Progetto C2

Modulo di reti e TLC

A.A. 21/22

Prof. F.Martignon

3 GIUGNO

Nicola Zambelli
1053015

Descrizione del Caso

Si consideri un centralino telefonico con N linee in uscita non in grado di accodare le chiamate. Il traffico in arrivo che chiede di usare le linee sia di Poisson, mentre la durata delle chiamate sia una variabile casuale di tipo a, b o c (le distribuzioni di a, b e c sono a scelta dello studente). Si calcoli la probabilità di blocco sistema al variare del traffico A(erlang) offerto dimostrando che non varia al variare delle distribuzioni (confrontare con la B-Erlang).Il software accetta in input il traffico A, N e il tipo di distribuzione e i suoi eventuali parametri.

Implementazioni nel codice

- `Buffer.h` : sono stati inseriti due campi privati di tipo double, `LineeS` e `tot_rejected`, il primo indica i serventi al lavoro nel buffer, il secondo è un contatore di pacchetti rifiutati dal sistema.
- `Buffer.cpp:` inizializzazione dei nuovi campi di `Buffer.h` a 0.0.
- `Queue.h:` inserimento di una variabile `Sstat PBlocco`, che calcolerà la probabilità di blocco di ogni Run.
- `Queue.cpp:` inserimento delle variabili `Nlinee`, numero di linee richieste in input, e `prob_block`, che contiene la probabilità di blocco del sistema teorica. Tale variabile sarà utile per confrontare il valore ricavato dalla simulazione ed è calcolata dalla funzione:

  ```
  inline double erlang(int c, double a)
  ```

  che calcola la B-erlang in modo ricorsivo:

  $$p_m = B\left(m, \frac{\lambda}{\mu}\right) = \frac{\left(\frac{\lambda}{\mu}\right)^m \frac{1}{m!}}{\sum_{n=0}^{m} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}}$$

  Le modifiche ai parametri in input sono state fatte per chiedere all'utente la tipologia di distribuzione del servizio (durata delle chiamate) e il numero di linee. Le distribuzioni implementate sono :

  a. Di Poisson.
  b. Deterministica .
  c. Erlangiana di ordine 2.

  Un altro cambiamento degno di nota è l'aggiornamento della variabile statistica `PBlocco` che è calcolata come:

  $$PB = \frac{tot_{rejected}}{tot_{pack}} = \frac{tot_{rejected}}{tot_{risolti} + tot_{rejected}}$$

  Nel codice la variabile `tot_packs` indica le chiamate risolte, altre modifiche sono state fatte per gestire l'output dei risultati.

- `Packet.h:` al fine di implementare la Erlang 2 è stato inserito un campo stadio che contiene il riferimento allo stadio attuale del pacchetto, il costruttore setta lo stadio a uno.

- `Event.cpp:` nelle funzioni `body()` di `arrival` e `service` sono stati implementati dei `Case-switch statement` per migliorare la leggibilità del codice (caso 1: poisson, 2: deterministico, 3: Erlang 2).
  Per quanto riguarda il `body()` della classe `arrival,` i tre casi sono abbastanza simili come implementazione, se non per la generazione dei servizi, che nel primo caso è generata da una Poisson con media `duration`, nel secondo caso è pari a `duration` e infine, nel terzo caso, è pari ad $\frac{1}{2\mu}$ (`0.5 * duration`, primo stadio della Erlang 2). In tutti e tre i casi possono verificarsi due scenari: sistema non bloccato e sistema bloccato. Nel primo scenario, inseriamo il pacchetto nel buffer, incrementiamo i serventi al lavoro e generiamo il servizio. Nella seconda ipotesi, i serventi al lavoro sono uguali alle linee, allora scartiamo il pacco e aggiorniamo il contatore di pacchetti rifiutati.
  Il caso 1 e 2 del `body()` della classe `service` sono identici: si preleva il pacchetto dal buffer, si incrementa il contatore dei pacchetti risolti e si libera un servente. Nel caso 3, si preleva dal buffer il pacchetto e possono verificarsi due scenari: il pacchetto è allo stadio 1 o allo stadio 2. Nella prima ipotesi, si incrementa lo stadio del pacchetto e lo si reinserisce nel buffer senza liberare il servente, creando un nuovo servizio con una Poisson di media $\frac{1}{2\mu}$ (`0.5*duration`, secondo stadio della Erlang 2). Altrimenti, si è al secondo stadio, perciò, si aggiorna il contatore dei pacchetti risolti e si libera il servente.

Risultati

L'obiettivo del lavoro è dimostrare che la probabilità di blocco non dipende dal tipo di distribuzione di servizio (lunghezza delle chiamate), a tale scopo, si svolgono delle simulazioni con N fissato e si modifica di volta in volta la distribuzione del servizio, valutando la `Pblocco` al variare di `load` .
Tutte le simulazioni sono state effettuate con:

- Transient length (s) =10000
- Run length (s)=10000
- Number of runs=100

Tali valori permettono di migliorare la qualità statistica di `PBlocco`, ottenere un valore basso del *p-value* e intervalli di confidenza estremamente ridotti.
I valori ottenuti sono stati salvati nel file di testo `ValoriSimu.txt.`
Attraverso il software R, sono stati fatti i grafici dei vari scenari, il codice per generarli è contenuto nel file `plotReti.r`.
I Valori sono stati raccolti sui seguenti parametri del `load:`

- 0.1
- 0.3
- 0.5
- 0.7
- 0.9
- 0.999

- N=1

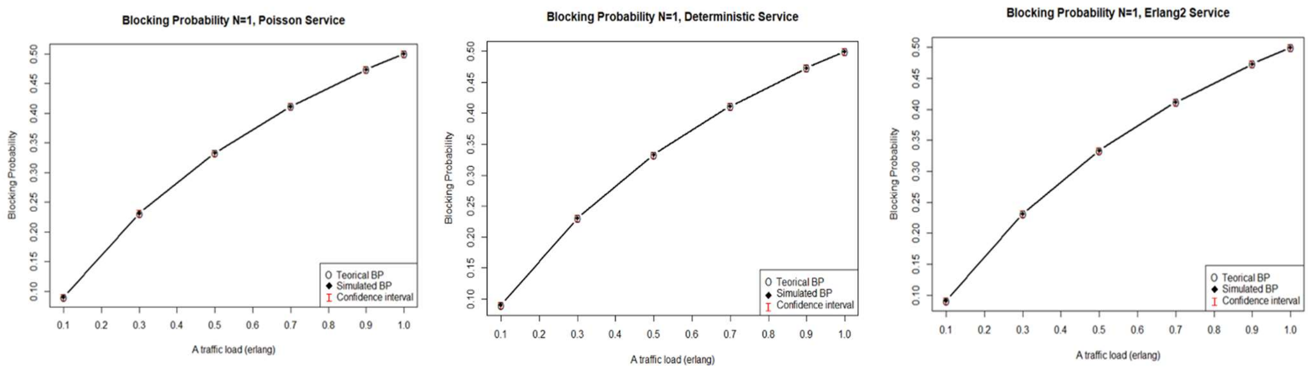

Figura 1: la probabilità di blocco al variare di A di un processo di servizio di Poisson(sinistra), Deterministico(centro) e Erlang2(sinistra), con Nlinee=1

- N=2

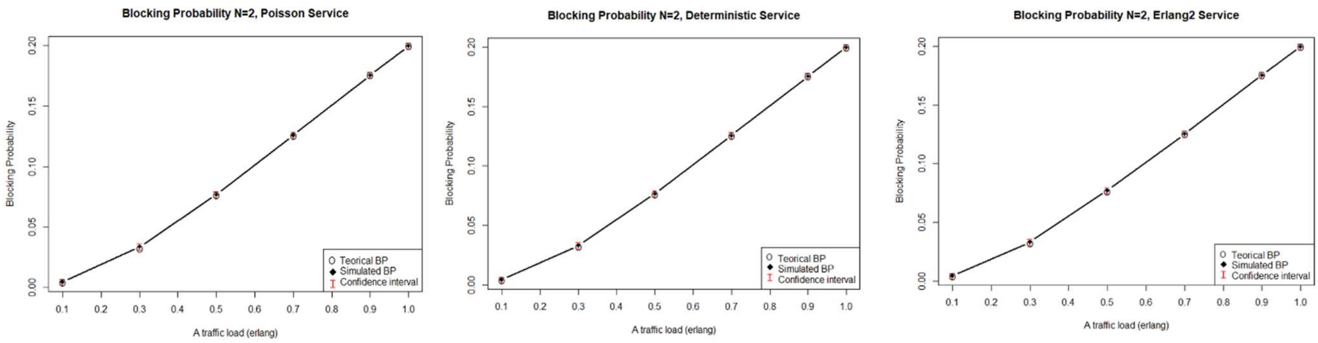

Figura 2: la probabilità di blocco al variare di A di un processo di servizio di Poisson(sinistra), Deterministico(centro) e Erlang2(sinistra), con Nlinee=2

- N=3

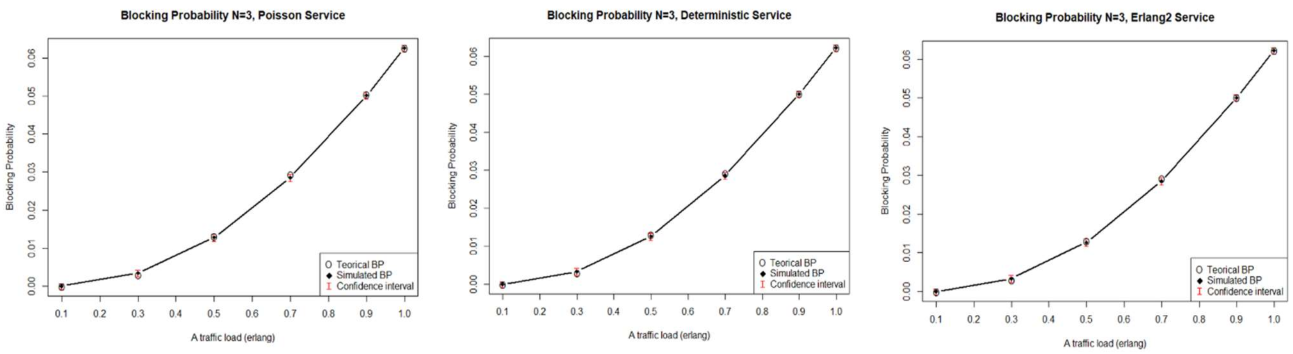

Figura 3: la probabilità di blocco al variare di A di un processo di servizio di Poisson(sinistra), Deterministico(centro) e Erlang2(sinistra), con Nlinee=3

Conclusioni

Il simulatore ha permesso di ottenere risultati vicinissimi alla realtà, con ampiezza degli intervalli di confidenza dell' ordine di $1 * 10^{-3}$, nei casi peggiori.
È facile notare come la probabilità di blocco non è influenzata dalla distribuzione del processo di servizio o, nel caso di linee telefoniche, dalla distribuzione della lunghezza delle chiamate. Gli unici parametri che influenzano la probabilità di blocco sono ρ (A, traffic-load [erlang]) e N il numero di linee (o serventi). In particolare, al crescere di ρ cresce la probabilità di blocco, viceversa, al crescere di N diminuisce la probabilità di blocco.
Una volta fissato ρ, il parametro $\frac{1}{\mu}$ (`duration`) è ininfluente sulla probabilità di blocco.
Tale valore viene chiesto ugualmente come input nel caso si vogliano implementare delle soluzioni per l'analisi dei tempi, ritardi ecc.