ scelto uniformemente a random nell'intervallo $[A, B]$.

Se il customer decide di fare una transizione (che lo riporta nello stato 0) allora invia sceglie un server s a random tra S server (numerati 1, 2, ..., S) ed invia al server s un valore $i > 0$. Questo vuol dire che il customer ha richiesto il prodotto i al server s .

Si noti che quando il customer permane nello stato 0 nulla viene inviato ai servers.

Ci sono N prodotti, quindi l'output del customer è $i \in \{1 \dots N\}$.

I servers sono tutti uguali.

Ciascun server ha un DB che, per ogni prodotto, contiene il numero di items disponibili. Quando un server riceve una richiesta per il prodotto j dalla FIFO di input si comporta come segue. Se nel DB ci sono items per il prodotto j richiesto il server aggiorna il DB e invia al customer il messaggio j . Se nel DB non ci sono items disponibili per il prodotto j , allora manda al customer il messaggio $-j$, che denota *mancata vendita*.

Per ognuno degli N prodotti, il DB di ciascun server è inizializzato a random con valori compresi tra 0 e K .

Quando un server invia al customer un messaggio del tipo $-j$ si comporta come segue:

- con probabilità p richiede il rifornimento di Q items del prodotto j

Per produrre il suo output il server impiega un tempo F quando non richiede items al fornitore. Impiega invece un tempo $G \geq F$ quando richiede items al fornitore.

1 Obiettivo

Il numero di mancate vendite al tempo t è il numero totale di mancate vendite entro il tempo t , cioè il numero di messaggi del tipo $-j$ ricevuti dal customer entro il tempo t .

Il rate di mancate vendite al tempo t è numero di mancate vendite al tempo t diviso t .

Il rate di mancate vendite è rate di mancate vendite alla fine della simulazione.

Si vuole riportare in uscita il valore atteso del rate di mancate vendite. Cioè il valore medio del rate di mancate vendite usando M simulazioni Montecarlo.

2 Formato dei parametri di input

I parametri della simulazione sono forniti nel file `parameters.txt`. Le righe del file `parameters.txt` sono formattate come segue.

- H <orizzonte di simulazione>
definisce l'orizzonte di simulazione (in secondi).
- T <valore in secondi del time step>
definisce il valore del parametro T .
- M <Montecarlo>
definisce il numero di simulazioni Montecarlo.
- N <numero prodotti>
definisce il numero di prodotti.
- A <valore minimo tempo attesa del customer>
definisce il valore minimo per il tempo τ .
- B <valore massimo tempo attesa customer>
definisce il valore massimo per il tempo τ .
- p <valore probabilità rifornimento>
definisce il valore del parametro p .
- S <numero di servers>
definisce il valore del parametro S .
- F <tempo calcolo del server>
definisce il valore del parametro F .
- G <tempo calcolo del server con fornitore>
definisce il valore del parametro G .

Un esempio di file `parameters.txt` è:

```
H 1.8  
T 0.5  
M 100  
N 3  
A 1.0  
B 2.0  
p 0.7  
S 4  
F 1  
G 2
```

Il file sopra indica che: l'orizzonte H di simulazione è 1.8 secondi, il time step è $T = 0.5$ secondi, il numero di simulazioni Montecarlo è 100, ci sono $N = 3$ prodotti, il tempo minimo tra un richiesta e l'altra è $A = 1.0$ secondi, il tempo massimo tra un richiesta e l'altra è $B = 2.0$ secondi, la probabilità di rifornimento è $p = 0.7$, ci sono $S = 4$ servers, con un tempo di calcolo $F = 1$ e $G = 2$ quando è richiesto un rifornimento.

3 Formato di output

L'output dell'esercizio è memorizzato nel file `results.txt` la cui prima riga è formattata come indicato nelle istruzioni generali.

Le rimanenti righe del file `results.txt` hanno il formato

$V < \text{rate mancate vendite} >$

Un esempio di file `results.txt` per il file `parameters.txt` dato sopra è:

```
2025-01-09-Mario-Rossi-1234567  
V 12.3
```