Universidad ORT Uruguay
Facultad de Ingeniería

Frequency regulation in electric power systems using deferrable loads

Entregado como requisito para la obtención del título de Master en ingeniería

Federico Bliman #181245
Tutor: Fernando Paganini
Tutor: Andrés Ferragut

2016

DECLARACIÓN DE AUTORÍA

Yo, Federico Bliman, declaro que el trabajo que se presenta en esta obra es de mi propia mano. Puedo asegurar que:

- La obra fue producida en su totalidad mientras realizaba el Master en Ingeniería;
- Cuando he consultado el trabajo publicado por otros, lo he hecho con claridad;
- Cuando he citado obras de otros, he indicado las fuentes. Con excepción de estas citas, la obra es enteramente mía;
- En la obra, he acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, he explicado claramente qué fue contribuido por otros, y qué fue contribuido por mi;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

Federico Bliman
xx de Junio 2016

AGRADECIMIENTOS

Primero que nada quisiera agradecer a mis tutores Fernando y Andrés por guiarme en este proceso y compartir conmigo su conocimiento. Sin duda que el resultado de estos años de trabajo no hubiera sido el mismo sin ellos. También quiero agradecerle a Daniel Kofman por haberme introducido en el tema y haberme dado la oportunidad de entrar en el mundo de la investigación.

El agradecimiento prinicipal es para Mari, por haberme apoyado en los momentos de crisis durante estos años que fueron de mucho aprendizaje y cambio, no solo en lo académico y profesional, sino también en la forma de percibir la vida, el mundo y quienes lo habitamos.

ABSTRACT

Electric power systems are changing. The current trend is to abandon fuel, carbon and gas generators in pursuit of greener generators that are supplied by wind or sun. Although this trend has many benefits, electric systems are not prepared to receive much energy from these new sources as they are not fully controllable because they depend on factors external to human control. Up to now electric grids were controlled from generation side. As there is almost no electric storage in the grids all the power being consumed at a given instant must me generated at the same moment. In order to complete the transition to a greener system this control paradigm must change and the demand side must play an important role in this new control logic.

In this thesis we study the problem of a load aggregator that manages a set of loads from its customers and exploit the flexibility of the loads to provide frequency regulation to the grid. We study the problem from a macroscopic point of view without entering into individual load details. We propose a set of ODE models to predict the evolution of the power consumed by the cluster of loads and we design controllers for this models in order to be able to follow external power references. We finish by suggesting some possible algorithms in order to implement the control to individual loads. Simulations show that this system could provide valuable services to electric grids if sufficient communications infrastructure is available.

Key words: Frequency regulation; Demand response; Smart Grid; Optimal h2 control.

Content

Chapter 1

Introduction

1.1 BACKGROUND

1.1.1 Electric power systems

Electric power systems are one of the largest and more complex human creations. Each electric grid is composed by thousands or millions of components that interact with each other to move energy from generators to consumers.

From the point of view of the power grid, individual customers and their appliances are small, numerous, and hardly discernible as distinct loads. While consumers typically think of their electricity usage in terms of a quantity of energy (in kilowatt-hours) consumed over the course of a billing period, the quantity of interest to system operators and planners is the power (in kilowatts or megawatts, measuring the instantaneous rate of energy flow) demanded at any given time. The term demand thus refers to a physical quantity of power, not energy. Serving that instantaneous demand under diverse circumstances is the central challenge in designing and operating power systems, and the one that calls for the majority of investment and effort [1].

Electric systems have the particularity that they have almost no electric storage capacity. This does not mean that there is no energy storage to feed the grid, there is energy stored in fossil fuels, carbon, water, etc. The lack of storage in the grid itself means that the consumed electric energy in a given moment must also be generated at the same instant. Generation must be controlled to keep the system at its nominal working point; this means that the energy generated should be the one demanded by all the loads connected to the system under nominal conditions ($230V$ and $50Hz$ in Uruguay) plus the losses in transmission. If there are imbalances between generation and nominal demand this would impact the equilibrium of the system moving voltage, frequency, temperatures, etc, out of their nominal

parameters with the associated risk to the system.

It is interesting to reflect upon the historical service philosophy that considers demand as the independent variable that is to be met by supply at any costs. The assumption embedded in both the hardware design and the operating culture of electric power systems is that customers freely determine how much power they want, and that it is the job of power system designers and operators to bend over backwards if necessary to accommodate this demand [1].

Matching the generation to demand is a complex task that involves several actors and time scales. Going from the longer term to the shorter term we have: infrastructure planning, long term contracts, day ahead planning, intraday planning and real-time balancing. Infrastructure planning includes mainly predicting how demand will evolve along the years and having always enough generation and transportation capacity to meet peak demands.

Having the necessary infrastructure, the daily routine of system operators (SOs) starts the day before dispatching by scheduling the generators to meet the predicted demand. The task of scheduling the generators depends on the structure of the market and the physical limitations of the system. In vertically integrated companies the decision is based mainly in efficiency while in deregulated markets prices play an important role, in any case the physical limitations of the system must be taken into account. This day ahead planning sets a baseline for the operation in real time but is not strictly fulfilled, as predictions are not exact as demand varies freely in most cases. In order to match generation and demand new predictions are done hours or minutes before dispatching and the base scheduling is modified. However at the moment of dispatching there will still be differences between generation and demand, and some generators with reserved power capacity, will be responsible of correcting these small differences by continuously modifying their output. This task is commonly known as frequency regulation because these imbalances are indirectly calculated by measuring the deviations of the frequency of its nominal value (50Hz or 60Hz depending on the system). If frequency exceeds its nominal value it means that there is over generation and generators receive the order of decreasing their output, and vice versa. To perform this task efficiently SOs must know in advance how frequency changes as a function of the imbalance between generation and demand, $f(\Delta P)$, for the particular setup of the grid. Using the inverse of this function the SO can calculate the needed change in generation by measuring the deviation of the frequency from its nominal value.

Frequency regulation did not become simpler through the years; especially in the last decades there were some circumstances that required new solutions. We will make reference to two of them that are directly related to our thesis. One is related to the growth of the grids due to interconnections and the market deregulation that followed it [1]. Small grids where regulated by a single generator; with the continuous growth of grids it reached a point where this task had to be shared by more than one generator. In vertically integrated power companies the decision of

which generators would regulate frequency and how it would be done depended on one actor and it could be done as optimally as possible. As vertically integrated companies were fragmented and markets were deregulated, frequency regulation and other ancillary services become a product traded in particular markets and the price become the decisive factor, not always leading to the best solution. At this point, system operators, responsible of maintaining the system and providing the service to final users, had to design methods to coordinate all actors participating in the system and controlling that they provide the promised service. For the specific case of frequency regulation most of SOs work in the same way. The day before dispatching the SO calculates the power it must reserve for frequency regulation in order to assure a proper operation for each hour. This quantity is measured in power units (MW) and it refers to an amount of power that generating units commit to reserve in order to be available when the SO asks for it. This power can be for up or down regulation depending if the unit reserves capacity to increase or decrease its output. Each generator willing to participate in the market makes an offer of up or down regulation for a given quantity at a given price and the SO chooses the smallest bids up to the needed quantity. At dispatch time the SO in a centralized way measures the system frequency and generates a regulation signal in [-1,1] that is sent to all generators providing regulation; each of them must modify its output away from its nominal working point an amount proportional to the signal received and the committed regulation. For example if a generator is producing $500MW$ and has $100W$ of committed regulation, if it receives a regulation signal of -0.5 it must modify its output to $450MW$.

The second factor that complicates balancing tasks is the increasing number of non-controllable generators in power systems. The last decades have seen an exponential growth of renewable energies connected to the grids which have their positive and negative effects. Most of us are aware of the positive side of renewable energy but not of its effects on power systems. These generators vary their output depending on external factors as wind or sun. Although these variables can be predicted they are not controllable and hence they increase the need of frequency regulation of the system. At the same time, as more and more energy comes from theses sources, there is less coming from controllable generators and this implies there are fewer generators available for providing balancing services. At this point is where our contribution comes into place.

1.2 DEMAND CONTROL PARADIGM

If we project ourselves to the near future we could imagine electric power systems with a very high percentage of the energy coming from renewable non-controllable sources. In such a scenario the classical paradigm of generation following demand is not longer feasible and we could expect opposite control logic where demand adapts to the available power. As we previously commented there are still no grid-scale storage technologies. If we could have enough storage we would not need to

control demand, we would just produce enough energy, store it and consume it when necessary. Even if this large scale storage does not exist, there is still some available storage that we could exploit to better utilize energy from non controllable sources. There is plenty of storage distributed in houses and industrial sites. In a normal house we have water heaters, fridges, heating devices, AC's, eventually electric cars and other smaller storages in portable devices. Most of these appliances allow us to decouple consumption of electric power from the use of the device; some of them only minutes as ACs, other a couple of hours like water heaters or maybe days for an electric car. Other devices do not have storage but their use can be deferred in time without affecting user experience, the most classical examples are washing machines or water pumps. The use of loads as part of the control of the grid is known as "Demand Response" (DR), and will be our main subject of investigation.

The new control paradigm for the electric grids should exploit as much as possible this intrinsic flexibility of the loads. There are numerous ways to harness this flexibility, from scheduling loads the day before their use to real-time control for frequency regulation.

1.3 AGGREGATORS AND SMART GRID

In order to make possible this change in electric grids new infrastructure and actors are needed. In regard to infrastructure a communication network will be necessary that connects individual loads, directly or indirectly, with the SO in order to coordinate generation and consumption. Individual loads will need to be modified to communicate with the network and to be able to take actions depending on specific variables to be defined. This infrastructure will enable to deploy algorithms over the whole grid and coordinate all the actors involved. This change in the infrastructure of electric grids is frequently referred to as "Smart Grid" and basically consists of the use of information and communication technologies all over the grid. Demand response is one of the potential uses of this smart grid, but it has plenty of other uses, one of the most important being the security and reliability of the system. These aspects are improving with the use of such technologies, as they enable continuous monitoring of the whole grid and make it possible to take automated actions to solve problems in distant places.

Going back to our main focus, DR, it is important to notice the scale of the problem. Electric grids connect millions of users, each of them with several different appliances, which means that a DR solution could potentially involve hundreds of millions of loads. Nowadays SOs deal with hundreds or maybe thousands of generators. These generators provide energy and ancillary services to the grid and are coordinated by the SO. If loads will start providing ancillary services it would mean going from thousands of actors to millions, and it may be impossible for a single central operator to coordinate them all. At this point we imagine that a possible solution would imply a hierarchical organization where intermediate agents between

SO and final users will be in charge of managing a cluster of loads and provide services to the grid in a simpler way. We will call these actors load aggregators or simply aggregators. Our investigation will be focused from the aggregator's point of view. Our hypothesis is that the SO will continue managing the system in the same way as today and aggregators will use loads to provide ancillary services (frequency regulation in our case), competing against generators in the corresponding markets.

In this thesis we explore solutions for using deferrable and interruptible loads for frequency regulation. Notice that in this category we can include all type of loads with storage as they can also be seen as deferrable loads, where the deferability capacity depends on the amount of storage.

1.4 PREVIOUS WORK

Before entering into our proposal we will briefly present a selection of related work in this area. Demand Response has been a popular research area in the last years; we will not focus in the infrastructure needed to implement DR, but on the candidate loads and the algorithms needed for using them in an efficient way.

There are two categories of loads that are the most popular in this research area, electric vehicles (EV's) and thermostatically controlled loads (TCL's). As we explained before there are many other candidate loads for DR, but these two groups are among the ones with highest potential. We will review some proposals using this kind of loads and we will then comment some more ideas using generic loads.

TCL's have been very popular in DR because of their intrinsic thermal storage. Most TCL's do not work in a specific temperature point but maintain temperatures using a hysteretic ON-OFF control. The typical operation of a TCL, a refrigerator for instance, that is set to maintain the temperature between $3°C$ and $5°C$ is the following. The cooling device turns on until the temperature reaches $3°C$, and then it remains off until the temperature reaches the upper limit, $5°C$, and turns on again to take it to $3°C$. We could think of this process as a battery that is full when the temperature is $3°C$ and empty when it is at $5°C$. The operation of the fridge could be modified to follow any other ON-OFF pattern as long as we keep the temperature within the range. Most TCL's work in a similar way and can be treated as batteries. The size of this storage depends on the range of permitted temperatures, thermal capacity and resistance of the storage.

In [2, 3, 4, 5] the focus is on TCLs for providing frequency regulation. In [2] they use Markov chains to derive a linear time-invariant representation of an heterogeneous population of TCLs. They propose two methods for determining the aggregated model parameters, either from the parameters of individual TCLs or by observation of the temperature dynamics of some or all of the loads of the population. They close by proposing a control method using model predictive control and ana-

lyze the impact of incomplete TCL state information has on controller performance. In [3] a collection of TCLs is characterized by an equivalent battery model and in [4] the potential of this approach for providing frequency regulation is demonstrated with practical data from California. [5] focuses on Commercial building HVAC systems to provide frequency regulation. Instead of aggregating small individual loads it exploits the potential of big HVAC systems with its different components, fans, coolers, thermal storages. In [6] a more general approach shows how generic TCL's as AC or fridges can be controlled in a decentralized way to follow a power reference.

Much research has been done in how to optimally charge EV's. We could imagine that in a near future most cars will be electric and this means millions of medium size batteries that are most of the time unused (when cars are parked). These batteries can be seen as small independent batteries and charge them independently one from another or they can be viewed as a large distributed battery that can smooth variation in generation and load. Of course coordinating all the batteries must take into account several other variables as, users requirements (ex: battery full before certain hour), distribution grid limitations, battery life cycles, mobility, etc.

[7] studies how to charge a fleet of vehicles under an aggregator control to maximize the frequency regulation capacity while maximizing the profit for the aggregator. By using dynamic programming, an optimal charging control is pursued for each vehicle and the optimality of the results is verified by simulations. In [8] it is shown how to estimate the frequency regulation capacity of a fleet of EVs while being charged. They find the optimal value for the average charge rate and the maximum allowed deviation from the average, while trying to maximize the value of the regulation service that can be offered to the grid.

More relevant to this paper is another line of work [9, 10, 11] that exploits the time deferability of generic loads, characterized by arrival times, deadlines, power and energy requirements. In particular [9] investigates different options for *scheduling* such deferrable loads, comparing classical approaches from processor scheduling (earliest deadline first, least laxity first, see e.g. [12]) with a model predictive control proposal tailored to the power setting. In [10] the authors attempt to characterize the aggregate flexibility provided by such load arrival profile, again in terms of electricity storage. [11] uses generic deferrable loads to make the aggregate total power consumption on the grid as smooth as possible with a decentralized control.

1.5 DOCUMENT OUTLINE

The thesis is organized as follows. In Chapter 2 we introduce our load aggregator model and analyze its potential and limitations. Chapter 3 presents a modification to the model that takes into account individual loads deadlines and analyzes how this change affects the potential of the aggregate of loads. In Chapter 4 the focus is

on optimizing the control of the system, an H2 optimal control is implemented. In Chapter 5 we provide extensive simulations to validate our model in diverse situations and we test different algorithms that could be used in a real implementation. We finish with conclusions and future research in Chapter 6. Appendices and bibliography follows.

Chapter 2

Load aggregator model

In this chapter we present a model for a load aggregator which will be the base for the rest of our work. We start by modeling individual loads and then infer a fluid model of an aggregate of loads. Using this model we will explore solutions for reducing regulation needs of the aggregate of loads and furthermore using the loads as frequency regulation providers.

2.1 DEFERRABLE LOADS

Electric loads can be classified between deferrable and non-deferrable. This classification is given by the ability of the load to decouple its consumption from the time when it is activated. In the previous chapter we introduced some examples of deferrable loads and some of its potential uses. This type of loads are the basis for our model.

Non-deferrable loads are all of those that need to be used in the moment that they are activated such as lights, TV, household devices, etc.

2.1.1 Individual load model

In this thesis we will work with a generic deferrable load model; although it does not fit exactly every type of load, it is a reasonable approximation for most. The first assumption we will make is that each load has a fixed rated power, that it consumes while being on. There are some loads like washing machines that have different stages during a cycle of use and consume a different amount of power in each stage; these are only a small fraction of loads so we will neglect these special cases and take their average power as their nominal power. The model for individual loads is similar to the one used in [10]. Each load j is modeled as a task parameterized

by its total service time needed τ_j, arrival time a_j, departure time d_j, and nominal power p_j. This means that at time a_j the load requests to be serviced before time d_j for a total time τ_j at rate p_j, or equivalently, the load needs to be serviced an amount of energy $Q_j = \tau_j p_j$.

Loads are also characterized by the possibility of being or not interrupted during their service, and in the case of being able to be interrupted the number of times they can be interrupted or the minimum time between interruptions. For the time being we will not consider this restriction but we will examine it later in chapter 5.

2.2 FLUID MODEL OF LOAD DEFERRALS

Let us now consider a load aggregator that manages a large set of loads from its customers. The aggregator exploits the flexibility of individual loads to reduce the cost of supplying the demands. We could think of different uses depending on the time scale; on the long term the aggregator could move consumption from times of higher prices to lower prices and in the short term the flexibility can be used to provide regulation. We will focus in the use of loads flexibility for providing and reducing the need of frequency regulation.

To begin with, this agent should have a prediction of the aggregate power demand profile for a time period (e.g. the following day), and will use the forecast to purchase this average power in the day-ahead market. Regulation comes into play to deal with real-time variations around these predicted values.

Assume first that none of the loads are deferrable: in that case the aggregator has a randomly varying load profile that deviates from the forecast and so it becomes a *consumer* of regulation services, which must be obtained through the SO. Suppose instead that a portion of the loads is deferrable in time; this flexibility can be exploited to align as much as possible the consumed power to the forecast, reducing the regulation power requirement. When load deferability is large the aggregator could eliminate the need to purchase regulation, as long as the total energy requirement has no bias. Furthermore, it could exploit the flexibility to become a *supplier* of regulation services to others in the network.

2.2.1 Aggregator model

Our model for a load aggregator takes into account the loads that are under control of this agent. Although each load has its own parameters as discussed above, the model we propose only takes into account the average parameters.

Suppose demands arrive at the aggregator at constant rate λ, which is directly associated with the power profile of the cluster of loads and can be estimated by

the aggregator. In a real scenario the assumption of a constant arrival rate is only valid for a relatively short period of time but we will not deal with this issue now. Individual loads are modeled as explained in section 2.1.1, τ being the mean service time. Instead of serving the loads at full power at their arrival the aggregator can choose to serve loads at a fraction of the power, or alternatively, serve only a fraction of the loads and defer the others. Our macroscopic model will be agnostic to these details. Let $n(t)$ denote the number of loads at disposal in the system to be serviced and $u(t) \in [0, 1]$ be the service fraction. A simple model for the evolution of the number of loads in the system is the following ODE:

$$\dot{n}(t) = \lambda - \frac{1}{\tau} n(t)u(t). \tag{2.1}$$

Here, the number of dispatchable loads grows as new demands arrive, whereas the second term accounts for service completions: $n(t)u(t)$ is the number of active loads, each completed at rate $1/\tau$, or alternatively we can think of $n(t)$ active loads being served at a fraction of their power $u(t)$ which gives a completion rate of $u(t)/\tau$.

We can also think of this model as the fluid-flow counterpart to the Markov chain in Figure 2.1. Figure 2.1 represents a $M/M/\infty$ queue with arrival rate λ and individual service times $exp(\frac{u}{\tau})$.

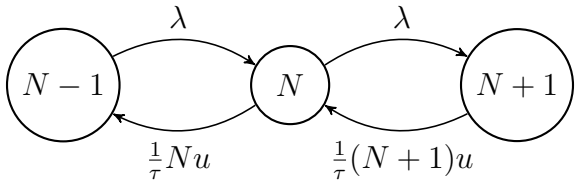


Figure 2.1. Markov state diagram with transition rates

Some remarks have to be made about this model; it is a fluid model of a discrete system, being valid only if the number of loads is large enough. It is also not trivial that the departure rate is always proportional to $u(t)$ if this value changes continuously. The error associated with changes in this parameter can be neglected if $u(t)$ stays close to its nominal working point and further if the loads' service time is distributed exponentially, because of the memory loss property of this distribution.

Equation 2.1 models the dynamics for the number of loads under the aggregator's control. What really interests us is the consumption of these loads so we should focus on this value. If u represents the fraction of each load's nominal power then the total power consumed by the cluster is

$$P(t) = u(t) \sum_{j=1}^{N(t)} p_j.$$

Alternatively if u is the fraction of loads being served, the total power would be

$$P(t) = \sum_{j=1}^{\lceil u(t)N(t) \rceil} p_{f(j)},$$

where $f(j)$ is some permutation of $N(t)$ (depending on the scheduling algorithm). In both cases if we take $E[P(t)] = p(t)$ we get:

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

assuming that the power of the loads, p_j, are independently distributed with mean p_0, and the scheduling algorithm is independent from the power of the loads.

The complete model is then:

$$\dot{n}(t) = \lambda - \frac{1}{\tau} n(t) u(t), \tag{2.2a}$$

$$p(t) = p_0 n(t) u(t). \tag{2.2b}$$

In Appendix B further justification of the model is included. Here we validate by simulations. We compare the prediction of model 2.2 with a discrete system simulation which better represents the real system.

We considered a random profile of loads arriving at the aggregator as a Poisson process, of rate $\lambda = 10$ jobs per minute, with (exponentially distributed) service time request of mean $\tau = 1800s$, and power $p_0 = 1kW$ when serviced. $u^* = 0.5$ for both scenarios.

The first simulation, Fig. 2.2 shows the case of constant $u(t) \equiv u^*$ with the system starting from zero initial conditions. The second simulation, Fig. 2.3 is for the case of varying $u(t)$, in this particular case we choose a sinusoidal signal and we simulated the system in steady state.

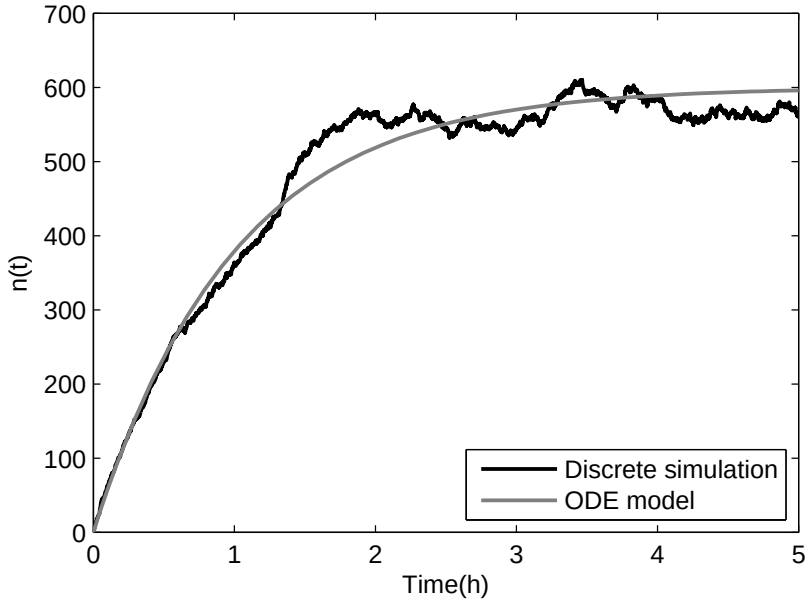


Figure 2.2. Comparison between the ODE model and a discrete loads simulation for constant $u(t)$.

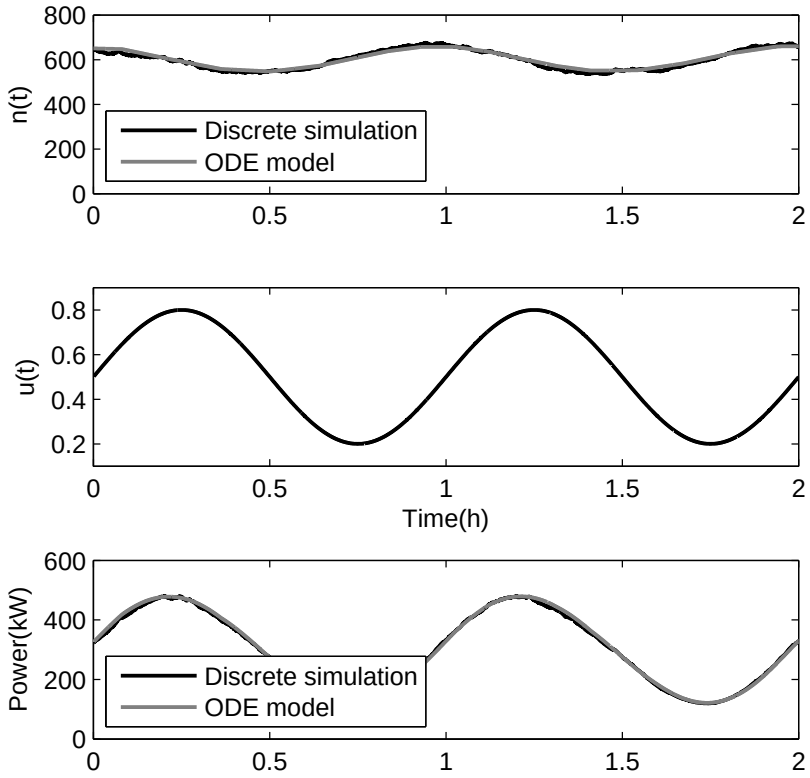


Figure 2.3. Comparison between the ODE model and a discrete loads simulation for $u(t) = u^* + 0.3\sin(\frac{2\pi}{3600s}t)$.

In this analysis we left out one of the loads' parameters, their deadline. As we mentioned in section 2.1.1 loads have and arrival and departures time, a_j and d_j respectively, which define a time window for the load to receive its service, τ_j. The difference between the available time to service a load and the time needed by the load will be called *laxity* and it is a measure of the flexibility of each load, $L_j = (d_j - a_j) - \tau_j$. We will define L as the mean laxity of the loads, which in turns defines h the mean deadline of the loads, which can be calculated as $h = E[d_j - a_j] = \tau + L$.

2.3 LINEAR ANALYSIS

To begin our analysis let us first look at the system with a fixed value of $u(t) = u^*$. Imposing equilibrium in (2.2) we obtain:

$$n^* = \frac{\lambda\tau}{u^*}, \quad p^* = p_0\lambda\tau = \lambda Q_0. \tag{2.3}$$

In equilibrium, the amount of serviceable loads in the system is increased by the deferral action $u^* \leq 1$. Note however that the average power consumed by the system is independent of u^*, and equal to the average energy per request (Q_0) times

the frequency of requests. This amount of power is the *predictable* component of the demand and can be purchased in advance for the time-period considered.

A second conclusion of (2.3) by applying Little's Law is that the average time spent by each request in the system is τ/u^*. We would like this to be below the deadline h, which imposes a first condition on the choice of u^*:

$$u^* > \frac{\tau}{h} = \frac{Q_0}{p_0 h} =: \eta. \tag{2.4}$$

Here $\eta \in [0, 1]$ is a measure of deferability of the loads (more deferability for smaller η).

We would like to analyze this system with input $u(t)$ and output $p(t)$, in order to understand which class of signals can be tracked by using the power fraction as a control input, while keeping with the deadline constraint. As we are working with a non-linear system a first approach for this analysis will be done by linearizing the system around the equilibrium point n^*, p^*, u^*. Denoting by $\delta n, \delta p$ and δu the deviation of variables from equilibrium, the linearized dynamics are:

$$\dot{\delta n} = -\frac{1}{\tau} u^* \delta n - \frac{1}{\tau} n^* \delta u, \tag{2.5a}$$

$$\delta p = p_0 \left(u^* \delta n + n^* \delta u \right). \tag{2.5b}$$

The transfer functions associated with the above system in the Laplace domain can be readily computed to yield:

$$G_{un}(s) := \frac{\hat{\delta n}}{\hat{\delta u}} = \frac{-\frac{n^*}{\tau}}{s + \frac{u^*}{\tau}},$$

$$G_{up}(s) := \frac{\hat{\delta p}}{\hat{\delta u}} = \frac{p_0 n^* s}{s + \frac{u^*}{\tau}}.$$

We can note that the transfer function from u to n is a low pass filter and from u to p is a high pass filter, both having the same pole at the frequency of completion of the loads u^*/τ. The effect of altering $u(t)$ in the power is immediate as it fixes the amount of active loads, whereas the effect on the number of loads is retarded as it implies a longer time, around τ/u^*, to complete the service of each load.

One important aspect that this model still doesn't reflect is that the arrival rate is not constant as it depends on non-controllable factors, load users. Arrivals are actually discretely distributed, λ being its average rate. Also departures happen in a discrete way. We will now discuss how to incorporate randomness in the model.

2.4 RANDOMNESS IN LOADS ARRIVALS AND DEPARTURES

To study the impact of randomness in loads arrivals and departures we use the aid of stochastic analysis tools that are out of the scope of this thesis. A summary

can be found in appendix B. Although we will not go deeply into the justification of the model for incorporating randomness, we will show later in this chapter numerical analysis to validate it.

2.4.1 The impact of uncertainty on regulated power

The presence of noise at the input of our system means that the output power will deviate from its intended value. To evaluate this impact we begin with the situation of a *fixed* deferral policy $u \equiv u^*$, which in the absence of noise would produce a constant consumption of power $p \equiv p^*$. In the presence of randomness, the output variations in power are characterized by incorporating noise into (2.5):

$$\dot{\delta n} = -\frac{1}{\tau}u^*\delta n + \underbrace{v_1 + v_2}_{v}, \tag{2.6a}$$

$$\delta p = p_0 u^* \delta n; \tag{2.6b}$$

where v_1 and v_2 are independent white noises of constant power spectral density λ, corresponding to fluctuations in arrivals and departures. So $v(t)$ is white noise of power spectral density $S_v(\omega) \equiv 2\lambda$. The transfer function from the noise input to the output δp is given by

$$G_{vp}(s) = \frac{p_0 u^*}{s + \frac{u^*}{\tau}}. \tag{2.7}$$

The noise variance at the output of this stable filter can be found (see e.g., [13]) from the corresponding $\mathcal{H}_2$ norm:

$$\begin{aligned} E\left[(\delta p)^2\right] &= \int_{-\infty}^{\infty} |G_{vp}(j\omega)|^2 S_v(\omega) \frac{d\omega}{2\pi} \\ &= \|G_{vp}(s)\|_{\mathcal{H}_2}^2 \, 2\lambda \\ &= (p_0 u^*)^2 \frac{\tau}{2u^*} 2\lambda = p^* p_0 u^*. \end{aligned} \tag{2.8}$$

To validate this conclusion in Fig. 2.4 we can see 2 hours of a 24-hour simulation with the same parameters as in section 2.2.1, and constant $u^* = 0.5$. According to Eq. 2.8 $\sigma(p) = \sqrt{p^* p_0 u^*} = 12.25kW$, which is consistent with the empirical standard deviation for this simulation which is $11.56kW$.

A first conclusion of 2.8 is that choosing $u^* < 1$ can reduce the variability of the instantaneous power consumption $p(t)$, with respect to the case of non-deferrable loads. Here we see the favorable impact of the flexibility of deferring loads in smoothing out the power profile, even if this deferral is chosen in a fixed, uncontrolled way. It appears one should work with u^* as small as possible, but of course this runs against the deadline constraint expressed in mean value by $u^* > \eta$; indeed, as $u^* \to \eta$ there will be increased probability of loads missing their deadlines.

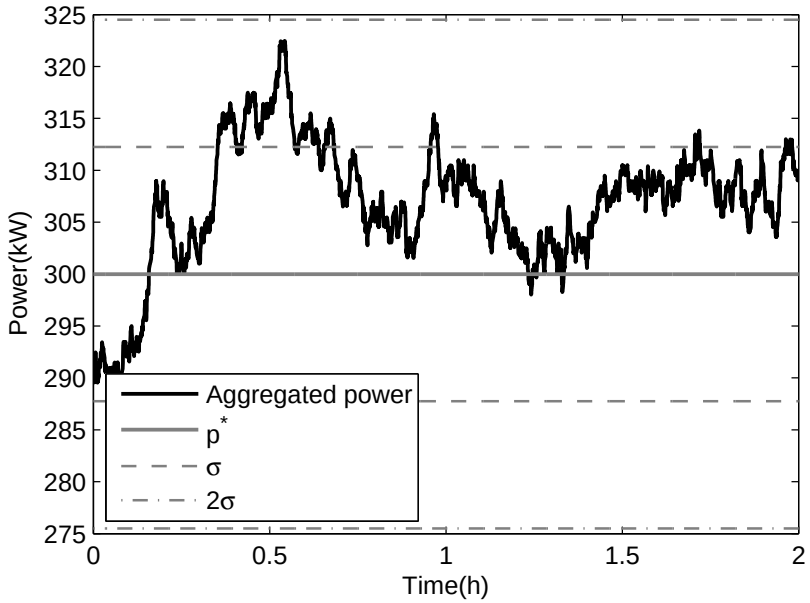


Figure 2.4. Power of the cluster of loads under fixed $u(t) \equiv 0.5$.

To further optimize the system to minimize this possibility, the scheduling of the loads must be taken into account. Proposals such as earliest deadline first (EDF) or least laxity first (LLF) [12] should be incorporated to cope with the deadlines. Here we analyze three possibilities:

- *Equal sharing:* The load aggregator chooses to serve all present jobs with power $p_0 u^*$. While not all loads will allow this policy in practice, it serves as a reference point for analysis. It corresponds to the Processor Sharing discipline of queueing theory.
- *Random:* The load aggregator chooses a fraction u^* of the available jobs at random. This policy is very easy to implement in a decentralized environment, by distributing the value of u^* and the loads choose whether to become active or not based on a local random variable.
- *Least-Laxity-First (LLF):* The load aggregator chooses a fraction u^* of the loads ordered by decreasing *laxity*, i.e. the remaining amount of time before the job needs to become active in order to meet its deadline.

We simulated the system using these scheduling policies and different values of u^*. In Fig. 2.5 we show confidence intervals (obtained through multiple runs) for the standard deviation of the measured noise power, and compare them with the theoretical value. We observe that power variability is oblivious to the exact scheduling performed, and correctly captured by the analysis. In other words, the main knob a load aggregator has to reduce variability in power consumption is reducing the fraction of serviced loads u^*.

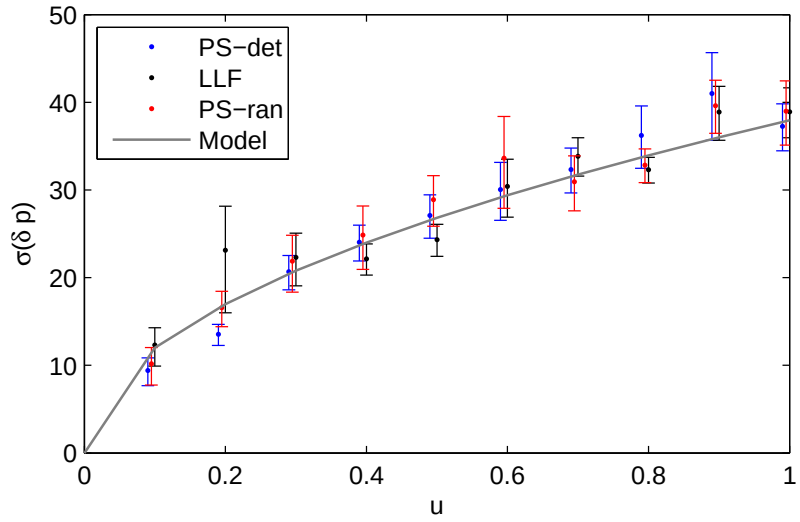


Figure 2.5. Variability of power output as a function of u^*, for different scheduling policies.

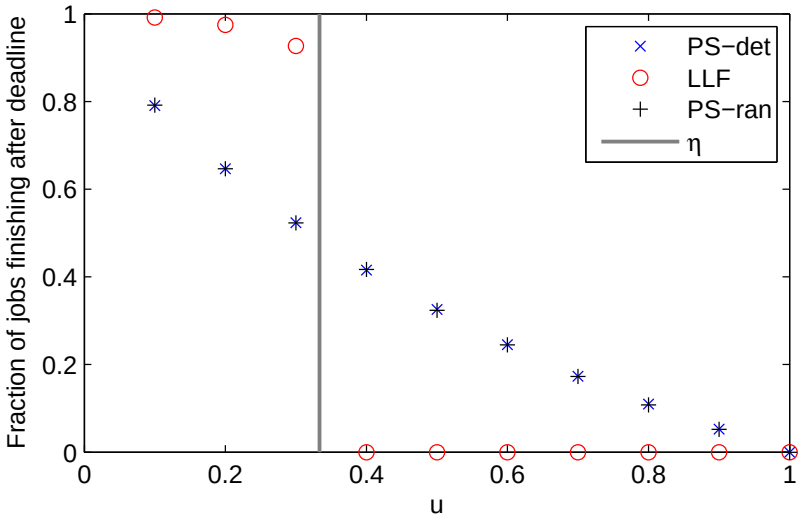


Figure 2.6. Fraction of jobs completed after the deadline with varying u^*, and different scheduling policies.

Scheduling *does* have an impact, however, in meeting the load deadline requirements. In Fig. 2.6, we plot the fraction of loads that finish with expired deadlines for the different scheduling policies. The equal sharing and random policies behave in the same way, with a smooth decrease of expired jobs as a function of u^*. In the case of LLF, which takes deadlines explicitly into account, there is a sharp decrease in expired jobs after $u^* > \eta$. This means that, provided the system can implement a suitable scheduling policy, the value of u^* can be reduced towards the minimum η, thereby reducing regulation requirements while meeting deadlines.

2.4.2 Optimizing regulation requirements through feedback

In our analysis of system noise so far we only considered a fixed, static choice of the load deferral fraction, captured by the parameter u^*. However, further improvements could be sought by controlling the variable $u(t)$ in feedback, in this case using as natural measurement the state $n(t)$.

We now analyze such a scenario. Consider again the linearized system from (2.5) restoring the input δu and adding the noise v, with output δp:

$$\dot{\delta n} = -\frac{1}{\tau}u^*\delta n - \frac{1}{\tau}n^*\delta u + v, \tag{2.9a}$$

$$\delta p = p_0\left(u^*\delta n + n^*\delta u\right); \tag{2.9b}$$

here again $v(t)$ is white noise of power spectrum 2λ.

Since we are working with stochastic noise and signal variances for performance, a natural feedback design strategy is $\mathcal{H}_2$-optimal control, seeking to minimize for instance

$$J := E[(\delta p)^2 + \beta(p^*)^2(\delta u)^2], \tag{2.10}$$

weighted sum of the regulation error variance with a penalty on control effort. The latter penalization is natural to induce the control input to stay within its saturation limits [1].

Noting that we are in a state-feedback situation, the optimal $\mathcal{H}_2$ control (see Appendix C) will have the form of a static state feedback $\delta u = -K\delta n$; in this scalar case we can work directly with the gain K, more conveniently written as

$$K = \frac{u^*}{n^*}a \tag{2.11}$$

with the parameter $0 \leqslant a < 1$.

Substituting the feedback law in the linearized state-space model (2.9), we arrive at the closed loop:

$$\dot{\delta n} = -\frac{u^*}{\tau}(1-a)\delta n + v, \tag{2.12a}$$

$$\delta p = p_0 u^*(1-a)\delta n, \tag{2.12b}$$

$$\delta u = -\frac{u^*}{n^*}a\delta n. \tag{2.12c}$$

The closed loop transfer function from noise to state is

$$G^a_{vn}(s) = \frac{1}{s + \frac{u^*}{\tau}(1-a)}, \tag{2.13}$$

[1] Other control designs that explicitly incorporate L_∞ bounds on u would be more precise, we choose this version for simplicity. β is dimensionless.

from where we compute the stationary state variance

$$E\left[(\delta n)^2\right] = \|G^a_{vn}(s)\|^2_{\mathcal{H}_2}\, 2\lambda = \frac{\tau\lambda}{u^*(1-a)} = \frac{n^*}{1-a}.$$

Expressions for the variances in (2.10) follow from (2.12):

$$E[(\delta p)^2] = (p_0 u^*(1-a))^2 \frac{n^*}{1-a} = \frac{(p^*)^2}{n^*}(1-a), \tag{2.14}$$

$$E[(\delta u)^2] = \frac{(u^*)^2 a^2}{n^*(1-a)}. \tag{2.15}$$

Therefore our cost from (2.10) becomes

$$J = \frac{(p^*)^2}{n^*}\left[(1-a) + \beta(u^*)^2 \frac{a^2}{1-a}\right].$$

The above expression clearly expresses the tradeoff between regulation and control effort as a function of the gain parameter $a \in [0, 1)$. If $a = 0$ there is no feedback control and we are back in the situation of Section 2.4.1, with the same performance. Setting $a \to 1$ would eliminate noise from the regulated power output, but make the control signal variance explode beyond its constraints. Intermediate values could potentially reduce the regulation variance while still keeping control within its admitted bound.

We now simulate the system with this fixed value of $u^* = 0.5$ against a system that continuously updates u to the deviations in δn following equation (2.12c). For this simulation we choose $a = 0.8$, which is a compromise between noise reduction and deviations in the control signal that may move the system far away from the nominal values.

In Fig. 2.7 we plot the results, showing the value of the control signal $u(t)$ (above) and the output power $p(t)$ (below). We can see that the state feedback is able to achieve an important reduction in the power variability, while the control signal $u(t)$ stays near the nominal value u^*.

As an additional remark, simulations with different scheduling policies show that, again in this case, the results are agnostic to the exact job scheduling policy.

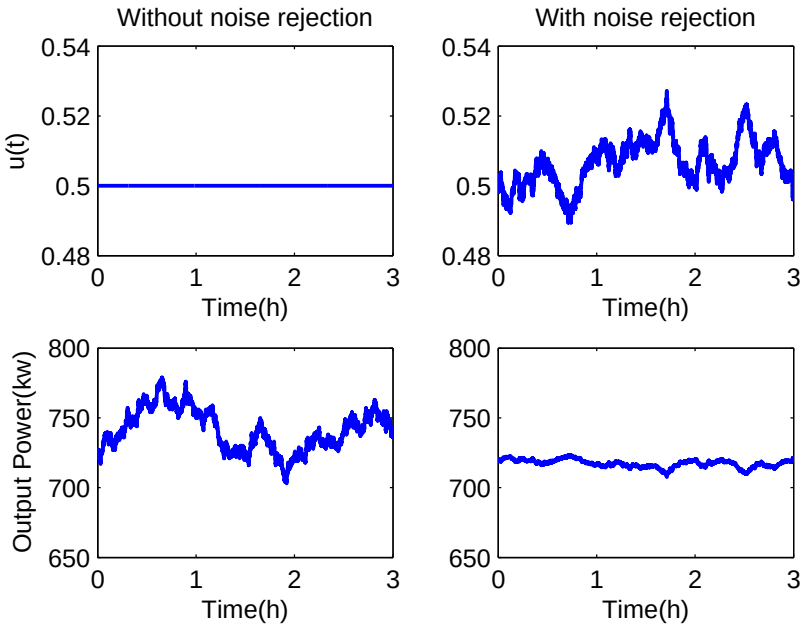


Figure 2.7. Noise rejection via state feedback.

2.5 PROVIDING REGULATION

Up to now we focused in reducing the regulation needs of a cluster of loads. Now we will replace the objective of keeping a constant power with a more ambitious one: controlling the aggregate of deferrable loads to follow a power reference signal provided exogenously by the SO. [2]

Specifically, a provider of this ancillary service must commit to varying its power consumption up to a fraction θ of its nominal power p^*, in response to a real-time signal $\rho(t) \in [-1, 1]$ that it receives every few seconds from the SO. Upon receiving this signal the load should ideally become

$$p(t) = p^*(1 + \theta\rho(t)) = p^* + \underbrace{\theta p^* \rho(t)}_{r(t)}. \tag{2.16}$$

2.5.1 Maximum offered regulation

A frequency regulator provider gets rewarded by the maximum deviation θp^* it is able to offer, thus it is to our convenience to make θ as large as possible. The maximum theoretical value is $\theta = 1$, which would imply varying the power in the range $[0, 2p^*]$. Our system could potentially offer this maximum value of regulation but in order to achieve this objective the value of u^* should be carefully chosen. The lower bound for the consumed power of the aggregate of loads is 0 because we can

[2] In control parlance, this is a *tracking* rather than a *regulation* problem. The section title uses the power systems terminology.

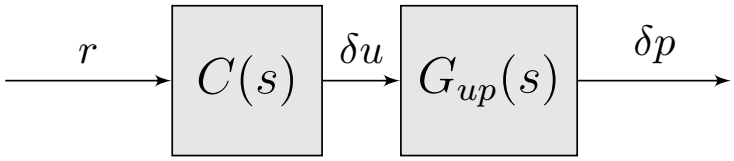


Figure 2.8. Controller design for tracking the regulation signal

always set $u(t) = 0$ momentarily. The higher bound is not fixed as it depends on the state of the system $n(t)$, being the maximum achievable power $p(t) = n(t)p_0$, by setting $u(t) = 1$. We can get an estimate of this bound by applying the equilibrium values:

$$p(t) \leq p_0 n^* = \frac{p^*}{u^*}.$$

This means that in order to offer a value of $\theta = 1$ we should choose $u^* < 0.5$.

2.5.2 Tracking a reference signal

A first proposal to achieve the desired tracking is through a feedforward controller, as depicted in Fig. 2.8. Observe that by choosing $C(s)$ as:

$$C(s) = \frac{1}{G_{up}(s)} = \frac{s + \frac{u^*}{\tau}}{p_0 n^* s} = \frac{1}{p_0 n^*}\left(1 + \frac{u^*}{\tau s}\right), \tag{2.17}$$

the linearized system should be able to match deviations in the regulation signal r.

Noting that $p_0 n^* = p_0 \lambda \tau / u^* = p^*/u^*$, the above proportional-integral law can be expressed in the time domain as follows:

$$\delta u(t) = \frac{u^*}{p^*}\left[r(t) + \frac{u^*}{\tau}\int_0^t r(w)\, dw\right].$$

Now replacing with (2.16) and noting $\tau = \eta h$ leads to

$$\delta u(t) = u^* \theta \left[\rho(t) + \frac{u^*}{\eta h}\int_0^t \rho(w)\, dw\right]. \tag{2.18}$$

Of course, the control input is subject to the saturation constraint $u(t) \in [0, 1]$. Consequently, the above expression indirectly constrains the class of regulation signals $\rho(t)$ that our system can track; in particular $\rho(t)$ must have mean-value zero, otherwise the integral term will necessarily lead to saturation; in fact $\rho(t)$ should not have a persistent sign for too long, in relation to the mean deadline h.

To illustrate the behavior of the proposed controller, we simulated the system driven by a real-life regulation signal $\rho(t)$ taken from [14]. We considered a random profile of loads arriving at the aggregator as a Poisson process, of rate $\lambda = 4$ jobs per

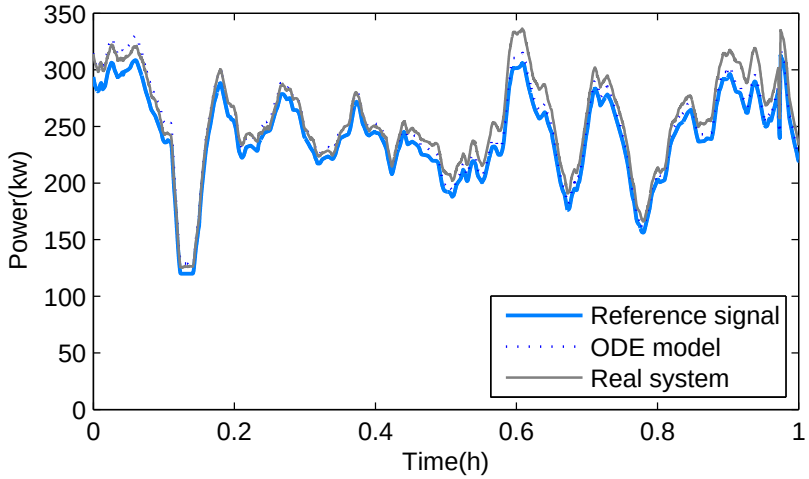


Figure 2.9. Tracking of a power regulation signal via service deferrals.

minute, with (exponentially distributed) service time of mean $\tau = 1800$s, and power $p_0 = 2kW$ when serviced. The fraction of serviced loads $u(t)$ is driven by (2.18), the output of the linear controller around a fixed equilibrium value of $u^* = 0.5$ and includes the effect of the saturation constraint $u(t) \in [0, 1]$. Of the possible scheduling policies based on $u(t)$ (described in more detail in Section 2.4.1) we chose here the *equal sharing* algorithm; however this choice has minimal impact.

Simulation results are shown in Fig. 2.9, corresponding to $\theta = 0.5$; we see that the aggregator output closely matches the regulation request. Thus the aggregator can offer 50% regulation around the average power $p^* = 240$ kW.

We do notice, however, some tracking errors which are attributed to the randomness in the system, for this particular setting the RMS value of the error is $16.37kW$. Depending on their entity, such errors may result in practice in penalties for not following the correct profile [15]. An effort should be made to try to minimize tracking errors.

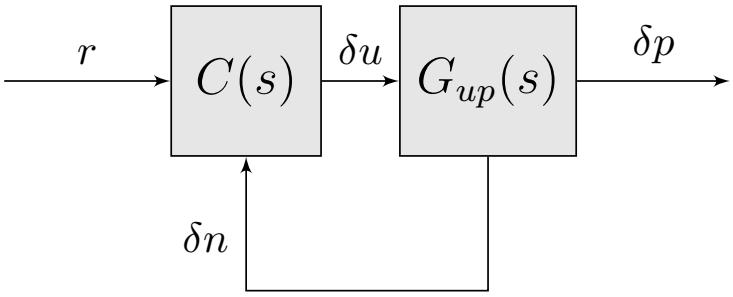


Figure 2.10. Controller design for tracking the regulation signal

We pursue this issue using again the linearized model around the nominal operating point. The open loop model coincides with (2.9), but now we must consider

in addition an external reference $r(t)$ which the power output must track, so the performance specification will involve the *tracking error* $e(t) := \delta p(t) - r(t)$. The controller, see Fig. 2.10, should have access to measurements of the state variable $\delta n(t)$, and also to the external reference $r(t)$, producing an action $\delta u(t)$ on the plant (2.9) so as to minimize the error variance, while keeping a check on control effort. This could be framed as a joint $\mathcal{H}_2$-optimal control design with cost

$$J' := E[e^2 + \beta(p^*)^2(\delta u)^2] \tag{2.19}$$

which generalizes (2.10). A complete design of this kind would require a characterization of the class of reference signals to be tracked, for instance through a frequency weighting function. At this point we will opt for the simpler strategy of combining the feedback and feedforward components from the earlier sections, using a controller of the form

$$\delta u(t) = -K\delta n + \tilde{u}(t),$$

where K has the form (2.11) and $\tilde{u}$ is a function of the reference input. Substitution into (2.9) gives:

$$\dot{\delta n} = -\frac{1}{\tau}u^*(1-a)\delta n - \frac{1}{\tau}n^*\tilde{u} + v, \tag{2.20a}$$

$$\delta p = p_0[u^*(1-a)\delta n + n^*\tilde{u}],. \tag{2.20b}$$

which leads in the Laplace transform domain to

$$\delta p(s) = G^a_{\tilde{u}p}(s)\tilde{u}(s) + G^a_{vp}(s)v(s),$$

with

$$G^a_{\tilde{u}p}(s) = \frac{p_0 n^* s}{s + \frac{u^*}{\tau}(1-a)}, \quad G^a_{vp}(s) = \frac{p_0 u^*(1-a)}{s + \frac{u^*}{\tau}(1-a)}.$$

This suggests choosing the feedforward component $\tilde{u}$ as

$$\tilde{u}(s) = \frac{1}{G^a_{\tilde{u}p}(s)}\delta r(s) = \frac{1}{p_0 n^*}\left(1 + \frac{u^*(1-a)}{\tau s}\right)\delta r(s),$$

which results in the following closed loop transfer function from noise to tracking error:

$$e(s) = \delta p(s) - \delta r(s) = G^a_{vp}(s)v(s); \tag{2.21}$$

therefore the noise penalty on performance will be exactly the one computed in (2.14). The control effort will also have the noise term of (2.15), but in addition there is the impact of the reference signal analogous to (2.18).

To end this chapter we show a simulation of the complete system. We use the same signal and parameters of Fig. 2.9 adding the feedback for noise reduction. Again we choose $a = 0.8$. We can see in Fig. 2.11 that the tracking is clearly improved, the RMS value of the error decreased to $11.09kW$.

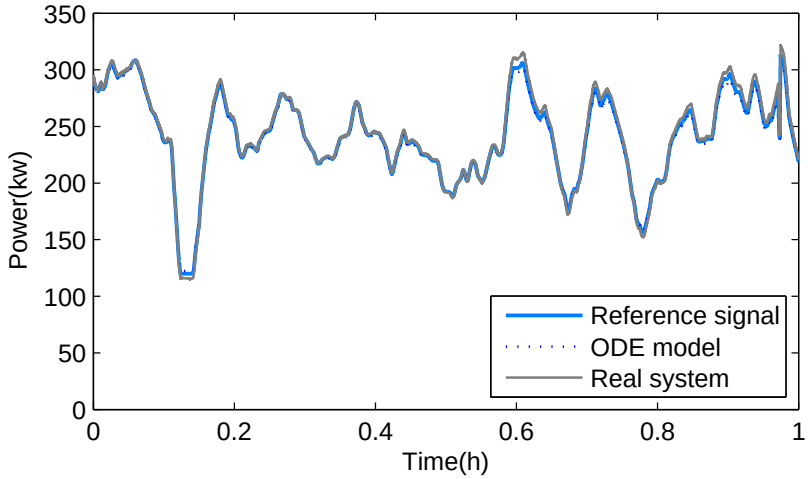


Figure 2.11. Tracking a reference signal for the system with noise rejection.

2.6 SUMMARY

In this chapter, we analyzed a model for a load demand aggregator that manages a large number of consumer deferrable loads and is capable of adjusting the number of active jobs to control their aggregated power. The proposed macroscopic model is oblivious to the exact management of the loads and captures the essential behavior of the system through the service fraction the aggregator provides. Using this model, we were able to analyze the impact of variability in the demands, and design tracking and noise rejection controllers. These simple mechanisms enable a load aggregator to reduce its need for regulation services, and even offer regulation services to others. The results were evaluated through fine-grained simulation, illustrating the performance of the designed mechanisms.

Several problems still remain unsolved. One of them is guaranteeing each individual load deadline. We will study this problem more in detail in the next section.

Chapter 3

Coping with deadlines

In the previous chapter we presented a macroscopic fluid model of load aggregation. One of the results that we found from analyzing this model is that in order to cope with individual load deadlines, specific scheduling policies must be used which require detailed information of every load state. This type of scheduling goes against our objective of keeping the load aggregation model and scheduling algorithms as simple as possible. In this chapter we will present an alternative solution by modifying the model to take into account the deadline issue.

3.1 SEPARATING THE CRITICAL POPULATION

Loads enter the system with a needed service time and a dealine before which they must be served. Loads not being served consume their laxity, and may reach a point when meeting the deadline requires turning on the load immediately. After an interval of time dt, a load served at power up_0 will have reduced its required service time by udt, but also will have consumed $dt - udt = (1 - u)dt$ of its spare time. Figure 3.1 shows the trajectory in (service-time/laxity) space when u is constant.

A trajectory reaching the vertical axis completes service and leaves the system. If, instead, the horizontal axis is reached first, laxity expires and to keep its deadline the load can no longer be deferred. We will denote by $n(t)$ the population of loads that at time t still have remaining laxity, and thus are served at level u; the remainder of loads $m(t)$ with expired laxity must be turned on immediately and served at full power in order to meet their deadline. We propose this course of action, which can be implemented in a decentralized fashion (e.g. a thermal load which decides to start consuming power since the temperature has become too low). Setting $L = h - \tau$ the mean laxity, a dynamic model for this new system is:

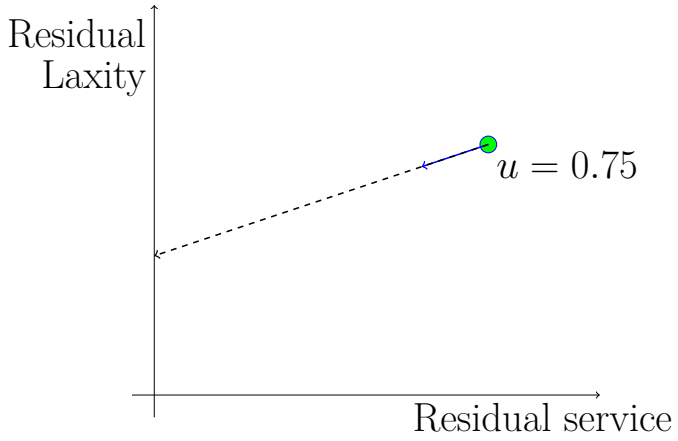


Figure 3.1. Service-laxity trajectory under service level u

$$\dot{n}(t) = \lambda - \frac{1}{\tau}n(t)u(t) - \frac{1}{L}n(t)(1-u(t)), \tag{3.1a}$$

$$\dot{m}(t) = \frac{1}{L}n(t)(1-u(t)) - \frac{1}{\tau}m(t), \tag{3.1b}$$

$$p(t) = p_0(n(t)u(t) + m(t)). \tag{3.1c}$$

Now $u(t)$ represents the fraction of loads with positive laxity that are being served. Loads can exit the first queue in two ways: a fraction of the loads, represented by the term $\frac{1}{\tau}n(t)u(t)$ get completely served before their deadline; the rest, represented by $\frac{1}{L}n(t)(1-u(t))$, are automatically turned on when they run out of laxity and move from n to m. The departure rate from the second queue is represented by $\frac{1}{\tau}m(t)$, as they are always on.

Model 3.1 is consistent with the original model 2.2 under the scheduling policy described above; all the loads with expired laxity are turned on and served at full power. Model 2.2 resembles model 3.1 when $L \to \infty$. A more detailed stochastic analysis of this model is provided in Appendix B.3.

To analyze this new model we will proceed as in the previous section by first studying the system in equilibrium and linearizing to analyze the power variability and design suitable controllers.

Fixing $u(t) = u^*$ and imposing equilibrium in (3.1) gives the following values for n and m:

$$n^* = \frac{\lambda}{\frac{u^*}{\tau} + \frac{(1-u^*)}{L}}, \quad m^* = \frac{\lambda\tau\frac{(1-u^*)}{L}}{\frac{u^*}{\tau} + \frac{(1-u^*)}{L}}. \tag{3.2}$$

Analyzing now the relationship between the number of loads and the service level is a bit trickier. First of all it's important to remark that u^* does not reflect the

percentage of active loads as in (2.2). If we denote as u_r the fraction of active loads, then

$$u_r^* = \frac{u^* n^* + m^*}{n^* + m^*}.$$

When $u^* = 1$ then also $u_r^* = 1$ because all the loads are on and $m^* = 0$ as loads are serviced immediately, $n^* = \lambda\tau$. If on the other hand $u = 0$ then

$$u_r^* = \frac{m^*}{n^* + m^*} = \frac{\tau}{\tau + L} = \eta,$$

which turns out to be the minimum percentage of active loads admitted by this system and is consistent with (2.4), as we do not allow any load to be serviced after its deadline. With regard to the number of loads, n^* is strictly increasing/decreasing with u^* depending if $\tau < L$ or $\tau > L$ respectively. m^* is always decreasing with u^* as it would be expected because as more loads are served early, fewer get their laxity expired. The number of active loads is $u^* n^* + m^* = \lambda\tau$ which does not depend on u^* and is proportional to the mean power $p^* = p_0\lambda\tau$ as in our 1-state model, and the total number of loads can be inferred from (3.1) $n^* + m^* = \frac{u^* n^* + m^*}{u_r^*} = \frac{\lambda\tau}{u_r^*}$.

We end this section by validating the model through simulations as in section 2.2.1, with the same parameters. The value of u^* is not the same as we want $u_r^* = 0.5$ to have the same effective service level. We can calculate u^* from eq. 3.1 which gives $u^* = 1/3$, for this particular setting. Figure 3.2 shows the evolution of the number of loads for both models under the same process of arrival and using the equal sharing algorithm. The simulation is consistent with the new model and the mean number of loads in the system is the same for both models, 600 for this setting.

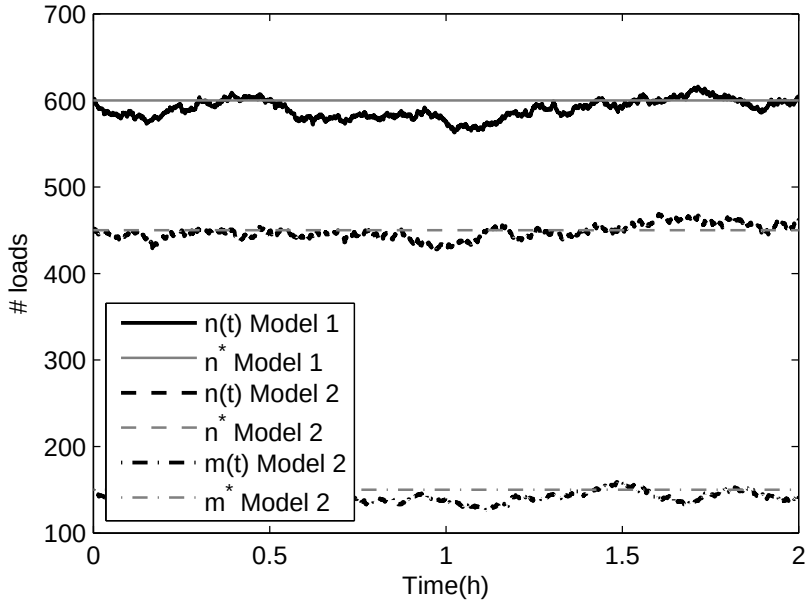


Figure 3.2. Number of loads for both models under the same arrival process.

If we look at the service level in Fig. 3.3 we can see that although we are working with a fixed u the effective service level is not constant. This variability comes from the imposition of serving at full power all loads with no laxity, which in turn depends on load service time and laxity. In the next section we will see that this restriction that guarantees that loads are served on time introduces more variability in the output power.

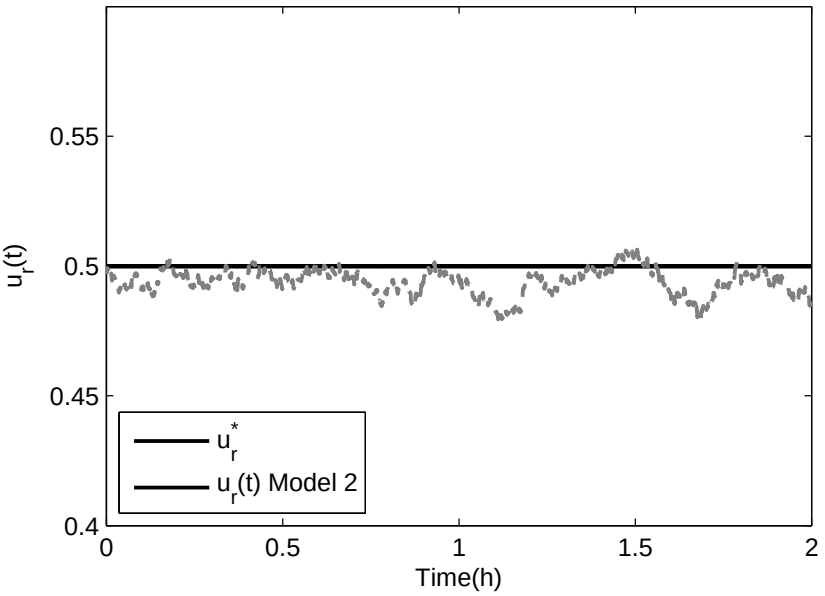


Figure 3.3. Comparison between u_r^* and $u_r(t)$ when imposing deadlines.

The last figure, Fig. 3.4 compares the consumed power for both models. Although both simulations are done with the exact same loads the difference in the scheduling strategies makes the consumed power to by slightly different. We can still see form the figure that they are qualitatively very similar.

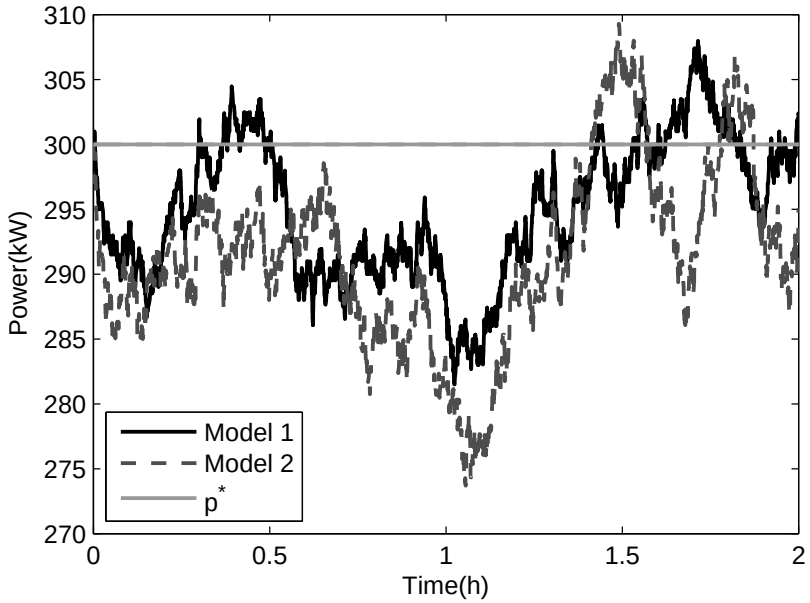


Figure 3.4. Comparison between the power consumed for both models, with and without enforcing deadlines.

3.2 MODELING RANDOMNESS IN LOAD ARRIVALS AND DEPARTURES

Analyzing consumed power variability in this model is somewhat more complicated. We will again linearize the model and add the noise sources with the same assumptions as in (2.6). The linear model is the following:

$$\dot{\delta n} = -\left[\frac{u^*}{\tau} + \frac{1-u^*}{L}\right]\delta n + \left[\frac{n^*}{L} - \frac{n^*}{\tau}\right]\delta u + v_1 - v_2 - v_3, \tag{3.3a}$$

$$\dot{\delta m} = \frac{1-u^*}{L}\delta n - \frac{1}{\tau}\delta m - \frac{n^*}{L}\delta u + v_3 - v_4, \tag{3.3b}$$

$$\delta p = p_0(u^*\delta n + \delta m) + p_0 n^* \delta u. \tag{3.3c}$$

Now v_1 stands for the arrival noise in n, v_2 and v_3 for the two departure noises. v_3 is also the arrival noise for m, and v_4 the departure noise for m.

To calculate the variability in the consumed power we will resort to the state space representation of the system:

$$\overbrace{\begin{bmatrix}\dot{\delta n}\\ \dot{\delta m}\end{bmatrix}}^{\dot{x}} = \overbrace{\begin{bmatrix}-\frac{u^*}{\tau} - \frac{1-u^*}{L} & 0\\ \frac{1-u^*}{L} & -\frac{1}{\tau}\end{bmatrix}}^{A}\overbrace{\begin{bmatrix}\delta n\\ \delta m\end{bmatrix}}^{x} + B_1\overbrace{\begin{bmatrix}w_1\\ w_2\\ w_3\\ w_4\end{bmatrix}}^{w} + \overbrace{\begin{bmatrix}-\frac{n^*}{\tau} + \frac{n^*}{L}\\ -\frac{n^*}{L}\end{bmatrix}}^{B_2}\delta u; \tag{3.4a}$$

$$\delta p = \overbrace{\begin{bmatrix}p_0 u^* & p_0\end{bmatrix}}^{C}\begin{bmatrix}\delta n\\ \delta m\end{bmatrix} + \overbrace{p_0 n^*}^{D_2}\delta u, \tag{3.4b}$$

where

$$B_1 = \begin{bmatrix} \sqrt{\lambda} & -\sqrt{\alpha\lambda} & -\sqrt{(1-\alpha)\lambda} & 0 \\ 0 & 0 & \sqrt{(1-\alpha)\lambda} & -\sqrt{(1-\alpha)\lambda} \end{bmatrix}.$$

and α is the probability that a job is completeky served before its laxity expires (see Appendix B.4 for further explanation). To calculate α explicitly we compute the probability that the job is served (at level u^*) before the laxity expires:

$$\alpha = P\left[\frac{\tau_k}{u^*} \leq \frac{L_k}{1-u^*}\right] = \frac{\frac{u^*}{\tau}}{\frac{u^*}{\tau} + \frac{1-u^*}{L}},$$

where we invoked the exponential distribution of τ_k, L_k with respective means τ, L.

We have now a stable state-space system driven by vector valued white noise. In steady-state, the covariance matrix Q of the state is (see e.g. [16]) the solution to the Lyapunov equation

$$AQ + QA^T + BB^T = 0, \tag{3.5}$$

and the resulting variance of the output p is $E\left[(\delta p)^2\right] = CQC^T$. The value of Q can be directly calculated from the linear equation 3.5, resulting in:

$$Q = \frac{\lambda}{\frac{u^*}{\tau} + \frac{1-u^*}{L}} \begin{bmatrix} 1 & 0 \\ 0 & \frac{\tau}{L}(1-u^*) \end{bmatrix}.$$

Which allow us to calculate the variance of consumed power:

$$E\left[(\delta p)^2\right] = p^* p_0 \left[1 - \frac{1}{\frac{1}{1-u^*} + \frac{\tau}{Lu^*}}\right]. \tag{3.6}$$

This model behaves in a different way to the one in Chapter 2: only the deferrable portion of the load population is within the scope of the service level u^*, hence this parameter can take any value in $[0, 1]$, without violating job deadlines. In fact we find from (3.6) that both extremes 0 or 1 have the same effect on the output variance, which becomes $E\left[(\delta p)^2\right] = p^* p_0$. This makes sense because the only difference between both cases is that for $u^* = 1$ loads are turned on when they arrive, whereas for $u^* = 0$ they do after their laxity expires, but in any case the time in service is the same, hence the steady-state output variability is the same.

Choosing an intermediate value of u^* allows us to lower the variance but with a lower bound, which is the price to pay for keeping all deadlines. The optimal value of u^* that minimizes the variance can be calculated to be:

$$u^*_{opt} = \frac{\sqrt{\tau}}{\sqrt{L} + \sqrt{\tau}}. \tag{3.7}$$

We show in Fig. 3.5 the power variability of the loads predicted by the model for the case $\eta = 1/3$, and simulation results for the equal sharing algorithm.

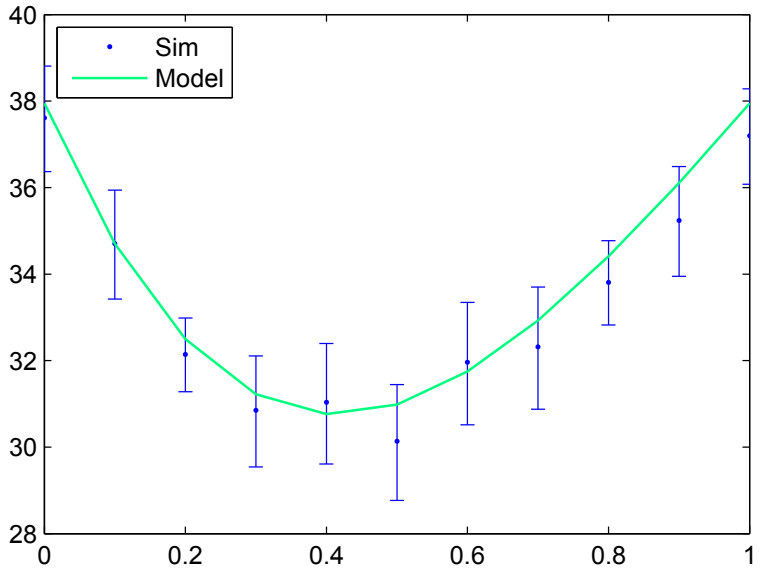


Figure 3.5. Variability of power output as a function of u^* for model (3.1).

The main conclusion of this analysis is that, with a simpler mechanism that does not resort to scheduling, we can nevertheless reduce the regulation requirements. By comparing the results of Figs. 2.5 and 3.5, we can see that although we cannot reduce the power variability as much as in the one state model with $u^* = \eta$, by carefully choosing the service level u^* of the second system, we can still achieve a significant reduction in the power variability, but now with deadlines automatically attained.

3.3 PROVIDING REGULATION BY ADAPTING DEFERABILITY

As with the previous model our objective is still to reduce regulation needs and offer frequency regulation to the system. Here we will proceed using similar techniques to the ones already applied in Chapter 2 and in Chapter 4 we will focus more in detail in the controller design.

3.3.1 Maximum offered regulation

As in section 2.5.1 we want to find the maximum θ achievable by out system. The maximum theoretical value is $\theta = 1$, which would imply varying the power in the range $[0, 2p^*]$.

In our system of deferrable loads this value is not achievable, because consumed

power must lie within the bounds

$$p_0 m(t) \leq p(t) \leq p_0[n(t) + m(t)];$$

in particular the lower bound is always positive since we have chosen not to defer the loads m with expired laxity. Also the upper bound is constrained by the loads $n(t)$ present at any given time. Both bounds are intrinsically time-varying, but we can get an estimate of the achievable margin by applying the equilibrium values:

$$p_0 m^* \leq p(t) \leq p_0[n^* + m^*].$$

Writing the lower bound as $p^*(1-\theta)$ and recalling $p^* = p_0(n^* u + m^*)$ we arrive at the bound

$$\theta \leq \frac{n^* u^*}{n^* u^* + m^*} = \frac{L u^*}{L u^* + \tau(1-u^*)}.$$

Similarly, the upper bound in power gives

$$\theta \leq \frac{n^*(1-u^*)}{n^* u^* + m^*} = \frac{L(1-u^*)}{L u^* + \tau(1-u^*)}.$$

The above bounds are, respectively, increasing and decreasing in u^*, and they become equal in $u^* = \frac{1}{2}$; therefore this choice is the value that provides the maximum (symmetric) regulation capability, namely $\theta_{\max} = \frac{L}{L+\tau}$. We will use this choice of u^* in what follows, despite the fact that it need not coincide with the value from (3.7) providing minimal open-loop power variability.

3.3.2 Controller design

To finish this chapter we present a first simple approach to a controller for providing regulation. We will resemble the controller used in Chapter 2, adapting it to the new model.

The transfer function of the linearized version of the plant (3.1) is:

$$G_{up}(s) := \frac{\hat{\delta p}}{\hat{\delta u}} = \frac{p_0 n^* s}{s + \frac{u^*}{\tau} + \frac{1-u^*}{L}}. \tag{3.8}$$

As a first solution to our tracking requirement we used the inverse of this plant as a feedforward controller. Again we tested the tracking capability of the loads with this controller simulating the response of the system to a real life regulation signal taken from the PJM Interconnection [14]. We compare the reference signal against the prediction of model (3.1) and a discrete system simulation which better represents the real system. In the latter loads arrive at random times and we schedule them using the *equal sharing* algorithm according to the signal u, until they are served or they run out of laxity and are turned on automatically. The setting for

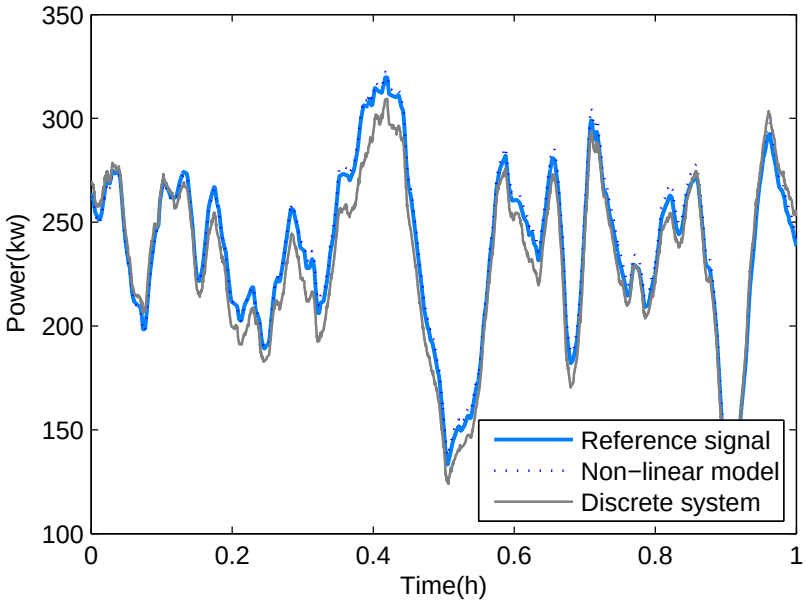


Figure 3.6. Tracking a real life reference signal.

the simulation are the same as the ones used in chapter 2. The results of this simulation in Fig. 3.6 show that the system is able to set the consumed power very close to the reference, the RMS value of the error is $19.20kW$.

Still, there are some differences between the reference and the output, which may not be tolerable if we want to be regulation providers. To improve tracking we consider adding feedback to the design, to compensate for the deviations introduced by randomness in arrivals and departures. In Fig. 3.7 we depict such a controller, using state feedback of the variables δn and δm.

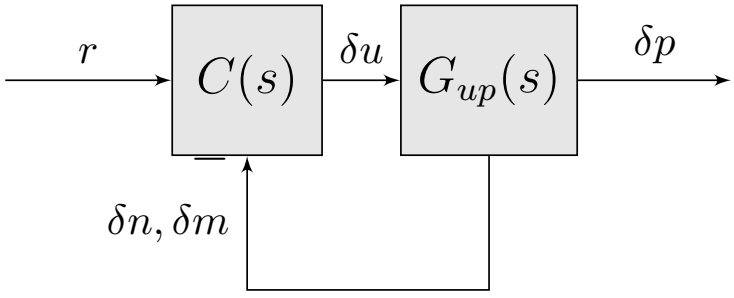


Figure 3.7. Controller design for tracking the regulation signal

Now the controller is the sum of two terms, one that tracks $r(t)$, plus a noise reducing term with inputs $n(t)$ and $m(t)$. The final form of the controller is, in Laplace transform notation:

$$\delta u = \frac{s + \left(\frac{u^*}{\tau} + \frac{1-u^*}{L}\right)(1-a)}{p_0 n^* s} r - \frac{u^*}{n^*} a \left(\delta n + \frac{\delta m}{u^*}\right). \tag{3.9}$$

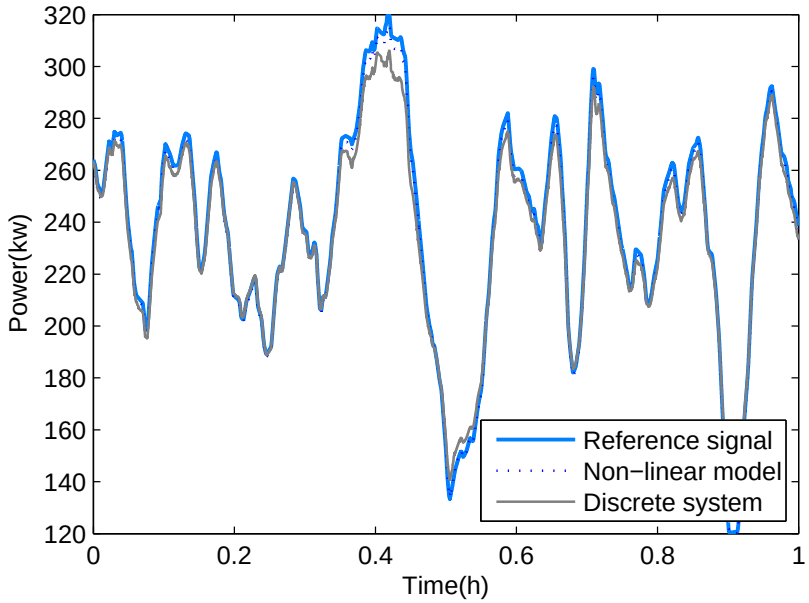


Figure 3.8. Tracking a reference signal for the system with noise rejection.

The parameter a fixes the feedback term for noise reduction, being 0 for no feedback. Setting $a = 1$ would make the system internally unstable, so we choose a strictly less than 1. In our simulation experiments we used $a = 0.7$.

The last simulation of this chapter, shown in Fig. 3.8, illustrates how the system is capable of tracking the same signal we used before. We see there is a notorious improvement in tracking after we add the noise reducing feedback, the RMS value of the error decreased to $11.77kW$. Still we can see that is slightly worse than in the case we do not take into account deadlines, which is reasonable as we do not control all loads.

3.4 SUMMARY

In this chapter we focused on the problem of coping with individual loads deadlines. We tackle this issue by proposing that each load will turn on automatically when it runs out of laxity in order to meet its deadline. This lead to a new model which is consistent with the one presented in chapter 2 with the difference of the distributed control to prevent loads from expiring. Noise disturbance and simple control strategies where analyzed for this solution. With this new model we propose studying more in depth better control strategies which we will present in the next chapter.

Chapter 4

Optimal $\mathcal{H}_2$ control

In the previous chapter we proposed and analyzed a second model for a load aggregator with the main difference from the one in chapter 2 that it took into account loads deadlines. We will now explore more in detail how to control this system in order to accomplish our two objectives: minimize the need for regulation and provide frequency regulation.

4.1 CONTROLLABILITY

A first observation in regard to control is that our state-space system is not fully controllable from the input δu. We can see this through the change of coordinates

$$\tilde{x} = \begin{bmatrix} u^* & 1 \\ \frac{1}{L} & -\frac{1}{\tau} + \frac{1}{L} \end{bmatrix} \begin{bmatrix} \delta n \\ \delta m \end{bmatrix}, \tag{4.1}$$

which leads to the state-space system

$$\begin{aligned} \dot{\tilde{x}} &= \tilde{A}\tilde{x} + \tilde{B}_1 w + \tilde{B}_2 \delta u, \\ \delta p &= \tilde{C}\tilde{x} + D_2 \delta u, \end{aligned}$$

where

$$\left[\begin{array}{c|c} \tilde{A} & \tilde{B}_2 \\ \hline \tilde{C} & D_2 \end{array}\right] = \left[\begin{array}{cc|c} -\frac{u^*}{\tau} - \frac{1-u^*}{L} & 1-u^* & -\lambda \\ 0 & -\frac{1}{\tau} & 0 \\ \hline p_0 & 0 & p_0 n^* \end{array}\right]. \tag{4.2}$$

In this realization we find that the second state variable $\frac{1}{L}\delta n + (\frac{1}{L} - \frac{1}{\tau})\delta m$ is uncontrollable[1]. As a consequence, the transfer function from δu to δp will be of first order. Since the uncontrollable state is also stable, one is tempted to ignore it for

[1] In fact, it can be verified that even in the nonlinear dynamics (3.1) the same linear combination of the states is uncontrollable.

feedback design. Note however that the uncontrollable state is excited by noise, and observable, so its effect must be considered when designing a control strategy to reduce power variance.

4.2 $\mathcal{H}_2$-CONTROL FOR REDUCING POWER VARIANCE

We will start by studying the problem of reducing the power variance which means needing less regulation service. We'll pose this problem as optimal $\mathcal{H}_2$-control regulation problem. This consists of a state-feedback controller that minimizes a compromise between output variance and control effort, expressed by the weighted objective

$$J := E[(k_1\delta p)^2 + (\delta u)^2], \tag{4.3}$$

Note the importance of penalizing the variations δu to avoid non-linear effects and most importantly saturation, since we recall that $u(t)$ is confined to the interval $[0, 1]$.

It could be argued that an $\mathcal{H}_\infty$-control would better adjust to this type of problem. This could be true for the problem of reducing regulation needs because frequency regulation providers are paid for their committed capacity more than for the energy they use for regulation. On the other hand if we look at the criterion used by SO, PJM for instance, to evaluate regulation providers (see apendix A) an $\mathcal{H}_2$-control may seem appropriate. We chose to use $\mathcal{H}_2$-control as it better adapts to the tracking problem and its construction is simpler.

Setting up the problem in the standard form for $\mathcal{H}_2$-control (see apendix C), we have a generalized plant

$$G(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ I & 0 & 0 \end{array}\right], \tag{4.4}$$

where A, B_1, B_2 are the same as in (3.4a), and we introduce

$$C_1 = \frac{1}{k_2}\begin{bmatrix} k_1 C \\ 0 \end{bmatrix} = \frac{1}{k_2}\begin{bmatrix} k_1 p_0 u^* & k_1 p_0 \\ 0 & 0 \end{bmatrix},$$
$$D_{12} = \frac{1}{k_2}\begin{bmatrix} k_1 D \\ 1 \end{bmatrix} = \frac{1}{k_2}\begin{bmatrix} k_1 p_0 n^* \\ 1 \end{bmatrix}.$$

The penalized output corresponds to the cost in (2.10), except for the constant $k_2 = (1 + (k_1 p_0 n^*)^2)^{\frac{1}{2}}$ which is included to satisfy the normalization $D_{12}^* D_{12} = 1$, and simplifies the expressions to follow.

We assume that the state (n, m) is available for feedback, i.e. the aggregator keeps track of the number of loads in each of the categories. Under these conditions

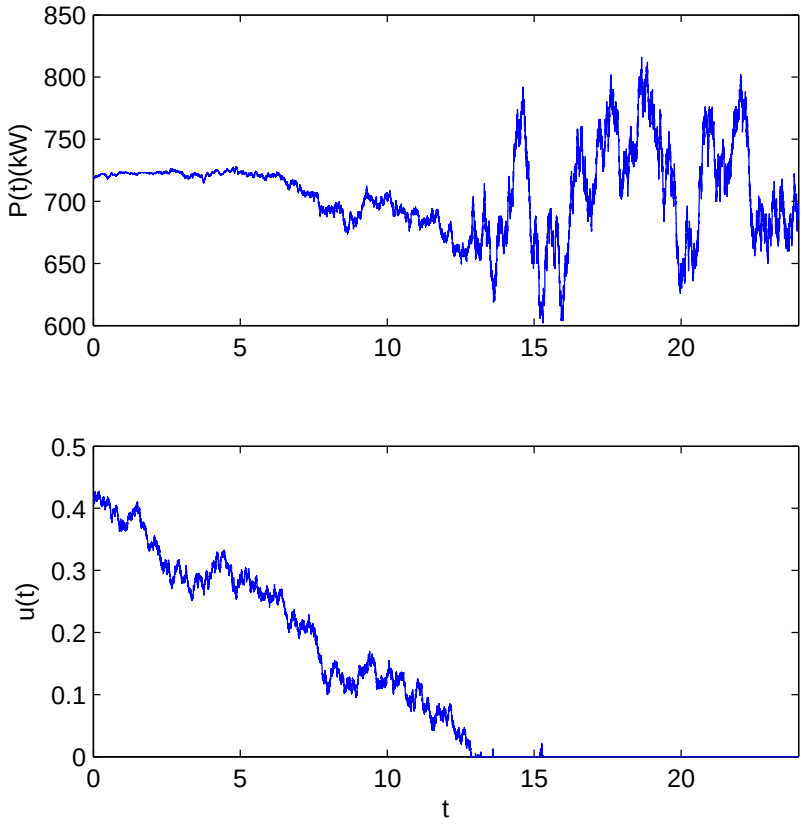


Figure 4.1. One day simulation for the system with $k_1 = \frac{10}{p_0 n^*}$.

it is shown in [13] that the $\mathcal{H}_2$-optimal control law is the static state feedback

$$\delta u = -Fx = -(B_2^* X + D_{12}^* C_1) \begin{bmatrix} \delta n \\ \delta m \end{bmatrix} \tag{4.5}$$

where X is the stabilizing solution to the Algebraic Riccati Equation

$$(A^* - C_1^* D_{12} B_2^*)X + X(A^* - B_2 D_{12}^* C_1) - X B_2 B_2^* X + C_1^*(I - D_{12} D_{12}^*) C_1 = 0. \tag{4.6}$$

Obtaining a parametric solution of this equation would be cumbersome so we choose to analyze it numerically. The parameters we will use are $\lambda = 0.2\, loads/s$, $\tau = 1800s$, $L = 3600s$, $p_0 = 2kW$. For u^* we choose the optimal value from (3.7) in the uncontrolled case, which yields $u^* = 0.41$. It follows that $n^* = 511$, $m^* = 151$ and $p^* = 720kW$.

To evaluate the correct choice of k_1 we note the following. Since our real objective is reducing power variance, we would like to increase this weight as much as possible in relation to control effort, the only limitation being that if $\delta u(t)$ becomes too large nonlinear effects come into play, in particular saturation. For instance, in Fig. 4.1 we show the time trajectory $u(t)$ for the case $k_1 = \frac{10}{p_0 n^*}$, which exhibits saturation,[2]

[2]Indeed, there is actuator "windup" and the system never leaves saturation.

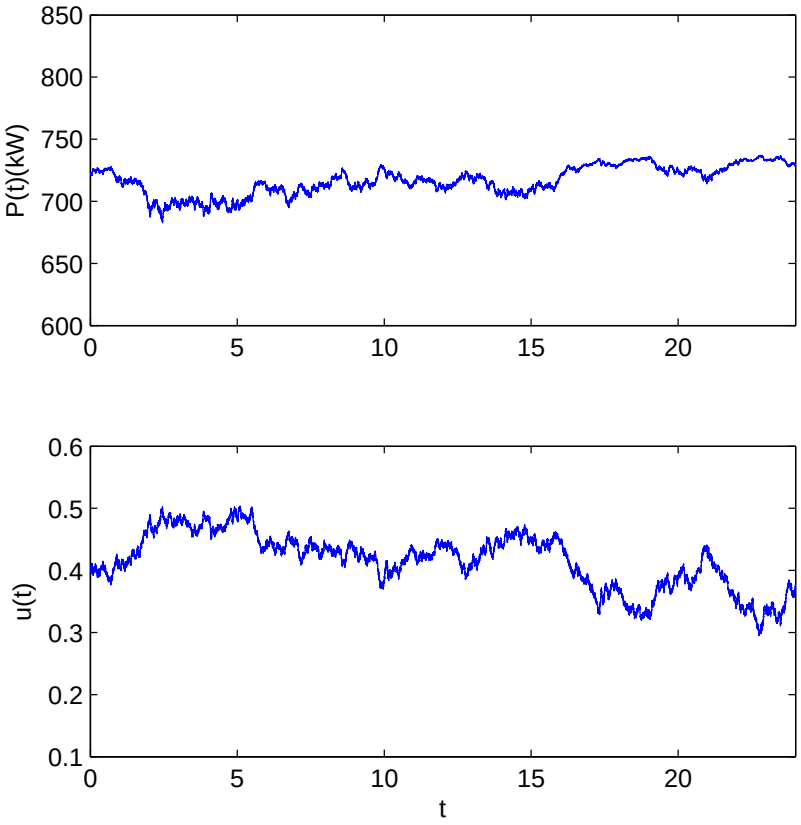


Figure 4.2. One day simulation for the system with $k_1 = \frac{5}{p_0 n^*}$.

and consequently a deterioration of performance. We thus carried out a linear search for k_1, simulating the nonlinear dynamics and computing the RMS value of δp and δu; results are shown in Fig. 4.3. From this evaluation, a good choice for the weight appears to be $k_1 = \frac{5}{p_0 n^*}$, which makes the optimal feedback matrix

$$F = [\frac{0.735u^*}{n^*} \frac{0.838}{n^*}]$$

and gives $\overline{\delta p} \approx 9.89kW$, $\overline{\delta u} \approx 0.0471$ and $\overline{J} \approx 0.0046$. If we look at these values in relative terms compared to the uncontrolled case we find that the variance of the consumed power decreased 67.8% from $30.76kW$ to $9.89kW$. And in Fig. 4.2 we see that the resulting trajectory for $u(t)$ contains moderate variations around its nominal value.

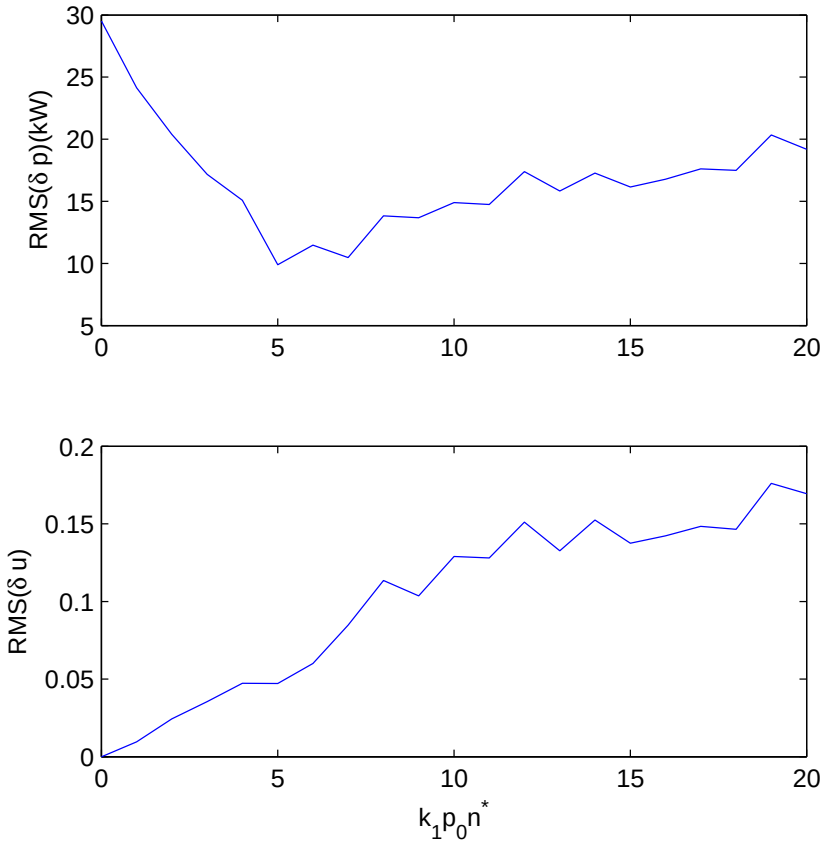


Figure 4.3. Consumed power variability and control effort for $\mathcal{H}_2$-optimal controlled system for different values of objective function.

4.3 PROVIDING FREQUENCY REGULATION

We will now apply the same theory for our final objective of controlling the aggregate of deferrable loads to follow a power reference signal provided exogenously by the system operator (SO). As we have already mentioned in section 2.5 the consumed power of the cluster of loads should be

$$p(t) = p^*(1 + \theta\rho(t)) = p^* + \underbrace{\theta p^* \rho(t)}_{r(t)}, \tag{4.7}$$

where θ is the amount of regulation offered and we calculated its maximum value in section 3.3.1, and ρ is the dimensionless signal sent by the SO and takes values in the range [0,1].

The problem we are facing now is, in control parlance, a *tracking* rather than a *regulation* problem. The input of the system will be $r(t)$ the reference we want to track and the output is the error in tracking, $r(t) - \delta p(t)$, which we want to minimize. An $\mathcal{H}_2$-optimal control seeks to minimize the $\mathcal{H}_2$-norm of the desired

transfer function, in order for this solution to be optimal the power spectral density of the input should be constant (see apendix C). This is not the case for regulation signals which are band-limited and their spectral density depends on the electric system characteristics and the control implemented by the SO. One way to tackle this issue is to model regulation signals as filtered white noises, which have constant power spectral density, and modify our system by incorporating the filter and taking the noise as the input. At the moment of implementing the system we ignore this filter and use directly the regulation signal.

4.3.1 Regulation signal characterization

For this purpose we turn again to the particular family of real-life regulation signals $\rho(t)$ taken from the PJM interconnection [14]. We performed a spectral density estimation based on these PJM signals using MATLAB's signal identification toolbox. A first observation is that they have band limited energy, with cutoff frequency $\omega_r \approx 1.65 \times 10^{-2}\, rad/s$, after which they present a roll-off of $40\, db/dec$, indicating a second-order filtering. A closer inspection shows a resonance in the cutoff frequency with a damping factor of $\zeta \approx 0.4$. We therefore approximated the practical signals as generated by white noise through the frequency weighting filter

$$W_\rho(s) = \frac{\kappa_r \omega_r^2}{s^2 + s2\zeta\omega_r + \omega_r^2}, \tag{4.8}$$

where $\kappa_r \approx 3$ was chosen to match the mean signal power.

In Fig. 4.4 we can see a 2 hour simulation of filtered white noise along with a real regulation signal, with a qualitatively similar behavior. The artificial signal stays in $[-1, 1]$, which of course need not always happen with a linear model; in fact the real signal in Fig. 4.4 features such a saturation around $T = 1.4h$. If such events are infrequent they can be ignored for the purpose of control design.

4.3.2 $\mathcal{H}_2$-optimal control

We now set-up our $\mathcal{H}_2$-optimal control problem through the generalized plant model in Fig. 4.5. Here, the input weight is $W_r := \theta p^* W_\rho$ with W_ρ in (4.8), consistent with $r(t)$ in (4.7), and is driven by a white noise signal $w_r(t)$, independent of the previously considered noise signals for the loads. The tracking error signal is:

$$e(t) = r(t) - \delta p(t) = p^*(1 + \theta\rho(t)) - p(t); \tag{4.9}$$

the penalized variables for $\mathcal{H}_2$ control correspond to the cost function

$$J_2 := E[(k_1 e)^2 + (\delta u)^2]. \tag{4.10}$$

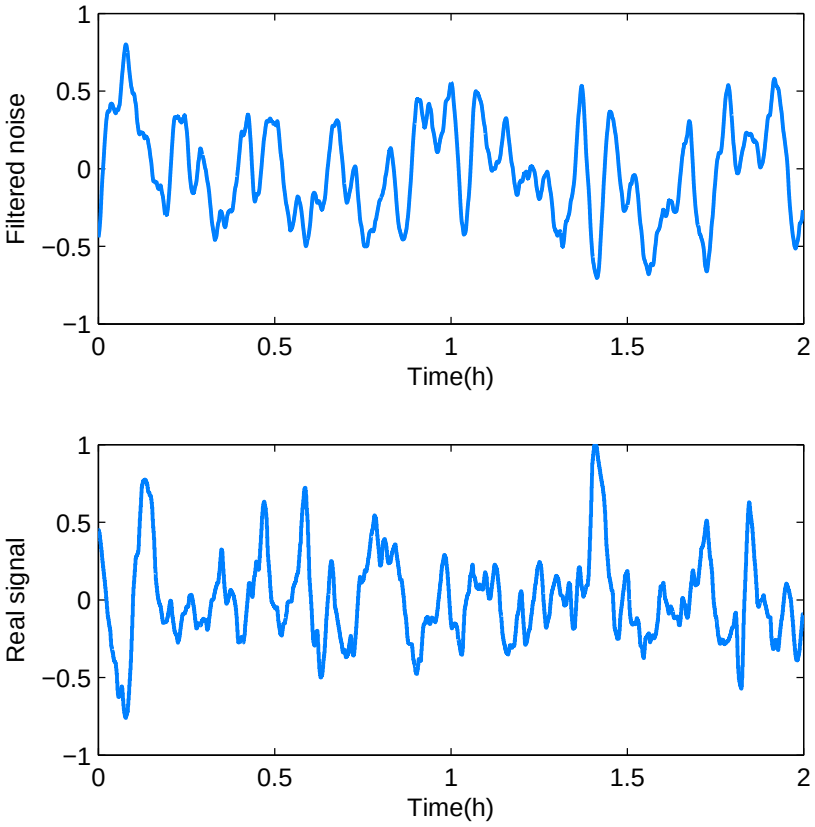


Figure 4.4. Artificial regulation signal from filtered white noise, in comparison with a real regulation signal from PJM.

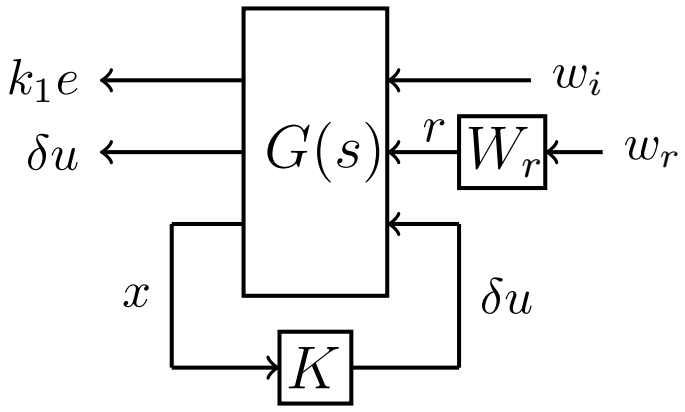


Figure 4.5. Controller design for tracking the regulation signal

We now augment our state-space realization (4.4) to incorporate the reference signal weight W_r, as follows:

$$G^r(s) = \left[\begin{array}{c|cc} A^r & B_1^r & B_2^r \\ \hline C_1^r & 0 & D_{12}^r \\ I & 0 & 0 \end{array}\right]. \tag{4.11}$$

The state vector is now $x = [\delta n,\, \delta m,\, r, \dot{r}]^T$, the last two variables corresponding to the frequency weight, and the augmented matrices are:

$$A^r = \left[\begin{array}{c|c} A & 0 \\ \hline 0 & A_{22}^r \end{array}\right],\ B_1^r = \left[\begin{array}{c|c} B_1 & 0 \\ \hline 0 & B_{12}^r \end{array}\right],\ B_2^r = \left[\begin{array}{c} B_2 \\ \hline 0 \end{array}\right],$$
$$\text{with}\ \ A_{22}^r = \begin{bmatrix} 0 & 1 \\ -\omega_r^2 & -2\zeta\omega_r \end{bmatrix},\quad B_{12}^r = \begin{bmatrix} 0 \\ 3\omega_r^2 \end{bmatrix};$$
$$C_1^r = \frac{k_1}{k_2}\begin{bmatrix} -p_0 u^* & -p_0 & \theta p^* & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, D_{12}^r = \frac{1}{k_2}\begin{bmatrix} -k_1 p_0 n^* \\ 1 \end{bmatrix}.$$

We are assuming again that the state is available for feedback. In addition to the load quantities (n, m), the aggregator must have the regulation signal r, which is received from the SO, and also its derivative $\dot{r}$. While the latter is typically not directly available, we note the following: in practical systems, $r(t)$ is communicated very frequently, e.g. every $T_s = 4$ seconds, i.e. a sampling rate of $0.25Hz$. On the other hand the bandwidth of r is much lower, in the previously fit model we have $f_r \approx 2.6.10^{-3}Hz$. This implies that a simple estimate $\dot{r}(t) \approx \frac{r(t)-r(t-T_s)}{T_s}$ has high accuracy for control purposes, so it is not justified to employ a more sophisticated method (e.g. the corresponding Kalman filter) solely for tracking this state variable.

The resulting $\mathcal{H}_2$-optimal control law is

$$\delta u = -F^r x = -(B_2^{r*} X^r + D_{12}^{r\,*} C_1^r) \begin{bmatrix} \delta n \\ \delta m \\ r \\ \dot{r} \end{bmatrix}, \tag{4.12}$$

where X^r is the solution to the corresponding Algebraic Riccati Equation.

As in the first case the solution depends on the parameter k_1 which sets the relative weight between the tracking error and the control effort. Once again, we can fix this parameter with the aid of simulations to find the value that minimizes the error and keeps $u(t)$ from saturating. The same load parameters as in the previous section were used, for an offered regulation of $\theta = \theta_{max} = 0.66$.

We will also consider for the choice of k_1 a performance score used by PJM to rank regulation resources [17]. This score is calculated comparing the reference signal with the actual response from the system and is the average of three components: correlation, delay and precision; all of them measured in a scale from 0 to 1. We note in this regard that a value of 0.75 is required for participation in the market, and that values above 0.9 are considered excellent (more details in appendix A).

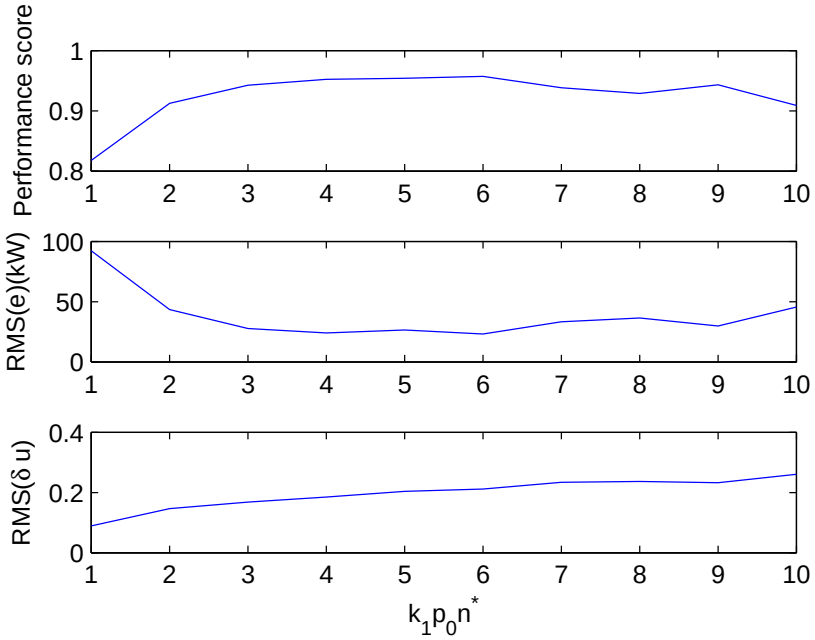


Figure 4.6. Performance of $\mathcal{H}_2$-optimal controlled system for different values of objective function .

In Fig. 4.6 we plot the RMS values of e and δu, along with the PJM regulation performance score, as a function of k_1. We see that up to a certain point the error goes down and the score improves, but towards the higher values performance degrades due to saturation of the signal u, which makes our model invalid. The optimum value for this particular case its around $k_1 = \frac{6}{p_0 n^*}$.

In figure 4.7 we show a simulation of the system tracking the refernce signal. The setting for the simulation is the same as the one used in chapters 2 and 3. In this case the RMS value of the error is $10.18kw$, which is better than in the 2 previous scenarios.

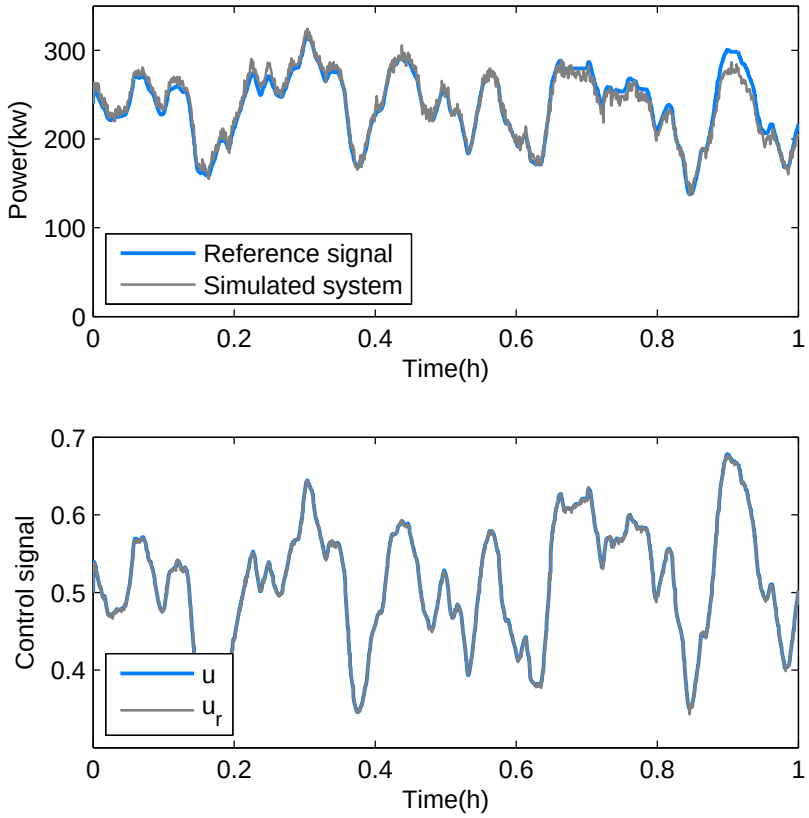


Figure 4.7. Tracking a real life signal under $\mathcal{H}_2$-optimal control.

4.4 SUMMARY

In this Chapter we focused in optimizing the controller for the load aggregator model we proposed in Chapter 3. Using optimal $\mathcal{H}_2$-control tools we added the control signal information to the controller and we were able to tune the importance of the control effort in relation to the output error. The results showed an improvement in relation to the simple controller used in the previous chapters.

This chapter closes the first part of this thesis, the design af a complete controller to provide frequency regulation to electric systems using deferrable loads. In the next chapter we will present a more empirical investigation on how to adapt this controllers to different scenarios.

Chapter 5

Implementations and simulations

We arrive now at the final part of this investigation, exploring how to adapt the controllers we design in the previous chapters to different scenarios. This chapter has a more empirical approach than the previous ones and the objective is to explore the application range of our proposal. For this purpose we developed a MATLAB based simulator (see apendix D) in order to study the effects of the different parameters of the system and explore different scheduling algorithms. Along the chapter we will get into more specific details of the implementation to show the real utility of our design. We will assume that a secure communication infrastructure exists in order for the aggregator to communicate with the individual loads and to receive the regulation signal from the SO. Anyway we will focus in developing algorithms that are the least intensive in communications as possible and as decentralized as possible. We will also emphasize keeping the control logic as simple as possible.

Most of the simulations will be done with a constant arrival rate which means that the average power of the system will be the same the whole simulation. This could be a realistic situation for short time periods, up to a couple of hours, although not for a whole day. At the end of this chapter we will show some more complete simulations using real power profiles with their corresponding arrival rates.

5.1 DATA USED FOR SIMULATIONS

In order to do a good validation of our design we should test it in a environment as close as possible to the real world. This has some difficulties as there is part of the data we need that is not easily available. This data includes statistics on individual loads deferability, measurement of individual loads activation time, correlation between service time and laxity, etc. Other datasets are more easily available such as real regulation signals or hourly energy consumption [14], and were used in the simulations.

As for the data we did not obtain we took the strategy of testing in many varied possible scenarios so as to validate the robustness of the techniques. These data includes service times, laxity, nominal power and arrival times. We will clarify in each case how these parameters were generated.

Our main objective are individual loads as household devices, water heaters, AC's, cars batteries, etc. These loads are in the range of few kW in power and service time up to a few hours. The distribution of these parameters within the cluster of loads may vary depending the specific system or the time of the day. We will explicit in each case which distribution we are using.

Laxity is a more complicated parameter as it is not as objective as power or service time. As this is one of the most important parameters and there is not much information on how it is distributed, we will make special emphasis on it and test several different scenarios.

Another parameter of the loads that we did not consider up to now is the number of interruptions loads can suffer during a service interval. Most TCL's or batteries can be interrupted continuously without affecting the quality of the service, but there are another loads that cannot be freely interrupted. In any case interrupting service always has at least a negative effect on the life-span of the loads. We will try to take into account this issue at the time of designing the scheduling algorithms in order to minimize the number of interruptions.

5.2 SCHEDULING ALGORITHMS

We will start by analyzing possible scheduling algorithms. In all cases we will use the same parameters and metrics in order to do a comparison as unbiased as possible.

The performance criteria will be the ones we have been using: tracking error, control effort and PJM score, and we will add the mean number of interruptions of the loads.

As for the settings for the simulations we will use a Poisson arrival process of intensity $1\,load/s$, the service time will be distributed exponentially with mean $1800s$, the laxity will also be exponentially distributed, independently of the service time, with mean $3600s$ and the nominal power of the loads will follow a uniform distribution in the range $[1kw, 3kw]$ independent of all other parameters. We will use this particular setup for testing the algorithms although later we will use different setups for testing the robustness of the control system. The justification for these particular parameters is quite simple. The arrival rate depends mainly on the size of the aggregator and the amount of customers it has. We chose $1\,load/s$ which means an average of around 3600 active loads (depends on the choice of u^*), which

is not large for a real system but enough for a proof of concept. If the number of loads controlled by the aggregator in real life is larger the system will probably work better because of the law of large numbers as most decisions are taken in a decentralized manner by the loads. The service time and laxity were chosen to be exponentially distributed as it represents the worst case for this parameters to be distributed; being the exponential distribution the one with largest differential entropy among all continuous probability distributions. Anyway we will test against other distributions. The nominal power was chosen in the range of the typical appliances named before.

One last key parameter is k_1 which sets the feedback matrix of the $\mathcal{H}_2$-optimal controller. For the particular settings of these first simulations the optimum value is the one calculated in section 4.3.2, $k_1 = \frac{6}{p_0 n^*}$. As we change the setting for other simulations we should recalculate the optimum value of k_1 in order to always work with the best possible controller.

5.2.1 Algorithm 1: Broadcasting u

This first proposal is the simplest and it could be said that it follows naturally from the model. The aggregator receives the regulation signal from the SO, calculates the control signal u and broadcasts it to all loads. When loads receive the signal they chose to turn on or off, each of them independently from the others, with probability u.

In order for this algorithm to work the aggregator must keep track of n and m, this task can be accomplished with very little communication. Loads must send at the most 3 messages during their life span, when they arrive, when they run out of laxity and when they complete their service. With this information the aggregator can keep track of n and m.

This algorithm has one big drawback because it does not take into account the number of interruptions of the loads that can be extremely high if loads are deciding every few seconds to turn on or off.

In Fig. 5.1 and 5.2 we can see 1 hour of two 24-hour simulations using this algorithm. Fig. 5.1 is with a constant power reference and in Fig. 5.2 the loads are following a regulation signal. In the frequency regulation case the offered regulation is $\theta = 0.66$, the maximum calculated in section 3.3.1. As we could expect the system performed well but the number of interruptions is too high. In the constant power case the RMS value of δp was $27kw$ which represents a 0.75% of the nominal power and a improvement of 56% compared to the best performance with constant u. The RMS value of δu is 0.019. In the case in which we offer frequency regulation we got a score of 0.976 using the PJM metric, $54kW$ the RMS value of the error which represents a 1.5% of the nominal power and a control effort of 0.18. As for the interruptions we find that the average was 150 in the first case and 130 in the

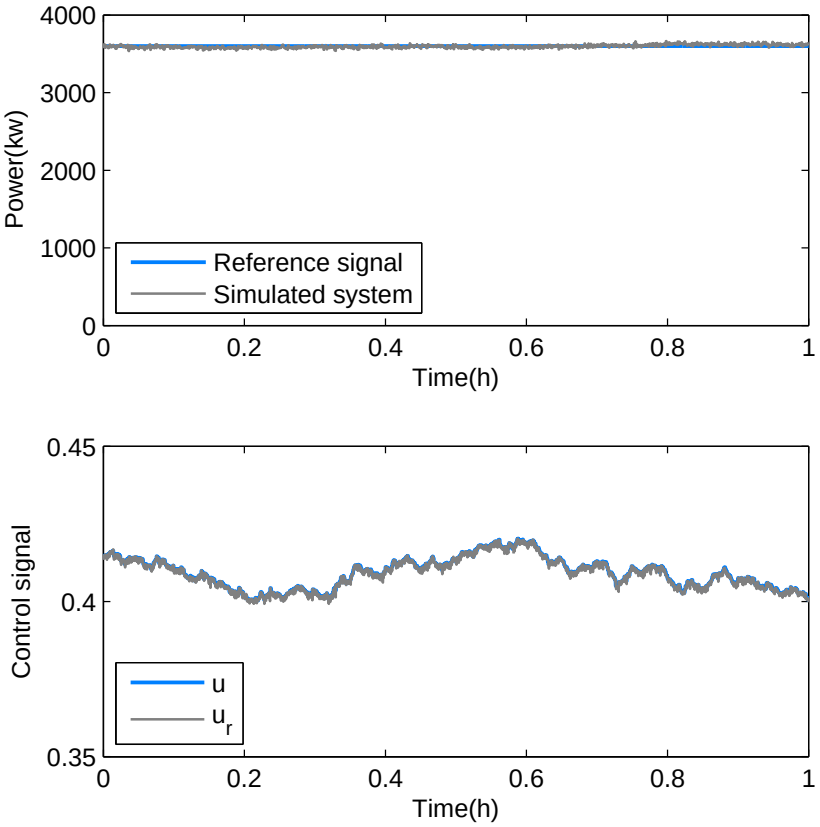


Figure 5.1. 1-hour simulation of deferrable loads tracking a constant power reference using algorithm 1.

second, both far from acceptable.

5.2.2 Algorithm 2: Zero-interruptions

This second proposal is an attempt to deal with the interruptions problem. We will move to an extreme opposite where loads cannot be interrupted at all and adapt our control system to this scenery. The basic idea is that loads receive only one order from the aggregator once they arrive to the system. The aggregator may chose to turn them on immediately as they arrive or to postpone their service till they run out of laxity, which will happen automatically.

In order for the aggregator to decide whether to turn a load on or not when it arrives, the aggregator must keep track not only of n and m, but also the active loads in n, n_a. With this information it computes the real fraction $u_a = \frac{n_a}{n}$ of active loads in n. When a new load arrives it notifies the aggregator, and the latter sends the order to start now or to delay the service depending on the value of u_a and u. If $u_a > u$ the service will be delayed and u_a decreases, in the case $u_a < u$ the load

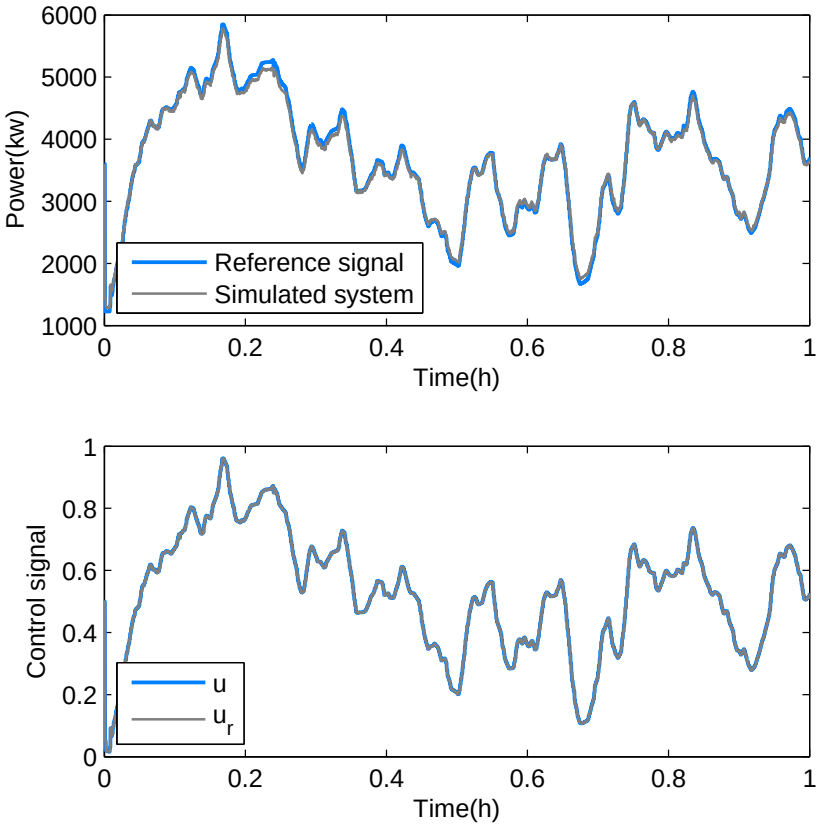


Figure 5.2. 1-hour simulation of deferrable loads tracking a regulation signal using algorithm 1.

receives the order to start the service immediately and u_a increases. In the case loads are delayed they will be served automatically when they run out of laxity as before. The loads only need to notify the aggregator when they arrive, run out of laxity and get served in order for the aggregator to maintain the state variables and u_a.

In the case of keeping a constant power reference this algorithm performs quite well because the changes needed in u are slow and its enough with the arriving loads as we can see in Fig. 5.3. The RMS value of δp was $39kw$ slightly larger than with algorithm 1, but with the advantage of not interrupting loads.

The results are not as good when we try to provide regulation using this algorithm. In Fig. 5.4 we show 1 hour of a day-long simulation of the aggregated power tracking the reference signal along with the control variable u and the percentage of active loads u_a. The amount of regulation offered $\theta = 0.66$ was the same as in the previous algorithm. We see that in this case our system is unable to track the reference.

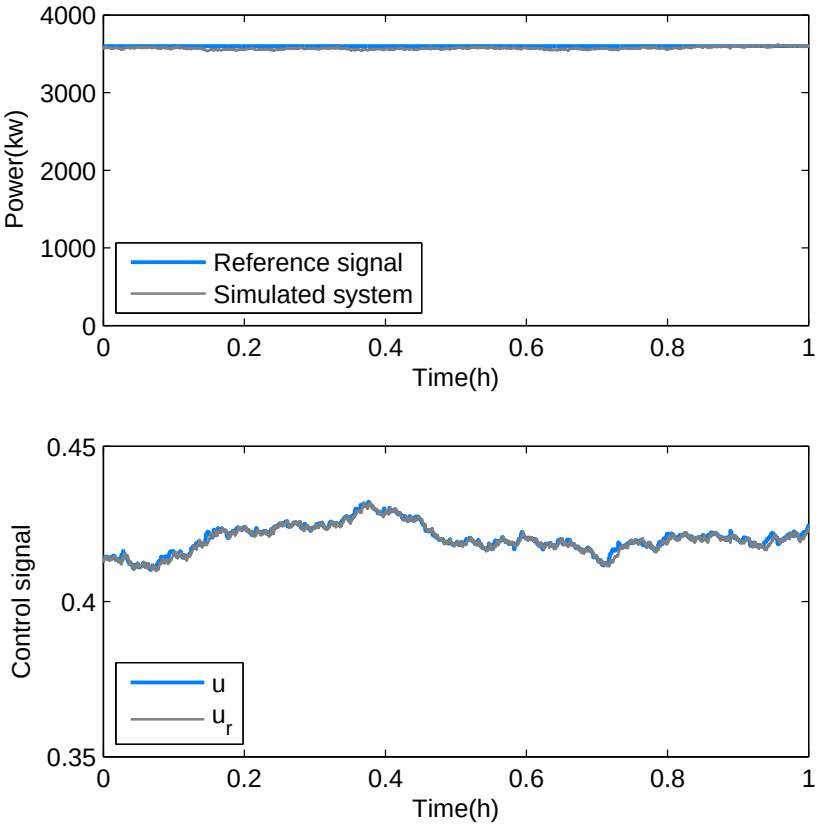


Figure 5.3. Non-interruptible loads tracking a constant reference.

To understand this problem, we see from Fig. 5.4 that there is a maximum and minimum slope that the output of the system can achieve. These values are given by the arrival and departure rates. In the case $u_a < u$ all the loads that enter the system are immediately turned on and this makes the power increase at rate λp_0, while there are active loads that are getting fulfilled at a rate $\approx \frac{u^* n}{\tau}$ and represent a decrease in the consumed power of $\approx p_0 \frac{u^* n}{\tau}$, so the maximum rate at which the power can increase is $\approx \lambda p_0 (1 - \frac{u^* n}{\tau \lambda})$. In the case $u_a > u$ the loads that enter the system are delayed and the decrease in power is given by the active loads that exit n, $\approx p_0 \frac{u^* n}{\tau}$. As the changes in u_a are slow, and all the loads in m are always on, the dynamics in m can be neglected for this approximation.

We ask the question of whether the system could not provide a more modest regulation target, i.e. offer a smaller value of θ. The choice of u^* can help by equalizing the upward and downward slopes, in particular choosing $u^* = \frac{\tau}{\tau + L}$, which follows from matching the increase and decrease rates at steady state. After experimenting we found that up to a value $\theta \approx 0.08$ the system is able to reach an acceptable performance score in PJM terms. Fig. 5.5, shows the system with these parameters.

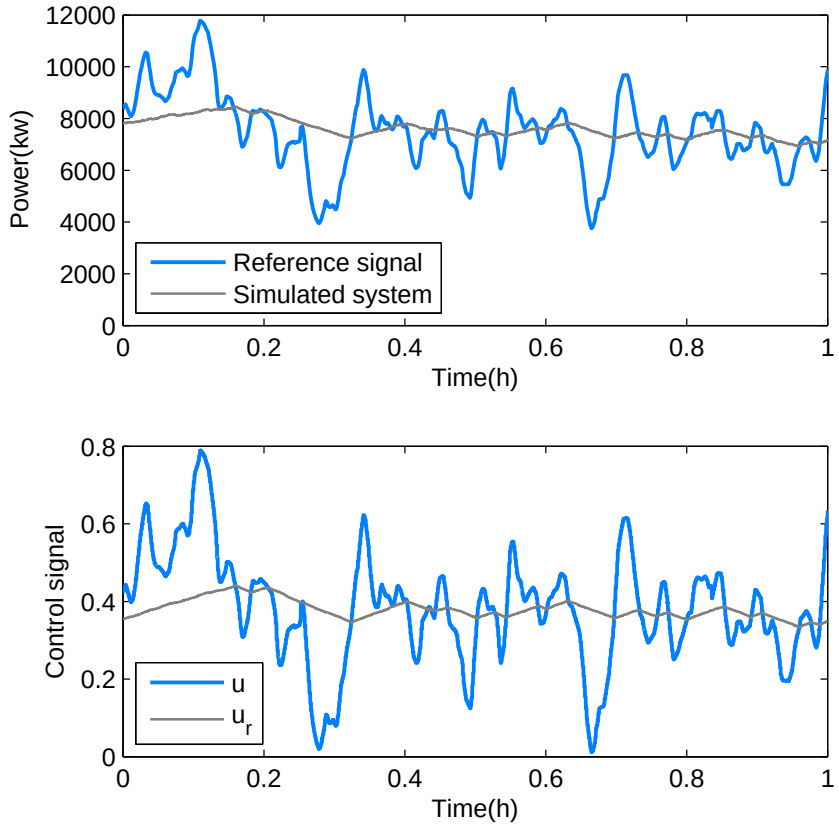


Figure 5.4. Non-interruptible loads failing to track the regulation signal.

So we find that, although the system is much slower than the interruptible case it can still provide some regulation.

5.2.3 Algorithm 3: Limited interruptions

The two algorithms already presented have each of them some problems, the first one has an excellent performance but interrupts the loads too much and the second one does not interrupt loads but has poor performance. This third algorithm takes some ideas form the previous two and combines them in a solution that limits the number of interruptions per load while still achieving a good performance.

As in the first algorithm the aggregator will broadcast a control signal to all loads, coordinated with the regulation signal, and each of them individually will decide their action, turning on or off, based on this signal. The difference is that now the signal will not be directly u, instead the aggregator will send a signal, $u_c = f(u - u_a)$ (with u_a as defined in algorithm 2), that takes values in $[-1, 1]$ and represents the probability of a load changing its status. The function $f : [-1, 1] \rightarrow$

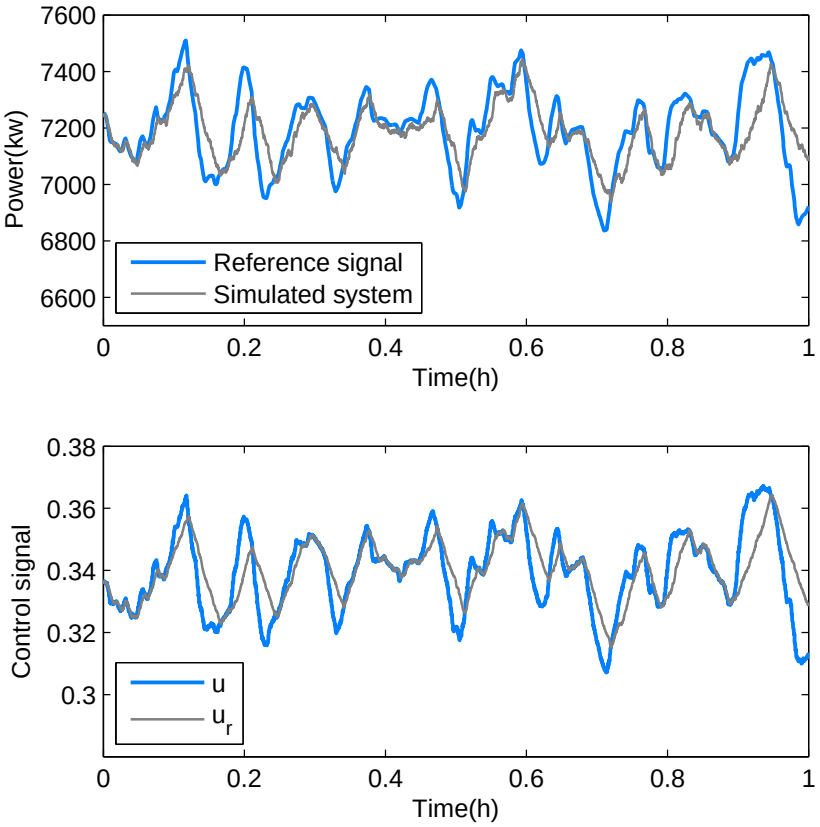


Figure 5.5. Non-interruptible loads tracking the regulation signal.

$[-1, 1]$, is monotonically increasing and satisfies $f(0) = 0$. There exists several candidate functions that can affect the response time and precision of the system. We choose to use $f(x) = sat^{0.5}_{-0.5}(2x)$ after trying various options, however there is place for improvement in this function. If $u_c < 0$ means that $u_a > u$ and more loads should turn off and vice versa. Only loads that will contribute to the needed change will react to the signal. In addition loads will have as restriction a maximum number of permitted interruptions or a minimum continuous service interval each time they are activated. In the case the restriction comes as a maximum number of interruptions it will be treated as a minimum service interval, which can be calculated as: $\tau_j/(\#_{maxint} + 1)$.

We will show an example to clarify. A load arrives at the system asking for a service of $2000s$ with a laxity of $1000s$, it consumes $1kw$ and can stand a maximum of 3 interruptions. Upon arrival the load is off and receives u_c, if u_c is negative there is not action for the load to take because $u_a > u$ and the system needs for loads to turn off. When u_a drops below u then u_c will become positive and each time the load receives $u_c > 0$ it choses with probability u_c whether to turn on or not. Once the loads decides to turn on it must remain in this status for at least

$2000s/(3+1) = 500s$, to meet the interruptions constraint. After this lapse the load will start responding to u_c again, but only if the value of $u_c < 0$ which means that loads that are on should start turning off again. Once the load turns off it must remain in this status for $1000s/(3+1) = 250s$, applying the same logic as when it is on. This process ends when the load runs out of laxity or completes its service.

Now there is a new variable that will affect the performance of the system, the maximum number of interruptions of each load, which can take different values and distributions. There are plenty of loads that cannot be interrupted once they started, such us washing machines, and others that can be interrupted quite often, for instance resistive loads like water heaters. There might be also some correlation between the service time and the maximum number of interruptions, loads that need only some minutes of service will probably not tolerate many interruptions, whereas a battery or heating system that needs hours of service can be interrupted plenty of times during their service interval. In this first instance of comparison between algorithms we will assume no correlation between the parameters and we will test against two different distributions for the number of interruptions, geometric and uniform.

The constant reference case is not very interesting to test as we have already shown in the previous algorithm that even with no interruptions the system is able to track a constant reference. Anyhow the simulations results showed an improvement from the no-interruptions case. We tested the system using both distributions, geometric and uniform, with a mean $\#_{maxint} = 5$. The results were very similar, the RMS value of δp was $27kw$, as with algorithm 1, but this time without interrupting loads at all.

The frequency regulation case presents more interesting results. We simulated the system using the same parameters as in the previous algorithms with a mean number of maximum interruptions of 5, using two distributions, geometric and uniform. In Fig. 5.6 we show the simulation results for the geometric case (the uniform case is almost identical). The first remark is that although the mean $\#_{maxint} = 5$, in practice the loads were interrupted on average 1.5 and 1.7 times in the geometric and uniform cases respectively. The PJM score was 0.84 and 0.85, both of them more than acceptable. These results shows that there exists reasonable ways of scheduling the loads while respecting the interruption and deadline limitations.

Looking at the simulation more into detail we can see that the system responds rapidly to changes and is able to keep close to the reference while it is around the working point ($3600kw$ in this case). The problems appear when there are long ramps, up or down, because there are always some loads that cannot change state because of the interruptions limitation. Once most of the loads that were available to change to the desired status did change, then the consumed power keeps approaching the reference at a slower pace. Each time a load finishes the minimum time restriction it starts responding to the control signal again and it will eventually change to the desired status.

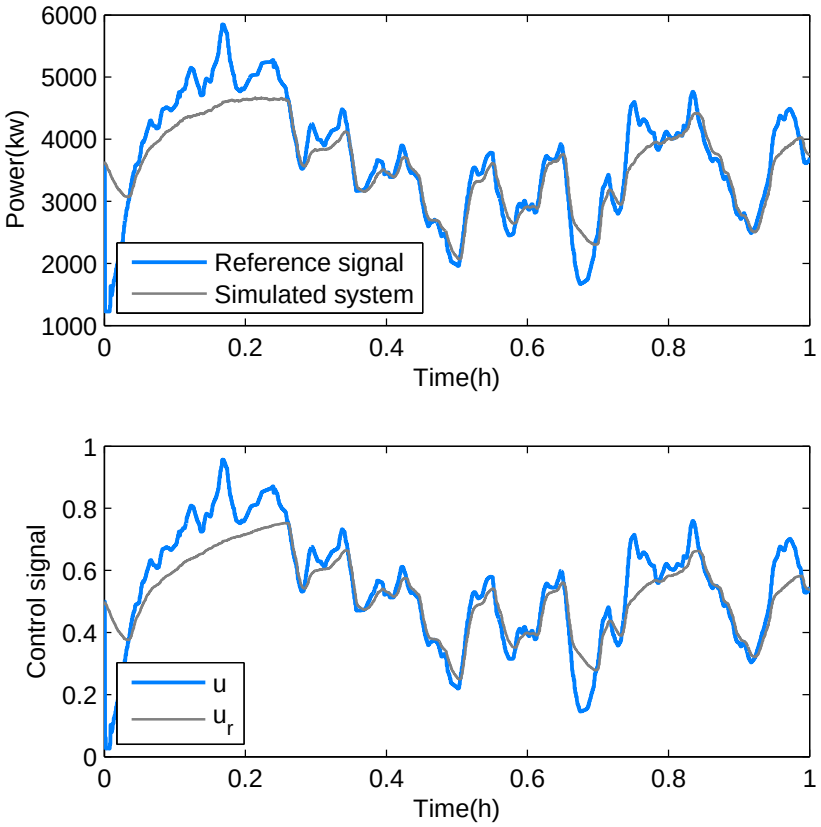


Figure 5.6. 1-hour simulation of deferrable loads tracking a regulation signal using algorithm 3. Mean $\#_{maxint} = 5$

The quality of the response will clearly improve if the mean $\#_{maxint}$ increases. In Fig. 5.7 we show the same system but with a higher number of allowed interruptions, 20 in this case. It is clear the improvement of the performance compared to the previous case. We will study the effect of this parameter more in detail further in the chapter.

From the analysis of these 3 algorithms it follows that an implementation that takes into account deadlines and interruptions could be possible using this model and control logic. Algorithm 3 appears to be the closest to a feasible implementation and all the simulations from now on will be based on this algorithm.

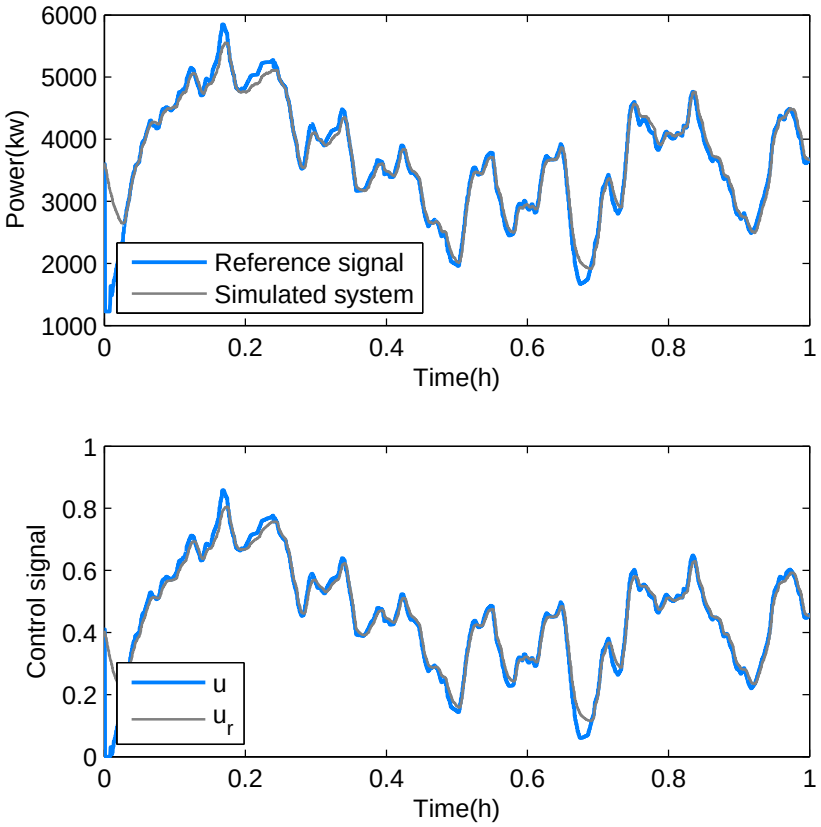


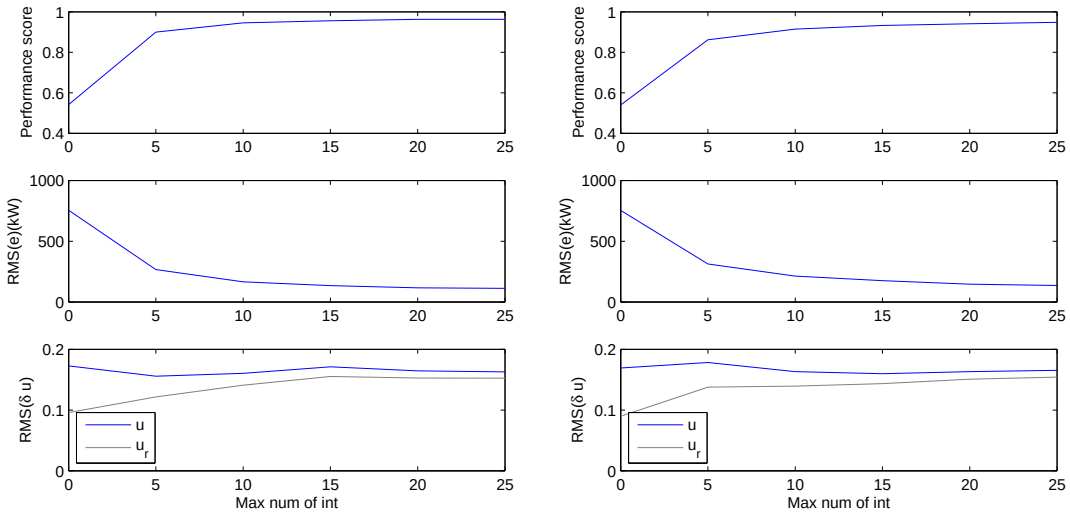
Figure 5.7. 1-hour simulation of deferrable loads tracking a regulation signal using algorithm 3. Mean $\#_{maxint} = 20$

5.3 EFFECT OF NUMBER OF INTERRUPTIONS

The algorithm designed for scheduling loads strongly depends on how many loads allow interruptions, this implies the mean $\#_{maxint}$ and how it is distributed. If $\#_{maxint} \to 0$ it resembles algorithm 2 and if $\#_{maxint} \to \infty$ then it is almost as algorithm 1. We will now explore by means of simulations the compromise between performance and load interruptions in order to find reasonable working points for this algorithm. We could imagine that a load aggregator that manages a portfolio of loads can chose from the available loads to group them in packages in order to sell their flexibility to the SO. The way in which aggregators will put together loads will depend on their individual parameters in order to arrive at the desired performance level.

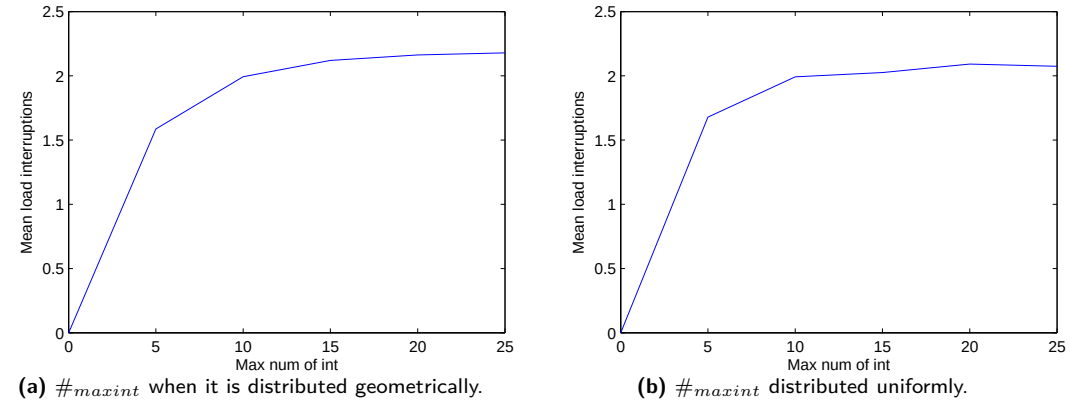
In Figs. 5.8 and 5.9 we can see the effect of the mean $\#_{maxint}$ in both the cases it is distributed uniformly and geometrically.The first observation, as it could be expected, is that the performance of the system increases monotonically with $\#_{maxint}$. The most interesting result that follows form this simulation is the relation

of the real number of interruptions with the $\#_{maxint}$. We can see that the algorithm does not exploit all the possible interruptions, moreover, it seems that it has an upper limit in the number of interruptions, being for this particular setting around 2.3. This is a positive result as it means that loads are being interrupted much less than expected, which is a desired operation. Another possible conclusion is that if we would like to improve the performance of the the system we could overestimate the $\#_{maxint}$ of the loads as in practice the system do not exploit all the available interruptions. This observation however is a bit trickier as we are not looking at individual loads, some of which reach their $\#_{maxint}$.



(a) $\#_{maxint}$ distributed geometrically. **(b)** $\#_{maxint}$distributed uniformly.

Figure 5.8. Performance of the system as a function of the mean $\#_{maxint}$.

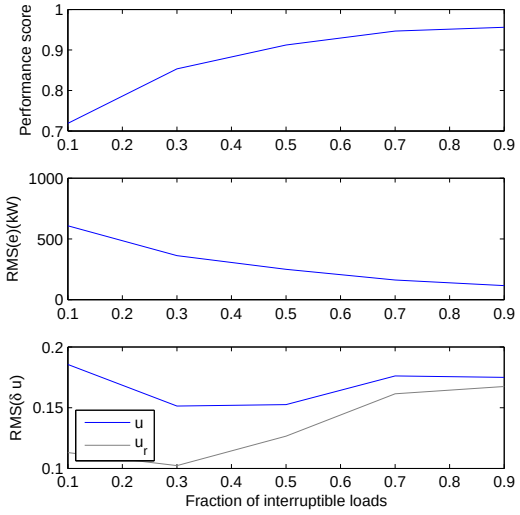


(a) $\#_{maxint}$ when it is distributed geometrically. **(b)** $\#_{maxint}$ distributed uniformly.

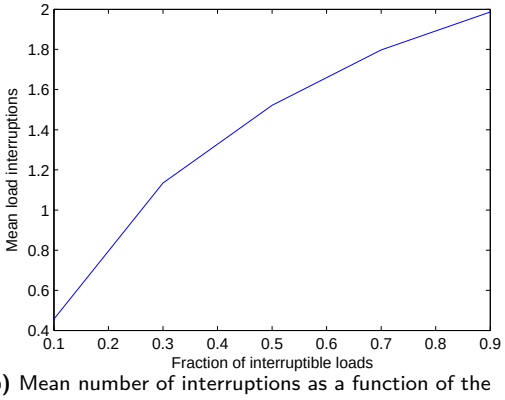
Figure 5.9. Mean number of interruptions as a function of the mean $\#_{maxint}$.

Another interesting setup would be a group of loads were only a fraction of them allow to be interrupted. Figure 5.10 shows the results for a system where only a fraction of the loads allow to be interrupted unlimited times. As it can be expected

the performance improves with the fraction of interruptible loads. It also shows that only with around 20% on interruptible loads the system is able to provide an acceptable service under these settings.



(a) Performance of the system as a function of the fraction of the loads that allow interruptions.



(b) Mean number of interruptions as a function of the fraction of the loads that allow interruptions.

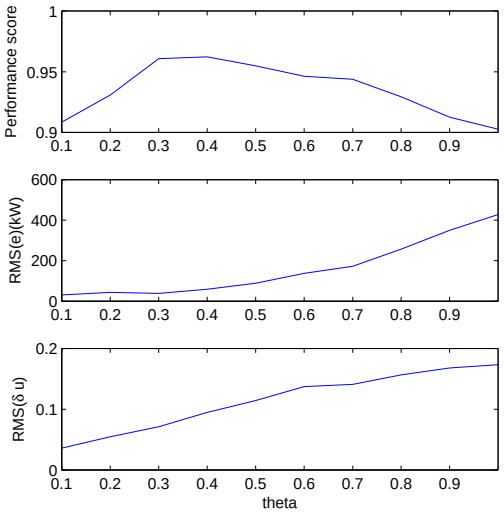
Figure 5.10

5.4 EFFECT OF OFFERED REGULATION

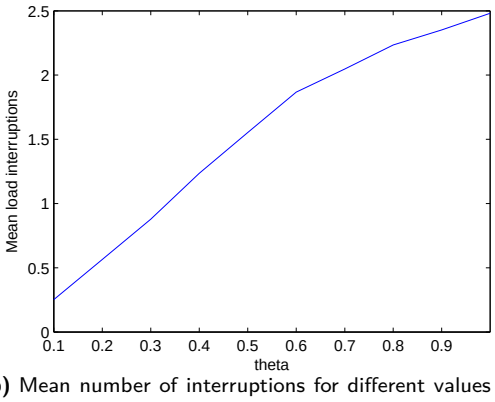
As we mentioned in section 3.3.1 the load aggregator will look to maximize the offered regulation, in order to maximize its profit, while staying above the minimum score allowed. The theoretical bound that we found was $\theta_{\max} = \frac{L}{L+\tau}$, but not necessarily it will always be possible to reach this value in practice. On the other hand it could also be possible to commit more regulation than this value, although we know in advance that we will not be able to reach the needed power in every moment, we could still have an acceptable performance.

In Fig. 5.11a we show the performance for the basic setup with different values of offered regulation. We can see that although the absolute error increases with θ, the system still performs well even with $\theta \geq \theta_{\max}$. This can be explained because the error only accounts for 1/3 of the score (see appendix A), and it is averaged with precision and delay, in which the system performs very well.

SO's are interested in acquiring regulation of the best quality as possible and hence they develop metrics as the one we explained previously. In the case of PJM the payout to regulation providers depends not only in the amount of regulation but also in the score the provider reaches. Figure 5.12 shows the product of θ times the obtained score as a measure of the payout for the services. This indicates that the aggregator should try to compromise as much regulation as possible.



(a) Performance metrics for different values of offered regulation.



(b) Mean number of interruptions for different values of offered regulation.

Figure 5.11

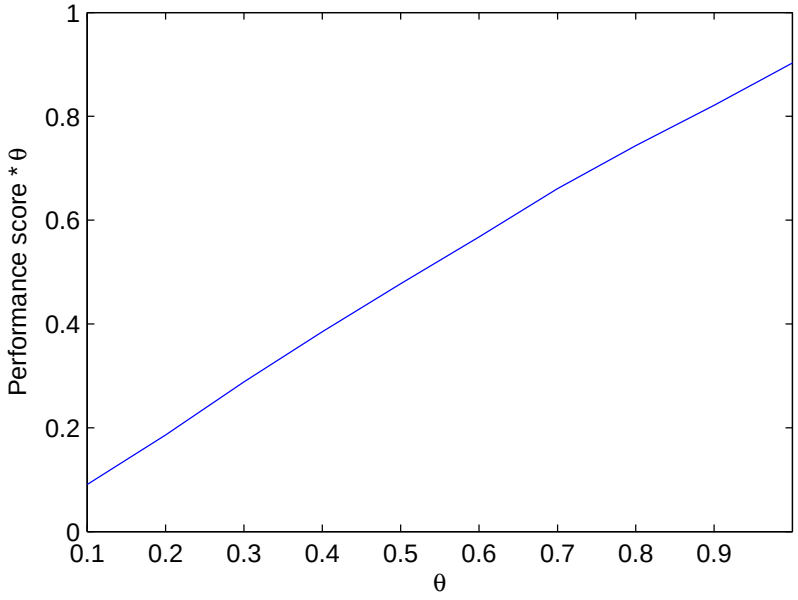


Figure 5.12. Performance score times θ for different values of offered regulation.

5.5 EFFECT OF LAXITY

The laxity of the loads is one of the key parameters of this system. The more flexible loads are, the more regulation they will be capable of offering. The laxity of the aggregate of loads is defined by the deferability factor $\eta = \frac{\tau}{\tau+L}$, as defined in 2.4. But this value does not give us complete information on the laxity of individual loads, how it is distributed among loads and the correlation with service time and

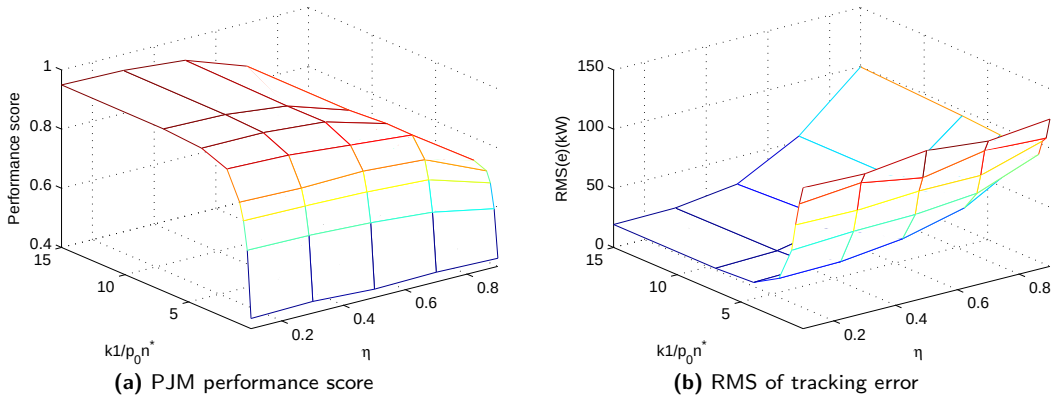
power is missing. In this section we will try to briefly study the effect of laxity on the system performance.

We will start by taking the basic setup and test it with different values of η. It will be logic to expect that more laxity (smaller η) will imply better performance. In order to do a meaningful simulation we also have to test the system with different values of the performance weight, k_1, as each particular setting will have a different optimal value for the controller's constant, see Section 4.2.

Figures 5.13 and 5.14 show the simulations results for the system providing a regulation of $\theta = 0.5$. We can see the performance metrics for different values of η and k_1. The four metrics we are using clearly show the benefit of having a smaller η, which means more laxity.

Fig. 5.13a shows how the performance score improves for smaller values of η independently of k_1. We can also see the different behavior the system presents when varying the performance weight depending on the laxity. If loads do not have much laxity there is a maximum k_1 the system can tolerate before nonlinear effects came into play, in particular saturation.

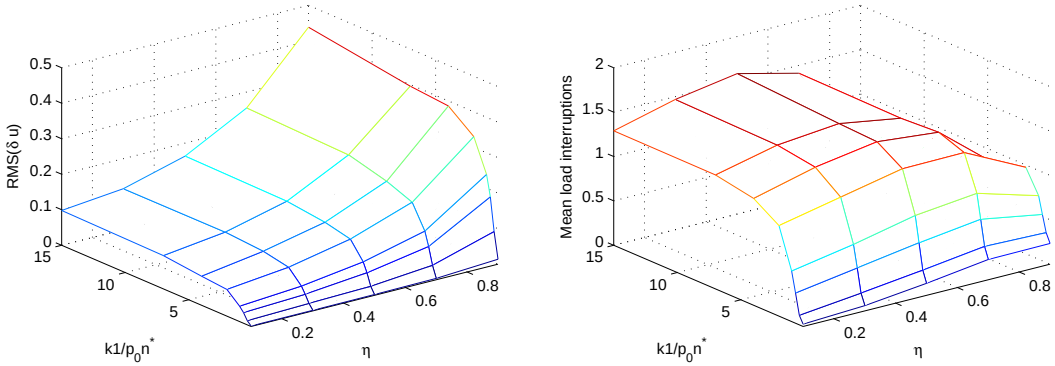
Fig. 5.13b supports the same conclusion showing how the error diminishes as the laxity increases. In Fig. 5.14a we can see the effect of η on the control effort. As loads are more flexible they respond better to control actions meaning that less control effort is necessary. Larger control efforts means that the system will be further away from the linear zone and more likely to saturate the control signal. Fig. 5.14b shows a very interesting effect. As loads become more flexible they are likely to be interrupted more often as they spend more time under the aggregator's control. This effect is up to a point (around $\eta = 0.3$ for these settings) where load interruptions decrease.



(a) PJM performance score **(b)** RMS of tracking error

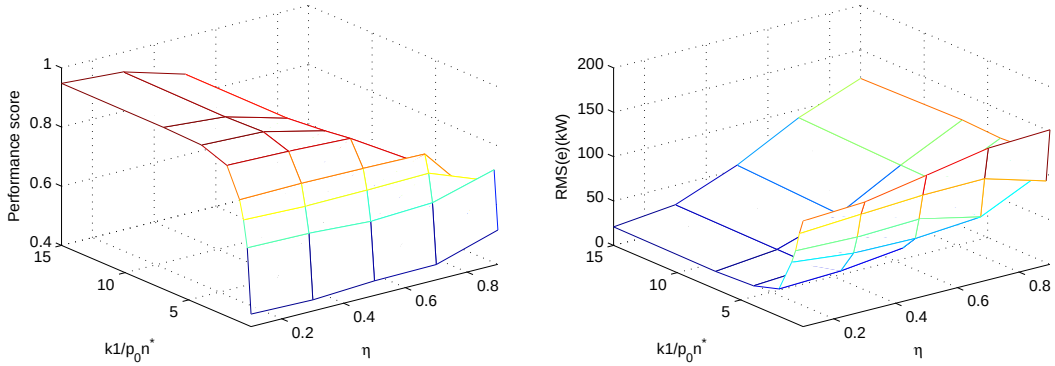
Figure 5.13. PJM performance score and RMS of tracking error for different values of η and k_1.

We would also like to explore the effect of the distribution of the laxity among loads on the system performance. We performed the same simulation as before



(a) RMS of control effort **(b)** Mean number of interruption

Figure 5.14. RMS of control effort and mean number of interruptions for different values of η and k_1.

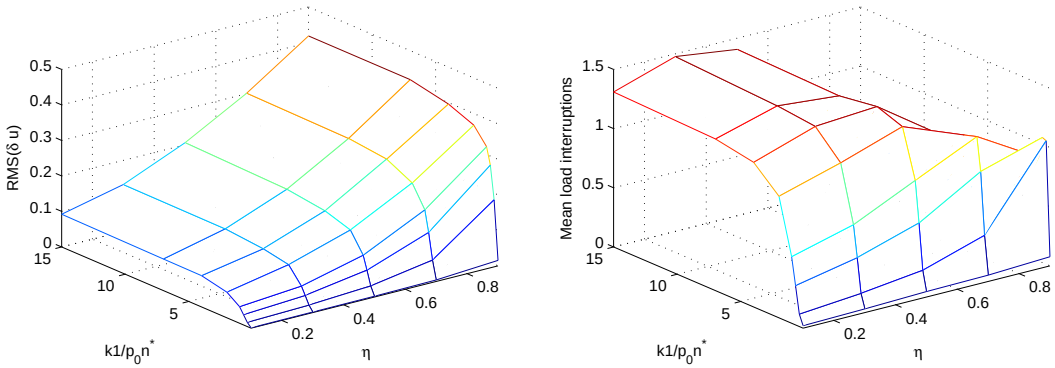


(a) PJM performance score **(b)** RMS of tracking error

Figure 5.15. PJM performance score and RMS of tracking error for different values of η and k_1. Laxity uniformly distributed.

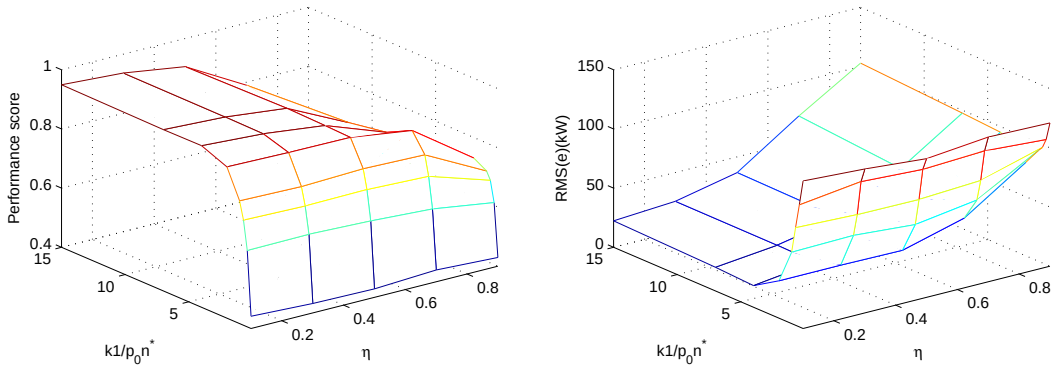
but instead of the laxity being distributed exponentially we tried with a uniform distribution and with fixed laxity.

A first conclusion is that the system is quite oblivious to the exact distribution of the laxity among the loads.



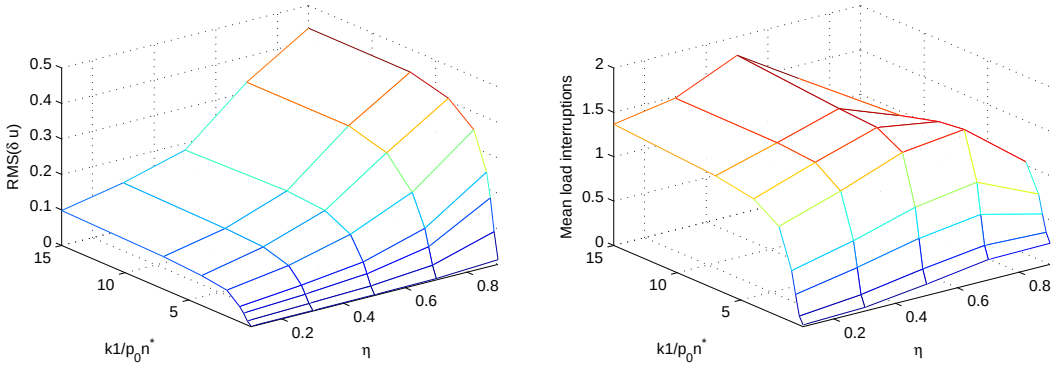
(a) RMS of control effort **(b)** Mean number of interruption

Figure 5.16. RMS of control effort and mean number of interruptions for different values of η and k_1. Laxity uniformly distributed.



(a) PJM performance score **(b)** RMS of tracking error

Figure 5.17. PJM performance score and RMS of tracking error for different values of η and k_1. Fixed Laxity



(a) RMS of control effort **(b)** Mean number of interruption

Figure 5.18. RMS of control effort and mean number of interruptions for different values of η and k_1. Fixed Laxity

5.6 VARIABLE LAMBDA

We would now like to go one step further into a real system. We can not expect that in real life the rate at which loads arrive into our system will be constant along the whole day. Although this rate will not be constant, system operators, energy retailers or aggregators will have an estimate of how this parameter varies along the day, which is directly related to how power consumption varies along the day. Using real data from PJM and assuming that the rest of loads mean parameters are constant along the day we can infer the rate of arrival, $\lambda(t)$, by using the ODE 3.1. The data we have to calculate λ is $p(t)$ and that we can assume $u = 1$ as there is no control on loads. Under these settings $n(t)$ is proportional to $p(t)$, $m(t) = 0$ and hence we can calculate $\lambda(t)$.

The setup for this simulation is as follows:

- The aggregator estimates using classical methods the power it should buy in the case loads are not under its control, $p_{u=1}(t)$.
- Using $p_{u=1}(t)$ and an estimate of loads mean parameters, p_0 and τ, the aggregator can infer the rate of arrival of loads, $\lambda(t)$.
- The aggregator chooses $u(t)$ for the whole day, taking into account the estimate laxity of the loads and if it intends to sell regulation or not.
- With all the previous data ($\lambda(t)$, $u(t)$, p_0, τ, L) and model 3.1, it will have the information on how much power to buy for next day, $p(t)$.
- The aggregator can now set the amount of regulation it is able to offer θ.

At operation time the aggregator will continue broadcasting the control signal to all loads. The difference is that now the control loop parameters will not be constant. As the variations in λ will be slow compared to the control loop, $4s$, we decided to refresh the value of the feedback constants every $10min$, which seems to be enough to compensate for the variations in λ and does not overload the system.

Figure 5.19 shows the system with variable lambda for a whole day. The reference signal is the power purchased the day before (orange line) plus the regulation power. In Fig. 5.20 we can see a zoom in to appreciate how the system is capable of following the regulation signal. The PJM score for this case was 0.91 which is more than acceptable.

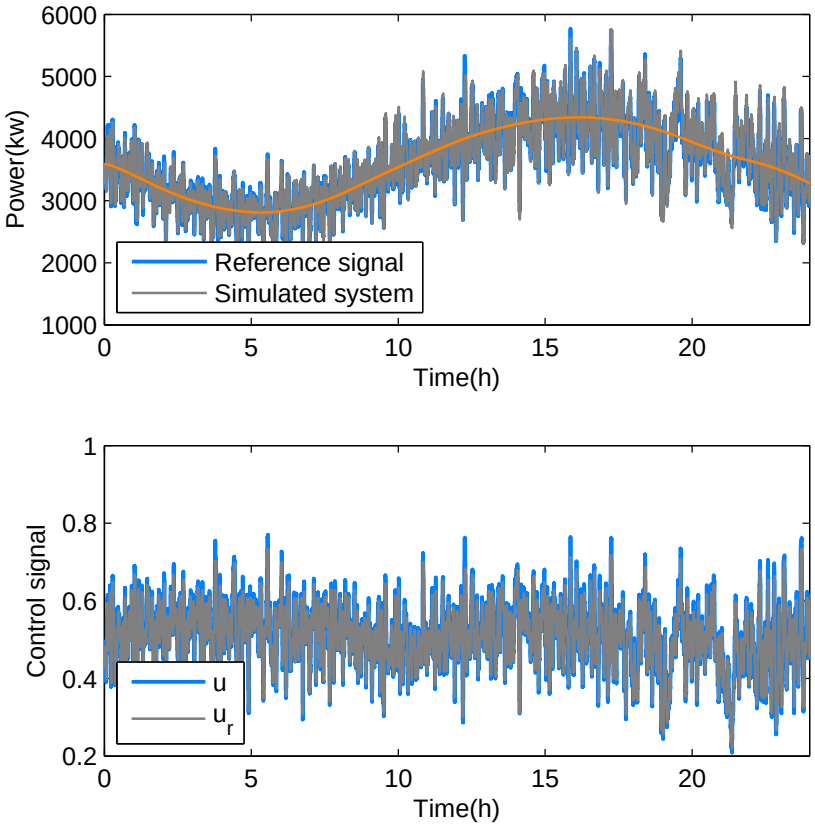


Figure 5.19. 1-day simulation of the system with variable λ and $\theta = 0.33$.

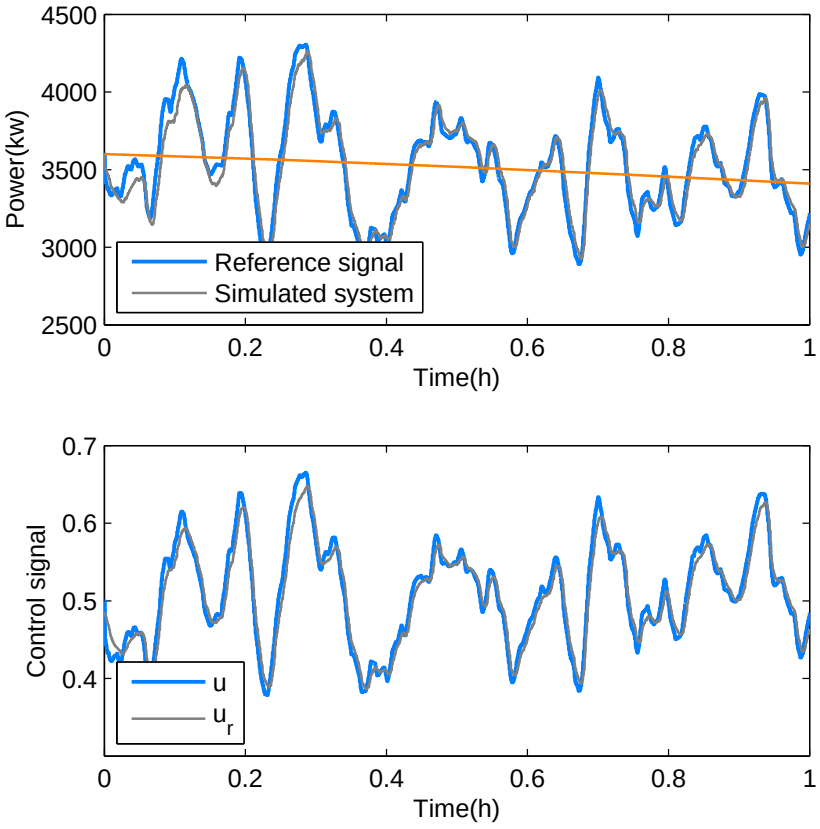


Figure 5.20. 1 hour of a 1-day simulation of the system with variable λ and $\theta = 0.33$.

5.7 SUMMARY

In this section we focused on analyzing possible implementations for the control techniques proposed in this thesis. Three different approaches were presented, being one of them quite feasible for implementation, the "limited interruptions" algorithm. Using this algorithms we explore the effects of the different parameters in the performance of the system and simulated a scenario closer to a real implementation.

Chapter 6

Conclusions and future work

In this thesis we analyzed the problem of a load aggregator which manages a cluster of deferrable loads and uses their flexibility to provide frequency regulation to the System Operator. By the use of simple ODE models we quantified the potential of the loads and we design controllers for the proposed objective. Algorithms for implementing the proposed controller were also analyzed showing the feasibility of our proposals.

A first analysis on the deferment of loads service showed the potential of a simple fixed deferral action on the variability of the consumption in electric systems. Further analysis showed that an active control on the service level that can be implemented by simple algorithms exploits this flexibility to the point of not only eliminating the need for frequency regulation but also being able to provide regulation to the grid. The design of the controllers was based on optimal control theory with the aid of simulations. What we leave is a tool for designing the controller for frequency regulation with loads which depends on loads parameters and the frequency spectrum of the regulation signals.

Several lines of work remain open. With respect to the model there are some theoretical results that are not proved although extensive simulations validate them. It would be also interesting to extend the model so it covers a wider range of scenarios with respect to load parameters distributions and correlation between them. The possibility of adding storage to the system and study the optimal way to manage the loads combined with storage is also an open line.

With respect to algorithms and simulations there is plenty of work to do. We presented an algorithm that is capable of implementing the desired control with a high level of accuracy and low connectivity requirements. Still it remains open the possibility of improving the algorithm and formalizing its performance. With respect to simulations one of the setbacks we met was the lack of real data. In order to have a more realistic outline of the potential of the loads, data from individual

loads uses should be gathered. *Laxity*, which is the key parameter to quantify the potential of load flexibility, is not easily measured and would be important in order to design a system like the one proposed. Other data such as service time, nominal power, number of interruptions or the correlation between them are also important, but can be more easily extracted from existing data.

Bibliography

[1] A. von Meierl, *Electric Power Systems, A Conceptual Introduction.* John Wiley & Sons, 2006.

[2] 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.

[3] 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.

[4] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "Frequency Regulation from Flexible Loads: Potential, Economics, and Implementation," in *Proc. of the American Control Conference (ACC)*, 2014.

[5] Y. Lin, P. Barooah, S. Meyn, and T. Middelkoop, "Demand Side Frequency Regulation from Commercial Building HVAC Systems: An Experimental Study," in *Proc. of the American Control Conference (ACC)*, 2015.

[6] S. Tindemans, V. Trovato, and G. Strbac, "Decentralized Control of Thermostatic Loads for Flexible Demand Response," *IEEE Transactions On Control Systems Technology*, vol. 23, pp. 1685–1700, 2015.

[7] S. Han, S. Han, and K.Sezaki, "Development of an Optimal Vehicle-to-Grid Aggregator for Frequency Regulation," *IEEE Transactions on Smart Grid*, vol. 1, pp. 65–72, 2010.

[8] H. Zarkoob, S. Keshav, and C. Rosenberg, "Optimal contracts for providing load-side frequency regulation service using fleets of electric vehicles," *Journal of Power Sources*, vol. 241, pp. 94–111, 2013.

[9] 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.

[10] 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.

[11] S. Goguri, J. Hall, R. Mudumbai, and S. Dasgupta, "A distributed, real-time and non-parametric approach to demand response in the smart grid.," in *Proc. of Annual Conference on Information Sciences and Systems (CISS)*, 2015.

[12] J. Hong, X. Tan, and D. Towsley, "A performance analysis of minimum laxity and earliest deadline scheduling in a real-time system," *IEEE Transactions on Computers*, vol. 38, pp. 1736–1744, 1989.

[13] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*. Prentice Hall, 1996.

[14] PJM, "Fast Response Regulation Signal." `http://www.pjm.com/markets-and-operations/ancillary-services/mkt-based-regulation/fast-response-regulation-signal.aspx`.

[15] New York Independent System Operator, "Ancillary services manual." `http://www.nyiso.com/public/webdocs/markets_operations/documents/Manuals_and_Guides/Manuals/Operations/ancserv.pdf`.

[16] J. Burl, *Linear Optimal Control: $\mathcal{H}_2$ and $\mathcal{H}_\infty$ methods*. Menlo Park, CA: Addison Wesley, 1999.

[17] PJM, "PJM Manual 12: Balancing Operations." `http://www.pjm.com/~/media/documents/manuals/m12.ashx`.

[18] P. Robert, *Stochastic networks and queues*. Berlin: Springer-Verlag, 2003.

[19] F. Paganini and A. Ferragut, "PDE models for population and residual work applied to peer-to-peer networks," in *Proc. of 46th Annual Conference on Information Sciences and Systems, Princeton, NJ*, 2012.

[20] A. Ferragut and F. Paganini, "Content dynamics in P2P networks from queueing and fluid perspectives," in *Proc. of 24th International Teletraffic Congress, Krakow, Poland*, 2012.

[21] A. Ferragut and F. Paganini, "Queueing analysis of service deferrals for load management in power systems," in *Proc. of the 53rd Annual Allerton Conference on Communication, Control, and Computing*, 2015.

Appendix A

PJM performance score

One of the metrics that we used along the thesis to rate our proposals for frequency regulation is the score used by PJM Interconnection, a Regional Transmission Organization in the United States [14].

In order to rate the different resources that apply to provide frequency regulation PJM uses it own score [17], from 0 to 1, which depends on the delay, the correlation and the precision of the response compared to the reference signal. This score is used in order for a resource to able to participate in the market, for which it must score over 0.75, but also it determines the payout, which increases together with the score. The calculations are done a posteriori as they are non-causal. The performance score is the average of the 3 scores all of them normalized in a scale form 0 to 1:

$$\begin{matrix}\text{Performance}\\ \text{Score}\end{matrix} = \frac{1}{3}\left(\begin{matrix}\text{Delay}\\ \text{Score}\end{matrix} + \begin{matrix}\text{Correlation}\\ \text{Score}\end{matrix} + \begin{matrix}\text{Precision}\\ \text{Score}\end{matrix}\right).$$

The score is calculated on hourly basis from samples taken every 10 seconds. The 3 scores are calculated for each sample and then the final score is the average over the hour.

Delay and Correlation scores are calculated together using the 5 minutes periods of the reference signal and 10 minutes of the response. The correlation score is calculated as the normalized cross-correlation of the 5 minutes reference signal and the 5 minutes time-shifted response starting at the sample time plus the delay (δ). The delay score is linear with the time-shift being 1 for zero delay and 0 for 5 minutes delay. The delay (δ) is chosen as the one that maximizes the sum of both scores.

$$\begin{aligned} \underset{\text{Score}}{\text{Correlation}} &= \max_{\delta=0\,to\,5min} C\left(Signal\left(0:5min\right), Response\left(\delta:\delta+5min\right)\right) \\ \underset{\text{Score}}{\text{Delay}} &= \left|\frac{\delta - 5Minutes}{5Minutes}\right| \end{aligned}$$

Precision score is calculated based on the relative error in the signal 1-norm, where the reference is averaged over the 1 hour period:

$$\underset{\text{Score}}{\text{Precision}} = 1 - \frac{\|error\|_1}{\|reference\|_1}.$$

The minimum score for qualifying to participate in the regulation market is 0.75. As a reference we have that the average score for steam generators is slightly above 0.75, hydro generators score slightly higher around 0.8, batteries are one of the best resources scoring over 0.9, whereas other demand response resources score over all the range from 0.7 upwards.

Appendix B

Stochastic analysis

This appendix has the purpose of supporting some of the models we used in this thesis based on stochastic analysis tools. Along this Appendix we assume all arrivals follow a Poisson process and service times are distributed exponentially among loads as well as laxity, independently of each other.

The Appendix is divided in four sections. First we analyze the model of Chapter 2 in two parts; we start by studying how the expected value of the number of loads evolves and then we focus in the variations around this value. Then we repeat the procedure with the model of Chapter 3.

B.1. AGGREGATOR MODEL

The first model we will analyze is:

$$\dot{n}(t) = \lambda - \frac{1}{\tau} u^* n(t). \tag{B.1}$$

As explained in Chapter 2 this equation models the dynamics of arrivals and departures of a cluster of loads for a fixed service rate.

Let $A(t)$ be the arrival process of the loads; if we know that this process happens at mean rate λ, then $E[A(t)] = \lambda t$. If loads are serviced at a fraction u^* of their nominal power, then each of them will leave the system τ_j/u^* after its arrival generating a departure process $D(t)$. The difference between the two processes will be the number of loads in the system at a given time, $N(t) = A(t) - D(t)$.

To model the exact dynamics of the system we should know the arrival process and the service time of each load. As we aim to work without knowing the details of every load we will start by considering the dynamics of the expected value of the number of loads, $E[N(t)] = n(t)$.

In Fig. B.1 we show an example of load arrivals (up to time T) and their departures. The shaded area corresponds to the total service time requested by the loads arriving before time T. Note that at time T not all the loads were totally served, so part of the shaded area is to the right of this time. One way to calculate this area is as the sum of all the service time requests until time T,

$$S_T = \sum_{j=1}^{A(T)} \tau_j / u^*.$$

We can also calculate the same area as the integral of $N(t)$ from time 0 until the last load exits the system (assuming no loads arrive after time T) or equivalently we can calculate this area as the integral of $N(t)$ from 0 to T (this would be the shaded area to the left of T) plus the remaining service time of the loads still active at T, $\sum_{j=1}^{N(T)} \tau_j^r(T)/u^*$ (the area to the right of T), being $\tau_j^r(T)$ the remaining service time of load j at time T. So

$$S_T = \sum_{j=1}^{A(T)} \frac{\tau_j}{u^*} = \int_0^T N(t)dt + \sum_{j=1}^{N(T)} \frac{\tau_j^r(T)}{u^*}.$$

If we now take expectation at both side of the equation, assuming $E[\tau_j] = \tau$, we have

$$E[S_T] = E[A(T)]\frac{\tau}{u^*} = \lambda T \frac{\tau}{u^*} = E\left[\int_0^T N(t)dt\right] + E[N(T)]\frac{\tau}{u^*} = \int_0^T n(t)dt + n(T)\frac{\tau}{u^*},$$

where $E\left[\tau_j^r(T)\right] = \tau$ because of the memory loss property of exponential tasks. Now differentiating both sides of the equation with respect to T, we arrive at

$$\lambda \frac{\tau}{u^*} = n(T) + \dot{n}(T)\frac{\tau}{u^*}.$$

Now if we multiply by $\frac{u^*}{\tau}$ and reorder the terms we arrive at equation B.1, which is our model for tracking the estimated number of loads at the system if they are served at rate u^*:

$$\dot{n}(t) = \lambda - \frac{1}{\tau} u^* n(t).$$

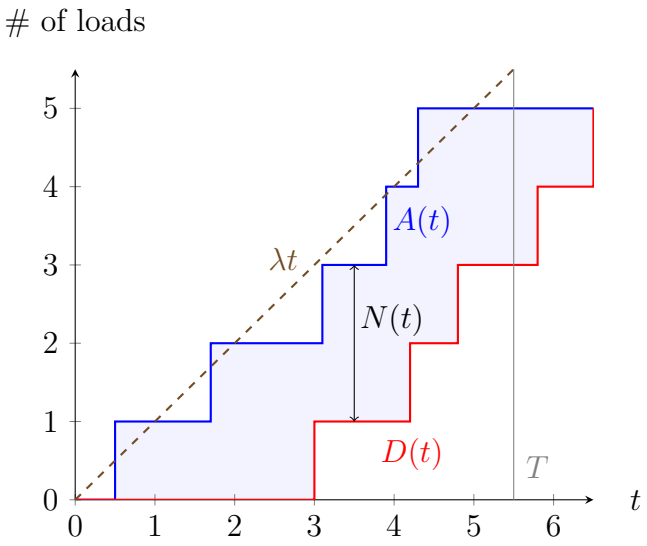


Figure B.1. Loads arrival and departure process

B.2.

The next step is to study how the actual number of loads vary around the expected value. The number of loads present in the system $N(t)$ behaves as an $M/M/\infty$ queue [18] with arrival rate λ and service completion rate:

$$\mu = \frac{u^*}{\tau}.$$

If the arrival rate λ is large, the random process $N(t)$ can be well approximated by a deterministic trajectory following the ODE (B.1); perturbations around the equilibrium value can be approximated by a random noise input to (B.1).

In mathematical terms, let q a scaling factor, and $N_q(t)$ represent the random process with arrival rate $q\lambda$. Provided $N_q(0)/q \to n(0)$, the rescaled random process satisfies:

$$\bar{N}_q(t) = \frac{N_q(t)}{q} \to_{q\to\infty} n(t)$$

uniformly over compact sets, where $n(t)$ is the solution of (B.1) with initial condition $n(0)$ [18].

As for variability around equilibrium, a diffusion approximation can be performed. If the initial condition satisfies $N_q(0)/q = n^*$, then $n(t) = n^* \ \forall \ t$ and the random process

$$\delta N_q(t) = \frac{N_q(t) - qn^*}{\sqrt{q}}$$

converges in distribution [18] to the solution of the following stochastic differential equation:

$$\dot{\delta n} = -\mu \delta n + \sqrt{2\lambda} v_0(t),$$

where $v_0(t)$ is stationary white noise of unit power spectral density[1]. The factor 2λ in the net noise power spectrum comes from the two sources of variability, arrivals and departures, each contributing a term λ, associated with the Poisson arrival and departures. This type of variability analysis has been used before to track population profiles in P2P network applications in [19, 20], which have the same type of fluid model as our present system.

B.3. TWO-STATE MODEL

This section supports the model presented in Chapter 3. $N(t)$ denotes the population of loads with remaining laxity which are served at level u, and $M(t)$ are the loads with no laxity which are served at full power. The behavior of the population variables over time is described by the continuous-time Markov chain with transition rates depicted in Figure B.2. This represents basically two $M/M/\infty$ queues with a partial transition between the two, as is now explained.

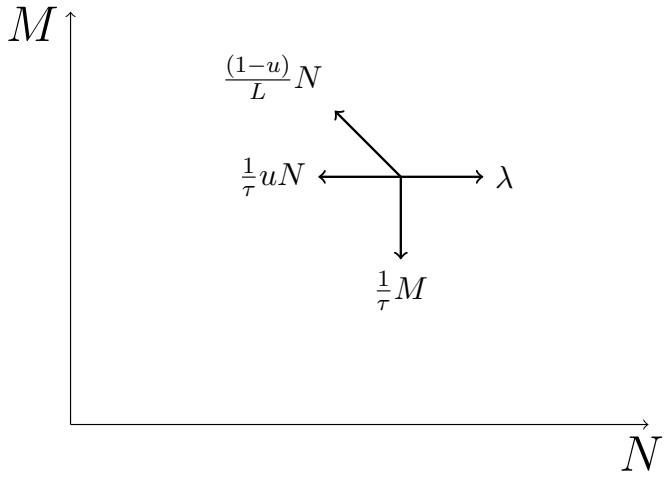


Figure B.2. Markov state diagram with transition rates

- $(N, M) \mapsto (N + 1, M)$ is a new Poisson(λ) arrival.
- $(N, M) \mapsto (N, M - 1)$ represents a departure from the M queue. Service here is at full power, so service times are $exp(\frac{1}{\tau})$, invoking the memoryless property of exponentials. The minimum of M such exponentials is $exp(\frac{M}{\tau})$, justifying the transition rate $\frac{M}{\tau}$.
- $(N, M) \mapsto (N - 1, M)$ represents a load from the N queue completing service. Since these are served at fractional power u, their individual service time is $exp(\frac{u}{\tau})$, which yields the transition rate $\frac{Nu}{\tau}$.

[1] A more formal version is $d(\delta n) = -\mu\delta n dt + \sqrt{2\lambda}dW$, where $W(t)$ is standard Brownian motion. Such δn constitutes an Ornstein-Uhlenbeck process. For second order analysis, however, the above description suffices.

- $(N, M) \mapsto (N-1, M+1)$ represents the transition between the N and M queues due to expiration of laxity. Since laxity is consumed at rate $(1-u)$, the time for this occurrence in one load is distributed as $exp(\frac{1-u}{L})$, which yields the transition rate $\frac{N(1-u)}{L}$.

In [21] some analysis was provided for the above Markov chain in the case of a fixed u. However in this thesis we are interested in *controlling* u to achieve a desired regulation objective; for this purpose a more tractable model involves replacing the Markov chain by a differential equation.

The fluid-flow counterpart to the Markov chain in Figure B.2 is obtained by interpreting N and M as continuous variables, and replacing the transition rates with different contributions to their drift, giving place to the following model:

$$\dot{n}(t) = \lambda - \frac{1}{\tau} n(t)u(t) - \frac{1}{L} n(t)(1-u(t)), \tag{B.2a}$$

$$\dot{m}(t) = \frac{1}{L} n(t)(1-u(t)) - \frac{1}{\tau} m(t), \tag{B.2b}$$

$$p(t) = p_0(n(t)u(t) + m(t)). \tag{B.2c}$$

The formal relationship between the two models is beyond the scope of this thesis. We state briefly that the solution to the differential equation can be seen as the limit of the scaled stochastic processes $(\frac{1}{q}N_q(t), \frac{1}{q}M_q(t))$ as $q \to \infty$, where the (N_q, M_q) correspond to the Markov chain under scaled arrival parameter $q\lambda$, and suitably scaled initial condition, as in the previous Section. For details on such procedure we refer to [18].

B.4. RANDOMNESS IN TWO-STATE MODEL

To study the effect of noise in the system we will work with the linearized model around the equilibrium point:

$$\dot{\delta n} = -\left[\frac{u^*}{\tau} + \frac{1-u^*}{L}\right]\delta n + \left[\frac{n^*}{L} - \frac{n^*}{\tau}\right]\delta u; \tag{B.3a}$$

$$\dot{\delta m} = \frac{1-u^*}{L}\delta n - \frac{1}{\tau}\delta m - \frac{n^*}{L}\delta u; \tag{B.3b}$$

$$\delta p = p_0(u^*\delta n + \delta m) + p_0 n^* \delta u. \tag{B.3c}$$

The preceding models are purely deterministic, having removed all randomness from the original Markov chain. For a more accurate description around the operating point we will introduce random noise, that results from a diffusion approximation

of the Markov chain dynamics. Formally, (see [18]) this noise process is the limit in distribution of

$$\sqrt{q}\left(\frac{N_q(t)}{q} - n(t), \frac{M_q(t)}{q} - m(t)\right)$$

where (N_q, M_q) is the scaled process mentioned before, and (n, m) the fluid limit. Again this is outside our scope, but we can motivate our noise model by reviewing the case of a Poisson process $a(t)$, such as the arrivals to our system. Its diffusion approximation satisfies the stochastic differential equation $da = \lambda dt + \sqrt{\lambda} dW$, where $W(t)$ is Brownian motion. More informally we can write equation

$$\dot{a} = \lambda + \sqrt{\lambda}\, w(t),$$

where $w(t)$ is unit white noise. In classical terms, the fluid model $\dot{a} = \lambda$ for Poisson arrivals is modified by additive white noise of power spectral density equal to the arrival rate itself. What we are looking for is the analogous modification to the model (B.3) to track the fluctuations of the process (n, m), locally around its equilibrium (n^*, m^*).

First, the Poisson arrivals will introduce a noise term $v_1(t) = \sqrt{\lambda}\, w_1(t)$ in (B.3a). The two departure terms in (B.2a) will also introduce noise terms, with power spectral density equal to the transition rate, evaluated at equilibrium:

$$v_2 = \sqrt{\frac{n^* u}{\tau}}\, w_2, \quad v_3 = \sqrt{\frac{n^*(1-u)}{L}}\, w_3;$$

here w_2, w_3 are independent unit white noises. With some algebra we can also rewrite the above as

$$v_2 = \sqrt{\alpha\lambda}\, w_2, \quad v_3 = \sqrt{(1-\alpha)\lambda}\, w_3, \tag{B.4}$$

$$\text{where} \quad \alpha := \frac{u^*}{\tau\left[\frac{u^*}{\tau} + \frac{1-u^*}{L}\right]} = P\left[\frac{\tau_k}{u^*} \leq \frac{L_k}{1-u^*}\right]$$

is the probability that a load finishes service before expiring its laxity. The interpretation of (B.4) is that departures of the n queue are equivalent to two Poisson processes, where the rate is "thinned" by the probability of, respectively, leaving the system and joining the m queue. The term v_3 will also appear as noise in arrivals to the dynamics (B.3b) for $m(t)$, and

$$v_4 = \sqrt{\frac{m^*}{\tau}}\, w_4 = \sqrt{(1-\alpha)\lambda}\, w_4$$

will represent noise in departures from this second queue. The resulting dynamics with noise is thus

$$\dot{\delta n} = -\left[\frac{u^*}{\tau} + \frac{1-u^*}{L}\right]\delta n + \left[\frac{n^*}{L} - \frac{n^*}{\tau}\right]\delta u + v_1 - v_2 - v_3, \tag{B.5a}$$

$$\dot{\delta m} = \frac{1-u^*}{L}\delta n - \frac{1}{\tau}\delta m - \frac{n^*}{L}\delta u + v_3 - v_4, \tag{B.5b}$$

$$\delta p = p_0(u^*\delta n + \delta m) + p_0 n^* \delta u. \tag{B.5c}$$

Appendix C

$\mathcal{H}_2$-optimal control

In chapter 4 we proposed a solution for designing the controllers using $\mathcal{H}_2$-optimal control. The motive of this appendix is to present a small background of the theory used for these solutions.

C.1. $\mathcal{H}_2$-NORM

We should start by defining the $\mathcal{H}_2$-norm of a MIMO system.

Definition C.1.1. Let $G(s)$ be a stable and strictly proper transfer matrix of dimension $p \times m$. The set of stable and strictly proper transfer matrices is denoted $\mathcal{RH}_2^{n_y x n_u}$. For any transfer matrix $G(s) \in \mathcal{RH}_2^{n_y x n_u}$, we define the $\mathcal{H}_2$-norm as:

$$\|G(s)\|_2 = \left(Trace \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} G(j\omega) G^*(j\omega) d\omega \right) \right)^{1/2},$$

or its equivalent in the time domain, by the use of Parseval's relation:

$$\|G(s)\|_2 = \left(\int_0^{\infty} g(t) g^T(t) dt \right)^{1/2} = \|g(t)\|_2,$$

where $g(t)$ is the impulse response of the system, and the last equality is the $\mathcal{L}_2$-norm of the function.

In the problems we are working our objective is to minimize the variance of the output:

$$\sigma^2(z) = E \left[\lim_{T \to \infty} \frac{1}{T} \int_0^T \|z(t)\|^2 dt \right],$$

which can be also be expressed in terms of the power spectral density (PSD) matrix

$$\sigma^2(z) = Trace \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} S_{zz}(j\omega) d\omega \right).$$

The basic idea behind using $\mathcal{H}_2$-optimal control is that if we have a system which its only input is a disturbance with constant PSD $S_{ww}(j\omega) = I$, the case for white noise, then the PSD of the output will be $S_{zz}(j\omega) = T_{zw}(j\omega)S_{ww}(j\omega)T_{zw}^*(j\omega) = T_{zw}(j\omega)T_{zw}^*(j\omega)$. Then if we minimize the $\mathcal{H}_2$-norm of $T_{zw}(j\omega)$:

$$\|T_{zw}\|_2^2 = Trace\left(\frac{1}{2\pi}\int_{-\infty}^{\infty} T_{zw}(j\omega)T_{zw}^*(j\omega)d\omega\right) \tag{C.1}$$

$$= Trace\left(\frac{1}{2\pi}\int_{-\infty}^{\infty} S_{zz}(j\omega)d\omega\right) \tag{C.2}$$

$$= \sigma^2(z), \tag{C.3}$$

we are also minimizing the variance of the output. So if we define the output as: the deviations from the nominal working point, the error at tracking, the control effort or any combination of them; by minimizing the $\mathcal{H}_2$-norm of the transfer function we also minimize variance of the desired output.

C.2. STANDARD $\mathcal{H}_2$ PROBLEM

The system we will consider is the one shown in Fig. C.1. The Plant G is a LTI system driven by white noise, w, and controlled in feedback by u. The $\mathcal{H}_2$ problem is defined as follows:

Definition C.2.1. The $\mathcal{H}_2$ control problem is to find a proper, real-rational controller K which stabilizes G internally and minimizes the $\mathcal{H}_2$-norm of the transfer matrix T_{zw} from w to z.

We will not cover here the solution to the general $\mathcal{H}_2$ problem (for the complete solution to this problem the reader can refer to [13]) but we will outline the solution

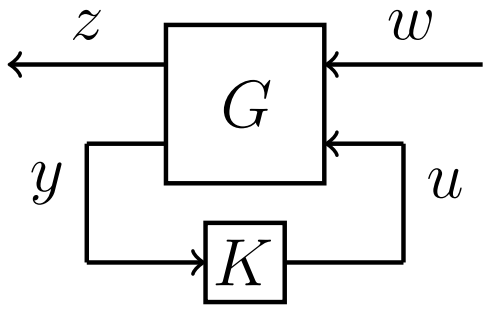


Figure C.1. Block diagram for the standard $\mathcal{H}_2$ problem

for the particular set up we are using. The transfer matrix G we consider here has the following form

$$G(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & 0 & 0 \end{array}\right]. \tag{C.4}$$

and we make the following assumptions:

1. (A, B_2) is stabilizable and (C_2, A) is detectable;
2. D_{12} has full column rank with $D_{12}^* D_{12} = I$ unitary;
3. $\begin{bmatrix} A - j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix}$ has full column rank for all ω.

Note that the noise only affects the state variables but the output and the variables for the feedback loop are measured without noise.

Under these hypotheses the optimal control strategy is constant gain state feedback:

$$u = -B_2^* X y,$$

where $X \geq 0$ is the solution to the corresponding Ricatti equation:

$$(A^* - C_1^* D_{12} B_2^*)X + X(A^* - B_2 D_{12}^* C_1) - X B_2 B_2^* X + C_1^*(I - D_{12} D_{12}^*) C_1 = 0. \tag{C.5}$$

C.3. $\mathcal{H}_2$-OPTIMAL CONTROLLERS

In the specific case we are studying the plants, G, are the ones defined in chapter 4, (4.4) the one for constant power case and (4.11) for the frequency regulation case. The first problem fits exactly in the $\mathcal{H}_2$ problem hypothesis because the only external signals are the 4 white noises given by the uncertainties in the loads and the outputs are the deviations from nominal power and the control signal. In the frequency regulation case one of the inputs is the regulation signal. To adapt this scenario to an $\mathcal{H}_2$ problem we modeled the regulation signal as filtered white noise, as explained in section 4.3.1, and we defined as outputs the tracking error and the control signal. After these modifications the problem fitted this hypothesis of the $\mathcal{H}_2$ problem.

Appendix D

Simulator

Along the research process we built a MATLAB based simulator which comes together with this document. This appendix was added to briefly explain how the simulator was done and how to use it.

The simulator is based on the model of Chapter 3 and was used to make the simulations of that Chapter onwards. It is constructed as a discrete-event simulator with the possibility of setting various scenarios and contrasting them.

D.1. SIMULATION ENGINE

The most important part of the simulator is the simulation engine of which their are several versions for different scenarios. All of them are named "mginf2*.m", where "*" is replaced for the different version. The engine was constructed as a discrete-event simulator. A discrete-event simulation models the operation of a system as a discrete sequence of events in time. Each event occurs at a particular instant in time and marks a change of state in the system, for instance load arrivals and departures. Between consecutive events, changes in the system can be calculated; thus the simulation can directly jump in time from one event to the next.

Each time an event happens the parameters of the aggregator and all the loads are updated. The simulator handles 5 types of events:

- New load arrival: Each time a load arrives its parameters are chosen at random depending on the mean value of the parameters and their distributions. The load is added to the existing loads and the next arrival time is computed and scheduled.
- Service completion of load with laxity: The load with no service remaining is

removed from the array of current loads and the statistics from its service are added to the array of completed jobs.

- Laxity expiration of loads with remaining service: The load with no laxity remaining but with pending service is moved from the array of loads with laxity to the array of loads with no laxity remaining.
- Service completion of load with no laxity: The load with no service remaining is removed from the array of loads and the statistics from its service are added to the array of completed jobs.
- New regulation signal value: Each time the SO sends a new regulation signal value the aggregator refreshes the control loop and broadcasts the corresponding signal depending on the algorithm used. Each load change its status or not, also depending on the algorithm in use.

The simulation starts by generating the initial array of loads with random values based on the chosen distribution and its parameters. Then the first arrival time is drawn and scheduled and the simulation starts. Each time an events happens a time stamp is created and all the state variables are updated. Then the next event is calculated and the simulation jumps to this event. Also at each event time traces are created for the relevant statistics, as explained below.

The time traces are used for the output of the simulation which includes:

- a time vector (T) with the time stamp of each event,
- the number of loads with laxity remaining after each event (X),
- the number of loads with no laxity remaining after each event(Y),
- the total consumed power after each event (P),
- the value of u after each event (U),
- the value of u_r after each event (U_r),
- the value of r and $\dot{r}$ after each event (R and R_dot),
- two vectors with the length off all completed loads with the number of maximum interruptions allowed for each load and the effective interruptions that each load suffered.

D.1.1 Main.m

The simulator runs from file "Main.m". This is the only file which the user has to interact with to run simulations. The user starts by setting "sim_name" which will be the name of the folder where the results of the simulation are saved.

Four groups of parameters follow: *global, loads, aggregator and simulation.* Global parametrs include the length of the simulation, "Tfinal", and the time step, "Dt". In the loads parameters the service time (τ), laxity (L), power (p) and number of interruptions ($\#_{maxint}$) are set. Each parameter is specified by setting its mean, distribution and variance if needed. The only parameter that is not set directly is laxity, which is indirectly set by the deferability factor (η). The third group are the aggregator parameters, this include the arrival rate (λ), the nominal service level (u^*), and the offered regulation (θ). In this section it is also possible to set the regulation signal from 4 different options: constant, sinusoidal, real signal o square wave. The amplitude of the signals was already fixed by θ. The last parameter to be set is the type of control being able to choose between the PI controller from Chapter 3 Eq. 3.9 or the $\mathcal{H}_2$-optimal control.

After setting the parameters 4 different types of simulations are offered.

- Time simulation: a single run of the aggregator is simulated under the parameters set. Output include ploting of power and control signal and a text file with all the settings a results including performance metrics.
- Parameter simulation: this mode if for comparing the effect of a parameter in the system. A range of values is chosen for a given parameter and several simulations are ran for each value to compare. The parameters that can be selected to analyze are: θ, k_1, a, η, u^* and $\#_{maxint}$. Output includes plotting of the different performance metrics (error, control effort, PJM score and number of interruptions) for each value fo the analyzed parameter. Also a text file with the results of each individual run and settings of the simulations is created.
- Real simulation: this simulation is very similar to the time simulation with the difference that the arrival rate λ is not constant. More details on this simulation is provided on Chapter 5, section 5.6.
- 2-parameters simulation: this mode is similar to parameter simulation with the difference that 2 different parameters can be compared at the same time.