Analysis of the Energy-Response Time Tradeoff for Delayed Mobile Cloud Offloading

Huaming Wu
Free University of Berlin,
Germany
huaming.wu@fu-berlin.de

Yi Sun
Free University of Berlin,
Germany
yi.sun@fu-berlin.de

Katinka Wolter
Free University of Berlin,
Germany
katinka.wolter@fu-berlin.de

ABSTRACT

We develop a delayed offloading model to leverage the complementary strength of WiFi and cellular networks when choosing heterogeneous wireless interfaces for offloading. Optimality analysis of the energy-delay tradeoff is carried out by using a queueing model with impatient jobs and service interruptions, which captures both energy and performance metrics and also intermittently available access links.

1. INTRODUCTION

There are several ways to offload tasks to a dedicated cloud server, e.g., via a cellular (e.g., 2G or 3G) connection or via intermittently available WLAN hotspots. By delaying transmission until a fast and energy-efficient network (e.g., WiFi) becomes available, it is possible to reduce the transmission time even if extra waiting time is introduced, which could lead to energy savings [2]. However, delayed offloading is still a matter of debate, since it is not known to what extent users are willing to delay a transmission [3]. Different applications usually give different relative importance to the factors of response time and energy consumption. We aim at guiding the decision of how to balance the delay and energy savings for different types of scenarios like delay-tolerant applications (e.g., iCloud and dropbox) and delay-sensitive applications (e.g., chess game and face recognition).

2. DELAYED OFFLOADING MODEL

Figure 1 shows a delayed offloading model, which consists of an *Offload Queue* for data offloading, a *Local Queue* denoting the local processing on the mobile device and a *Remote Queue* representing the remote processing on a cloud server. The jobs are offloaded either via a cellular connection or a WiFi network to the cloud. We consider an $M/M/1$ modulated queue in a 2-phase (fast and slow) Markovian random environment, with impatient jobs as a mathematical abstraction of the scenario. The transmission speed of the fast phase (WiFi network) is s_w with service rate $\mu_w = s_w/\mathbb{E}[X]$, its operating power is p_w when serving jobs and zero whenever idle, where $\mathbb{E}[X]$ is the average job size. Similarly, the corresponding speed for the slow phase (cellular network) is s_c with service rate $\mu_c = s_c/\mathbb{E}[X]$ ($\mu_c \leq \mu_w$), and operating power p_c.

We assume that offloading jobs arrive to the system according to a Poisson process with rate λ, and the modulating process $f \in \{f_{\text{ON}}, f_{\text{OFF}}\}$ determines the service rates:

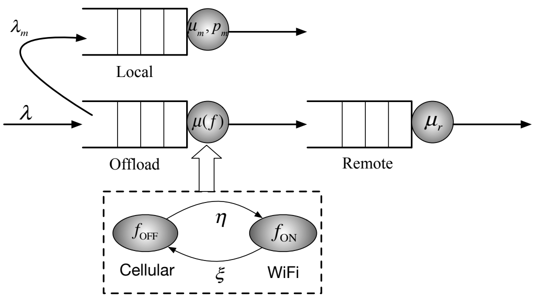

Figure 1: The delayed offloading model

$$\mu(f) = \begin{cases} \mu_c, & \text{if } f = f_{\text{OFF}} \\ \mu_w, & \text{if } f = f_{\text{ON}} \end{cases}. \tag{1}$$

We model the intermittent availability of WiFi hotspots with occasional server break-down [1], either in ON-state where the WiFi network is processing the existing jobs, or in the OFF-state during which the job is served over the cellular network (the cellular network is assumed to be always available). However, when the job stays in the cellular network for too long time, it abandons the *Offload Queue* and is then processed locally. We assume that the sojourn time in a hotspot and the time to move from one hotspot to another are exponentially distributed with parameters ξ (failure rate) and η (recovery rate), respectively.

In the slow phase jobs may become impatient. A deadline T_d, is associated with each job in this phase. That is, each job, upon arrival, activates an individual 'impatience timer', exponentially distributed with an abandonment rate γ $(= T_d^{-1})$. If the job in the *Offload Queue* is completely transmitted before the assigned deadline has expired, we say that the job is successfully offloaded. If the system does not change its environment from the slow phase to the fast phase before the deadline expires, the job will be removed from the *Offload Queue* and join the *Local Queue* for immediate local processing rather than offloaded to the cloud.

Since there is no waiting time before entering service, the $M/M/\infty$ queue of the cloud is occasionally referred to as a delay (sometimes pure delay) station, the probability distribution of the delay being that of the service time.

3. QUEUEING ANALYSIS

Given the previously stated assumptions, the delayed offloading model can be modeled with a 2D Markov chain, as shown in Fig. 2.

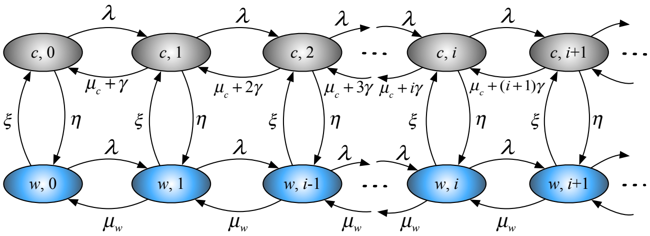

Figure 2: The 2D Markov chain for the delayed offloading model with cellular and WiFi networks

The states with cellular network are denoted with $\{c, i\}$, and the states with WiFi connectivity are denoted with $\{w, i\}$. The variable i corresponds to the number of jobs in the system (queuing and in service). During the WiFi phase, the system drains at rate μ_w and during the cellular phase, the system drains at rate $\mu_c + i \cdot \gamma$ since any of the i queued jobs can abandon the *Offload Queue* [4]. Writing the balance equations for the cellular and WiFi phases gives:

$$(\lambda + \eta)\pi_{c,0} = (\mu_c + \gamma)\pi_{c,1} + \xi\pi_{w,0} \tag{2a}$$

$$(\lambda + \eta + \mu_c + i\gamma)\pi_{c,i} = \lambda\pi_{c,i-1} + \big(\mu_c + (i+1)\gamma\big)\pi_{c,i+1} + \xi\pi_{w,i} \ (i > 0) \tag{2b}$$

$$(\lambda + \xi)\pi_{w,0} = \mu_w\pi_{w,1} + \eta\pi_{w,0} \tag{2c}$$

$$(\lambda + \xi + \mu_w)\pi_{w,i} = \lambda\pi_{w,i-1} + \mu_w\pi_{w,i+1} + \eta\pi_{c,i} \ (i > 0) \tag{2d}$$

The steady-state probability of finding the offloading system in some region without WiFi availability (with only cellular access) is $\pi_c = \frac{\mathbb{E}[T_{\text{OFF}}]}{\mathbb{E}[T_{\text{ON}}]+\mathbb{E}[T_{\text{OFF}}]} = \frac{\xi}{\eta+\xi}$. Similarly, the steady-state probability for the periods with WiFi is $\pi_w = \frac{\mathbb{E}[T_{\text{ON}}]}{\mathbb{E}[T_{\text{ON}}]+\mathbb{E}[T_{\text{OFF}}]} = \frac{\eta}{\eta+\xi}$, which equals to the availability ratio AR.

The probability generating functions for both cellular and WiFi states are defined as:

$$G_c(z) = \sum_{i=0}^{\infty} \pi_{c,i} z^i \text{ and } G_w(z) = \sum_{i=0}^{\infty} \pi_{w,i} z^i, \quad |z| \leq 1.$$

After some calculation and algebraic manipulations in (2), we obtain:

$$G_w(z)\beta(z) = \eta z G_c(z) - \mu_w(1 - z)\pi_{w,0},$$

where $\beta(z) = (\lambda z - \mu_w)(1 - z) + \xi z$. The roots z_1, z_2 of the quadratic polynomial $\beta(z) = -\lambda(z - z_1)(z - z_2)$ are $z_{1,2} = \frac{\lambda+\mu_w+\xi\mp\sqrt{(\lambda+\mu_w+\xi)^2-4\lambda\mu_w}}{2\lambda}$ [5].

According to [5], we obtain:

$$\pi_{c,0} = \frac{S\gamma\xi\kappa_2(1)}{\mu_c(\xi + \eta)(SV - TU)}, \tag{3}$$

$$\pi_{w,0} = -\frac{T\gamma\kappa_2(1)}{\mu_w(\xi + \eta)(SV - TU)}, \tag{4}$$

where we define $S = \int_0^{z_1} \frac{\kappa_1(x)}{\beta(x)} dx$, $T = \int_0^{z_1} \frac{\kappa_1(x)}{x} dx$, $U = \int_{z_1}^1 \frac{\kappa_2(x)}{\beta(x)} dx$ and $V = \int_{z_1}^1 \frac{\kappa_2(x)}{x} dx$. Accordingly, $\kappa_1(z)$ and $\kappa_2(z)$ are represented as follows:

$$\kappa_1(z) = e^{-\frac{\lambda z}{\gamma}} z^{\frac{\mu_c}{\gamma}} (z_1 - z)^{\frac{\eta}{\gamma}\frac{z_1(z_2-1)}{z_2-z_1}} (z_2 - z)^{-\frac{\eta}{\gamma}\frac{z_2(z_1-1)}{z_2-z_1}}, z \leq z_1,$$

$$\kappa_2(z) = e^{-\frac{\lambda z}{\gamma}} z^{\frac{\mu_c}{\gamma}} (z - z_1)^{\frac{\eta}{\gamma}\frac{z_1(z_2-1)}{z_2-z_1}} (z_2 - z)^{-\frac{\eta}{\gamma}\frac{z_2(z_1-1)}{z_2-z_1}}, z \geq z_1.$$

By the definitions of $\kappa_1(z)$, $\kappa_2(z)$ and $\beta(z)$, it follows that $T, U, V > 0$ and $S < 0$. Therefore, $\pi_{c,0}$ and $\pi_{w,0}$ are positive. It can be shown formally that the system is ergodic.

Let μ be defined as: $\mu = \pi_c \cdot \mu_c + \pi_w \cdot \mu_w$. According to [5], we obtain:

$$\mathbb{E}[N_c] = \frac{\lambda - \mu + \mu_c\pi_{c,0} + \mu_w\pi_{w,0}}{\gamma}, \tag{5}$$

$$\mathbb{E}[N_w] = \frac{\eta(\lambda - \mu) + \gamma(\lambda - \mu_w)\pi_w + \eta\mu_c\pi_{c,0} + \mu_w(\eta + \gamma)\pi_{w,0}}{\xi\gamma}. \tag{6}$$

As shown in Fig. 2, the expected number of jobs served per unit of time in the slow phase and the fast phase are $\mu_c(\pi_c - \pi_{c,0})$ and $\mu_w(\pi_w - \pi_{w,0})$, respectively. An arbitrary job arriving to the *Offload Queue* may leave and join the *Local Queue* due to impatience in the slow phase. Therefore, the rate of jobs executed locally on the mobile device, λ_m, is given by:

$$\begin{aligned} \lambda_m &= \lambda - \mu_c(\pi_c - \pi_{c,0}) - \mu_w(\pi_w - \pi_{w,0}) \\ &= \lambda - \mu + \mu_c\pi_{c,0} + \mu_w\pi_{w,0} \\ &= \gamma \cdot \mathbb{E}[N_c]. \end{aligned} \tag{7}$$

Further, the probability to abandon the queue is defined as:

$$\Pr\{\text{abandonment}\} = \frac{\lambda_{\text{m}}}{\lambda} = \frac{\lambda - \mu + \mu_c\pi_{c,0} + \mu_w\pi_{w,0}}{\lambda}, \tag{8}$$

where Pr denotes the probability operation.

4. METRIC-BASED ANALYSIS

4.1 Mean Response Time

The total cost for offloading a job is composed of the cost for sending the job to the cloud and idly waiting for the cloud to complete the job.

By Little's Law, $\mathbb{E}[N] = \lambda\mathbb{E}[T]$, the mean response time can be calculated as:

$$\mathbb{E}[T] = \mathbb{E}\big[\mathbb{E}[T_j]\big] = \sum_j \frac{\lambda_j}{\lambda}\mathbb{E}[T_j] = \frac{1}{\lambda}\sum_j \mathbb{E}[N_j], \tag{9}$$

where $j \in \{c, w, m, r\}$ represents the cellular phase, the WiFi phase, the mobile device and the remote cloud, respectively.

The average number of jobs in the *Local Queue* and *Remote Queue* can be calculated as: $\mathbb{E}[N_m] = \rho_m/(1 - \rho_m)$ and $\mathbb{E}[N_r] = \lambda_r/\mu_r$, respectively, where $\rho_m = \lambda_m/\mu_m$ and $\lambda_r = \lambda - \lambda_m$ is the arrival rate to the *Remote Queue.*

4.2 Mean Energy Consumption

Since $\mathbb{E}[P] = \lambda\mathbb{E}[\mathcal{E}]$ is the mean power consumption, we can calculate the mean energy consumption for the delayed offloading model as:

$$\mathbb{E}[\mathcal{E}] = \mathbb{E}\big[\mathbb{E}[\mathcal{E}_k]\big] = \sum_k \frac{\lambda_k}{\lambda}\mathbb{E}[\mathcal{E}_k] = \frac{1}{\lambda}\sum_k \mathbb{E}[P_k], \tag{10}$$

where $k \in \{c, w, m\}$ represents the cellular phase, the WiFi phase and the mobile device, respectively. The corresponding average power consumption can be calculated as:

$$\mathbb{E}[P_k] = p_k \cdot \Pr\{N_i > 0\} = p_k \cdot \rho_k. \tag{11}$$

Since the utilization of the queue is the probability that the server is busy, we have $\Pr\{N_k > 0\} = \rho_k$, i.e., the energy cost is only incurred during the fraction of the time the server is busy. The parameters ρ_c and ρ_w are the utilization of the cellular and WiFi network. According to Fig. 2, they can be separately calculated as: $\rho_c = \pi_c - \pi_{c,0}$ and $\rho_w = \pi_w - \pi_{w,0}$.

4.3 The ERWP Metric

We define a new metric, the Energy-Response time Weighted Product (ERWP) as:

$$ERWP = \mathbb{E}[\mathcal{E}]^{\omega} \cdot \mathbb{E}[T]^{1-\omega} \tag{12}$$

where $\mathbb{E}[T]$ and $\mathbb{E}[\mathcal{E}]$ are the mean response time and mean energy consumption, respectively, and ω (ranging between 0 and 1) is a weighting parameter that represents the relative significance of energy consumption and response time for the mobile device. To focus on performance, ω should be less than 0.5; to focus on power consumption, ω should be greater than 0.5. When ω equals 0.5, the focus is on both increasing performance and reducing power consumption.

Further, by substituting (9) and (10), into (12), we can formulate the explicit expressions and the optimization problem of the ERWP metric:

$$\gamma^{*} = \arg\min_{\gamma} \; ERWP \tag{13}$$

we seek the abandonment rate γ^{*} such that ERWP is minimised.

5. PERFORMANCE EVALUATION

Using measurements from real traces in [2], the average data rates of the cellular and WiFi networks are set as s_c=200 Kbps and s_w=2 Mbps, respectively. The average duration of WiFi availability period is 52 min ($\xi = 1/52$ min^{-1}), while the average duration with only cellular network coverage is 25.4 min ($\eta = 1/25.4$ min^{-1}). The availability ratio is thus 67%. The mean job size is assumed to be 10 MB. We set the power coefficients $p_c = 2.5$ W, $p_w = 0.7$ W and $p_m = 2$ W, respectively. Besides, suppose that the total job arrival rate is $\lambda = 0.5$ packet/min, the mobile service rate $\mu_m = 0.2$ and the cloud service rate $\mu_r = 1$.

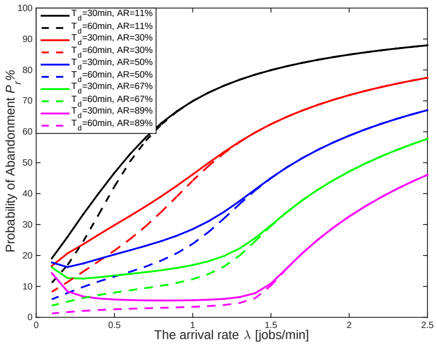

Figure 3: The abandonment probabilities for the delayed offloading models

As shown in Fig. 3, we find that jobs abandon the queue very often especially when the availability ratio (AR) of the WiFi network is relatively small. As *AR* increases, the percentage of jobs that abandon the *Offload Queue* declines rapidly. As the deadline increases from 30 min to 1 hour, jobs are more likely to be offloaded via the WiFi network, and thus the abandonment probability decreases at lower arrival rates. However, at high arrival rates, the abandonment probabilities stay the same under different deadlines.

We then compare the delayed offloading model with the non-delayed offloading model in [3], i.e., when there is WiFi available, all jobs are offloaded through the WiFi network; otherwise, they are offloaded via the cellular interface. Since jobs with higher deadlines have more chance to be transmitted with the fast WiFi network, leading to smaller response time; while with lower deadlines jobs leave the queue earlier, leading to smaller queueing delays. Therefore, by optimally choosing the abandonment rate γ, the delayed offloading model can achieve the smallest ERWP value as depicted in Fig. 4. As shown in Fig. 5, the non-delayed offloading model is more sensitive to the job arrival rate. When considering response time as a more important factor (e.g., for delay-sensitive applications), it is better to use the delayed offloading model; while at higher ω when considering energy consumption more important (e.g., for delay-tolerant applications), the non-delayed offloading model is preferred.

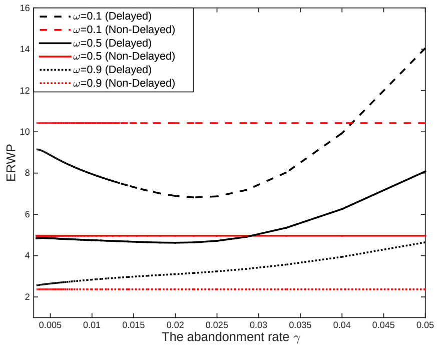

Figure 4: Comparison of ERWP values for the offloading models under different abandonment rates

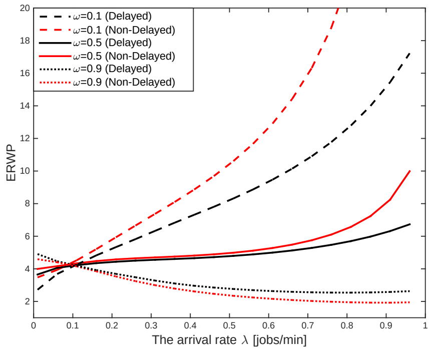

Figure 5: Comparison of ERWP values for the offloading models under different arrival rates

6. REFERENCES

[1] E. Hyytiä, T. Spyropoulos, and J. Ott. Optimizing offloading strategies in mobile cloud computing. 2013.

[2] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong. Mobile data offloading: How much can wifi deliver? *Networking, IEEE/ACM Transactions on*, 21(2):536–550, 2013.

[3] F. Mehmeti and T. Spyropoulos. Performance analysis of "on-the-spot" mobile data offloading. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 1577–1583. IEEE, 2013.

[4] F. Mehmeti and T. Spyropoulos. Is it worth to be patient? analysis and optimization of delayed mobile data offloading. In *INFOCOM, 2014 Proceedings IEEE*, pages 2364–2372. IEEE, 2014.

[5] N. Perel and U. Yechiali. Queues with slow servers and impatient customers. *European Journal of Operational Research*, 201(1):247–258, 2010.