Queueing analysis of service deferrals for load management in power systems

Andrés Ferragut and Fernando Paganini
Universidad ORT Uruguay

***Abstract*— With the advent of renewable sources and Smart-Grid deployments, it is increasingly common to control demands in order to reduce power consumption variability and thus the need for regulation, with load aggregators now exploiting the deferability of some power loads to smooth the consumption profile.**

In this paper, we analyze the impact of service deferrals and scheduling on power consumption variability using tools from queueing theory. We consider a generic model for a load aggregator that receive job requests, involving a certain amount of energy to be provided and a deadline. We analyze different scheduling policies and examine the impact of service deferrals, quantifying the tradeoff between variance reduction and attained deadlines.

I. Introduction

One of the basic functions of a power system is to maintain instantaneous balance between demand and supply, due to the limited availability of storage. Since the first-order effect of any imbalance is a deviation of AC frequency from its nominal value, the term *frequency regulation* is normally used to describe the real-time actions by the system operator to correct such imbalance.

Historically, frequency regulation has been implemented by adjusting the amount of power injected by fast ramping generators (hydro or gas turbines) in real time into the power grid [1]. This procedure comes from the traditional rationale of an exogenous uncertain demand, and supply under control of the operator. However, with the advent of renewable energy sources, supply is now also becoming unpredictable, which may require the installation of extra regulation capacities [2]. On the other hand, Smart-Grid deployments open the possibility of controlling power demand in real time by signaling users. While such *demand response* could include modifying mean consumption levels, in the context of short-term regulation it is more meaningful to focus on merely *deferring* some consumption (heating, AC, EV battery charging, etc.) to contribute to power balance.

Such a regulation service on the demand side could be provided by a *load aggregator* (e.g., [3], [4]) making dispatch decisions on behalf of a large enough number of individual loads. Some recent references exploring this potential for thermostatically controlled loads are [5]–[7]. More generically, [8], [9] analyze a collection of loads characterized by arrival times, deadlines, and power and energy requirements. In [8] different *scheduling* methods are studied from a numerical perspective, comparing classical approaches from processor scheduling (earliest deadline first, least laxity first [10]) with a model predictive control proposal. In [9] the authors attempt to characterize the aggregate flexibility provided by such load arrival profile in deterministic terms, invoking an equivalent electricity storage.

From a control perspective, recently [11] propose an architecture which enables the approximation of an aggregation of loads by a linear time invariant model to enable tracking of an external reference signal. Our previous work [12] pursues a similar objective with fluid models of the load arrival/service process, similar to those of large-scale queueing systems[1]. Also in [12] is a study, from a fluid aggregate perspective, of a basic tradeoff for stochastically arriving loads: deferring service can reduce consumed power variability (and thus helps with regulation), but also impacts quality expectations of users in terms of meeting deadlines.

This tradeoff is the central focus of the present paper, this time using more detailed stochastic queueing models, and considering a wider variety of service disciplines. The problem formulation is laid out in Section II, setting the main assumptions and parameters. In Section III we assume a fixed level of service deferral, and study the impact of this choice on both the consumption variance and the probability of missing deadlines. In particular we analyze an equal sharing policy with tools of $M/G/\infty$ queues, and provide approximate results for the LLF policy. In Section IV we consider an alternate situation in which deadlines are strictly enforced, and the remaining laxity is administered to reduce consumption variance. We use tools of point-process theory and Markov processes to characterize the variability for two relevant policies. Conclusions are given in Section V.

II. Queueing model for fixed deferred service

We now formulate the problem and notation by means of an initial queueing model, which will be later extended in Section IV. Consider a load aggregator entity which receives random energy requests, arriving as a Poisson process of intensity λ. The nominal power of each request is for simplicity taken to be a common parameter p_0, but the amount of energy Q_k of request k is random. Its nominal service time is thus given by

$$\sigma_k = \frac{Q_k}{p_0}.$$

This work was partially supported by AFOSR under grant FAA9550–15–1–0183 and ANII-Uruguay under grant FSE_1_2014_1_102426.

E-mail:ferragut@ort.edu.uy.

[1] See also [13] for queueing analysis of aggregation, without deferability.

Individual load requests may also have a *deadline*, before which they should have received full service. We model this by assuming that request k has an initial *laxity* ℓ_k, which is the amount of spare time or slackness it has on arrival. In other words, each load has a service deadline of $d_k = \sigma_k + \ell_k$ time units after its arrival. If not served after ℓ_k time units, laxity expires and service must be provided at full power to meet the deadline. If remaining laxity becomes negative, the job will miss its deadline even at full power.

Our initial focus is on the case where both σ_k and ℓ_k are independent exponential random variables, with $E[\sigma_k] = 1/\mu$ and $E[\ell_k] = 1/\gamma$. Nevertheless other cases will be considered below and we will indicate when the results remain valid for more general distributions.

The load profile is thus characterized by the parameters λ, μ and γ, and the nominal power of each load p_0. The mean aggregate power needed for a given load profile is given by:

$$\bar{p} = \lambda E[Q_k] = \lambda p_0 E[\sigma_k] = p_0 \frac{\lambda}{\mu}, \tag{1}$$

and this quantity is independent of any decision on load deferral or scheduling. In particular, given the expected energy requirements this amount of power would be purchased in advance by the load aggregator.

At any given time t, the load aggregator has a queue of $n(t)$ jobs present in the system, all of which may receive service or be deferred. To model service deferrals, we consider a fixed *service level* $u \in (0, 1]$, which represents the fraction of nominal power used by the system. Under this policy, the power consumed by the load aggregate at any time t is:

$$p(t) = p_0 n(t) u.$$

Setting $u = 1$ represents no service deferral, all loads are served at full power upon arrival. Choosing $u < 1$ means that either fewer loads are served at any given time, or that they are served with less power and consuming laxity, leading to deadline misses. Once the service level is chosen, the system has also the choice of which loads to serve, i.e. the detailed scheduling mechanism.

If we assume that all loads are eventually served (before or after deadline), then the average consumed power should match the average power of the load profile given by (1) and therefore in steady state:

$$\bar{p} = p_0 \frac{\lambda}{\mu} = E[p(t)] = E[p_0 n(t) u],$$

meaning that for a fixed service level u the expected number of loads in the system is:

$$\bar{n} = \frac{\lambda}{u\mu}. \tag{2}$$

We now state the two main objectives of the load aggregator in mathematical terms. The main purpose of the aggregation is to smooth the consumed power profile, minimizing deviations from the mean power $\bar{p}$. Let us denote by $\delta p(t) = p(t) - \bar{p}$ this real-time deviation. Then our first objective is to reduce the steady state *variance* of $p(t)$, $E[(\delta p)^2]$, a measure of the regulation cost.

The second objective is to serve as many load requests as possible before their deadlines. To quantify this aspect we introduce the *deferability factor* of the load profile:

$$\Delta := \frac{E[\ell_k]}{E[\sigma_k]} = \frac{\mu}{\gamma}. \tag{3}$$

Let T_k denote the time load k spends in the system, and $\bar{T}$ its mean. Applying Little's law, we have from equation (2):

$$\bar{n} = \lambda \bar{T} \Longrightarrow \bar{T} = \frac{1}{u\mu} = \frac{E[\sigma_k]}{u}.$$

This imposes a first constraint on u: in order to meet deadlines on average we should have $\bar{T} < E[\sigma_k + \ell_k]$ i.e.

$$u > \frac{E[\sigma_k]}{E[\sigma_k] + E[\ell_k]} = \frac{1}{1+\Delta}. \tag{4}$$

Here $\eta := \frac{1}{1+\Delta}$ is the minimum service level that loads should receive to meet their deadlines, in terms of the deferability factor. As deferability increases, $\eta \to 0$ and the system gains in flexibility, meaning that loads arrive with larger slack time. Of course, to evaluate system performance, other metrics should also be included, mainly the *missed deadline probability* $\alpha := P(T_k > \sigma_k + \ell_k)$.

Using this characterization of the load aggregator as a queueing system, we now explore the aforementioned tradeoffs between consumed power variability and missed deadlines, in terms of the service level u and the scheduling policies implemented by the system.

III. Fixed service deferral–impact on power consumption variability and missed deadlines

Our definition of the queueing system states that the total output power for service level u is given by $p(t) = p_0 n(t) u$. We now analyze two simple scheduling policies operating under this constraint: first we consider an *equal sharing* policy, where every load is served at individual rate $p_0 u$. While this is not always possible in the context of energy distribution, it serves as a baseline to compare other policies. Also, it can be approximately implemented by serving a subset of the current jobs taken at random from the population, with probability u. This scheduling was analyzed through a fluid model by the authors in [12].

The second case is the *least-laxity-first (LLF)* policy, introduced by [10] in the context of scheduling and discussed in [6], [7] in the context of power systems. Here the idea is to serve first the loads with least spare time remaining, up to the aggregate power determined by the service level.

A. Equal sharing policy

Under the equal sharing policy, every load present in the system is served at the reduced rate $p_0 u$, thus taking longer to finish. The time in the system is given by:

$$T_k = \frac{Q_k}{p_0 u} = \frac{\sigma_k}{u}$$

Since all requests are served in parallel, the system behaves as an infinite server queue, with arrival rate λ and average service time $E[T_k] = 1/(u\mu)$. In particular, the number of loads in steady state satisfies:

$$n(t) \sim \text{Poisson}\left(\frac{\lambda}{u\mu}\right).$$

We remark here that, since the steady state population in an $M/G/\infty$ queue is insensitive to the detailed characteristics of the job size distribution, the above result holds for general distribution of σ_k.

We conclude that, in steady state,

$$\bar{n} = E[n(t)] = \frac{\lambda}{\mu u}, \quad \bar{p} = E[p(t)] = E[p_0 n(t) u] = \frac{p_0 \lambda}{\mu}$$

consistently with our previous analysis.

Our analysis can however now go further, computing the *variance* of consumed power

$$\begin{aligned} E[(\delta p)^2] &= E[p_0^2 u^2 (n(t) - \bar{n})^2] = p_0^2 u^2 \text{Var}(n(t)) \\ &= p_0 \bar{p} u, \end{aligned} \tag{5}$$

where we have invoked the variance of the Poisson distribution. A more normalized way of expressing variability is the *coefficient of variation* $cv^2(p)$ defined by

$$cv^2(p) = \frac{\text{Var}(p)}{\bar{p}^2},$$

which can be readily computed from (1), (5) to yield:

$$cv^2(p) = \frac{p_0}{\bar{p}} u. \tag{6}$$

So we find that variability reduces linearly with the service level u. Also, as aggregation grows large in the sense that $\bar{p}/p_0$ is large, the system reduces its variability in consumed power.

We now turn our attention to deadline misses. The probability of missing the deadline is:

$$\begin{aligned} \alpha &= P\left(T_k > \sigma_k + \ell_k\right) = P\left(\frac{\sigma_k}{u} > \sigma_k + \ell_k\right) \\ &= P\left(\frac{\sigma_k}{u} > \frac{\ell_k}{1-u}\right). \end{aligned} \tag{7}$$

The previous equation can be reinterpreted as follows: when a job arrives with service time σ_k and initial laxity ℓ_k and is served at rate u, after a time dt the remaining service time will be $\sigma' = \sigma_k - u dt$. Since its deadline is $\sigma_k + \ell_k - dt$ time units ahead, the remaining laxity after a time dt will be:

$$\ell' = \sigma_k + \ell_k - dt - (\sigma_k - u dt) = \ell_k - (1-u)dt.$$

This means that for service level u, laxity is consumed at rate $1-u$, and (7) simply states that laxity is consumed before service. A depiction of the equal sharing policy in the service-laxity space is given in Fig. 1: all loads present in the system consume service and laxity in certain fixed proportions, therefore points move following the same vector.

Under the exponential assumptions, α can be readily calculated by observing that $\frac{\sigma_k}{u} \sim \exp(u\mu)$ and $\frac{\ell_k}{1-u} \sim \exp((1-u)\gamma)$. Using the minimum of two exponential random variables we get:

$$\alpha = \frac{\gamma(1-u)}{\mu u + \gamma(1-u)} = \frac{(1-u)}{\Delta u + (1-u)}. \tag{8}$$

Deadline misses are decreasing in u as expected. In particular, for $u = \eta = 1/(1+\Delta)$, $\alpha = 1/2$.

Analogous calculations can be performed for any joint distribution in (σ, ℓ). For comparison purposes we compute also the probability for deterministic service time $\sigma_k \equiv \frac{1}{\mu}$ which yields:

$$\alpha^{ds} = P\left(\frac{1}{\mu u} > \frac{\ell_k}{1-u}\right) = 1 - e^{-\frac{1}{\Delta}\frac{1-u}{u}}.$$

For deterministic laxity $\ell_k \equiv \frac{1}{\gamma}$ the corresponding expression is:

$$\alpha^{dl} = P\left(\frac{\sigma_k}{u} > \frac{1}{\gamma(1-u)}\right) = e^{-\Delta\frac{u}{1-u}}.$$

The three cases are depicted in Fig. 2 for a deferability parameter of $\Delta = 2$. As we can see, deadline misses are rather high even for moderate values of $u > \eta$. We turn to an alternate policy that seeks to minimize deadline misses.

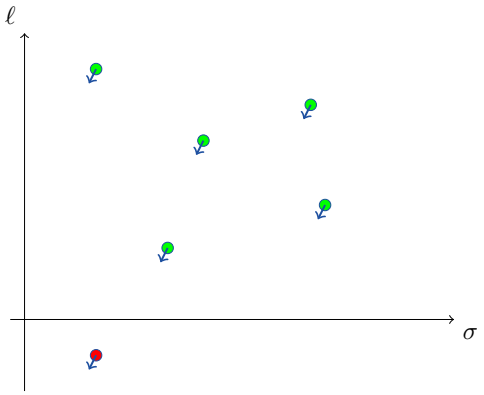


Fig. 1. Equal sharing scheduling for $u = 1/3$.

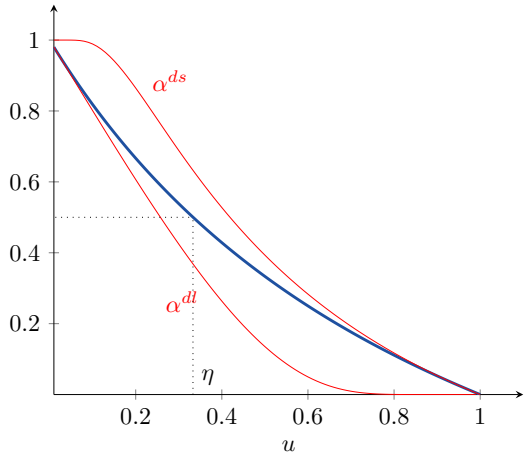


Fig. 2. Missed deadline probability as a function of u for $\Delta = 2$.

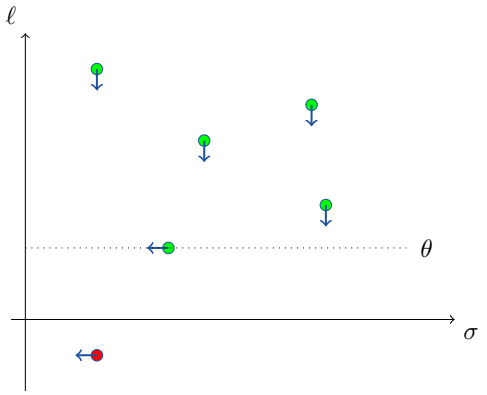


Fig. 3. LLF scheduling for $u = 1/3$.

B. Least-laxity-first

The previous policy is agnostic to whether the deadlines are about to expire. Assuming global information and central control, it would be better to schedule service taking into account the remaining laxity of the loads. The least-laxity-first (LLF) policy is defined in the following way: sort the current jobs by increasing laxity, and serve the first $k(t) = n(t)u$ at nominal power.[2] The remaining jobs will consume laxity until they are scheduled. The LLF policy was introduced in the context of processor time scheduling [10], and has been thoroughly analyzed in the case of single-server queues (cf. [14] for a recent treatment). The difference here is that we are dealing with an infinite server system.

A depiction of the LLF policy in service-laxity space is given in Fig. 3. It is convenient to define the *frontier process*

$$\theta(t) := \sup \left\{ \ell : \sum_{k=1}^{n(t)} \mathbf{1}_{\{\ell_k \leq \ell\}} < n(t)u \right\}. \qquad (9)$$

Then $\theta(t)$ represents the maximum laxity of the loads currently in service. Loads with laxity greater than $\theta(t)$ only consume laxity and are not served.

We now analyze the steady state occupation and power output of the LLF policy for exponentially distributed service times. We begin by the following:

Proposition 1: Under the LLF policy and exponential service times, the total occupation of the system $n(t)$ evolves as in the equal sharing policy.

Proof: Due to the LLF definition, at any time t there are $n(t)u$ loads in service. Due to the memoryless property of the exponential distribution, the service process of the system for occupation state $n(t)$ and service level u corresponds to $n(t)u$ exponential servers in parallel. Therefore, the total population evolves as a birth-death process with birth rate λ and death rate $\mu n u$, i.e. as in the $M/M/\infty$ of the equal sharing policy. ∎

In particular we conclude that in steady state, $n \sim \text{Poisson}\,(\lambda/(\mu u))$, the average system occupation is again $\bar{n} = \lambda/(\mu u)$ and the output variance is again given by (6), i.e. it is linear in u. This was observed empirically in [12].

[2]More precisely, $\lfloor n(t)u \rfloor$ are served at full power. The remaining power should be allocated proportionally to the difference between $n(t)u$ and $k(t)$ but this rounding error is negligible in a large scale system.

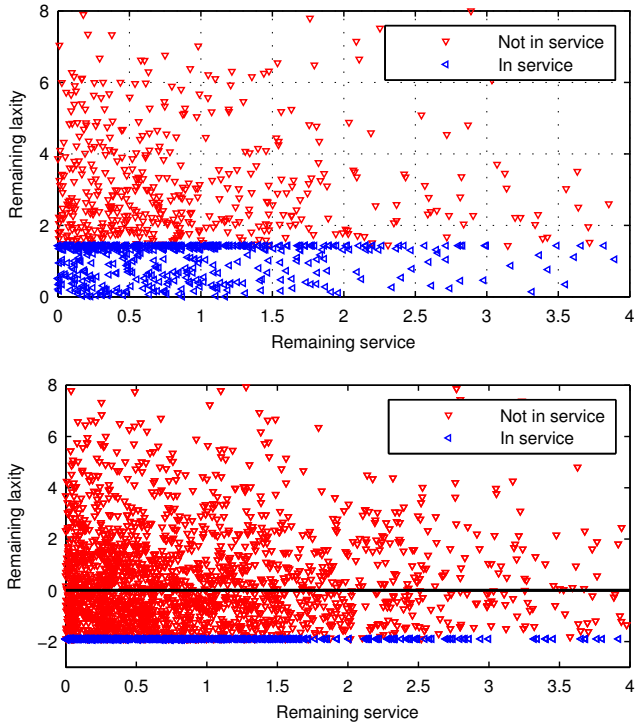


Fig. 4. Remaining service and laxities for LLF when $u > \eta$ (above) and $u < \eta$ (below), with $\lambda/\mu = 500$.

We now analyze the behavior of the system when the scale is large ($\lambda \to \infty$). It was also observed in [12] that under the LLF policy deadline misses show a sharp decline when the service level satisfies $u > \eta$. To understand this, we plot in Fig. 4 two simulation experiments for a system with $\lambda/\mu = 500$ and $\Delta = 2$ ($\eta = 1/3$). In the first case $u = 0.5 > \eta$ and the frontier process $\theta(t)$ finds a positive equilibrium θ^*. Loads arriving with laxity greater that θ^* consume laxity down to level θ^* and then they are served. In the second case, with $u = 0.2 < \eta$, loads expire their laxities before being served when they reach an equilibrium value of $\theta^* < 0$.

By applying Little's law, we can characterize the equilibrium value θ^* through a fixed-point analysis. We do so in the case where laxity is exponentially distributed. Recall that, from Proposition 1, the average number of clients in the system is $\bar{n} = \lambda/(\mu u)$. Therefore, the average time in the system by Little's law is $\bar{T} = 1/(\mu u)$. If $\theta^* > 0$ we can compute the average time as:

$$\bar{T} = \frac{1}{\mu u} = E[\ell_k - \theta^* \mid \ell_k > \theta^*] P(\ell_n > \theta^*) + E[\sigma_k].$$

The first term simply states that loads arriving with laxity greater that θ^* should wait to consume their laxity up to level θ^* before being served. If $\ell_k \sim \exp(\gamma)$ the above equation becomes:

$$\frac{1}{\mu u} = \frac{1}{\gamma} e^{-\gamma \theta^*} + \frac{1}{\mu},$$

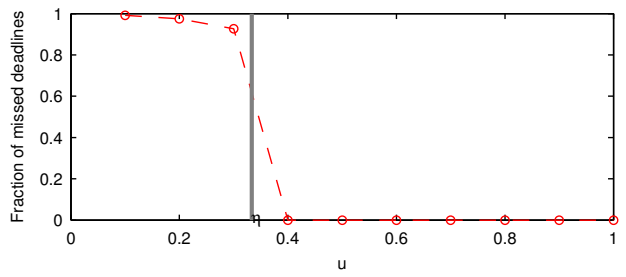


Fig. 5. Empirical missed deadline probability as a function of u for LLF scheduling with $\Delta = 2$ ($\eta = 1/3$) and $\lambda/\mu = 500$.

or equivalently:

$$\theta^* = \frac{1}{\gamma} \log \left[\frac{\Delta u}{1-u} \right]. \tag{10}$$

Note that (10) yields a positive solution provided $\Delta u > 1 - u$, i.e. $u > \frac{1}{1+\Delta} = \eta$. Therefore, provided $u > \eta$, the frontier converges to a positive equilibrium and deadlines are attained.

If $\theta^* < 0$, the average time must be computed as:

$$\bar{T} = \frac{1}{\mu u} = (E[\ell_k] - \theta^*) + \frac{1}{\mu},$$

which follows from the fact that loads must wait an extra time $|\theta^*|$ before receiving service. Solving for θ^* we get:

$$\theta^* = \frac{1}{\gamma} - \frac{1}{\mu} \frac{1-u}{u}. \tag{11}$$

Note that provided $u < \eta$, (11) gives a negative solution, as expected. Therefore, in steady state, the frontier converges to a negative equilibrium and all deadlines are missed.

While the above discussion is valid in the large scale limit (where the process $\theta(t)$ becomes constant), the approximation is indeed good for moderate values of λ, as depicted in Fig. 5.

The main conclusion of this analysis is that, for large scale systems using LLF, the service level can be reduced almost up to η (thereby reducing variance), without great impact on deadline misses. Of course, this comes at the cost of having a complex scheduling policy: the load aggregator should possess detailed information of the remaining laxity of each job request in order to perform scheduling. In the next Section we analyze a different class of policies that cope with deadlines in a way more amenable to decentralization.

IV. Policies that strongly enforce deadlines

In the previous section the focus was on scheduling policies that reduce the global service level by a fixed amount, and we analyzed the impact of this choice in the amount of variability in consumed power as well as on deadline misses. We now turn our attention to a different class of scheduling policies: in these, we impose that no deadlines are missed, and analyze the impact on steady state variance.

A. Exact scheduling

The first policy in this family is called *exact scheduling*. Here, service requests arrive as a Poisson process of intensity λ, each load bringing a service time σ_k and initial laxity ℓ_k given by a joint density $f(\sigma, \ell)$ in the positive orthant. Given σ_k and ℓ_k, the service level for load k is chosen as:

$$u_k = \frac{\sigma_k}{\sigma_k + \ell_k},$$

and therefore the time spent by load k in the system is:

$$T_k = \frac{\sigma_k}{u_k} = \sigma_k + \ell_k.$$

Namely, each request is served at exactly the amount of power needed to meet its deadline. A depiction of the policy is given in the first graph of Fig. 6. Note that this policy is very easy to decentralize provided that loads can tune their service level, since job requests are already aware of their energy requests and deadline.

We would like to compute the steady state variance of consumed power $E[(\delta p)^2]$ for this system. The main challenge here is that the service level is determined by the job request characteristics. In order to express the steady state characteristics it is better to perform the following change of variables:

$$z = \sigma + \ell,$$
$$u = \frac{\sigma}{\sigma + \ell}.$$

Here $z \geqslant 0$ represents the amount of time spent in the system and $u \in [0, 1]$ the service level. For given z and u we can recover the original variables as $\sigma = uz$, $\ell = (1-u)z$. The Jacobian is given by $|J_{(\sigma,\ell)}(z, u)| = z$. The joint distribution of (z_k, u_k) can be readily computed as:

$$g(z, u) = z f(uz, (1-u)z). \tag{12}$$

Note that in these new variables, requests are served in parallel at rate 1 along the variable z, while u is fixed throughout service, as depicted in the second graph of Fig. 6.

The state of the system at time t is expressed through the following counting measure [15] in $\mathbb{R}^+ \times [0, 1]$:

$$\Phi_t = \sum_{i=1}^{n(t)} \delta_{(z_i, u_i)}, \tag{13}$$

where z_i represents the remaining service time of job i and u_i its service level.

With the above definitions, Φ_t is a measure-valued Markov process with simple dynamics: current jobs are translated to the left at rate 1 while new jobs arrive as a Poisson process with marks z_k, u_k distributed as $g(z, u)$ given by (12). Since all jobs are served in parallel at rate 1 the system behaves as an $M/G/\infty$ queue. We have the following Theorem, derived from the steady state characteristics of the $M/G/\infty$ queue [15], [16]:

Theorem 1: Under the exact scheduling policy with requests distributed as $f(\sigma, \ell)$, the distribution of Φ_t in steady

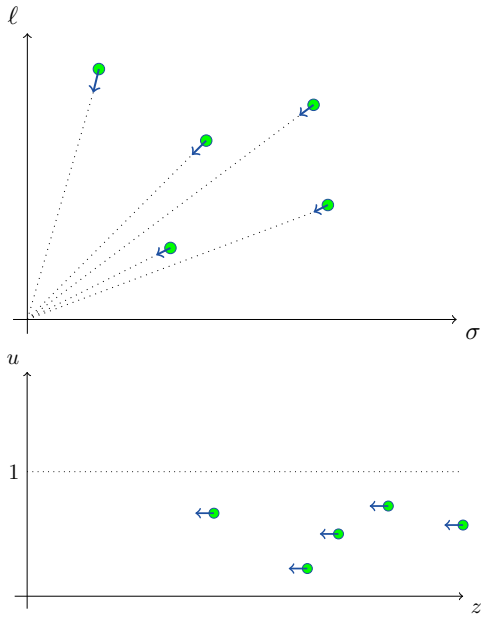


Fig. 6. Exact scheduling depicted in service-laxity space and in the time-service level coordinates.

state is a Poisson point process on $\mathbb{R}^+ \times [0,1]$ with mean measure density:

$$h(z,u) = \lambda \int_z^\infty g(w,u)dw. \tag{14}$$

As a consequence of this result, average characteristics of the steady-state process can be computed by suitable integrals with respect to the above density; we refer to [17] for these derivations. In particular:

- The mean number of loads present in the system is

$$\bar{n} = E[n(t)] = \int_0^\infty \int_0^1 h(z,u)dzdu. \tag{15}$$

- The mean aggregate power of the loads is given by

$$E[p(t)] = p_0 E\left[\textstyle\sum_i u_i\right] = p_0 \int_0^\infty \int_0^1 uh(z,u)dzdu. \tag{16}$$

- The variance of aggregate power is given by

$$\mathrm{Var}[p(t)] = p_0^2 \mathrm{Var}\left[\textstyle\sum_i u_i\right] = p_0^2 \int_0^\infty \int_0^1 u^2 h(z,u)dzdu. \tag{17}$$

To gain some intuition, it is worth specializing the above results to the case of exponential requests. In that case we have:

$$f(\sigma,\ell) = \mu\gamma e^{-\mu\sigma-\gamma\ell}, \quad \sigma,\ell > 0,$$

and in consequence:

$$g(z,u) = \mu\gamma z e^{(\mu u+\gamma(1-u))z}$$

Define $\nu := \mu u + \gamma(1-u)$, then the steady state density is given by computing the integral in (14) to yield:

$$h(z,u) = \lambda\mu\gamma \left[\frac{\nu z+1}{\nu^2}\right] e^{-\nu z}.$$

With this density we can carry out the calculations indicated in (15-17). In particular, it is easily checked that

$$\int_0^\infty h(z,u)dz = \frac{2\lambda\mu\gamma}{\nu^3},$$

so the mean number of customers is given by

$$\begin{aligned}\bar{n} &= \int_0^1 \frac{2\lambda\mu\gamma}{(\mu u+\gamma(1-u))^3}du \\ &= \frac{\lambda\mu\gamma}{\mu-\gamma}\int_\mu^\gamma \frac{2}{\nu^3}d\nu \\ &= \frac{\lambda\mu\gamma}{\mu-\gamma}\left(\frac{1}{\mu^2}-\frac{1}{\gamma^2}\right) \\ &= \lambda\left(\frac{1}{\mu}+\frac{1}{\gamma}\right).\end{aligned} \tag{18}$$

Equation (18) simply states that the average number of customers in the system is the arrival rate λ, times the expected service time $E[\sigma_k+\ell_k] = \frac{1}{\mu}+\frac{1}{\gamma}$, consistent with the fact that the system behaves as an $M/G/\infty$ queue. It can also be readily verified that $E[p(t)] = p_0 E[\sum_i u_i] = p_0\lambda/\mu = \bar{p}$, as expected.

While service level in this system is load dependent, an *effective* service level can be computed as point of comparison, dividing the mean power consumed by the mean nominal power of the loads present:

$$u_{\text{eff}} = \frac{E[p(t)]}{p_0 E[n(t)]} = \frac{\lambda/\mu}{\lambda\left(\frac{1}{\mu}+\frac{1}{\gamma}\right)} = \frac{\gamma}{\mu+\gamma} = \eta,$$

i.e. the exact scheduling policy works at a service level comparable to taking $u = \eta$ in the preceding policies.

More importantly, using (17) and integration by parts we can compute the steady state power variance to be

$$\begin{aligned}E[(\delta p)^2] = p_0^2\lambda\mu\gamma \Bigg[&-\frac{1}{(\mu-\gamma)\mu^2} - \frac{2}{(\mu-\gamma)^2\mu} \\ &+\frac{2}{(\mu-\gamma)^3}\log\left(\frac{\mu}{\gamma}\right)\Bigg].\end{aligned} \tag{19}$$

A more amenable measure is again the coefficient of variation $cv^2(p) = E[(\delta p)^2]/\bar{p}^2$ which can be derived from (19) and expressed in terms of the deferability factor:

$$cv^2(p) = \frac{p_0}{\bar{p}}\left[\frac{1}{1-\Delta} - \frac{2\Delta}{(1-\Delta)^2} - \frac{2\Delta^2\log(\Delta)}{(1-\Delta)^3}\right]. \tag{20}$$

We shall use the above expression to compare the performance of the different policies at the end of the Section.

B. Laxity expiring policy

Finally, let us consider the following simple policy:

- Apply a fixed service level $\tilde{u} \in (0,1]$ only to loads with positive remaining laxity[3].
- If laxity of load k expires, serve the load at full power.

A depiction of the trajectories of this policy is given in Fig. 7. The main advantage of this policy is that it is very easy to decentralize. The system operator fixes a service level $\tilde{u}$

[3]This is not an overall service level, hence the new notation $\tilde{u}$.

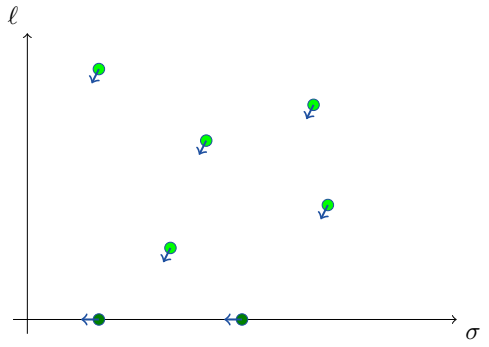
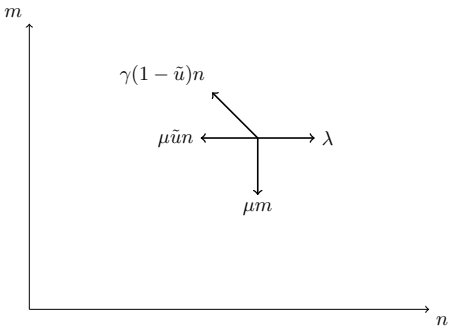


Fig. 7. Laxity expiring scheduling for $u = 1/3$.

for those loads that still have laxity, and distributes this as a common signal. When a given load reaches the point where it cannot be deferred any longer, it starts consuming power at maximum rate.

Under the exponential job/laxity size assumption, the above queue has a very simple model. Let $n(t)$ denote the number of jobs with positive laxity and $m(t)$ those whose laxity has expired. Then $(n(t), m(t))$ is a continuous time Markov chain with state space $\mathbb{N}^2$ and the following transition rates:

$$\begin{array}{ll} (n,m) \mapsto (n+1,m): & \lambda \\ (n,m) \mapsto (n-1,m): & \mu\tilde{u}n \\ (n,m) \mapsto (n-1,m+1): & \gamma(1-\tilde{u})n \\ (n,m) \mapsto (n,m-1): & \mu m \end{array} \tag{21}$$



The above Markov chain has a product form solution:

Proposition 2: The equilibrium distribution of the Markov process defined by (21) is given by:

$$\pi(n,m) = e^{-\rho_n - \rho_m} \frac{\rho_n^n}{n!} \frac{\rho_m^m}{m!}, \quad n,m \in \mathbb{N} \tag{22}$$

i.e. in steady-state n and m behave as independent Poisson random variables with parameters:

$$\rho_n = \frac{\lambda}{\mu\tilde{u} + \gamma(1-\tilde{u})}, \quad \rho_m = \frac{\gamma(1-\tilde{u})}{\mu}\rho_n$$

These average values can be rewritten in terms of the following:

$$\nu := \mu\tilde{u} + \gamma(1-\tilde{u}),$$
$$\alpha := \frac{\gamma(1-\tilde{u})}{\mu\tilde{u} + \gamma(1-\tilde{u})}$$

With the above definition, $1/\nu$ is the average time before either the laxity or the service of a given job ends, and α is the probability that the laxity expires before the job ends, and thus the job starts service in the second queue. Note that α has the same form as the missed deadline probability for the equal sharing policy in the previous section. We can then rewrite:

$$\rho_n = \frac{\lambda}{\nu}, \quad \rho_m = \frac{\alpha\lambda}{\mu}$$

Noting that the output power p is $p_0\tilde{u}$ for the first n loads and p_0 for the remaining m loads, we can compute:

$$\bar{p} = E[p(t)] = E[p_0(n\tilde{u} + m)] = p_0\left(\tilde{u}\frac{\lambda}{\nu} + \frac{\alpha\lambda}{\mu}\right)$$
$$= p_0\frac{\lambda}{\mu}\underbrace{\left[\frac{\tilde{u}\mu + \nu\alpha}{\nu}\right]}_{=1} = p_0\frac{\lambda}{\mu},$$

as expected.

We can also quantify the deviations from equilibrium in steady-state as:

$$\begin{aligned} E[(\delta p)^2] &= \mathrm{Var}[p_0(n(t)\tilde{u} + m(t))] \\ &= p_0^2(\tilde{u}^2\mathrm{Var}(n(t)) + \mathrm{Var}(m(t))) \\ &= p_0^2\left(\rho_n\tilde{u}^2 + \rho_m\right) \\ &= p_0^2\frac{\lambda}{\mu}\left[1 - \frac{\mu\tilde{u}(1-\tilde{u})}{\nu}\right]. \end{aligned}$$

where we have used that n and m are independent random variables in steady state due to the product form distribution.

We can see that $E[(\delta p)^2] \leqslant p_0^2\frac{\lambda}{\mu}$ and in fact this is achieved for $\tilde{u} = 0$ or $\tilde{u} = 1$. The case $\tilde{u} = 0$ corresponds to not giving any service until laxity expires, effectively delaying arrival for all jobs to the second queue and losing control on deferability. The case $\tilde{u} = 1$ corresponds to serving the loads at full power upon arrival, so in steady state $m \equiv 0$ and the system behaves as in the equal sharing policy with $u = 1$.

Again, it is better to express the variability in normalized units, by computing the coefficient of variation as:

$$cv^2(p) = \frac{E[(\delta p)^2]}{\bar{p}^2} = \frac{p_0}{\bar{p}}\left[1 - \frac{\Delta\tilde{u}(1-\tilde{u})}{\Delta\tilde{u} + (1-\tilde{u})}\right] \tag{23}$$

The above expression is minimized at:

$$\tilde{u}^* = \frac{1}{1+\sqrt{\Delta}},$$

and the minimal value of $cv^2(p)$ for this policy is:

$$cv^2(p)\big|_{\tilde{u}=\tilde{u}^*} = \frac{p_0}{\bar{p}}\left[1 - \frac{1}{(1+\sqrt{1/\Delta})^2}\right].$$

Note that both the optimal value of $\tilde{u}$ as well as the ratio between optimal and maximal variance do not depend on the arrival rate, and only on the ratio between the parameters μ and γ for the load service and laxity times. The effective service level in the optimal value can be computed as:

$$u_{\text{eff}}^* = \frac{\bar{p}}{p_0\ E[n(t)+m(t)]|_{\tilde{u}=\tilde{u}^*}} = \frac{1+\sqrt{\Delta}}{1+\Delta+\sqrt{\Delta}}.$$

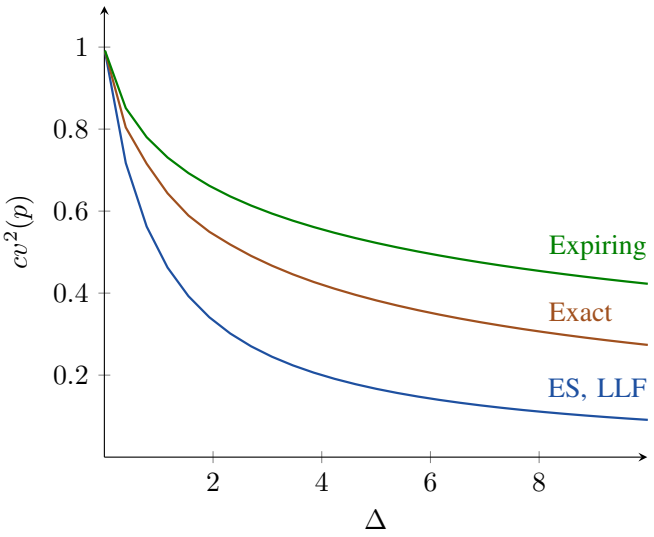


Fig. 8. Normalized coefficient of variation for the different scheduling policies.

It turns out that $u^*_{\text{eff}} > \frac{1}{1+\Delta}$, consistent with the fact that some of the loads are given full power and thus losing control on their deferral.

As a final comparison, we summarize below the variability measures for the four scheduling policies discussed in the paper. For Equal Sharing and LLF we use the coefficient of variation for the case $u = \eta$; this is the minimum value for which LLF can comply with deadlines with high probability, as shown before[4]. For the laxity expiring policy we use the optimal value u^* discussed above. The expressions are:

$$cv^2(p)_{ES} = cv^2(p)_{LLF} = \frac{p_0}{\bar{p}} \frac{1}{1+\Delta}, \tag{24a}$$

$$cv^2(p)_{ExS} = \frac{p_0}{\bar{p}} \left[\frac{1}{1-\Delta} - \frac{2\Delta}{(1-\Delta)^2} - \frac{2\Delta^2 \log(\Delta)}{(1-\Delta)^3} \right], \tag{24b}$$

$$cv^2(p)_{ExpL} = \frac{p_0}{\bar{p}} \left[1 - \frac{1}{(1+\sqrt{1/\Delta})^2} \right]. \tag{24c}$$

The above expressions are plotted for comparison in Fig. 8. As deferability increases (Δ grows), the best policy is LLF, at the cost of having a detailed information of the current system state. The laxity expiring policy is consequently the worst since it only controls a fraction of the loads, with the exact scheduling achieving intermediate results.

V. Conclusions

In this paper, we analyzed how deferring service of power loads can be used to reduce power consumption variability, an important problem in Smart-grid deployments aimed at reducing frequency regulation needs. We derived a queueing model for a load aggregator entity that manages service requests, characterized by service times and deadlines. We analyzed different queueing policies with different degrees of complexity and attention to deadlines. For these policies, we computed the coefficient of variation of power consumption in terms of the deferability characteristics of the load profile, as well as the probability of missed deadlines, therefore quantifying the tradeoff between variance reduction and quality of service.

Several lines of future work remain open. In the case of the least-laxity-first scheduling, a more detailed model for general service times and laxities is in order, as well as the transient behavior of in-service/not-in-service loads and the frontier process. In more general terms, an economic analysis of how users may be incentivized to cooperate by adjusting the service level and declare their true deadlines is an interesting line to pursue.

[4]For Equal Sharing the fraction of missed deadlines would be high.

Acknowledgements

The authors would like to thank Adam Wierman for many hours of insightful discussions and Federico Bliman for his help with the analysis of simulations.

References

[1] New York Independent System Operator, "Ancillary services manual," http://www.nyiso.com/public/webdocs/markets_operations/documents/Manuals_and_Guides/Manuals/Operations/ancserv.pdf .
[2] H. Holttinen and et al., "Design and operation of power systems with large amounts of wind power," IEA wind, Tech. Rep., 2009.
[3] EnergyPool, http://www.energy-pool.eu/en/.
[4] GoodEnergy, http://www.goodenergy.com.
[5] S. Koch, J. Mathieu, and D. Callaway, "Modeling and Control of Aggregated Heterogeneous Thermostatically Controlled Loads for Ancillary Services," in *Proc. of the 17th Power Systems Computation Conference*, 2011.
[6] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "A Generalized Battery Model of a Collection of Thermostatically Controlled Loads for Providing Ancillary Service," in *Proc. of the 51st Allerton conference on Communication, Control and Computing*, 2013.
[7] ——, "Frequency Regulation from Flexible Loads: Potential, Economics, and Implementation," in *Proc. of the American Control Conference (ACC)*, 2014.
[8] A. Subramanian, M. Garcia, D. Callaway, K. Poolla, and P. Varaiya, "Real-Time Scheduling of Distributed Resources," *IEEE Transactions on Smart Grid*, vol. 4, pp. 2122–2130, 2013.
[9] A. Nayyar, J. Taylor, A. Subramanian, K. Poolla, and P. Varaiya, "Aggregate Flexibility of a Collection of Loads," in *Proc. of the 52nd IEEE Conference on Decision and Control*, 2013.
[10] J. Hong, X. Tan, and D. Towsley, "A performance analysis of minimum laxity and earliest deadline scheduling in a real-time system," *IEEE Trans. on Computers*, vol. 38, pp. 1736–1744, 1989.
[11] S. Meyn, P. Barooah, A. Bušić, Y. Chen, and J. Ehren, "Ancillary service to the grid using intelligent deferrable loads," *IEEE Trans. on Automatic Control*, to appear.
[12] F. Bliman, A. Ferragut, and F. Paganini, "Controlling Aggregates of Deferrable Loads for Power System Regulation," in *Proc. of the 2015 American Control Conference, Chicago, IL, June*, 2015.
[13] G. D. Bella, L. Giarrè, M. Ippolito, A. Jean-Marie, G. Neglia, and I. Tinnirello, "Modeling Energy Demand Aggregators for Residential Consumers," in *Proc. of the 52nd IEEE Conference on Decision and Control*, 2013.
[14] H. C. Gromoll and L. Kruk, "Heavy traffic limit for a processor sharing queue with soft deadlines," *Annals of Applied Probability*, vol. 17, no. 3, pp. 1049–1101, 2007.
[15] P. Robert, *Stochastic networks and queues*. Springer, 2003.
[16] S. Zachary, "A note on insensitivity in stochastic networks," *Journal of Applied Probability*, vol. 44, pp. 238–248, 2007.
[17] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks, Volume I - Theory*. Now Publishers, 2009.