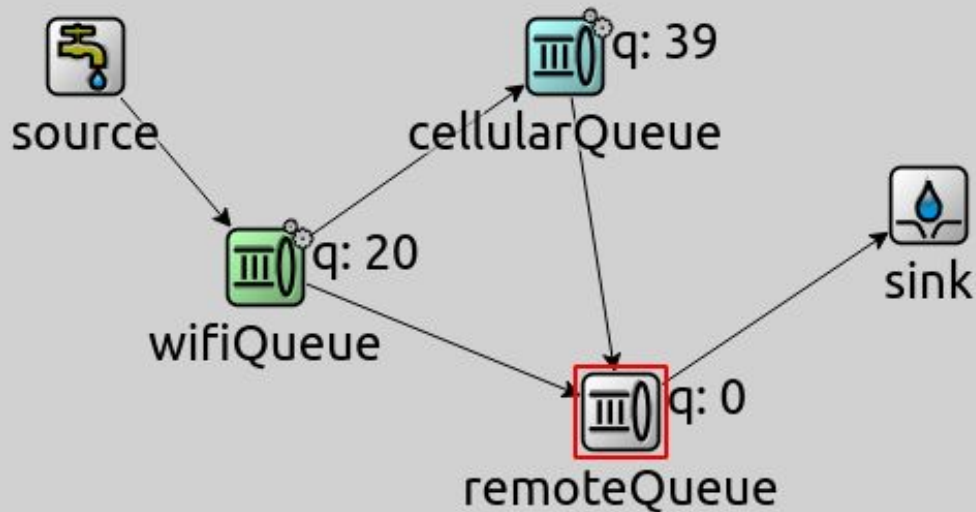


FullOffloadingNetwork



Stochastic Analysis of Delayed Mobile Offloading in Heterogeneous Networks

Matteo Del Vecchio

Corso di Simulazione di Sistemi

Informatica Magistrale - UniBo - A.A. 2018/2019

Indice

Indice	1
Introduzione	2
Strumenti	2
Full Offloading Model	3
Metriche	4
Mean Response Time (MRT)	4
Mean Energy Consumption (MEC)	4
Energy-Response Weighted Product (ERWP)	4
Implementazione	5
Simulazione	8
Parametri	8
Configurazioni	9
Transiente	9
Risultati	12
Intervalli di Confidenza	14
Conclusioni	23
Appendice A - Grafici dal Paper	24
Bibliografia	25

Introduzione

In questa relazione verrà illustrato il lavoro che è stato svolto nell'ambito del Corso di Simulazione di Sistemi, consistente nella simulazione di un modello reale e della relativa analisi statistica dei dati risultanti.

Partendo dal paper che dà il titolo a questa relazione, consultabile al seguente [link](#), è stato implementato e simulato un modello che permette l'offloading dell'elaborazione in remoto, utilizzando la rete WiFi o cellulare per la trasmissione. In particolare, si è posta l'attenzione sull'ottimizzazione del consumo energetico e dei tempi di risposta. Tale studio nasce dal presupposto che spesso, operazioni complesse e dispendiose sono deleterie per l'esecuzione su un dispositivo come un cellulare, sebbene oggi siano molto performanti.

Si parla infatti di Delay-Tolerant e Delay-Sensitive Applications; le prime rappresentano quell'insieme di applicazioni la cui trasmissione dei dati non richiede una particolare urgenza, rendendo quindi più rilevante il miglioramento del consumo energetico. Nel secondo caso, invece, la risposta è l'aspetto più rilevante per l'utilizzo, rendendo necessario ottimizzare il tempo di risposta e la latenza.

Nel paper in analisi vengono proposti due modelli: Partial Offloading Model e Full Offloading Model; entrambi si basano sul fatto di utilizzare un network veloce (WiFi) o lento (cellulare) per la trasmissione. La differenza sostanziale consiste nel fatto che nel modello Partial, i job vengono continuamente processati, indipendentemente dalla rete disponibile, mentre nel caso del modello Full, si assume di essere collegati o meno ad una rete WiFi e, solo in caso quest'ultima non sia disponibile, la computazione può ricadere sull'uso della rete cellulare.

Nelle sezioni successive verrà descritto nel dettaglio il modello analizzato, ovvero il Full Offloading Model.

Strumenti

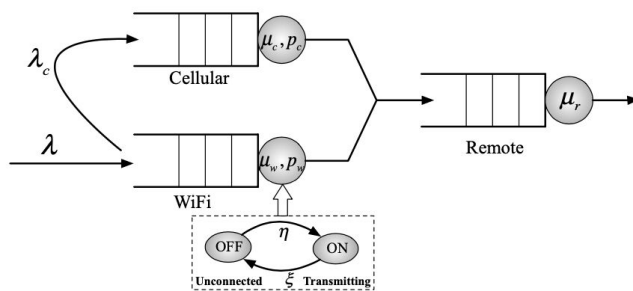
Gli strumenti utilizzati per la creazione di questo progetto sono principalmente tre: l'applicazione OMNeT++[\[2\]](#), il linguaggio Python ed il linguaggio Bash/Shell.

OMNeT++, versione 5.5.1, è un simulatore di eventi discreti, eseguito su una macchina virtuale con sistema operativo Linux Ubuntu 18.04; esso viene utilizzato per la simulazione di numerosi modelli reali, soprattutto in ambito di reti. Nel caso corrente è stato sfruttato per la creazione di una rete di code, attraverso la *queueinglib* fornita ed i moduli aggiuntivi implementati.

Una volta concluse le simulazioni, tutti i dati sono stati elaborati tramite svariati script: Bash per quanto riguarda l'esecuzione delle simulazioni e l'estrazione dei dati, mentre Python per l'analisi statistica vera e propria.

Inoltre, l'intero sviluppo del progetto è stato versionato tramite Git e l'implementazione è disponibile open source su GitHub, al seguente [link](#).

Full Offloading Model



Il modello Full Offloading preso in analisi in questo progetto è costituito da tre code, come mostrato nello schema. In particolare, le code WiFi e Cellular sono code M/M/1 mentre quella Remote è M/M/∞. Come ci suggerisce la notazione di Kendall, ci troviamo di fronte a code FIFO che descrivono un processo

poissoniano, quindi con tempi di interarrivo e di servizio dati da distribuzioni esponenziali, aventi un solo server (eccezione fatta per la coda Remote, la quale ha un numero infinito di server; in altre parole, ogni job viene immediatamente servito, senza la possibilità di formare una coda).

Tutti i job che entrano nel sistema sono automaticamente inviati alla coda WiFi per poterne fare offloading tramite la rete veloce; il server di tale coda però può trovarsi in due stati: ON, quando il dispositivo è connesso ad una rete WiFi e può trasmettere, oppure OFF, quando non c'è nessun tipo di servizio. L'avvicinarsi degli stati viene gestito tramite due distribuzioni esponenziali rispettivamente di parametri ξ e η , la prima per passare allo stato OFF e la seconda per passare allo stato ON.

Durante il periodo in cui lo stato è OFF, i job in coda possono diventare impazienti: nel momento in cui arrivano in coda, ad ogni job viene assegnata una deadline, anch'essa data da una distribuzione esponenziale di parametro r . Questo valore rappresenta il tempo massimo che il job è disposto ad aspettare prima di essere interamente servito. Se lo stato del server non ritorna ON prima della scadenza di tale deadline, il job decide di lasciare la coda WiFi per procedere all'offloading tramite rete Cellulare.

Va inoltre notato che la coda WiFi è gestita come una priority queue: il job da servire viene scelto in base al tempo rimanente della deadline, dando quindi priorità ai job arrivati durante lo stato OFF invece che a quelli arrivati durante lo stato ON.

Infine, il sistema è definito da un arrival rate λ , la coda WiFi da un service rate μ_w , quella Cellular μ_c e quella remote μ_r .

Metriche

In queste sezioni verranno brevemente descritte le metriche utilizzate per la valutazione dei risultati del modello. Esse sono derivate analiticamente ma verrà anche descritto come sono state calcolate partendo dai dati della simulazione.

Mean Response Time (MRT)

Il tempo medio di risposta del sistema viene descritto dalla relazione in basso; essa prende in considerazione il numero medio di job presenti in ogni coda (c : cellular, w : wifi, r : remote). Avendo a disposizione la simulazione, tale metrica è facilmente calcolabile come la media dei tempi di risposta di tutti i job presenti nel sistema. In particolare, questo dato risulta essere la somma del tempo totale in coda e di quello totale in servizio, accumulato in ogni coda in cui il job è passato.

$$\mathbb{E}[T] = \mathbb{E}[\mathbb{E}[T_i]] = \sum_{i \in \{c, w, r\}} \frac{\lambda_i}{\lambda} \mathbb{E}[T_i] = \frac{1}{\lambda} \sum_{i \in \{c, w, r\}} \mathbb{E}[N_i]$$

Mean Energy Consumption (MEC)

Il consumo energetico medio del sistema è definito dalla relazione in basso; in particolare, consiste nella sommatoria del tasso di utilizzazione delle code WiFi e Cellular, moltiplicate per un coefficiente energetico pari a 0,7 W e 2,5 W, rispettivamente per WiFi e Cellular. Partendo dalla simulazione e per ogni coda, si è calcolata la somma dei tempi di servizio per poi moltiplicarli per il relativo coefficiente. In seguito, la somma dei consumi delle code è stata divisa per il numero totale di job, in modo da ottenere un consumo medio.

$$\mathbb{E}[\xi] = \mathbb{E}[\mathbb{E}[\xi_i|i]] = \sum_{i \in \{w, c\}} \frac{1}{\lambda} \mathbb{E}[P_i] = \frac{1}{\lambda} \sum_{i \in \{w, c\}} p_i \cdot Pr\{N_i > 0\} = \frac{1}{\lambda} \sum_{i \in \{w, c\}} p_i \cdot \rho_i$$

Energy-Response Weighted Product (ERWP)

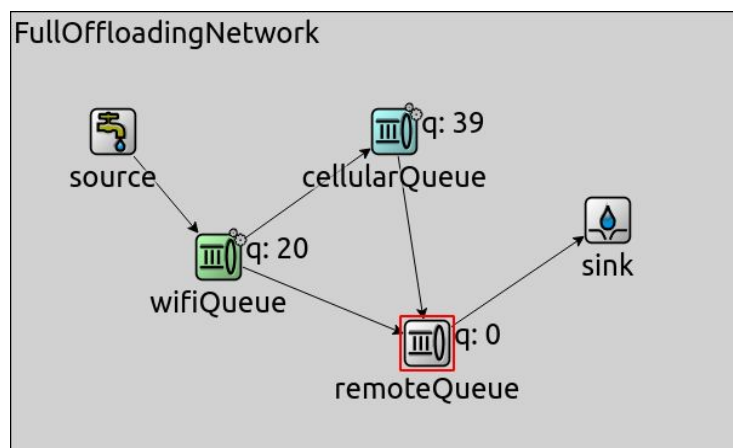
Tale metrica è stata introdotta per poter facilmente valutare ed ottenere il giusto compromesso tra il consumo energetico e il tempo di risposta, rimanendo facilmente trattabile analiticamente e mitigando il problema della diversa scala di valori.

Essa è definita come:

$$ERWP = \mathbb{E}[\xi]^\omega \cdot \mathbb{E}[T]^{1-\omega}$$

In particolare, si può notare come intervenendo sul parametro ω sia possibile dare maggior peso all'energia oppure al tempo; ponendo $\omega = 0,5$ si dà la stessa importanza ad entrambe le metriche.

Implementazione



L'immagine qui a lato rappresenta lo schema logico del sistema implementato nel simulatore. Sebbene la *queueinglib* metta a disposizione varie classi per modellare reti di code, tutti i moduli utilizzati sono stati implementati da zero (sulla falsariga della libreria), in modo da descrivere un comportamento il più fedele possibile al paper.

Come primo elemento c'è **source** (definito dalla classe *LimitedSource*), un generatore di job; la modifica al modulo originario consiste nell'aggiunta di alcuni controlli per essere certi che il numero totale di job (*numJobs*) venga considerato solo da quando lo stato del sistema sia stazionario.

Abbiamo poi **sink**, (definito dalla classe *LimitedSink*), in cui vengono inviati tutti i job nel momento in cui il sistema ha terminato la loro elaborazione e si occupa di registrare varie statistiche. La modifica consiste nell'introduzione del segnale *totalResponseTime* che tiene traccia del response time di ogni job, oltre ai controlli per terminare la simulazione nel momento in cui l'ultimo dei *numJobs* previsti sia stato elaborato.

cellularQueue e **remoteQueue** (entrambe definite dalla classe *QueueCustom*) sono code analoghe a quelle implementate nella *queueinglib*, con l'unica differenza che è stato aggiunto un segnale, *jobServiceTime*, che tenga traccia del service time di ogni job ma relativo alla singola coda; ciò si è reso necessario per poter successivamente calcolare il consumo energetico medio.

Infine, abbiamo **wifiQueue** (definita dalla classe *OffloadingQueue*) che è il modulo principale del sistema. I punti chiave riguardano l'implementazione del meccanismo di cambio stato del server e la relativa sospensione/ripresa del servizio, la gestione delle deadline associate ai job arrivati durante lo stato OFF, l'uso come priority queue attraverso un'apposita funzione di comparazione (*compareFunction*). Sono stati inoltre aggiunti vari segnali per poter raccogliere i dati necessari allo studio del transiente, relativamente a tutte le distribuzioni utilizzate in questo modulo.

Un aspetto particolarmente *challenging* si è rivelata proprio l'associazione delle deadline ai job e viceversa per due motivi: nel momento in cui si verificava una deadline, era necessario risalire velocemente al job per poterlo inviare alla coda Cellular, oppure, nel momento in cui il job terminava il servizio, la deadline doveva essere annullata. La soluzione consiste nell'utilizzare il *contextPointer* presente in ogni *cMessage*. Il seguente codice mostra il set della deadline al job in arrivo (Codice 1) e la gestione della deadline nel momento in cui essa si verifica (Codice 2).

Codice 1

```
void OffloadingQueue::arrival(Job *job) {
    job->setTimestamp();
    job->setQueueCount(job->getQueueCount() + 1);
    EV << job << " queue count: " << job->getQueueCount() << endl;

    // WIFI is OFF so add deadline to jobs
    if (!wifiAvailable) {
        cMessage *deadlineMsg = new cMessage("deadline_reached");
        deadlineMsg->setContextPointer(job);
        job->setContextPointer(deadlineMsg);

        simtime_t deadlineLength = par("deadlineDistribution").doubleValue();
        emit(deadlineDistrib, deadlineLength);
        simtime_t deadlineTime = simTime() + deadlineLength;
        EV << "Deadline set for job " << job << "; firing time: " << deadlineTime << endl;
        scheduleAt(deadlineTime, deadlineMsg);
    }
}
```

Codice 2

```
std::string jobName = msg->getName();
if (jobName.std::string::compare("deadline_reached") == 0) {
    if (msg->getContextPointer()) {
        Job *job = (Job *)msg->getContextPointer();
        msg->setContextPointer(nullptr);
        EV << "DEADLINE REACHED! WIFI status: " << wifiAvailable << " - Associated job: " << job << endl;

        if (hasGUI()) {
            std::string text = std::string("Deadline for ") + std::string(job->getName());
            bubble(text.c_str());
        }

        if (job == servicedJob) {
            if (endServiceMsg->isScheduled())
                cancelEvent(endServiceMsg);

            prepareNextJobIfAny();
        }
        else if (job == suspendedJob) {
            suspendedJob = nullptr;
            if (endServiceMsg->isScheduled())
                cancelEvent(endServiceMsg);
        }
        else
            queue.remove(job);

        emit(queueLengthSignal, length());
        send(job, "out", 0);
        delete msg;
    }
    else
        EV << "DEADLINE REACHED!" << endl;
}
```


Di seguito invece, viene mostrato il codice che gestisce il cambio di stato del server WiFi (Codice 3), quello che si occupa della sospensione del job in servizio, insieme all'aggiornamento delle varie informazioni temporali (Codice 4) e quello che schedula il nuovo cambio di stato (Codice 5). Va notato che, graficamente, lo stato ON viene indicato dalla coda colorata di verde mentre quello OFF di rosso.

Codice 3

```
else if (msg == wifiStatusMsg) {
    wifiAvailable = !wifiAvailable;
    EV << "WIFI STATUS CHANGED! Now is " << (wifiAvailable ? "ON" : "OFF") << "\n";

    // wifi OFF -> ON
    if (wifiAvailable) {
        if (suspendedJob) resumeService(suspendedJob);
        else prepareNextJobIfAny();
        updateNextStatusChangeTime();
    }
    // wifi ON -> OFF
    else {
        updateNextStatusChangeTime();
        if (servicedJob)
            suspendService(servicedJob);
    }
}
```

Codice 4

```
void OffloadingQueue::suspendService(Job *job) {
    cancelEvent(endServiceMsg);
    simtime_t startTime = job->getTimestamp();
    simtime_t elapsedTime = simTime() - startTime;
    simtime_t remainingTime = curJobServiceTime - elapsedTime;
    scheduleAt(nextStatusChangeTime + remainingTime, endServiceMsg);
    job->setTotalServiceTime(job->getTotalServiceTime() + elapsedTime);

    job->setTimestamp();
    suspendedJob = job;
    servicedJob = nullptr;
    EV << "Service SUSPENDED! Received: " << job << " - Suspended: " << suspendedJob << " - Serviced: " << servicedJob << endl;
    EV << "Elapsed service time for " << job << ": " << elapsedTime << endl;
    EV << "Remaining service time for " << job << ": " << remainingTime << endl;
    EV << "New scheduleAt time: " << nextStatusChangeTime + remainingTime << endl;
}
```

Codice 5

```
void OffloadingQueue::updateNextStatusChangeTime() {
    simtime_t nextChange = (wifiAvailable) ? par("wifiStateDistribution").doubleValue() : par("cellularStateDistribution").doubleValue();
    if (wifiAvailable) emit(wifiActiveTime, nextChange);
    else emit(cellActiveTime, nextChange);
    nextStatusChangeTime = simTime() + nextChange;
    scheduleAt(nextStatusChangeTime, wifiStatusMsg);
    EV << "Next WIFI status change time: " << nextStatusChangeTime << endl;
}
```

Infine, viene mostrato il codice che rende la coda una priority queue: in caso due job hanno la deadline viene data priorità a quello con tempo rimanente minore; se solo uno dei due ha deadline, questo avrà priorità sull'altro job, altrimenti viene data priorità al job creato prima (Codice 6).

Codice 6

```

int OffloadingQueue::compareFunction(cObject *a, cObject *b) {
    Job *jobA = check_and_cast<Job *>(a);
    Job *jobB = check_and_cast<Job *>(b);
    cMessage *aDeadline = (cMessage *)jobA->getContextPointer();
    cMessage *bDeadline = (cMessage *)jobB->getContextPointer();
    if (aDeadline && bDeadline) {
        simtime_t now = simTime();
        simtime_t timeLeftA = aDeadline->getArrivalTime() - now;
        simtime_t timeLeftB = bDeadline->getArrivalTime() - now;
        if (timeLeftA < timeLeftB) return -1;
        else if (timeLeftA > timeLeftB) return 1;
        else return 0;
    }
    else if (aDeadline || bDeadline) {
        if (aDeadline) return -1;
        else return 1;
    }
    else {
        simtime_t creationTimeA = jobA->getCreationTime();
        simtime_t creationTimeB = jobB->getCreationTime();
        if (creationTimeA < creationTimeB) return -1;
        else if (creationTimeA > creationTimeB) return 1;
        else return 0;
    }
}

```

Simulazione

D'ora in avanti verrà usata la notazione s per indicare l'unità di tempo utilizzata all'interno di OMNeT++. Le simulazioni vengono lanciate tramite script Bash che, in base al nome passatogli come argomento, esegue la relativa configurazione, fino all'analisi statistica dei dati. Per ottenere i grafici relativi alle metriche, il discriminante principale è il reneging rate r da cui dipendono i valori delle deadline assegnati alla relativa distribuzione esponenziale. Inizialmente, lo studio era previsto solamente per il rate $0,025 \pm 20\%$, ovvero 33, 40 e 50 min di deadline. Si è scelto invece di utilizzare 37 valori diversi, compresi tra 20 min e 300 min; inoltre, sono stati utilizzati 100 seed scelti randomicamente. Così facendo, ogni configurazione genera 3700 simulazioni diverse.

Parametri

Prima di procedere alla simulazione vera è propria è stato necessario calcolare i parametri da assegnare alle distribuzioni. Dall'articolo si evincono i seguenti dati:

- Job arrival rate: $\lambda = 0,5$ pkt/min quindi $0,008333$ pkt/s
- Service rate coda Remote: $\mu_r = 1$
- Coefficiente energetico coda WiFi: $p_w = 0,7$ W, coda Cellular: $p_c = 2,5$ W
- Durata media della disponibilità WiFi: $\xi = 52$ min = 3120 s
- Durata media della disponibilità solo Cellular: $\eta = 25,4$ min = 1524 s
- Dimensione media di un job: $E[X] = 10$ MB
- Velocità media rete WiFi: $s_w = 2$ Mbps

- Velocità media rete Cellular: $s_c = 200$ Kbps

Effettuando le opportune conversioni e applicando le formule indicate, otteniamo i parametri mancanti:

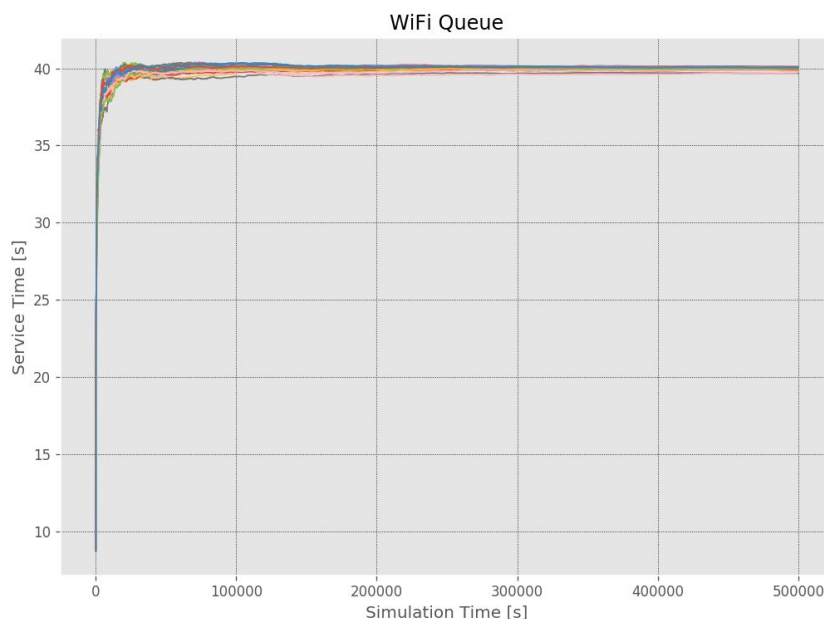
- Job interarrival time: $1 / \lambda = 120$ s
- Service rate WiFi: $\mu_w = s_w / E[X] = 0,025$ pkt/s quindi service time: $1 / \mu_w = 40$ s
- Service rate Cellular: $\mu_c = s_c / E[X] = 0,0025$ pkt/s quindi service time: $1 / \mu_c = 400$ s

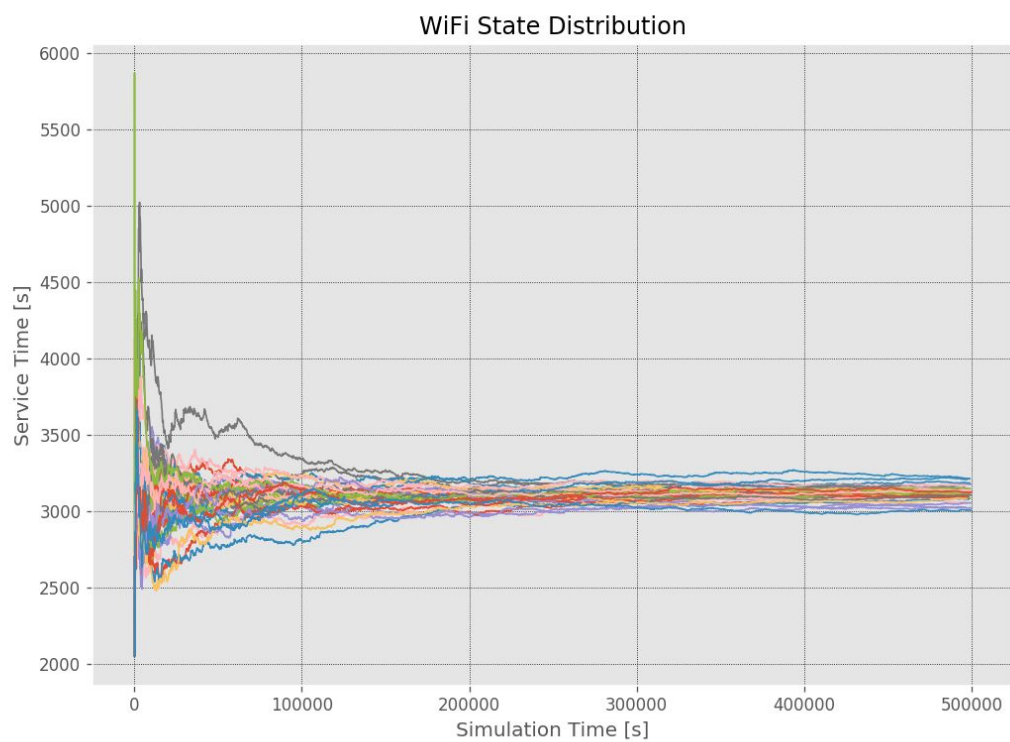
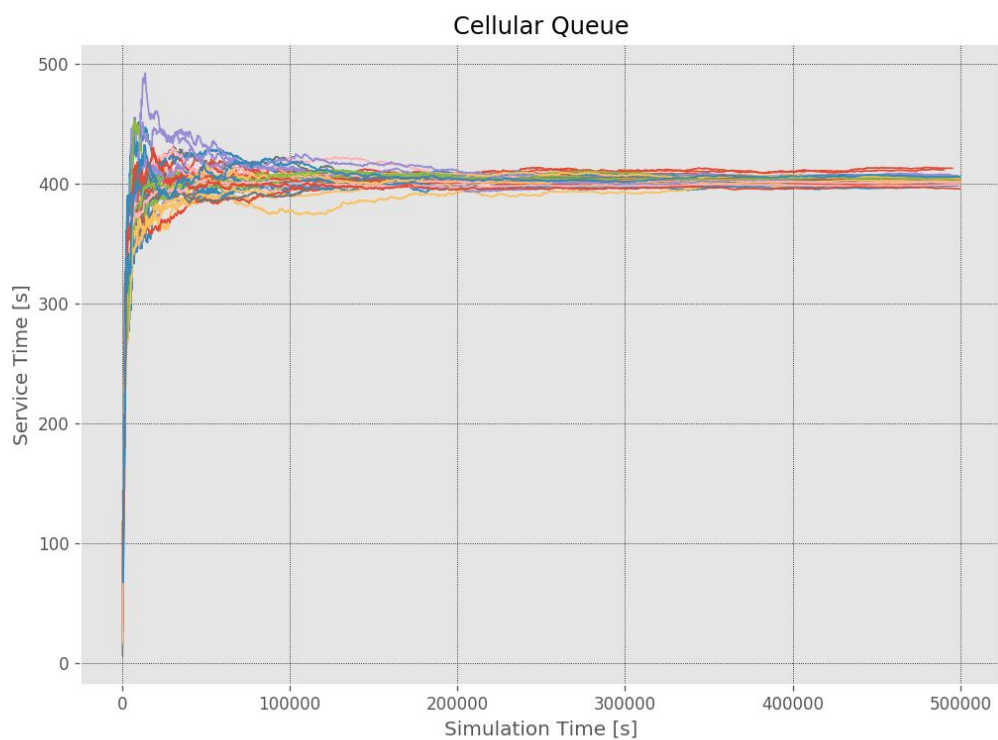
Configurazioni

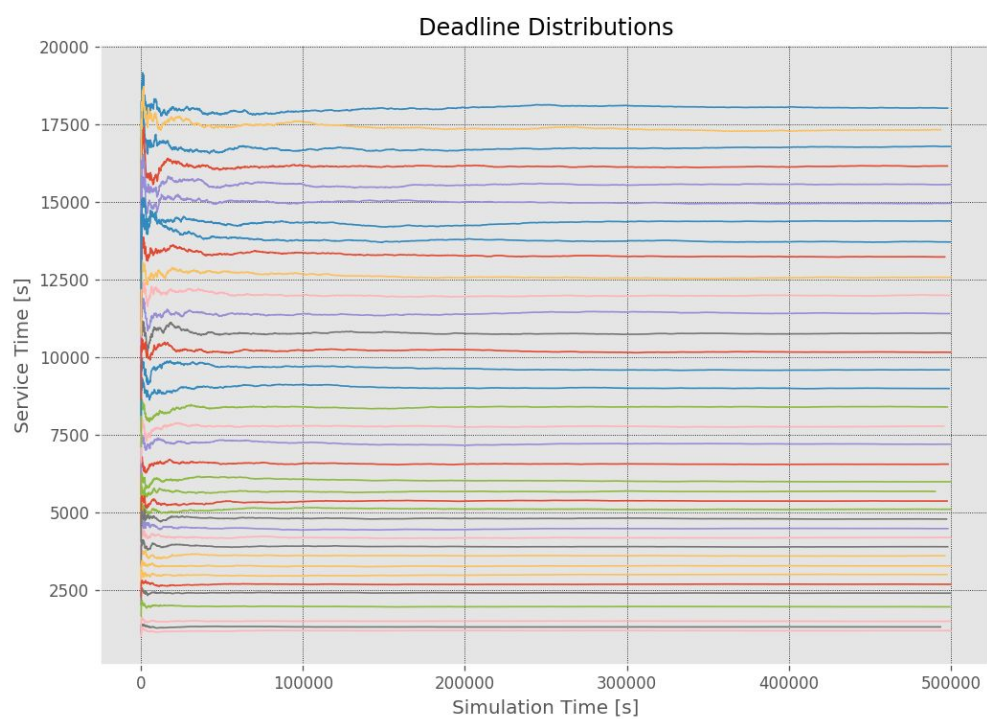
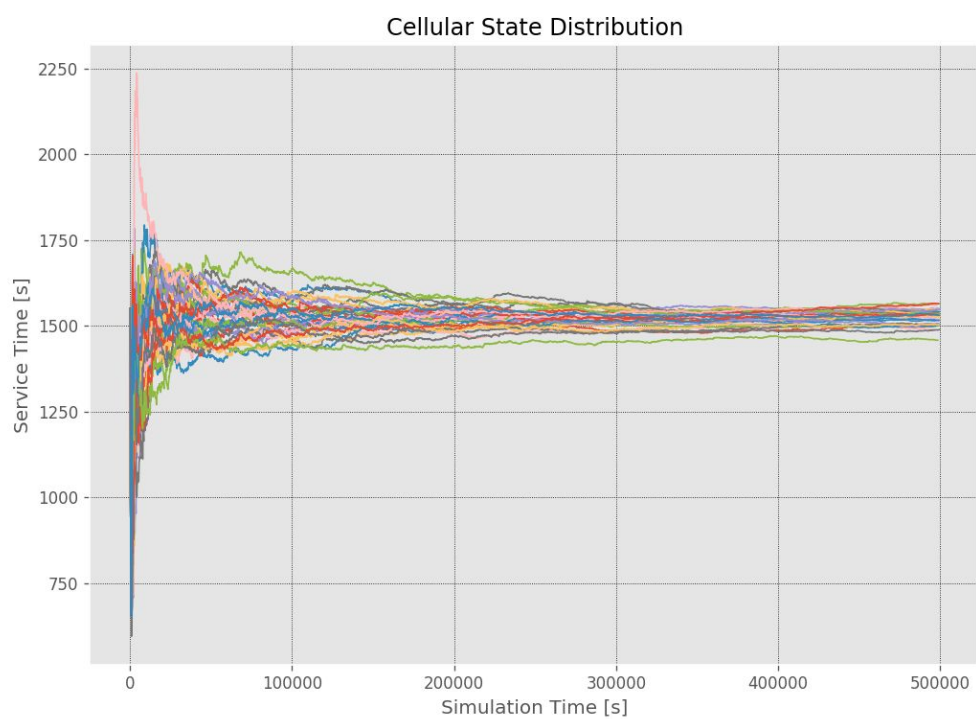
Il file `omnetpp.ini` è il punto principale in cui indicare i parametri delle simulazioni e le relative configurazioni, ognuna indicata tramite una sezione. La sezione `[General]` contiene tutti i dati condivisi tra tutte le simulazioni, come i tempi di servizio, i seed, le distribuzioni ed il nome della rete implementata. In seguito sono presenti la configurazione `SetupAnalysis`, per lo studio del transiente, e quella `BatchExecution`, per l'esecuzione delle simulazioni una volta che il sistema ha raggiunto lo stato stazionario. Quest'ultima contiene il numero di job totali da considerare per ogni simulazione (1000) ed il *warmup-period*, ovvero quel lasso di tempo necessario al sistema per stabilizzarsi, prima di iniziare a registrare i dati.

Transiente

Prima di poter registrare i dati è necessario studiare il transiente iniziale del sistema, ovvero quel lasso di tempo in cui esso ancora non è stabile e quindi i dati risultanti non presentano regolarità statistica. Questa fase consiste quindi nell'analizzare la convergenza delle svariate distribuzioni che governano il modello, attraverso grafici. Per ogni distribuzione viene mostrato il relativo grafico *averaged* rispetto a tutti i seed; tutte le simulazioni sono state eseguite per 500000 s.



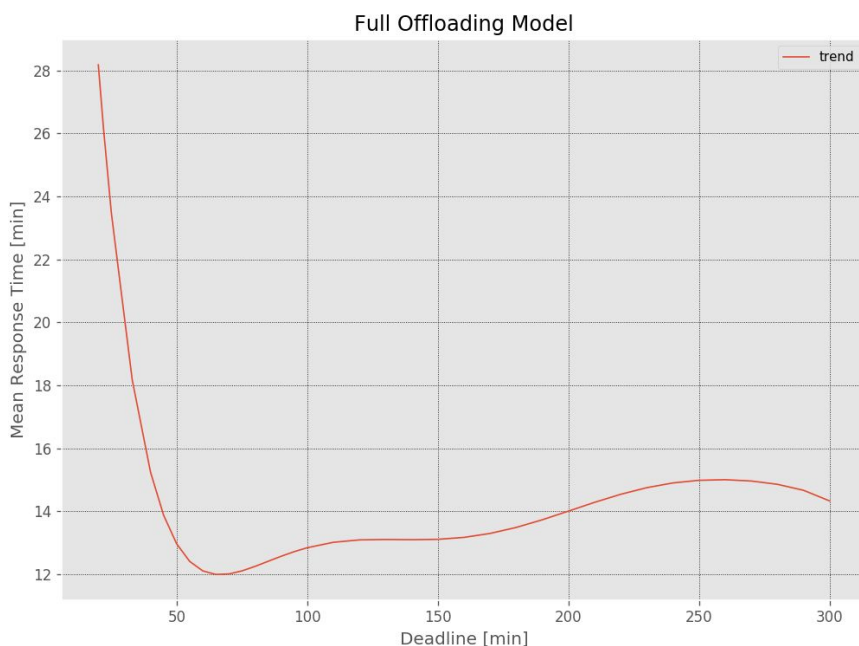




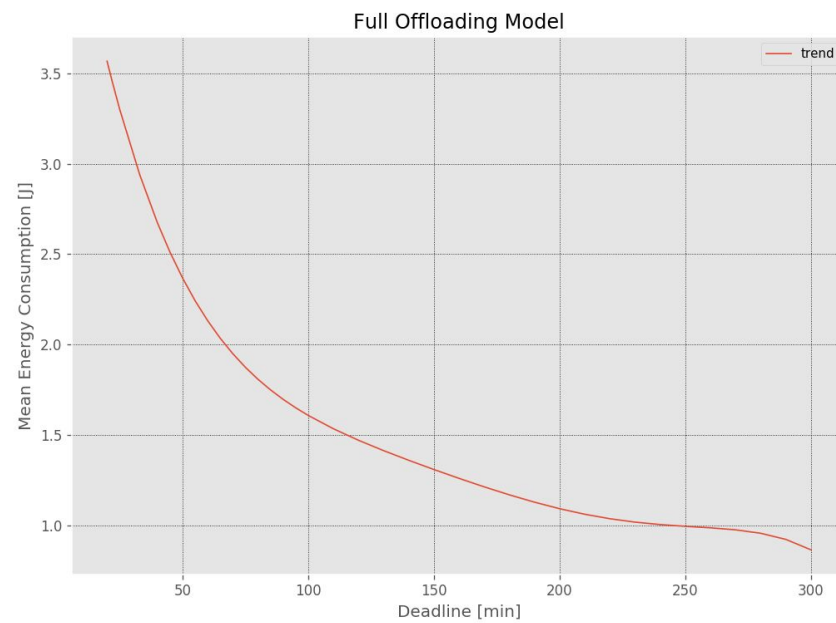
In seguito all'osservazione dei risultato, è stato scelto un *warmup-period* di 200000 s, sovrastimando il punto in cui tutte le distribuzioni iniziano a convergere.

Risultati

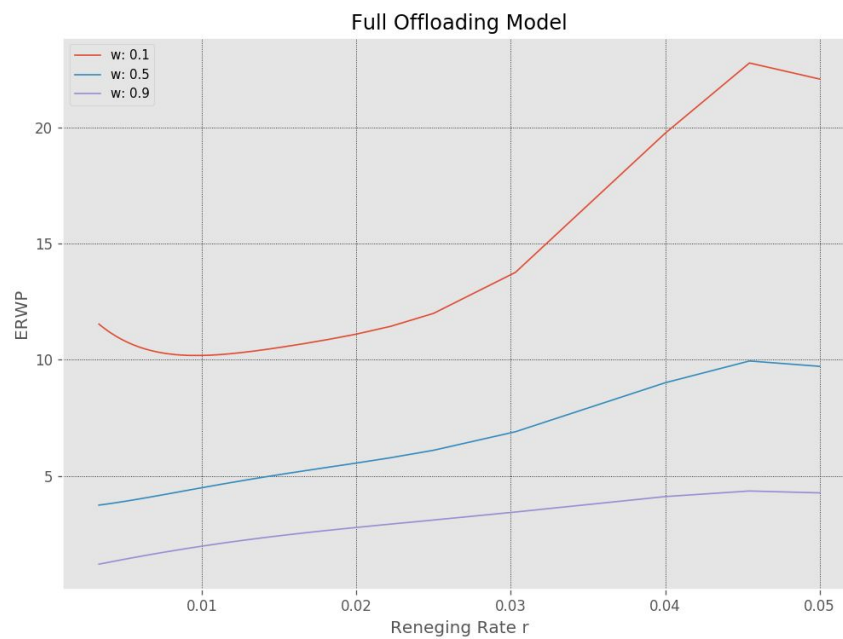
In questa sezione verranno mostrati i risultati ottenuti dalla simulazione del sistema nello stato stazionario. Come indicato precedentemente, il numero di job utilizzato per ogni run è 1000 mentre non è stata indicata una durata massima, in quanto l'esecuzione viene automaticamente terminata non appena tutti i 1000 job (a partire dalla fase stazionaria) sono stati elaborati. L'andamento dei grafici può essere paragonato con quelli del paper, inseriti nell'Appendice [A](#) di questa relazione.



Il trend del Mean Response Time risulta essere paragonabile a quello mostrato dal paper (range di valori stabilizzati 10-20 min): nel momento in cui vengono settate deadline con una durata media molto bassa, un numero elevato di job viene elaborato dalla coda Cellular, la quale ha un tempo di servizio medio 10 volte più grande di quello WiFi. Per questo motivo assistiamo al picco iniziale nel grafico. Se invece le deadline hanno una durata media alta, i job sono disposti ad attendere più tempo in coda WiFi e, ricordando che essa è una priority queue rispetto al tempo rimanente allo scadere della deadline, anche in questo caso assistiamo ad un aumento del response time medio.



Per quanto riguarda i consumi energetici, i risultati sono analoghi a quelli del paper, con anche lo stesso range di valori 1-4 Joule.



Infine viene mostrato il grafico che mette in relazione l'ERWP con il reneging rate r , calcolandolo con tre diversi esponenti: $\omega = 0.5$ comporta uno stesso peso tra energia e

response time, $\omega = 0.1$ comporta una maggiore attenzione al response time e, viceversa, $\omega = 0.9$ comporta un peso maggiore al consumo energetico.

Intervalli di Confidenza

Una volta ottenute i valori di tutte le metriche per ogni run, esse sono state analizzate attraverso il metodo batch; per ogni seed quindi, sono stati calcolati i valori MRT, MEC e ERWP e la stima dei relativi intervalli di confidenza è stata fatta per le seguenti combinazioni (numero di batch, numero di osservazioni): (5, 10) e (7, 12).

Il tutto consiste nel calcolare la media delle media per batch, la varianza ed i valori della distribuzione T-Student con una confidenza del 90%. Avendo 2 combinazioni e 3 metriche, di cui una con 3 argomenti differenti, è stato generato un totale di 10 file CSV, riproposti in seguito.

Per ogni CSV sono anche indicate le informazioni relative a deadline, reneging rate r , valore calcolato della metrica e un'indicazione se tale valore ricade nell'intervallo di confidenza stimato (*Included? OK*).

Intervalli di Confidenza, metrica **MRT**, combinazione (5, 10)

Deadline [s]	Deadline [min]	Reneging Rate r	MRT	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	26.7419	27.2811	164.8382	15.0405	39.5216	OK
1320	22	0.0455	29.1018	34.0658	23.9785	29.3972	38.7343	
1500	25	0.0400	23.0830	22.7340	48.9911	16.0609	29.4071	OK
1980	33	0.0303	15.6965	16.0477	34.0085	10.4878	21.6075	OK
2400	40	0.0250	15.7913	15.4923	5.8751	13.1814	17.8031	OK
2700	45	0.0222	13.8417	13.7042	16.6884	9.8095	17.5989	OK
3000	50	0.0200	13.0364	13.4629	6.9372	10.9518	15.9740	OK
3300	55	0.0182	12.8323	12.0753	2.4384	10.5865	13.5640	OK
3600	60	0.0167	11.9706	11.6085	0.5768	10.8844	12.3326	OK
3900	65	0.0154	12.9339	12.9640	8.4404	10.1942	15.7338	OK
4200	70	0.0143	12.5511	12.4987	4.1367	10.5596	14.4378	OK
4500	75	0.0133	12.9251	13.4095	2.2100	11.9922	14.8268	OK
4800	80	0.0125	12.1905	11.7824	0.9622	10.8472	12.7176	OK
5100	85	0.0118	11.3542	11.3506	0.6929	10.5570	12.1442	OK
5400	90	0.0111	12.0442	11.3407	1.2243	10.2858	12.3956	OK
5700	95	0.0105	12.2241	12.3346	2.7055	10.7664	13.9027	OK
6000	100	0.0100	12.7831	12.4894	0.3314	11.9406	13.0382	OK
6600	110	0.0091	13.1417	14.0961	6.8056	11.6089	16.5832	OK
7200	120	0.0083	12.6200	12.4105	2.0554	11.0437	13.7773	OK
7800	130	0.0077	12.5560	13.2357	1.1234	12.2252	14.2463	OK
8400	140	0.0071	13.8476	13.6402	2.7637	12.0553	15.2252	OK
9000	150	0.0067	14.0432	13.5808	5.1909	11.4087	15.7530	OK

9600	160	0.0063	13.1926	11.9733	6.0638	9.6256	14.3210	OK
10200	170	0.0059	14.1013	14.0984	1.9982	12.7507	15.4461	OK
10800	180	0.0056	13.0199	13.6332	2.4410	12.1436	15.1227	OK
11400	190	0.0053	13.0283	12.8391	2.5484	11.3172	14.3611	OK
12000	200	0.0050	13.6059	14.2034	0.2783	13.7005	14.7063	
12600	210	0.0048	14.4871	14.4539	1.4807	13.2938	15.6141	OK
13200	220	0.0045	14.1445	13.9943	0.4876	13.3285	14.6600	OK
13800	230	0.0043	14.9872	14.8857	2.7807	13.2959	16.4755	OK
14400	240	0.0042	14.7143	14.8284	8.4601	12.0554	17.6015	OK
15000	250	0.0040	15.3295	14.5078	2.7542	12.9255	16.0900	OK
15600	260	0.0038	15.8491	16.6536	7.6451	14.0175	19.2897	OK
16200	270	0.0037	14.8436	14.9143	5.3587	12.7073	17.1213	OK
16800	280	0.0036	14.0592	13.7783	1.2394	12.7169	14.8397	OK
17400	290	0.0034	14.6811	13.8320	0.5964	13.0957	14.5683	
18000	300	0.0033	14.5364	13.8239	1.9924	12.4781	15.1696	OK

Intervalli di Confidenza, metrica **MRT**, combinazione (7, 12)

Deadline [s]	Deadline [min]	Reneging Rate r	MRT	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	26.7419	24.0953	71.6239	17.8795	30.3110	OK
1320	22	0.0455	29.1018	29.5605	40.4211	24.8910	34.2300	OK
1500	25	0.0400	23.0830	22.0685	8.4559	19.9327	24.2042	OK
1980	33	0.0303	15.6965	16.2618	18.6130	13.0931	19.4304	OK
2400	40	0.0250	15.7913	15.3840	13.6236	12.6731	18.0949	OK
2700	45	0.0222	13.8417	14.4639	5.0058	12.8207	16.1072	OK
3000	50	0.0200	13.0364	13.2534	2.8851	12.0058	14.5009	OK
3300	55	0.0182	12.8323	12.9556	4.9627	11.3195	14.5918	OK
3600	60	0.0167	11.9706	11.7452	2.1261	10.6743	12.8161	OK
3900	65	0.0154	12.9339	12.9877	1.6426	12.0464	13.9290	OK
4200	70	0.0143	12.5511	12.5674	4.6242	10.9880	14.1467	OK
4500	75	0.0133	12.9251	13.2043	1.7334	12.2373	14.1712	OK
4800	80	0.0125	12.1905	12.1306	1.2423	11.3120	12.9492	OK
5100	85	0.0118	11.3542	11.4148	0.9697	10.6916	12.1381	OK
5400	90	0.0111	12.0442	12.3541	1.8132	11.3651	13.3431	OK
5700	95	0.0105	12.2241	12.3312	0.5821	11.7708	12.8915	OK
6000	100	0.0100	12.7831	12.7986	1.0218	12.0562	13.5411	OK
6600	110	0.0091	13.1417	13.1786	6.3820	11.3232	15.0340	OK
7200	120	0.0083	12.6200	12.5987	1.4710	11.7079	13.4895	OK
7800	130	0.0077	12.5560	12.3757	3.2637	11.0488	13.7025	OK
8400	140	0.0071	13.8476	14.0068	1.6214	13.0716	14.9421	OK
9000	150	0.0067	14.0432	13.7308	1.4463	12.8476	14.6141	OK
9600	160	0.0063	13.1926	12.8779	4.8708	11.2569	14.4988	OK
10200	170	0.0059	14.1013	14.3196	3.1432	13.0175	15.6217	OK
10800	180	0.0056	13.0199	13.2223	3.2833	11.8915	14.5531	OK
11400	190	0.0053	13.0283	12.6935	1.0414	11.9440	13.4430	OK
12000	200	0.0050	13.6059	13.8426	0.5238	13.3111	14.3742	OK

12600	210	0.0048	14.4871	14.0346	2.7269	12.8218	15.2474	OK
13200	220	0.0045	14.1445	13.7134	0.2976	13.3127	14.1141	
13800	230	0.0043	14.9872	15.3566	2.7589	14.1367	16.5766	OK
14400	240	0.0042	14.7143	14.6032	1.9292	13.5831	15.6233	OK
15000	250	0.0040	15.3295	15.2968	8.8487	13.1121	17.4816	OK
15600	260	0.0038	15.8491	15.8859	2.7809	14.6611	17.1106	OK
16200	270	0.0037	14.8436	14.7367	4.2508	13.2224	16.2509	OK
16800	280	0.0036	14.0592	13.8840	0.8735	13.1976	14.5705	OK
17400	290	0.0034	14.6811	14.1979	1.5234	13.2914	15.1044	OK
18000	300	0.0033	14.5364	14.4423	2.4467	13.2934	15.5911	OK

Intervalli di Confidenza, metrica **MEC**, combinazione (5, 10)

Deadline [s]	Deadline [min]	Reneging Rate r	MEC	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	3.4760	3.5350	0.2047	3.1036	3.9664	OK
1320	22	0.0455	3.5576	3.6560	0.0426	3.4592	3.8528	OK
1500	25	0.0400	3.3820	3.3748	0.0473	3.1674	3.5821	OK
1980	33	0.0303	2.8416	2.8337	0.1763	2.4334	3.2339	OK
2400	40	0.0250	2.7338	2.7005	0.0662	2.4551	2.9459	OK
2700	45	0.0222	2.4517	2.4586	0.0443	2.2580	2.6592	OK
3000	50	0.0200	2.3477	2.3625	0.0331	2.1890	2.5359	OK
3300	55	0.0182	2.2009	2.1737	0.0365	1.9916	2.3558	OK
3600	60	0.0167	2.0865	2.0175	0.0103	1.9205	2.1144	OK
3900	65	0.0154	2.1074	2.1269	0.0831	1.8521	2.4018	OK
4200	70	0.0143	1.9652	1.9541	0.0425	1.7575	2.1506	OK
4500	75	0.0133	1.9371	1.9628	0.0218	1.8221	2.1034	OK
4800	80	0.0125	1.8386	1.7781	0.0148	1.6620	1.8942	OK
5100	85	0.0118	1.6860	1.7174	0.0153	1.5995	1.8354	OK
5400	90	0.0111	1.6827	1.6375	0.0265	1.4822	1.7928	OK
5700	95	0.0105	1.6582	1.6871	0.0258	1.5340	1.8402	OK
6000	100	0.0100	1.6309	1.6343	0.0021	1.5906	1.6780	OK
6600	110	0.0091	1.5603	1.5991	0.0171	1.4744	1.7237	OK
7200	120	0.0083	1.4618	1.4633	0.0101	1.3675	1.5591	OK
7800	130	0.0077	1.3855	1.4775	0.0187	1.3472	1.6078	OK
8400	140	0.0071	1.3668	1.3778	0.0162	1.2565	1.4990	OK
9000	150	0.0067	1.3325	1.3176	0.0129	1.2091	1.4260	OK
9600	160	0.0063	1.2196	1.1446	0.0171	1.0201	1.2691	OK
10200	170	0.0059	1.2221	1.2080	0.0073	1.1266	1.2894	OK
10800	180	0.0056	1.1444	1.1855	0.0030	1.1336	1.2374	OK
11400	190	0.0053	1.0782	1.0830	0.0075	1.0004	1.1657	OK
12000	200	0.0050	1.0799	1.0866	0.0014	1.0508	1.1224	OK
12600	210	0.0048	1.0960	1.0823	0.0014	1.0470	1.1176	OK
13200	220	0.0045	1.0697	1.0296	0.0043	0.9669	1.0923	OK
13800	230	0.0043	1.0440	1.0609	0.0066	0.9836	1.1382	OK
14400	240	0.0042	1.0115	1.0219	0.0052	0.9528	1.0909	OK
15000	250	0.0040	1.0012	1.0030	0.0057	0.9308	1.0751	OK

15600	260	0.0038	0.9840	0.9821	0.0077	0.8983	1.0660	OK
16200	270	0.0037	0.9599	0.9681	0.0108	0.8692	1.0670	OK
16800	280	0.0036	0.9106	0.9097	0.0017	0.8704	0.9490	OK
17400	290	0.0034	0.9083	0.8695	0.0004	0.8498	0.8893	
18000	300	0.0033	0.8972	0.8759	0.0037	0.8175	0.9343	OK

Intervalli di Confidenza, metrica **MEC**, combinazione (7, 12)

Deadline [s]	Deadline [min]	Reneging Rate r	MEC	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	3.4760	3.4006	0.0985	3.1701	3.6311	OK
1320	22	0.0455	3.5576	3.5338	0.0722	3.3364	3.7312	OK
1500	25	0.0400	3.3820	3.3735	0.0132	3.2890	3.4580	OK
1980	33	0.0303	2.8416	2.8857	0.0966	2.6575	3.1140	OK
2400	40	0.0250	2.7338	2.6961	0.1048	2.4583	2.9338	OK
2700	45	0.0222	2.4517	2.4843	0.0108	2.4081	2.5604	OK
3000	50	0.0200	2.3477	2.3584	0.0142	2.2708	2.4460	OK
3300	55	0.0182	2.2009	2.2046	0.0161	2.1116	2.2977	OK
3600	60	0.0167	2.0865	2.0404	0.0340	1.9049	2.1758	OK
3900	65	0.0154	2.1074	2.1198	0.0296	1.9934	2.2462	OK
4200	70	0.0143	1.9652	1.9539	0.0474	1.7940	2.1137	OK
4500	75	0.0133	1.9371	1.9534	0.0253	1.8365	2.0703	OK
4800	80	0.0125	1.8386	1.8355	0.0180	1.7369	1.9342	OK
5100	85	0.0118	1.6860	1.7028	0.0081	1.6367	1.7688	OK
5400	90	0.0111	1.6827	1.7053	0.0175	1.6081	1.8025	OK
5700	95	0.0105	1.6582	1.6652	0.0064	1.6066	1.7239	OK
6000	100	0.0100	1.6309	1.6340	0.0031	1.5933	1.6747	OK
6600	110	0.0091	1.5603	1.5525	0.0289	1.4277	1.6774	OK
7200	120	0.0083	1.4618	1.4662	0.0078	1.4014	1.5310	OK
7800	130	0.0077	1.3855	1.3711	0.0337	1.2362	1.5060	OK
8400	140	0.0071	1.3668	1.3765	0.0071	1.3146	1.4385	OK
9000	150	0.0067	1.3325	1.3071	0.0115	1.2283	1.3860	OK
9600	160	0.0063	1.2196	1.1974	0.0172	1.1012	1.2937	OK
10200	170	0.0059	1.2221	1.2110	0.0094	1.1399	1.2821	OK
10800	180	0.0056	1.1444	1.1675	0.0058	1.1117	1.2233	OK
11400	190	0.0053	1.0782	1.0721	0.0036	1.0283	1.1159	OK
12000	200	0.0050	1.0799	1.0843	0.0013	1.0576	1.1109	OK
12600	210	0.0048	1.0960	1.0705	0.0040	1.0239	1.1171	OK
13200	220	0.0045	1.0697	1.0544	0.0038	1.0091	1.0997	OK
13800	230	0.0043	1.0440	1.0568	0.0021	1.0234	1.0903	OK
14400	240	0.0042	1.0115	1.0068	0.0018	0.9759	1.0377	OK
15000	250	0.0040	1.0012	0.9990	0.0056	0.9439	1.0542	OK
15600	260	0.0038	0.9840	0.9753	0.0025	0.9386	1.0121	OK
16200	270	0.0037	0.9599	0.9594	0.0060	0.9024	1.0164	OK
16800	280	0.0036	0.9106	0.9054	0.0018	0.8744	0.9365	OK
17400	290	0.0034	0.9083	0.8865	0.0011	0.8617	0.9114	OK
18000	300	0.0033	0.8972	0.8941	0.0041	0.8469	0.9413	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.1$, combinazione (5, 10)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	21.3687	21.8502	89.5284	12.8293	30.8712	OK
1320	22	0.0455	23.1137	26.6159	10.2120	23.5693	29.6626	
1500	25	0.0400	18.8331	18.5882	26.6224	13.6690	23.5074	OK
1980	33	0.0303	13.1368	13.3791	20.4715	9.0655	17.6928	OK
2400	40	0.0250	13.1687	12.9225	3.6654	11.0972	14.7478	OK
2700	45	0.0222	11.5491	11.4491	9.4058	8.5251	14.3730	OK
3000	50	0.0200	10.9493	11.2643	4.1544	9.3211	13.2076	OK
3300	55	0.0182	10.7109	10.1534	1.6219	8.9392	11.3676	OK
3600	60	0.0167	10.0312	9.7246	0.3838	9.1339	10.3152	OK
3900	65	0.0154	10.7680	10.7975	5.3652	8.5892	13.0058	OK
4200	70	0.0143	10.3937	10.3476	2.5522	8.8245	11.8707	OK
4500	75	0.0133	10.6743	11.0486	1.3963	9.9221	12.1752	OK
4800	80	0.0125	10.0758	9.7393	0.6271	8.9843	10.4942	OK
5100	85	0.0118	9.3699	9.3867	0.4684	8.7342	10.0392	OK
5400	90	0.0111	9.8768	9.3291	0.7958	8.4786	10.1796	OK
5700	95	0.0105	9.9963	10.0979	1.7014	8.8543	11.3414	OK
6000	100	0.0100	10.3868	10.1757	0.1988	9.7507	10.6008	OK
6600	110	0.0091	10.5983	11.3159	3.8530	9.4445	13.1873	OK
7200	120	0.0083	10.1552	10.0038	1.2150	8.9529	11.0547	OK
7800	130	0.0077	10.0573	10.6179	0.7247	9.8063	11.4295	OK
8400	140	0.0071	10.9641	10.8241	1.5573	9.6344	12.0139	OK
9000	150	0.0067	11.0739	10.7308	2.8961	9.1083	12.3533	OK
9600	160	0.0063	10.3780	9.4487	3.3989	7.6910	11.2064	OK
10200	170	0.0059	11.0208	11.0038	1.1447	9.9837	12.0238	OK
10800	180	0.0056	10.1868	10.6547	1.3013	9.5671	11.7422	OK
11400	190	0.0053	10.1326	10.0042	1.4250	8.8661	11.1423	OK
12000	200	0.0050	10.5346	10.9553	0.1264	10.6164	11.2943	
12600	210	0.0048	11.1626	11.1299	0.7020	10.3311	11.9288	OK
13200	220	0.0045	10.8919	10.7361	0.2449	10.2643	11.2080	OK
13800	230	0.0043	11.4391	11.3862	1.4851	10.2243	12.5480	OK
14400	240	0.0042	11.2216	11.3131	4.2776	9.3412	13.2849	OK
15000	250	0.0040	11.6291	11.0752	1.4753	9.9172	12.2332	OK
15600	260	0.0038	11.9592	12.4979	3.8704	10.6222	14.3735	OK
16200	270	0.0037	11.2547	11.3141	2.8777	9.6968	12.9315	OK
16800	280	0.0036	10.6712	10.4781	0.6221	9.7261	11.2300	OK
17400	290	0.0034	11.0730	10.4585	0.2870	9.9477	10.9693	
18000	300	0.0033	10.9637	10.4564	1.0572	9.4761	11.4367	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.1$, combinazione (7, 12)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
--------------	----------------	-----------------	------	------	----------	------------	-------------	-----------

1200	20	0.0500	21.3687	19.4987	39.1110	14.9055	24.0919	OK
1320	22	0.0455	23.1137	23.3862	20.3538	20.0727	26.6997	OK
1500	25	0.0400	18.8331	18.0912	4.1476	16.5955	19.5870	OK
1980	33	0.0303	13.1368	13.5828	11.2594	11.1183	16.0473	OK
2400	40	0.0250	13.1687	12.8487	8.4286	10.7165	14.9810	OK
2700	45	0.0222	11.5491	12.0246	2.5630	10.8487	13.2004	OK
3000	50	0.0200	10.9493	11.1169	1.7012	10.1589	12.0749	OK
3300	55	0.0182	10.7109	10.8005	2.8492	9.5608	12.0402	OK
3600	60	0.0167	10.0312	9.8388	1.4032	8.9688	10.7088	OK
3900	65	0.0154	10.7680	10.8147	1.0738	10.0536	11.5758	OK
4200	70	0.0143	10.3937	10.3948	2.9004	9.1440	11.6456	OK
4500	75	0.0133	10.6743	10.8902	1.1169	10.1140	11.6664	OK
4800	80	0.0125	10.0758	10.0312	0.8001	9.3743	10.6882	OK
5100	85	0.0118	9.3699	9.4249	0.5881	8.8616	9.9881	OK
5400	90	0.0111	9.8768	10.1189	1.1418	9.3341	10.9037	OK
5700	95	0.0105	9.9963	10.0806	0.3501	9.6460	10.5151	OK
6000	100	0.0100	10.3868	10.4017	0.5695	9.8474	10.9560	OK
6600	110	0.0091	10.5983	10.6196	3.7484	9.1977	12.0416	OK
7200	120	0.0083	10.1552	10.1435	0.8721	9.4576	10.8294	OK
7800	130	0.0077	10.0573	9.9183	2.0365	8.8702	10.9664	OK
8400	140	0.0071	10.9641	11.0856	0.9065	10.3863	11.7848	OK
9000	150	0.0067	11.0739	10.8303	0.8704	10.1451	11.5156	OK
9600	160	0.0063	10.3780	10.1379	2.8039	8.9081	11.3678	OK
10200	170	0.0059	11.0208	11.1648	1.7715	10.1872	12.1423	OK
10800	180	0.0056	10.1868	10.3504	1.8021	9.3645	11.3364	OK
11400	190	0.0053	10.1326	9.8947	0.5821	9.3343	10.4550	OK
12000	200	0.0050	10.5346	10.7034	0.2535	10.3336	11.0731	OK
12600	210	0.0048	11.1626	10.8252	1.4410	9.9435	11.7068	OK
13200	220	0.0045	10.8919	10.5764	0.1468	10.2951	10.8578	
13800	230	0.0043	11.4391	11.7052	1.3683	10.8461	12.5643	OK
14400	240	0.0042	11.2216	11.1359	0.9568	10.4175	11.8543	OK
15000	250	0.0040	11.6291	11.6029	4.3445	10.0721	13.1338	OK
15600	260	0.0038	11.9592	11.9715	1.3835	11.1076	12.8354	OK
16200	270	0.0037	11.2547	11.1773	2.2134	10.0846	12.2700	OK
16800	280	0.0036	10.6712	10.5445	0.4493	10.0522	11.0368	OK
17400	290	0.0034	11.0730	10.7229	0.7518	10.0861	11.3598	OK
18000	300	0.0033	10.9637	10.8922	1.2727	10.0636	11.7208	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.5$, combinazione (5, 10)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	9.1408	9.3551	7.2426	6.7893	11.9208	OK
1320	22	0.0455	9.6492	10.4779	0.5340	9.7812	11.1746	
1500	25	0.0400	8.5712	8.5142	2.0999	7.1327	9.8958	OK
1980	33	0.0303	6.5516	6.5917	2.6194	5.0487	8.1348	OK
2400	40	0.0250	6.4609	6.3523	0.5810	5.6256	7.0790	OK

2700	45	0.0222	5.7070	5.6888	0.9238	4.7724	6.6051	OK
3000	50	0.0200	5.4874	5.5761	0.5196	4.8889	6.2634	OK
3300	55	0.0182	5.2536	5.0970	0.3103	4.5658	5.6281	OK
3600	60	0.0167	4.9695	4.8120	0.0756	4.5498	5.0742	OK
3900	65	0.0154	5.1947	5.2211	0.8617	4.3361	6.1061	OK
4200	70	0.0143	4.9246	4.8989	0.3900	4.3036	5.4943	OK
4500	75	0.0133	4.9827	5.1091	0.2213	4.6606	5.5576	OK
4800	80	0.0125	4.7164	4.5602	0.1154	4.2362	4.8841	OK
5100	85	0.0118	4.3588	4.4012	0.0996	4.1002	4.7021	OK
5400	90	0.0111	4.4822	4.2887	0.1583	3.9094	4.6681	OK
5700	95	0.0105	4.4845	4.5473	0.2654	4.0561	5.0384	OK
6000	100	0.0100	4.5448	4.4991	0.0253	4.3474	4.6508	OK
6600	110	0.0091	4.5032	4.7213	0.3736	4.1385	5.3040	OK
7200	120	0.0083	4.2745	4.2404	0.1465	3.8754	4.6053	OK
7800	130	0.0077	4.1536	4.4083	0.1350	4.0580	4.7587	OK
8400	140	0.0071	4.3274	4.3113	0.1726	3.9153	4.7074	OK
9000	150	0.0067	4.3013	4.2037	0.2703	3.7081	4.6994	OK
9600	160	0.0063	3.9906	3.6810	0.3280	3.1350	4.2271	OK
10200	170	0.0059	4.1292	4.1026	0.1214	3.7704	4.4347	OK
10800	180	0.0056	3.8362	3.9948	0.0953	3.7004	4.2891	OK
11400	190	0.0053	3.7255	3.7062	0.1391	3.3506	4.0617	OK
12000	200	0.0050	3.8070	3.9003	0.0079	3.8158	3.9849	
12600	210	0.0048	3.9569	3.9321	0.0302	3.7664	4.0978	OK
13200	220	0.0045	3.8565	3.7532	0.0319	3.5829	3.9235	OK
13800	230	0.0043	3.9148	3.9316	0.1248	3.5949	4.2684	OK
14400	240	0.0042	3.8235	3.8592	0.2541	3.3786	4.3399	OK
15000	250	0.0040	3.8810	3.7847	0.1207	3.4535	4.1160	OK
15600	260	0.0038	3.9092	3.9995	0.2502	3.5226	4.4764	OK
16200	270	0.0037	3.7445	3.7706	0.2406	3.3029	4.2383	OK
16800	280	0.0036	3.5576	3.5200	0.0384	3.3332	3.7069	OK
17400	290	0.0034	3.6141	3.4406	0.0135	3.3297	3.5515	
18000	300	0.0033	3.5759	3.4479	0.0852	3.1696	3.7261	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.5$, combinazione (7, 12)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	9.1408	8.6713	3.2432	7.3486	9.9939	OK
1320	22	0.0455	9.6492	9.6499	1.4681	8.7601	10.5398	OK
1500	25	0.0400	8.5712	8.3852	0.2250	8.0368	8.7336	OK
1980	33	0.0303	6.5516	6.7202	1.4576	5.8335	7.6069	OK
2400	40	0.0250	6.4609	6.3376	1.2204	5.5262	7.1490	OK
2700	45	0.0222	5.7070	5.8638	0.1455	5.5836	6.1440	OK
3000	50	0.0200	5.4874	5.5439	0.2037	5.2124	5.8753	OK
3300	55	0.0182	5.2536	5.2772	0.2984	4.8761	5.6784	OK
3600	60	0.0167	4.9695	4.8675	0.2667	4.4882	5.2468	OK
3900	65	0.0154	5.1947	5.2209	0.2057	4.8878	5.5540	OK

4200	70	0.0143	4.9246	4.9074	0.4586	4.4100	5.4047	OK
4500	75	0.0133	4.9827	5.0567	0.1988	4.7292	5.3843	OK
4800	80	0.0125	4.7164	4.7031	0.1418	4.4265	4.9797	OK
5100	85	0.0118	4.3588	4.3929	0.0802	4.1849	4.6009	OK
5400	90	0.0111	4.4822	4.5701	0.1784	4.2599	4.8803	OK
5700	95	0.0105	4.4845	4.5151	0.0507	4.3497	4.6805	OK
6000	100	0.0100	4.5448	4.5536	0.0504	4.3888	4.7185	OK
6600	110	0.0091	4.5032	4.4981	0.4365	4.0129	4.9834	OK
7200	120	0.0083	4.2745	4.2781	0.1072	4.0375	4.5186	OK
7800	130	0.0077	4.1536	4.1039	0.3184	3.6895	4.5183	OK
8400	140	0.0071	4.3274	4.3678	0.0917	4.1455	4.5902	OK
9000	150	0.0067	4.3013	4.2117	0.1202	3.9571	4.4664	OK
9600	160	0.0063	3.9906	3.9083	0.3001	3.5060	4.3107	OK
10200	170	0.0059	4.1292	4.1428	0.1749	3.8357	4.4500	OK
10800	180	0.0056	3.8362	3.9052	0.1530	3.6179	4.1925	OK
11400	190	0.0053	3.7255	3.6691	0.0575	3.4930	3.8452	OK
12000	200	0.0050	3.8070	3.8475	0.0152	3.7570	3.9381	OK
12600	210	0.0048	3.9569	3.8514	0.1085	3.6095	4.0934	OK
13200	220	0.0045	3.8565	3.7692	0.0206	3.6637	3.8747	OK
13800	230	0.0043	3.9148	3.9854	0.0775	3.7810	4.1898	OK
14400	240	0.0042	3.8235	3.7962	0.0558	3.6227	3.9697	OK
15000	250	0.0040	3.8810	3.8715	0.2385	3.5129	4.2302	OK
15600	260	0.0038	3.9092	3.8945	0.0819	3.6843	4.1047	OK
16200	270	0.0037	3.7445	3.7266	0.1610	3.4319	4.0212	OK
16800	280	0.0036	3.5576	3.5244	0.0323	3.3925	3.6564	OK
17400	290	0.0034	3.6141	3.5148	0.0411	3.3659	3.6637	OK
18000	300	0.0033	3.5759	3.5547	0.0970	3.3260	3.7834	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.9$, combinazione (5, 10)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	4.1863	4.2629	0.4496	3.6236	4.9021	OK
1320	22	0.0455	4.3114	4.4724	0.0692	4.2217	4.7232	OK
1500	25	0.0400	4.0556	4.0441	0.1129	3.7238	4.3644	OK
1980	33	0.0303	3.3487	3.3437	0.3082	2.8145	3.8730	OK
2400	40	0.0250	3.2388	3.1958	0.1008	2.8931	3.4984	OK
2700	45	0.0222	2.8951	2.9002	0.0828	2.6259	3.1745	OK
3000	50	0.0200	2.7788	2.8006	0.0588	2.5694	3.0318	OK
3300	55	0.0182	2.6150	2.5756	0.0564	2.3491	2.8021	OK
3600	60	0.0167	2.4798	2.3985	0.0153	2.2807	2.5163	OK
3900	65	0.0154	2.5222	2.5432	0.1335	2.1949	2.8915	OK
4200	70	0.0143	2.3587	2.3454	0.0654	2.1015	2.5894	OK
4500	75	0.0133	2.3385	2.3751	0.0347	2.1976	2.5526	OK
4800	80	0.0125	2.2184	2.1454	0.0222	2.0033	2.2874	OK
5100	85	0.0118	2.0375	2.0720	0.0221	1.9302	2.2138	OK
5400	90	0.0111	2.0455	1.9838	0.0373	1.7996	2.1679	OK

5700	95	0.0105	2.0220	2.0561	0.0412	1.8627	2.2495	OK
6000	100	0.0100	2.0004	1.9999	0.0034	1.9445	2.0553	OK
6600	110	0.0091	1.9271	1.9839	0.0322	1.8130	2.1549	OK
7200	120	0.0083	1.8103	1.8088	0.0173	1.6835	1.9341	OK
7800	130	0.0077	1.7246	1.8376	0.0275	1.6794	1.9958	OK
8400	140	0.0071	1.7196	1.7294	0.0250	1.5787	1.8800	OK
9000	150	0.0067	1.6829	1.6600	0.0238	1.5129	1.8072	OK
9600	160	0.0063	1.5446	1.4445	0.0308	1.2772	1.6118	OK
10200	170	0.0059	1.5577	1.5412	0.0127	1.4335	1.6488	OK
10800	180	0.0056	1.4562	1.5100	0.0059	1.4367	1.5833	OK
11400	190	0.0053	1.3804	1.3838	0.0135	1.2731	1.4945	OK
12000	200	0.0050	1.3879	1.4014	0.0019	1.3601	1.4428	OK
12600	210	0.0048	1.4152	1.3996	0.0020	1.3570	1.4423	OK
13200	220	0.0045	1.3806	1.3312	0.0065	1.2543	1.4081	OK
13800	230	0.0043	1.3577	1.3763	0.0116	1.2737	1.4790	OK
14400	240	0.0042	1.3178	1.3311	0.0119	1.2272	1.4349	OK
15000	250	0.0040	1.3109	1.3064	0.0103	1.2096	1.4032	OK
15600	260	0.0038	1.2946	1.2983	0.0156	1.1792	1.4173	OK
16200	270	0.0037	1.2587	1.2691	0.0201	1.1340	1.4041	OK
16800	280	0.0036	1.1947	1.1913	0.0029	1.1398	1.2428	OK
17400	290	0.0034	1.1953	1.1434	0.0007	1.1176	1.1692	
18000	300	0.0033	1.1811	1.1503	0.0070	1.0707	1.2300	OK

Intervalli di Confidenza, metrica **ERWP** $\omega = 0.9$, combinazione (7, 12)

Deadline [s]	Deadline [min]	Reneging Rate r	ERWP	Mean	Variance	Left Value	Right Value	Included?
1200	20	0.0500	4.1863	4.0747	0.2098	3.7383	4.4111	OK
1320	22	0.0455	4.3114	4.2861	0.1284	4.0230	4.5492	OK
1500	25	0.0400	4.0556	4.0312	0.0198	3.9279	4.1345	OK
1980	33	0.0303	3.3487	3.4076	0.1698	3.1049	3.7102	OK
2400	40	0.0250	3.2388	3.1910	0.1719	2.8865	3.4955	OK
2700	45	0.0222	2.8951	2.9411	0.0133	2.8563	3.0259	OK
3000	50	0.0200	2.7788	2.7946	0.0241	2.6805	2.9087	OK
3300	55	0.0182	2.6150	2.6205	0.0290	2.4955	2.7456	OK
3600	60	0.0167	2.4798	2.4257	0.0512	2.2596	2.5919	OK
3900	65	0.0154	2.5222	2.5366	0.0431	2.3841	2.6890	OK
4200	70	0.0143	2.3587	2.3457	0.0744	2.1454	2.5460	OK
4500	75	0.0133	2.3385	2.3611	0.0379	2.2182	2.5040	OK
4800	80	0.0125	2.2184	2.2144	0.0269	2.0939	2.3349	OK
5100	85	0.0118	2.0375	2.0570	0.0124	1.9753	2.1386	OK
5400	90	0.0111	2.0455	2.0755	0.0278	1.9530	2.1980	OK
5700	95	0.0105	2.0220	2.0317	0.0094	1.9607	2.1028	OK
6000	100	0.0100	2.0004	2.0044	0.0049	1.9531	2.0556	OK
6600	110	0.0091	1.9271	1.9189	0.0496	1.7553	2.0825	OK
7200	120	0.0083	1.8103	1.8150	0.0131	1.7308	1.8992	OK
7800	130	0.0077	1.7246	1.7062	0.0525	1.5379	1.8745	OK

8400	140	0.0071	1.7196	1.7327	0.0113	1.6544	1.8109	OK
9000	150	0.0067	1.6829	1.6502	0.0182	1.5512	1.7493	OK
9600	160	0.0063	1.5446	1.5159	0.0306	1.3874	1.6444	OK
10200	170	0.0059	1.5577	1.5474	0.0168	1.4522	1.6427	OK
10800	180	0.0056	1.4562	1.4849	0.0113	1.4067	1.5631	OK
11400	190	0.0053	1.3804	1.3700	0.0061	1.3127	1.4273	OK
12000	200	0.0050	1.3879	1.3953	0.0019	1.3632	1.4275	OK
12600	210	0.0048	1.4152	1.3815	0.0078	1.3168	1.4462	OK
13200	220	0.0045	1.3806	1.3585	0.0054	1.3045	1.4124	OK
13800	230	0.0043	1.3577	1.3758	0.0042	1.3284	1.4232	OK
14400	240	0.0042	1.3178	1.3108	0.0034	1.2681	1.3535	OK
15000	250	0.0040	1.3109	1.3079	0.0119	1.2278	1.3881	OK
15600	260	0.0038	1.2946	1.2844	0.0049	1.2330	1.3357	OK
16200	270	0.0037	1.2587	1.2568	0.0116	1.1777	1.3358	OK
16800	280	0.0036	1.1947	1.1871	0.0030	1.1470	1.2272	OK
17400	290	0.0034	1.1953	1.1659	0.0022	1.1315	1.2004	OK
18000	300	0.0033	1.1811	1.1763	0.0077	1.1117	1.2409	OK

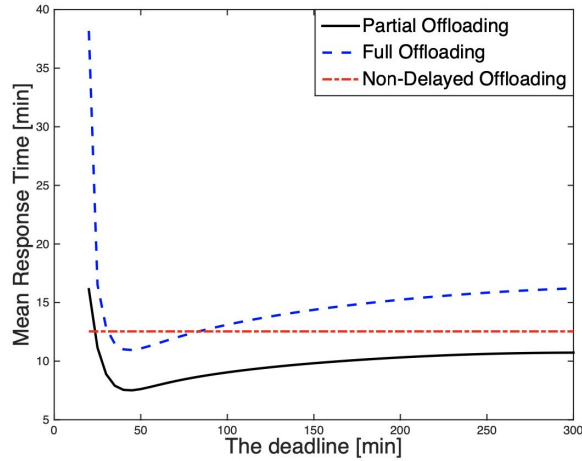
Conclusioni

In questo progetto è stato simulato un sistema reale tramite l'utilizzo dell'applicazione OMNeT++. Tale sistema ha come obiettivo quello di permettere l'offloading dell'elaborazione da dispositivi mobili su cloud, ma ottimizzandone il consumo energetico ed i tempi di risposta.

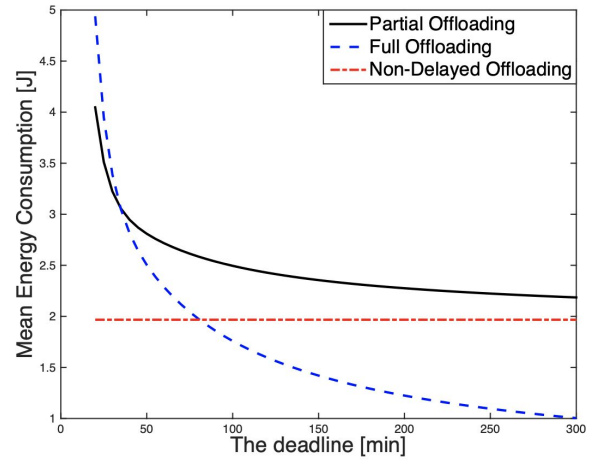
La simulazione è praticamente riuscita a replicare tutti i dati reali, aggiungendo anche un'analisi statistica. Si è infatti partiti dallo studio del transiente iniziale, per poi passare alle metriche caratteristiche del paper, Mean Response Time, Mean Energy Consumption e Energy-Response Weighted Product ed infine all'analisi batch per la stima di media, varianza ed intervalli di confidenza.

Appendice A - Grafici dal Paper

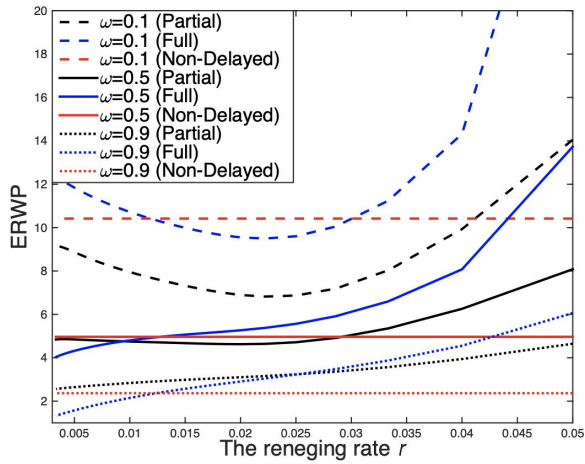
Va notato che, in ogni grafico seguente, i valori di riferimento sono quelli indicati in blu.



(a) Mean Response Time



(b) Mean Energy Consumption



(a) Reneging rate

Bibliografia

[1] Huaming Wu. 2018. Stochastic Analysis of Delayed Mobile Offloading in Heterogeneous Networks. IEEE Transactions on Mobile Computing 17, 2 (February 2018), 461-474. DOI: <https://doi.org/10.1109/TMC.2017.2711014>

[2] <https://omnetpp.org>

[3] <https://github.com/matteodelv/SdSFullOffloading>