

Performance Modeling Of Computer Systems And Networks

Relazione Progetto 2017/18

Simone Falvo
smvfal@gmail.com



UNIVERSITA' degli STUDI di ROMA
TOR VERGATA

<i>INDICE</i>	1
---------------	---

Indice

1	Introduzione	2
2	Modello Concettuale	2
3	Modello di Specifica	3
3.1	Metriche	3
3.1.1	Tempi di risposta	4
3.1.2	Popolazione media	5
3.1.3	Throughput	5
3.2	Eventi	5
3.3	Vincoli	6
4	Modello Analitico	6
5	Bibliografia	6

1 Introduzione

In questa relazione viene descritta una possibile soluzione ad un problema di edge computing in cui è richiesto di ottimizzare le prestazioni di un sistema di computazione cloud, calibrando i parametri di un algoritmo per l'inoltro di task utente verso un cloudlet ed un server remoto.

Il sistema è in grado di eseguire task di due diverse classi in un cloudlet, situato ad un "hop" di distanza dall'utente, fintanto che le risorse lo consentono, altrimenti vengono inoltrati ed eseguiti in un server remoto. Inoltre, se la somma del numero di job presenti nel cloudlet è uguale al valore di soglia S e se vi è almeno un job di classe 2 in esecuzione, questo viene fatto migrare dal cloudlet al server remoto per far posto ad un task di classe 1 in arrivo, che ha una maggiore "priorità" di esecuzione nel cloudlet.

L'obiettivo principale è quello di trovare il valore ottimale del parametro S affinché sia minimizzato il tempo di risposta medio.

Il problema è stato affrontato definendo un modello a code per il sistema in modo tale da poter valutare, tramite l'analisi e la simulazione, i tempi di risposta e le altre metriche di performance come il throughput e la popolazione media.

Il modello è stato validato analiticamente tramite lo studio dello stato del sistema a regime, calcolandone la distribuzione stazionaria e valutando le metriche di interesse in relazione ai possibili scenari.

La simulazione è stata realizzata implementando in linguaggio C un programma basato su eventi ed i risultati sono stati raccolti ed elaborati utilizzando il metodo "batch means" per avere una stima del comportamento del sistema a regime.

Nel seguito di questo documento verranno presentati i vari modelli di astrazione del sistema con i risultati che ne derivano, ed infine verrà fatto un confronto in relazione ai possibili scenari analizzati.

2 Modello Concettuale

Nel modello concettuale il sistema viene descritto come un sistema di code connesse tra loro (figura 1), di quest'ultimo si distinguono i seguenti componenti:

Cloudlet : nodo del sistema che comprende un numero N di server senza coda che operano in parallelo con tassi di servizio specifici per classi di job.

Cloud : nodo del sistema che comprende un numero infinito di server senza coda che operano in parallelo con tassi di servizio specifici per classi di job.

Centro di setup : nodo del sistema che modella la fase di setup di un job interrotto, composto da un numero infinito di server senza coda che operano in parallelo con un tasso di servizio pari a $1/E[S_{setup}]$. Un job interrotto transita per questo centro prima di prendere servizio nel cloud ed il tempo ivi trascorso corrisponde al tempo necessario alla ri-esecuzione del job.

Controllore : componente che implementa la logica di routing in base all'occupazione del cloudlet tenendo conto del parametro di soglia S , non costituisce un centro di servizio in quanto la sua funzione è limitata all'inoltro dei job.

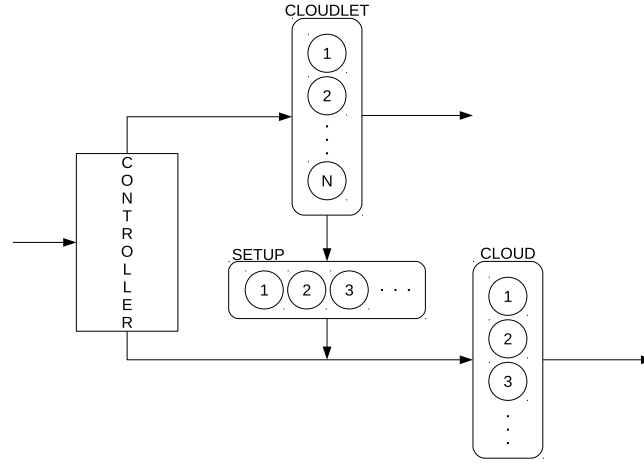


Figura 1: Modello concettuale del sistema

Ogni job che arriva al sistema passa prima per il controllore, il quale decide quale debba essere il nodo di esecuzione (cloud o cloudlet). Se arriva un job di classe 1 ed il cloudlet si trova nella condizione per cui almeno un suo server ha in esecuzione un job di classe 2 e la somma dei job presenti ha raggiunto il valore di soglia S , allora tale server deve sospendere il job e sostituirlo con quello appena arrivato di classe 1. Il job di classe 2 interrotto verrà poi inoltrato al server remoto (cloud), non prima però di aver trascorso un tempo necessario alla propria riesecuzione sul nuovo nodo, tale fase è modellata assumendo che il job venga eseguito in un centro intermedio (centro di setup).

3 Modello di Specifica

In questo modello vengono definite le variabili e le equazioni necessarie al calcolo delle metriche, inoltre viene descritta la logica con cui il sistema evolve a seguito degli eventi che si verificano ed i vincoli a cui esso deve sottostare.

3.1 Metriche

Di principale interesse sono le variabili che compongono lo stato del sistema, ovvero quelle che lo caratterizzano completamente, esse tengono conto del numero di job in servizio suddivisi per classe e per nodo di esecuzione ad ogni istante di tempo e sono indicate nella tabella 1. Le altre variabili utili al calcolo

$n_{1clet}(t)$	numero di job di classe 1 nel cloudlet al tempo t
$n_{2clet}(t)$	numero di job di classe 2 nel cloudlet al tempo t
$n_{1cloud}(t)$	numero di job di classe 1 nel cloud al tempo t
$n_{2cloud}(t)$	numero di job di classe 2 nel cloud al tempo t
$n_{setup}(t)$	numero di job in fase di setup al tempo t

Tabella 1: Variabili che compongono lo stato del sistema

delle statistiche del sistema sono indicate nella tabella 2 e riguardano i tempi di risposta ed i completamenti, sempre suddivise per classe e nodo di esecuzione.

$s_{1clet,i}$	tempo di servizio dell'i-esimo job di classe 1 eseguito nel cloudlet
$s_{2clet,i}$	tempo di servizio dell'i-esimo job di classe 2 eseguito nel cloudlet
$s_{1cloud,i}$	tempo di servizio dell'i-esimo job di classe 1 eseguito nel cloud
$s_{2cloud,i}$	tempo di servizio dell'i-esimo job di classe 2 eseguito nel cloud
$s_{intr,i}$	tempo di servizio nel cloudlet dell'i-esimo job interrotto
$s_{setup,i}$	tempo di setup dell'i-esimo job interrotto
$c_{1clet}(t)$	numero di job di classe 1 completati nel cloudlet al tempo t
$c_{2clet}(t)$	numero di job di classe 2 completati nel cloudlet al tempo t
$c_{1cloud}(t)$	numero di job di classe 1 completati nel cloud al tempo t
$c_{2cloud}(t)$	numero di job di classe 2 completati nel cloud al tempo t
$n_{intr}(t)$	numero di job interrotti al tempo t

Tabella 2: Variabili tempi di servizio e completamenti

3.1.1 Tempi di risposta

$$E[T_{jclet}] = E[S_{jclet}] = \frac{\sum_{i=1}^{c_{jclet}(t^*)} s_{jclet,i}(t^*)}{\left(c_{jclet}(t^*)\right)} \quad j = 1, 2 \quad (1)$$

$$E[T_{jcloud}] = E[S_{jcloud}] = \frac{\sum_{i=1}^{c_{jcloud}(t^*)} s_{jcloud,i}}{\left(c_{jcloud}(t^*)\right)} \quad j = 1, 2 \quad (2)$$

$$E[T_1] = E[S_1] = \frac{\sum_{i=1}^{c_{1clet}(t^*)} s_{1clet,i} + \sum_{i=1}^{c_{1cloud}(t^*)} s_{1cloud,i}}{\left(c_{1clet}(t^*) + c_{1cloud}(t^*)\right)} \quad (3)$$

$$E[T_2] = E[S_2] = \frac{\sum_{i=1}^{c_{2clet}(t^*)} s_{2clet,i} + \sum_{i=1}^{c_{2cloud}(t^*)} s_{2cloud,i} + \sum_{i=1}^{n_{intr}(t^*)} (s_{intr,i} + s_{setup,i})}{\left(c_{2clet}(t^*) + c_{2cloud}(t^*)\right)} \quad (4)$$

$$E[T] = E[S] = \frac{\sum_{i=1}^{c_{1clet}(t^*)} s_{1clet,i} + \sum_{i=1}^{c_{1cloud}(t^*)} s_{1cloud,i} + \sum_{i=1}^{c_{2clet}(t^*)} s_{2clet,i}}{\left(c_{1clet}(t^*) + c_{1cloud}(t^*) + c_{2clet}(t^*) + c_{2cloud}(t^*)\right)} +$$

$$+ \frac{\sum_{i=1}^{c_{2cloud}(t^*)} s_{2cloud,i} + \sum_{i=1}^{n_{intr}(t^*)} (s_{intr,i} + s_{setup,i})}{\left(c_{1clet}(t^*) + c_{1cloud}(t^*) + c_{2clet}(t^*) + c_{2cloud}(t^*)\right)} \quad (5)$$

3.1.2 Popolazione media

$$E[N_{jclet}] = \frac{1}{t_{stop} - t_{start}} \int_{t_{start}}^{t_{stop}} n_{jclet}(t) dt \quad j = 1, 2 \quad (6)$$

$$E[N_{jcloud}] = \frac{1}{t_{stop} - t_{start}} \int_{t_{start}}^{t_{stop}} n_{jcloud}(t) dt \quad j = 1, 2 \quad (7)$$

$$E[N_j] = E[N_{jclet}] + E[N_{jcloud}] \quad j = 1, 2 \quad (8)$$

$$E[N_{clet}] = E[N_{1clet}] + E[N_{2clet}] \quad (9)$$

$$E[N_{cloud}] = E[N_{1cloud}] + E[N_{2cloud}] \quad (10)$$

$$E[N] = E[N_{cloud}] + E[N_{clet}] = E[N_1] + E[N_2] \quad (11)$$

3.1.3 Throughput

$$X_{jclet} = \frac{c_{jclet}(t_{stop})}{t_{stop} - t_{start}} \quad j = 1, 2 \quad (12)$$

$$X_{jcloud} = \frac{c_{jcloud}(t_{stop})}{t_{stop} - t_{start}} \quad j = 1, 2 \quad (13)$$

$$X_j = X_{jclet} + X_{jcloud} \quad j = 1, 2 \quad (14)$$

$$X_{clet} = X_{1clet} + X_{2clet} \quad (15)$$

$$X_{cloud} = X_{1cloud} + X_{2cloud} \quad (16)$$

$$X = X_1 + X_2 = X_{clet} + X_{cloud} \quad (17)$$

3.2 Eventi

Lo stato del sistema evolve a seguito di eventi di vario tipo:

1. Arrivo di un job di classe 1
2. Arrivo di un job di classe 2
3. Partenza di un job
4. Setup

Algoritmo 1

Algorithm 1 Logica del sistema in base agli eventi

```

Arrivo di un job di classe 1:
if  $n_{1clet}(t) = N$  then
    esecuzione su cloud
     $s_{1cloud} \leftarrow s_{1cloud} + s_{1cloud,i}$ 
     $n_{1cloud}(t') \leftarrow n_{1cloud}(t) + 1$ 
else if  $n_{1clet}(t) + n_{2clet}(t) < S$  then
    esecuzione su cloudlet
     $s_{1clet} \leftarrow s_{1clet} + s_{1clet,i}$ 
     $n_{1clet}(t') \leftarrow n_{1clet}(t) + 1$ 
end if
 $S \leftarrow 0$ 
Arrivo di un job di classe 2:
Partenza di un job:
Setup:

```

3.3 Vincoli**4 Modello Analitico**

(figura 2)

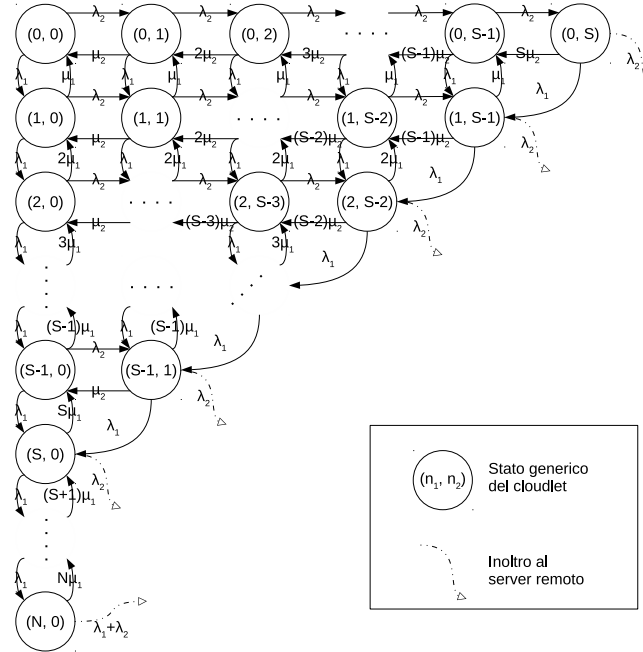


Figura 2: CTMC cloudlet

5 Bibliografia