NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Distributed Energy-Saving Algorithms for Wireless Networks

by

Javier Zazo Ruiz

Master Thesis

Supervisor: Prof. Douglas J.Leith

Hamilton Institute

National University of Ireland

Maynooth

Co. Kildare

October 2012

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Master of Science is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: _____

Date: 30th of October, 2012.

Contents

List of Figures

List of Tables

Abstract

The rapid growth of wireless networks has led to increasing interest in designing new algorithms that can efficiently reduce the energy consumption of routers and other devices. We present a new formulation of the Network Flow problem that takes into account the energy consumption of the data flows, and reduces the overall network energy expenditure.

We introduce an energy model for wireless connections and analyse its validity with real measurements. Then we propose a convex optimization problem that establishes energy constraints on the links, and encourages energy savings that induce sparsity (shut-off of links). We propose several algorithms that can be computed in a distributed fashion for different types of capacity constraints. Finally we justify the sparsity of the solution by using the theory of proximal methods and present simulations for different scenarios. Our algorithms have application both in wired networks as well as in TDMA and 802.11 wireless networks.

Chapter 1

Introduction

Energy saving is currently a subject of much interest, with efforts to reduce the energy consumption of Communication Technologies being targeted at both infrastructure and user devices. New protocols are being devised to increase the battery life of mobile terminals, and also algorithms that increase the efficiency of wired and wireless networks by using only as much energy as is needed. With this in mind, opportunities for saving energy are particularly great during idle periods of communication, or when traffic is low. These periods can be used to place the network elements to a lower energy consumption state, or sleep, and reduce the energy expenditure from the case when the device is always on.

In this thesis we consider the problem of how to allocate the traffic from users in networks with multiple routing paths. We analyse the potential energy savings that can be achieved while maximising network utility for users. We formulate this task as a convex optimization that allows a distributed implementation with network cooperation. The proposed distributed algorithm maximizes a utility function for every user [20], while respecting network capacity constraints and minimising the energy costs associated with data transmission.

Capacity constraints are studied for different network technologies, specifically we focus on constraints for wired networks, wireless TDMA (Time Division Multiple Access), and 802.11 networks. The first type of constraints are linear in

form, with traffic rates bounded by a capacity threshold. The 802.11 constraints use a Markovian model of successful packet transmissions, collisions, and idle periods that allow for a log-convex representation of the requirements [21, 22]. Both cases are analysed and solved in a distributed fashion.

We analyse energy constraints for different networks, producing a linear model that can faithfully represent the energy use of wireless networks such as 802.11 routers, UMTS or LTE basestations, and femtocell routers. We analyse real measurements of these technologies to validate the model, and explore the literature [31, 16, 14, 18]. This model is then included as a loss function in the network flow problem.

These energy constraints are specified as the norm of flows that run through links, which are added as penalties. The kind of norm chosen for the links influences the flows that are penalized in favour of others. We explore the effects that the l_1-norm encourages as well as the l_1/l_∞-norm exploiting group graph structure [4]. Our interest here is in solutions that encourage sparsity, meaning a complete shut-off of links, as well as to limit user rates so as to allow energy saving on underused links. Finally we present simulations to evaluate the algorithms convergence and performance.

Other works that study energy savings specifically for wireless networks are [10, 25, 29, 30], who consider cell zooming and size scaling of the cell size to adapt dynamically to traffic demands and save energy by shutting down redundant base stations. In [12] they consider the problem of sleep and wake-up transients of basestations, to dynamically react to user demands.

Regarding the problem of a whole network, [2] studied the routing problem of provisioning guaranteed flow rates for a given demand matrix while minimizing energy consumption. They analyse different models of energy consumption, linear and polynomial, and show that for most other functions it is a NP-hard problem. In a different work [1], the same authors present a routing and periodic scheduling algorithm, that given a network and traffic matrix, minimizes switching among routers and delays among packets.

In [15] the authors present a minimum edges routing problem of a network. Given a set of demands and capacity constraints on the links they show this is an NP-hard problem and it is impossible to find a polynomial-time constant-factor approximation algorithm to solve it. After this statement they present heuristics to find energy-efficient routing, consisting of balancing load from the links.

In [28], the authors present two forms of power management that reduce energy consumption over networks, either adapting the rate of network operation, or putting network elements to sleep during idle times with small and controlled increase in latency. With a similar idea but applied to LTE cells, the authors in [14] propose discontinuous transmission (DTX) on the base station side. This methodology would allow significant energy reductions in a lightly used LTE network without damaging service availability.

Our contribution in this area is to present fair solutions to the multipath problem considering power consumption of links and routers. Our algorithms optimize a whole network in a distributed fashion, choosing the traffic routes that minimize energy, while providing some traffic utility to every user, e.g. proportional fairness, rather than simply meeting a specified traffic demand.

Chapter 2

Energy Usage by Wireless Routers

2.1 Introduction

We present in this chapter an overview of energy measurements of wireless networks and analyse the validity of a linear model for these cases. More specifically, we use results from [31, 16] corresponding to the analysis of a femtocell transmitting router and a 802.11 Access Point, respectively.

There are not many other studies available in the literature that offer measurements for wireless networks. Apart from the ones already mentioned and their included references, [14] considers power consumption in 3G and LTE Networks and presents a linear model similar to the one in Figure 2.1. It discusses the possibility of DTX (Discontinuous Transmission) for existing HSPA and LTE networks and the corresponding energy savings, and analysed the potential savings in data acquired from 300 cells in a large European city.

Regarding 802.11n wireless networks, in [18] they analyse the energy consumption of MIMO routers controlling different aspects for transmission such as channel width, transmit power, rates, antenna selection, *etc.* In their results

they show that adding antennas to achieve higher rates changes the total power consumption to a high degree, but this is not so significant with a fixed selection of antennas. In addition, they analyse the heuristic of transmitting at highest speed and then racing to sleep, concluding that due to the increased power when using extra antennas, it might not be the optimal strategy.

More studies consider the energy savings from disconnecting router elements in idle states, or reducing transmission rates to save energy. For instance, [28] proposes an algorithm that buffers arriving data, and uses the long gaps from these acquisitions to set network interfaces in sleep mode while establishing a controlled delay for packets, an approach they call "*buffer-and-burst*". The article also proposes another alternative based on reducing traffic rate transmissions which can also deliver substantial savings. One reason they mention is the lower consumption of electronic components when operating more slowly, but also the possibility of using Dynamic Voltage Scaling (DVS) in lower operating frequencies, which also reduces power consumption. They measure energy saving gains in percentage of disconnected elements in time, and in rate reductions from the average for each method respectively.

2.2 Energy Model

We consider the following simple model of wireless device energy consumption,

$$P = w(d)sat(x, d) + v(d)H(x) + c \tag{2.1}$$

where P is the device power consumption in Watts, x is the offered load in Mbps, d is the datagram size in bytes and

$$sat(x,d) = \begin{cases} x & 0 \leq x \leq x_{max}(d) \\ 0 & x \leq 0 \\ x_{max}(d) & x > x_{max}(d) \end{cases} \tag{2.2}$$

$$H(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

In this model, $w(d)$ captures the dependence of power consumption on offered load, $v(d)$ the dependence on datagram length when the offered load is held fixed and c captures the baseline power consumption when the device is idle (powered on but no data transmitted or received). This is illustrated in Figure 2.1, where the "fixed-term" refers to $c + v(d)H(x)$ and the "variable term" refers to $w(d)sat(x,d)$.

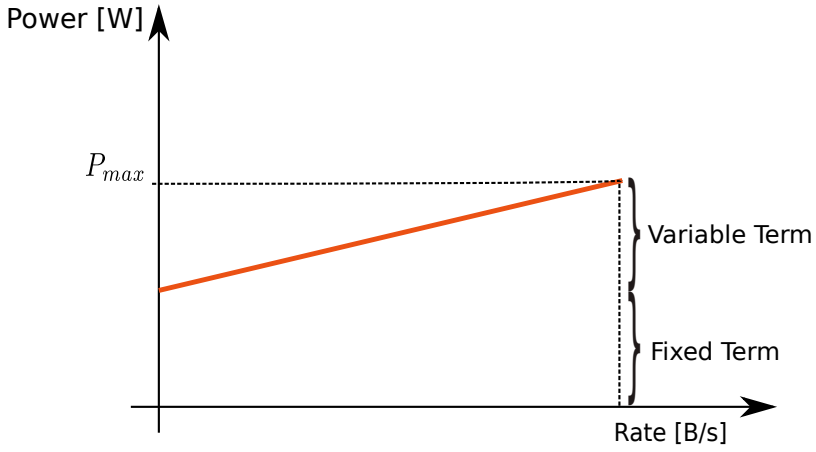


Figure 2.1: Energy Model

2.3 Femtocells

We report detailed measurements of the electrical energy consumption of a commercial 3G femtocell base station. The work in this section is based on data from [31].

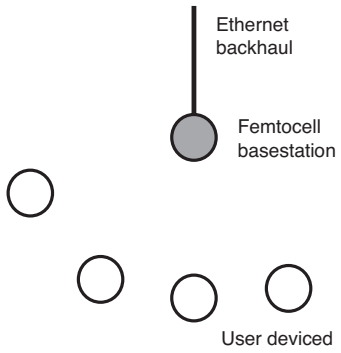


Figure 2.2: Network topology

2.3.1 Experimental Setup

2.3.1.1 Network setup

The test environment is composed of a single femto–cell and up to four end–user devices, see Figure 2.2.

The femtocell basestation is an Alcatel–Lucent device (model 9361 Home Cell V2-V). The femtocell acts as a standard 3G basestation and uses a SIM card which is active on the cellular network and that has been registered for use with the femtocell. The femtocell basestation is equipped with an Ethernet port that must be connected to a suitable broadband connection to provide backhaul access to the network operator. In the experiments the femtocell is connected to the campus network. During bootstrap the femtocell establishes an encrypted VPN connection to the network operator which is used to carry all the traffic to/from end–user devices in the cell. This traffic may include voice/video calls and data transfer sessions. The femtocell supports up to four simultaneous end–user devices.

End-user devices studied included one mobile broadband modem and three mobile phones. The mobile broadband modem used during the measurements is a Huawei K3770. This device supports HSUPA/HSDPA/UMTS standards on the 2100 MHz/900 MHz bands and the GSM/GPRS/EDGE standards on the 850/900/1800/1900 MHz bands. The device is rated for 2 Mbps HSUPA and 7.2 Mbps HSDPA date service. Mobile phones were Samsung model Galaxy S2.

2.3.1.2 Measuring electrical energy consumption

A custom Energino instrument was used to measure the electrical energy consumption of the femtocell basestation. Energino is a plugload meter designed to monitor the energy consumption of DC devices. It consists of a hardware and a software components both based on the Arduino platform. A management backend written in Python is used to configure Energinos operating parameters, *e.g.* sampling rate and resolution, to turn the monitored device on/off, and to gather the energy consumption statistics. Energino supports sampling rates up to 10 KHz and measures electrical power with an accuracy of approximately 1 mW. See [16, 17] for further details. In the experiments the Energino instrument was located between the electrical power plug of the basestation and the wall socket, and so measures the power consumption of the complete basestation.

2.3.1.3 Traffic generation

Several types of traffic were generated on the end-user devices, including 3G voice calls, SMS, MMS, 3G data (youtube, browsing) and CBR and VBR UDP data traffic. UDP data traffic was generated using iperf, with traffic transmitted from the end-user device to a public machine.

2.3.2 Measurements

In the following sections we present representative measurements of energy consumption when the basestation is idle (powered on but with no end-user devices associated to the cell), as the offered load is varied and as the datagram size used is varied.

2.3.2.1 Basestation idle

Figure 2.3 shows measured electrical power consumption vs time when the basestation is idle. It can be seen that the power consumption consists of a baseline

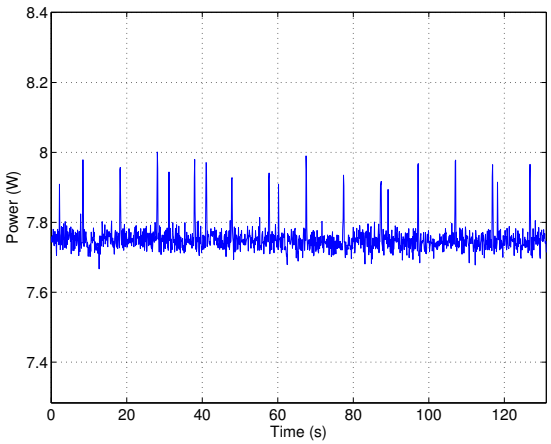
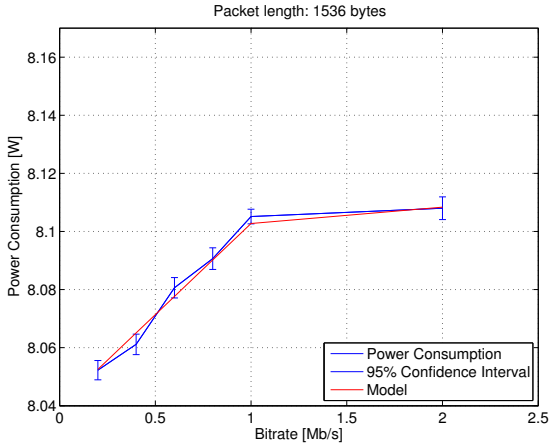


Figure 2.3: Measured power consumption of the femtocell basestation when idle (no end-user devices associated).

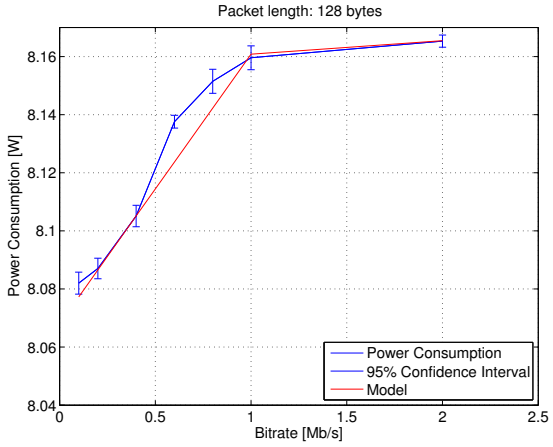
value of around 7.77 W (with some variability around this mean value), periodic spikes (with period 10s) and a number of less regular spikes. From inspection of tcpdump traces on the wired backhaul link, we find that the spikes in power are correlated with communication on this link and so appear to be related to network management functions.

2.3.2.2 Energy consumption vs offered load

Figure 2.4 presents measurements of the mean power consumption of the femtocell basestation when a single end-user device is associated and is transmitting UDP data traffic. Results are shown of power consumption as the offered load is varied. Figure 2.4 (a) shows measurements when the UDP datagram size is 1536 B, while Figure 2.4 (b) shows measurements when the UDP datagram size is 128 B. Also indicated are the 2σ confidence intervals.



(a)



(b)

Figure 2.4: Average power consumption at the Femtocall as a function of the bitrate size for a constant datagram length. Packet loss was lower than 5% for bitrates equal to or lower than 1 Mbps. The femtocell is acting as receiver.

It can be seen that the power consumption increases with offered load before reaching a plateau. For a given offered load, the power consumption is uniformly higher for the small datagrams than for the large datagrams, *e.g.* at 0.5 Mbps the mean power consumption is 8.07 W with 1536 B datagrams and 8.12 W with 128 B datagrams. The rate of increase with offered load is also somewhat higher with smaller datagrams.

The plateau in power consumption correlated with the offered load reaching the network capacity. With 1536 B datagrams the maximum network through-

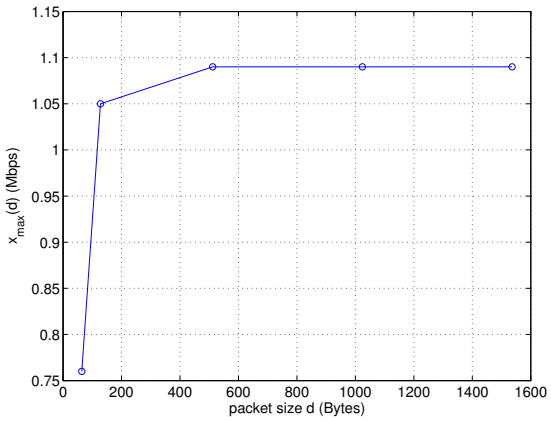


Figure 2.5: Measured maximum network throughput vs datagram size.

put is observed to be 1.09 Mbps – at offered loads above this level, significant packet loss is observed and the net goodput remains constant at 1.09 Mbps. With 128B datagrams, the maximum network throughput is observed to be 1.05 Mbps. Measured values for other datagram sizes are shown in Figure 2.5. It can be seen that the maximum throughput increases monotonically with datagram size, and is significantly reduced at the smallest datagram size of 64 B. This is as expected, since fixed network overheads (framing, ARQ and control overheads *etc*) are amortised across more data bits as the datagram size is increased.

2.3.2.3 Energy consumption vs datagram size

Figure 2.6 plots measurements of power usage vs UDP datagram size. Figure 2.6(a) shows data when the offered load is 0.2Mbps and Figure 2.6(b) shows the corresponding data when the offered load is 1Mbps. Also indicated are the 2σ confidence intervals. It can be seen that the power consumption tends to decrease as the datagram size is increased. As the datagram size is increased, the number of datagrams sent per second decreases when the offered load in Mbps is held fixed. Per datagram overheads (framing, ARQ, *etc.*) are therefore reduced and presumably this is the source of the reduction in power consumption. Observe that the power consumption appears to rise again for datagrams

above about 1470 B. We believe that this is due to fragmentation of these larger datagrams – from separate downlink tcpdump measurements, we estimate the wireless link MTU to be 1368 B.

2.3.2.4 Energy Model

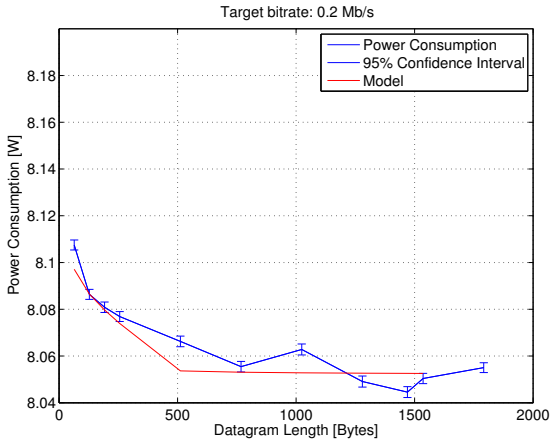
From Figure 2.3 parameter c is approximately 7.77 W. Figure 2.5 gives the measured values for x_{max}. When the dependence of $w(d)$ on datagram size is primarily due to the contribution of fixed overheads per datagram (framing *etc*, as already noted), we can select

$$w(d) = \omega_0(1 + \frac{\omega_1}{d}) \tag{2.4}$$

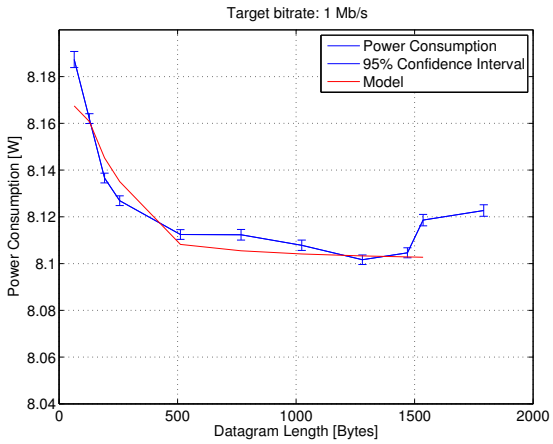
where ω_0, ω_1 are parameters. ω_1 can be thought of as the per datagram overhead, specified in bytes, while ω_0 is a factor converting between units of bytes and energy.

Using this choice of structure for $w(d)$, we find that with the parameter values given in Table 2.1 and the $v(d)$ values shown in Figure 2.7 this simple model provides a good fit to the measurements across the full range of operating conditions considered. For example, the model energy consumption predictions are indicated by the red lines in Figures 2.4 and 2.6. Similar predictive accuracy is obtained for other datagram sizes and offered loads.

Observe that structure of the proposed model is simple yet intuitively reasonable. c and x_{max} can be directly measured. The function $w(d)$ varies in accordance with fixed overheads. From Figure 2.7 it can be seen that $v(d)$ decreases with increasing datagram size until a datagram size of around 1300B is reached and then increases again. This increase is consistent with the onset of fragmentation commented upon previously.



(a) 0.2Mbps



(b) 1Mbps

Figure 2.6: Average power consumption at the Femtocall as a function of the datagram size for a constant traffic generation rate of 1Mb/s. Packet loss is lower that 2% for all measurements, except for the one with datagram length set to 64 bytes where packet loss is 26%. The femtocell is acting as receiver.

Parameter	Value
ω_o	0.06 W/Mbps
ω_1	70 B
c	7.77 W

Table 2.1: Energy model parameter values for femtocell basestation

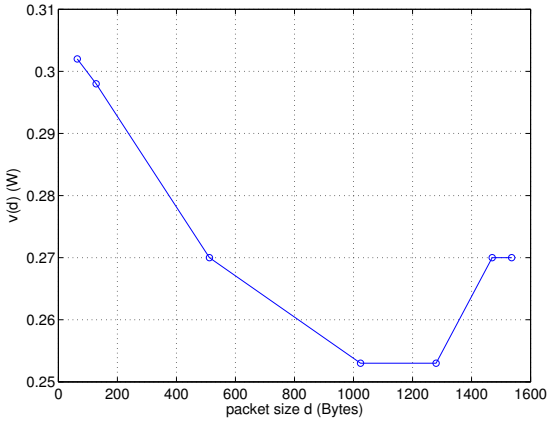


Figure 2.7: Energy model $v(d)$ values for femtocell basestation.

2.4 802.11

We present here an analysis of energy consumption in 802.11g networks, and how this data fits in the model from the previous section. The data for this analysis has been taken from [16], where we tried to motivate our energy model from the measurements in their figures.

2.4.1 Experimental Setup

The authors from [16] used a setup consisting of one computer transmitting to an Access Point (AP) connected to a plug load meter and to the power grid. Figure 2.8 shows their setup. They performed tests both when the AP was in receiver mode and in transmitter mode, but we will only refer to results from the later.

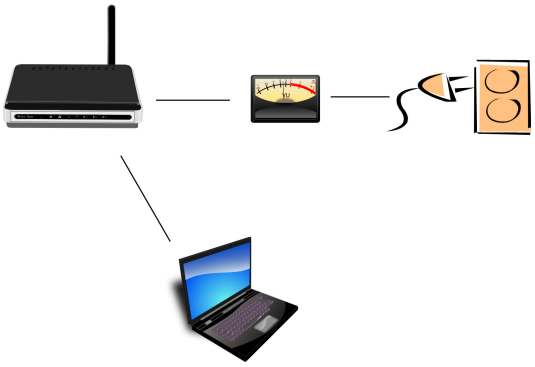


Figure 2.8: 802.11 Measurement Topology

The measurement testbed consisted of an Access Point built around a PCEngines ALIX 2C2, two processor board 802.11a/b/g wireless interfaces (Atheros AR5123A chipset), operating system OpenWRT 10.3.01-rcl and *MadWifi* Wireless NIC driver. The computer was a DELL D630 notebook equipped with a wireless adapter Atheros AR5212 chipset. Measurements were taken on frequency of operation 2.412 GHz (channel 1), rate control was set on auto and transmission power to 18dBm (˜63.1mW).

Traffic generation was performed at the AP using the freely available traffic generator Multi-Generator MGEN, that can inject both TCP and UDP traffic. Finally, power consumption was measured using a plug load *Watts up?* and connected through a USB port to acquire the measurements.

2.4.2 Model validation

Authors from [16] carry out a number of experiments to analyse the average power consumption behaviour of the AP in different scenarios. Namely, changing packet size while keeping the bitrate constant, and changing rates with fixed packet length.

The first observation is that the power consumption level in an idle state is a constant 5.3 W. This corresponds in our model to the defined constant c, and can be checked in all of their figures.

Furthermore, considering the test that analyses traffic load vs consumed

power for fixed packet size of 1280 bytes, a linear relation can be obtained from their data. More specifically, we present our fitted curve with the estimated data from their graphs, and show our model in Figure 2.9.

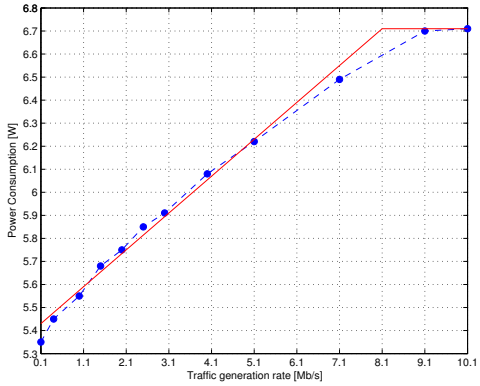


Figure 2.9: Average power consumption at the AP as a function of traffic rates, and constant message size of 1280 bytes.

The model parameters we found to fit this data are presented in Table 2.2. Additionally, we used values from Figure 2.10 for parameter $v(d)$, and $x_{max} = 6.7\,W$ was extracted from their figure (valid when transmitting 1280 B packets).

Parameter	Value
ω_0	0.15 W/Mbps
ω_1	70 B
c	5.3 W

Table 2.2: Energy Model parameter values for 802.11 AP.

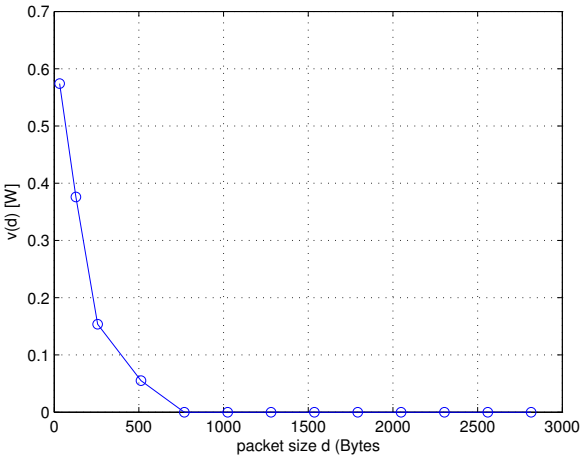


Figure 2.10: Energy model $v(d)$ values for 802.11 AP.

Another test analysed power consumption vs message size for a fixed transmission rate. They considered two transmission rates, 100 KB/s and 1 Mb/s. We exported their data to Figure 2.11, and fitted our model to their curves. Our curves reasonably represents this data.

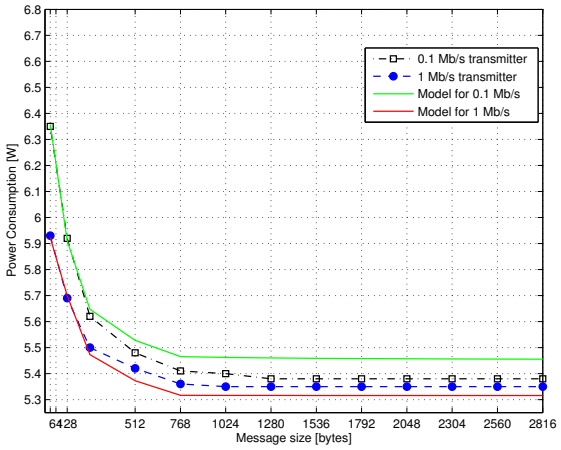


Figure 2.11: Average power consumption for different message sizes, and constant transmission rates.

Finally, in another experiment [16] considered two different transmission power levels, i.e. 10 and 18dBm for fixed packet size and transmission rate.

In this case they did not observe relevant power consumption variations for the constant message size (1280 bytes). This fact also confirms that our linear model does not need to take this aspect into account.

Chapter 3

Fair Energy-Optimal Scheduling

In this chapter we address the problem of fair rate control and routing in communication networks while taking into account the energy consumption of links and routers. We formulate this task as a utility-fair optimization problem that includes an energy cost component associated with the router's consumption (or link interfaces).

3.1 Problem Setup

3.1.1 Network Model

We consider a network with a set of stations N connected by a set of links L. We model this by a graph $G : (N, L)$. We let $\mathcal{F}$ denote the set of flows carried by the network, where each flow $f \in \mathcal{F}$ has a source, a destination, a route r_f consisting of a set of links in L connecting the source and destination, and a transmission rate x_f. We can summarise this routing information using a binary matrix $\mathbf{A}$, with rows indexed by flow identifiers and columns indexed by link identifiers. That is, the $(f, l)^{th}$ element $a_{fl} = 1$ if link $l \in r_f$, and zero

otherwise. We also define for each link $l \in L$ the set $F_l := \{f : f \in \mathcal{F}, l \in r_f\}$ consisting of the flows that use the link.

We model the network link capacity constraints using inequalities of the form

$$g_l(\mathbf{x}) \leq 0, \qquad l \in L \tag{3.1}$$

where $\mathbf{x}$ denotes the vector of network flow rates. The precise form of function $g_l(\cdot)$ depends on the characteristics of the link considered. We will analyse the following options.

Wired Networks

In wired networks, the router interfaces have a limit on the total rate they can transmit, where all flow rates in the link add towards that limit. We can express such restrictions as the linear constraints,

$$\sum_{f \in F_l} x_f \leq c_l, \quad l \in L \tag{3.2}$$

That is, $g_l(\mathbf{x}) := \sum_{f \in F_l} x_l - c_l$. Equivalently, in matrix form we have

$$\mathbf{A}^T \mathbf{x} \leq C \tag{3.3}$$

where C is the vector with elements consisting of the link capacities c_l, $l \in L$ and $\mathbf{A}$ is the routing matrix defined earlier.

TDMA Wireless Networks

In wireless networks that use Time Division Multiple Access (TDMA) the routers communicate on assigned slots which are decided by a scheduler, each flow getting an allocated slice of time to forward the information. Suppose the network consists of a set of cells L, with a single scheduler controlling each cell, and consider a scheduler that assigns a time slice $T_{f,l}$ in a round robin fashion for each flow f and wireless cell l, satisfying the constraint $\sum_{f \in F_l} T_{f,l} \leq T_l$ where T_l

is the period of schedule. Let the rate of transmission of flow f in cell l be given by $w_{f,l}$ in symbols/second, which is determined by the modulation, spectral bandwidth and coding used for the transmitting signal in the specified cell. Therefore, given the rate of transmission, $x_f = w_{f,l}T_{f,l}$ encoded symbols are sent through the link corresponding to flow f on every slot time. Considering the TDMA scheduling for all flows, we arrive at the following constraint per wireless cell

$$\sum_{f \in F_l} \frac{x_f}{w_{f,l}} \leq T_l, \quad l \in L \tag{3.4}$$

That is, $g_l(\mathbf{x}) := \sum_{f \in F_l} \frac{x_f}{w_{f,l}} - T_l$. Observe that this is similar in form to equation 3.2

802.11 Wireless Mesh Networks

Consider a wireless network composed of a set of 802.11 e/n WLANs. Within each WLAN we divide time in MAC slots, where each slot can be a physical idle period, a successful transmission, or a colliding transmission. Following [21, 22], we define the following:

N_l Set of stations in a WLAN l

L Set of WLAN cells

τ_i Probability that the station i is attempting transmission

$P_{succ,i} = \tau_i \cdot \prod_{j \neq i}(1 - \tau_j) \qquad i \in N$

$P_{idle} = \prod_{j \in N}(1 - \tau_j)$

$P_{coll} = 1 - \sum_{i \in N} \tau_i \cdot \prod_{j \neq i}(1 - \tau_j) - \prod_{j \in N}(1 - \tau_j)$

$E_s = P_{idle} \cdot \sigma + P_{coll} \cdot T_{coll} + \sum_{i \in N_c} P_{succ,i} \cdot T_{succ,i}$ Average time slot duration

T_{coll} the duration of a collision

$T_{succ,i}$ the duration of a successful transmission by station i

F_i is the set of flows going through station i, $F_i := \{f : f \in \mathcal{F}, i \in r_f\}$

d_i is the number of bits sent in a successful transmission by station i (datagram size)

The mean throughput of station i in cell l is given by

$$\begin{aligned}
&\sum_{f\in F_i} x_f = \frac{P_{succ,i}}{E_s} d_i \\
&= \frac{\tau_i \prod_{j\neq i}(1-\tau_j)\, d_i}{\prod_{j\in N_l}(1-\tau_j)\,\sigma + \left(1 - \sum_{i\in N_l}\tau_i \prod_{j\neq i}(1-\tau_j) - \prod_{j\in N_l}(1-\tau_j)\right) T_{coll} + \sum_{i\in N_l} T_{succ,i}\left(\tau_i \cdot \prod_{j\neq i}(1-\tau_j)\right)} \\
&= \frac{\frac{\tau_i}{1-\tau_i}\prod_{j\in N}(1-\tau_j)\, d_i}{\prod_{j\in N_l}(1-\tau_j)\,\sigma + \left(1 - \sum_{i\in N}\frac{\tau_i}{1-\tau_i}\prod_{j\in N_l}(1-\tau_j) - \prod_{j\in N_l}(1-\tau_j)\right) T_{coll} + \sum_{i\in N_l}\left(\frac{\tau_i}{1-\tau_i}\prod_{j\in N_l}(1-\tau_j)\right) T_{succ,i}} \\
&= \frac{\xi_i d_i}{\sigma + \left(\frac{1}{\prod_{j\in N_l}(1-\tau_j)} - \xi_i - 1\right) T_{coll} + \sum_{i\in N_l}(\xi_i)\, T_{succ,i}} \\
&= \frac{\xi_i \frac{d_i}{T_{coll}}}{\frac{\sigma}{T_{coll}} - 1 + \prod_{j\in N_l}(1+\xi_j) + \sum_{i\in N_l}\left(\frac{T_{succ,i}}{T_{coll}} - 1\right)\xi_i} = \boxed{\frac{d_i}{T_{coll}}\frac{1}{\chi_l}\xi_i} \qquad i \in N_l
\end{aligned} \tag{3.5}$$

where $\xi_i = \frac{\tau_i}{1-\tau_i}$ is a normalized transmission probability, and $\chi_l = \frac{\sigma}{T_{coll}} - 1 + \prod_{j\in N_l}(1+\xi_j) + \sum_{j\in N_l}\left(\frac{T_{succ,j}}{T_{coll}} - 1\right)\xi_j$ is independent of the particular station.

The network capacity constraints can therefore be written as

$$\sum_{f\in F_i} x_f \leq \frac{d_i}{T_{coll}}\frac{1}{\chi_l}\cdot\xi_i, \qquad i \in N_l \tag{3.6}$$

These constraints are non-convex, but taking logs of both sides and changing variables yields

$$\log\sum_{f\in F_i} e^{\tilde{x}_f} \leq \log\frac{d_i}{T_{coll}} - \log\chi_l + \tilde{\xi}_i \tag{3.7}$$

where $\tilde{x}_f = \log x_f$, $\tilde{\xi}_i = \log\xi_i$. That is,

$$g_i(\tilde{\mathbf{x}}) := \log\sum_{f\in F_i} e^{\tilde{x}_f} - \log\frac{d_i}{T_{coll}} + \log\chi_l - \tilde{\xi}_i \leq 0 \tag{3.8}$$

Since $\log\chi_l$ is convex in $\tilde{\xi}$ [21, 22] and the log of a sum of exponentials is convex [9], this constraint is convex.

For a WLAN with two stations, Figure 3.1 plots the set of achievable rates

when $T_{succ} = T_{col} = 10\sigma$.

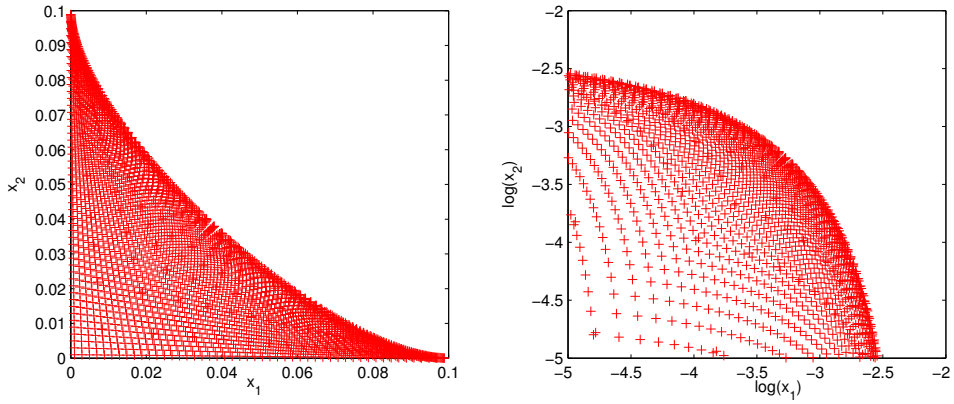


Figure 3.1: Non-convex rate, and log-convex transformation

3.1.2 Utility-Fairness

We consider the family of utility fair objective functions introduced by Kelly [20],

$$U(x) := \begin{cases} \frac{x^{1-\beta}-1}{1-\beta} & x \geq 0,\ \beta \geq 0,\ \beta \neq 1 \\ \log(x) & x \geq 0,\ \beta = 1 \end{cases} \tag{3.9}$$

These functions are strictly concave and continuously differentiable over x.

We collect all flows sharing the same source and destination into a bundle, and let $\mathcal{B}$ denote the set of such bundles. We associate each bundle with a user, and seek to achieve appropriate fairness between the aggregate rates allocated amongst bundles. That is, we seek to maximise the sum-utility $\sum_{b\in\mathcal{B}} U\left(\sum_{f\in b} x_f\right)$. See Figure 3.2 for an example, where the flow throughputs x_1 and x_2 belong to the same bundle (user), and x_3 belongs to a different one.

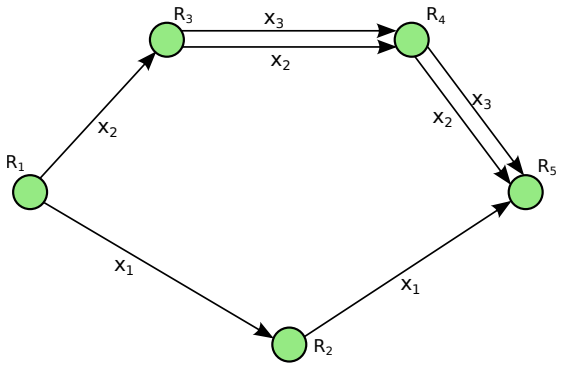


Figure 3.2: Example of Network with multipath flows

We will generally focus on the special case of proportional fairness, corresponding to a choice of $\beta = 1$ and sum-utility $\sum_{b\in\mathcal{B}} \log\left(\sum_{f\in b} x_f\right)$.

3.1.3 Energy Model

We let $P_l(\mathbf{x})$ denote the energy usage of link $l \in L$, which depends on the flow rates $\mathbf{x}$. We have seen in Chapter 2 that a linear model can be used to represent the energy consumption of links and routers when used for transmission, consisting in a fixed cost plus a term proportional to the rate when the link is on. The model captures the behaviour of a range of different technologies.

When $P_l(\cdot)$ is non-convex, the utility fair optimisation problem typically becomes difficult. For this reason, we will restrict our analysis to the case where $P_l(\cdot)$ is convex. For example, in equation 2.1 we neglect the "fixed term" $c+v(d)\delta$ and use

$$P_l\left(\mathbf{x}\right) = w_l \sum_{f\in F_l} x_f = w_l \left\|\mathbf{x}_{F_l}\right\|_p \tag{3.10}$$

where $\mathbf{x}_{F_l}$ denotes the vector with elements x_f, $f \in F_l$, and parameter w_l corresponds to the energy variation proportional to traffic flow. The saturation term of equation 2.1 can still be included as a separate inequality constraint in the problem formulation, as it is convex.

In addition to using the $l_1 - norm$ as an energy cost, we will also consider a number of different norms. For this reason, we introduced parameter p in

equation (3.10) to have a general representation. Specifically, the $l_\infty - norm$ defined as $\|\mathbf{x}\|_\infty = \max_i |x_i|$ adds a cost equal for all flows in the link equal to the maximum flow. Applying this norm to every link has the effect of generating sparsity among groups, while encouraging equality among the flows in the same group (they pay the same price). More generally, these norms are referred as $l_1/l_\infty - norm$ in the literature [5], for the linear cost among the groups, and maximum cost among flows within the group. In the following chapters, we will use the notation $l_1/l_q - norm$ for the general analysis, and particularize for the $l_1 - norm$ and $l_1/l_\infty - norm$ when necessary. Equation 3.10 is the penalty we will use in Chapter 4, which is convex.

3.2 Utility-fair Optimization

We are now in a position to formulate the following utility-fair energy optimization problem

$$\mathbf{P1}: \qquad \max \sum_{b \in B} U\left(\sum_{f \in b} x_f\right) - \gamma \sum_{l \in L} P_l(\mathbf{x}_{F_l}) \tag{3.11}$$

$$s.t. \quad g_l(\mathbf{x}) \leq 0 \qquad\qquad l \in L \tag{3.12}$$

$$\mathbf{x} \geq 0 \tag{3.13}$$

where γ is a price per energy unit, equation 3.12 are the network capacity constraints and equation 3.13 constrains all flow rates to be non-negative.

Observe that in equation 3.11 the energy cost $P_l(\mathbf{x}_{F_l})$ is treated as a penalty.

An alternative formulation is **P2**,

$$\mathbf{P2}: \qquad \max \sum_{b \in B} U\left(\sum_{f \in b} x_f\right) \tag{3.14}$$

$$s.t. \quad P_l(\mathbf{x}_{F_l}) \leq s_l \qquad l \in L \tag{3.15}$$

$$g_l(\mathbf{x}) \leq 0 \qquad l \in L \tag{3.16}$$

$$\mathbf{x} \geq 0 \tag{3.17}$$

where the energy requirements are now introduced as constraints.

Under mild assumptions and convexity, both problems have the same solutions when parameters γ and s_l are selected appropriately.

Proof. Assuming convexity for **P1** and **P2**, consider the (reduced) formulation of **P1** as

$$\max_{x_i} \sum_{b} U\left(\sum_{i \in b} x_i\right) - \gamma \sum_{l \in L} P_l(\mathbf{x}_{F_l})$$

where $\mathbf{x}_{F_l} = \left\{ [x_i \dots x_k]^T \, | i, ..., k \in F_l \right\}$ is the vector whose components are the rates going through link l.

Differentiating the unconstrained problem from above, we get

$$U'\left(\sum_{i \in b} x_i\right) - \gamma \sum_{l \in r_i} \frac{\partial}{\partial x_i} P_l(\mathbf{x}_{F_l}) = 0 \quad i \in N \tag{3.18}$$

and solving the system of equations we get the optimal solution, which we call $x_i^{\star 1}$.

The (reduced) formulation of **P2** is

$$\max_{x_i} \sum_{b} U\left(\sum_{i \in b} x_i\right)$$

$$s.t. P_l(\mathbf{x}_{F_l}) \leq s_l \qquad l \in L$$

and we choose $s_l = P_l(\mathbf{x}_{F_l}^{\star 1})$, whose components are the optimal solution of P1.

The Lagrangian and KKT conditions are

$$L = -\sum_b U\left(\sum_{i\in b} x_i\right) - \sum_l \lambda_l \left(P_l(\mathbf{x}_{F_l}) - P_l(\mathbf{x}_{F_l}^{\star 1})\right)$$

$$\frac{\partial L}{\partial x_i} = -U'\left(\sum_{i\in b} x_i\right) + \sum_{l\in r_i} \lambda_l \frac{\partial}{\partial x_i} P_l(\mathbf{x}_{F_l}) = 0 \qquad i \in N \tag{3.19}$$

$$\lambda_l \left(P_l(\mathbf{x}_{F_l}) - P_l(\mathbf{x}_{F_l}^{\star 1})\right) = 0 \qquad l \in L \tag{3.20}$$

We can see that the values $\mathbf{x}_{F_l}^{\star 2} = \mathbf{x}_{F_l}^{\star 1}$ and $\lambda_l^\star = \gamma$ for all l, satisfy equations 3.19 and 3.20 by comparison with 3.18, and therefore it is an optimal solution.

□

Chapter 4

Distributed Algorithms for Energy-Optimal Scheduling

4.1 Convex Optimisation

When the utility function $U(\cdot)$ is concave and the network constraints $g_l(\cdot)$ are convex, **P1** is a convex optimisation problem. When the interior of the network rate region is non-empty, Slater's condition is satisfied and strong duality holds [9]. In this chapter we focus on wired and TDMA wireless networks where the network capacity constraints are of the form in equation 3.3. The Lagrangian is then

$$L(\mathbf{x}, \lambda) = \sum_{b \in B} U\left(\sum_{f \in b} x_f\right) - \gamma \sum_{l \in L} w_l \left\|\mathbf{x}_{F_l}\right\|_p - \lambda^T \left(\mathbf{A}^T \mathbf{x} - \mathbf{c}\right) \tag{4.1}$$

where λ is the vector of multipliers, and the KKT conditions for optimality are

$$\frac{\partial L}{\partial x_f} = U'\left(\sum_{f \in b} x_f^\star\right) - \sum_{l \in r_f} \left(\gamma \partial_{x_f} w_l \left\|\mathbf{x}_{F_l}^\star\right\|_p + \lambda_l^\star\right) = 0 \quad f \in F \tag{4.2}$$

$$\frac{\partial L}{\partial \mu_l} = \lambda_l^\star \left(\sum_{f \in F_l} x_f^\star - c_l\right) = 0 \qquad l \in L \tag{4.3}$$

Note that the optimal solution may not be unique due to the multipath nature of the optimization problem.

In the remainder of this chapter we present several distributed algorithms for solving problem **P1**, and analyse the induced sparsity among the flows.

4.2 Lagrangian Algorithms

4.2.1 Alternating Direction Method of Multipliers

A standard method to solve convex optimisation problems is via the Alternating Direction Method of Multipliers using subgradients, and Proximal Point Methods (PPMs) [8, 7]. Algorithm 4.1 summarise the solution proposed in [23] for multi-path network flow problems.

Algorithm 4.1 Alternating Direction Method of Multipliers

1. Initialize $k = 0$, $y_i(0)$ and$\lambda\left(0\right)$.
2. Solve the Augmented Lagrangian problem for fixed $\lambda\left(k\right)$

$$\mathbf{x}(k+1) \quad = \quad \arg\max_{x} L(\mathbf{x}(k), \lambda(k)) - \frac{1}{2}\sum_{b\in\mathcal{B}}\sum_{f\in b}\left(x_i - y_i\left(k\right)\right)^2$$

3. Update the variables introduced to the augmented Lagrangian

$$\mathbf{y}\left(k+1\right) = \mathbf{y}\left(k\right) + \alpha\left(\mathbf{x}\left(k+1\right) - \mathbf{y}\left(k\right)\right)$$

4. Update the Lagrangian variables

$$\lambda\left(k+1\right) = \lambda\left(k\right) + \alpha\left(\mathbf{A}^T\mathbf{x} - \mathbf{c}\right)$$

5. Update $k \leftarrow k+1$.
6. Repeat steps 2 to 4 until convergence.

However, Algorithm 4.1 requires the solution of a convex optimisation (to find $\mathbf{x}(k+1)$) at every iteration, which is not necessarily distributed. In the next subsections, we propose a distributed algorithm that avoids this problem.

4.2.2 Lagrangian formulation with linear constraints

It was Kelly in [20] who first proposed a distributed algorithm for solving network flow problems based on an iterative update of flow rates, while updating the prices of the network links with a strictly concave function. His proof of convergence to an optimum consisted of finding a Lyapunov function for the dynamical system he proposed, and then showing asymptotic stability for the strictly concave problem. However, his proof was only valid for a relaxed version of the optimization problem, as the pricing coefficients he proposed depended on a parameter $\epsilon > 0$ which approximated the exact problem only as $\epsilon \to 0$. In addition, the generalization of his formulation to the multipath problem did not fulfil all the conditions necessary to prove asymptotic stability in the Lyapunov sense.

Due to the instability in the multipath version, Wang et. al. [33] presented a modified version of the distributed algorithm using Proximal Point Methods (PPMs), and referred for convergence proof to [3]. It was Feijer and Paganini in [13] who extended the proof of convergence of this algorithm to the multipath case, by adding a penalty that made the problem strictly concave. However, this proof only analysed the continuous case in the dynamical system, and had the limitation of not considering subgradient methods.

Our contribution here is to present a modified version of the original dynamical system which addresses the energy consumption problem, while providing asymptotic convergence to a ball around the optimum even without using PPMs (Proximal Point Methods) in discrete time. We propose Algorithm 4.2 with a system of difference equations and prove convergence for the general case. We include a positive projection on the variables to guarantee positivity, namely $[w]^+ := w$ if $w > 0$, and $[w]^+ := 0$ otherwise.

Algorithm 4.2 Lagrangian Method with linear constraints

1. Initialize all vectors $\mathbf{x}(0)$ and $\lambda(0)$, and $k = 0$
2. Update all variables

$$\begin{aligned} \mathbf{x}(k+1) &= [\mathbf{x}(k) + \alpha \partial_x L(\mathbf{x}(k), \lambda(k))]^+ \\ \lambda(k+1) &= [\lambda(k) + \alpha \partial_\lambda L(\mathbf{x}(k), \lambda(k))]^+ \end{aligned}$$

   where

$$\partial_{x_f} L(\mathbf{x}, \lambda) = U'\left(\sum_{f \in b} x_f\right) - \sum_{l \in r_f} \left(\gamma \partial_{x_f} w_l \left\|\mathbf{x}_{F_l}\right\|_p + \lambda_l\right) \quad f \in \mathcal{F}$$
$$\partial_\lambda L(\mathbf{x}, \lambda) = \left[\mathbf{A}^T \mathbf{x} - \mathbf{c}\right]_\lambda^+$$

3. Update $k \leftarrow k + 1$.
4. Repeat step 2 and 3 until convergence.

Note that we do not add the slack variables $\mathbf{y}$ present in Algorithm 4.1, but we will still have asymptotic convergence. The proof of convergence of Algorithm 4.2 is presented in the Appendix. Algorithm 4.2 makes use of subgradients in every optimization step, so we can only guarantee convergence to a ball around the optimum.

We present here the subgradients for the $l_1 - norm$ and the $l_\infty - norm$. In the case of the $l_1 - norm$, the subgradients for a vector $\mathbf{x} \in \mathbb{R}^F$ where we have F flows are given by

$$\partial_x \left\|\mathbf{x}\right\|_1 = \begin{pmatrix} \partial_x |x_1| & \dots & \partial_x |x_p| \end{pmatrix}^T$$

where the individual components $f = 1, \dots, F$ have the form

$$\partial_x |x_f| = \begin{cases} 1 & if\, x_f > 0 \\ -1 & if\, x_f < 0 \\ [-1, 1] & if\, x_f = 0 \end{cases}$$

This is illustrated schematically for the scalar case in Figure 4.1, where we have

drawn supporting planes to the epigraph of the function on the origin with normal vectors of the form $(\partial_x \|\mathbf{x}\|_1, -1)^T$.

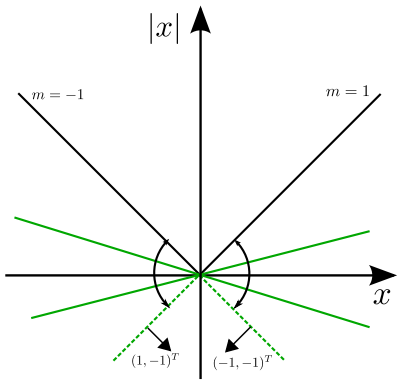


Figure 4.1: Subgradients of absolute value function

For the $l_\infty - norm$ of vector $\mathbf{x}$ the subgradient is

$$\partial_x \|\mathbf{x}\|_\infty = \begin{pmatrix} \delta_1^{\mathcal{K}} & \dots & \delta_F^{\mathcal{K}} \end{pmatrix}^T$$

where $\mathcal{K} = \{i | i = \arg\max_f \{x_f\}\}$ is the set of positions of the maximum components, and $\delta_f^{\mathcal{K}} := 1$ if $f \in \mathcal{K}$, and $\delta_f^{\mathcal{K}} := 0$ otherwise. The normalized level curves of the norms can be seen in Figure 4.2 for vector $\mathbf{x} \in \mathbb{R}^2$.

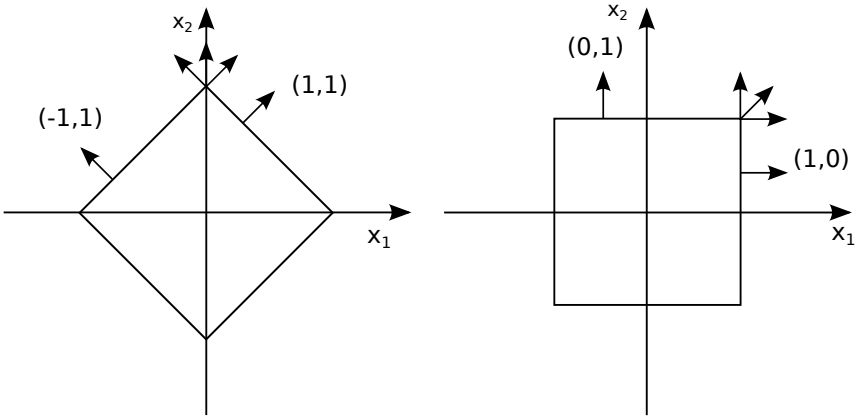


Figure 4.2: Level sets of l_1 and l_∞ norms

4.2.3 Lagrangian formulation with 802.11 constraints

The algorithm for 802.11 constraints presents some particular conditions that requires Algorithm 4.2 to be adapted to the new variables. Specifically, the range of variables $\tilde{x}_f = \log x_f$ and $\tilde{\xi}_f = \log \xi_f$ is $[-\infty, \infty)$, taking the $-\infty$ value when the flow is shut off. This condition is unsuitable for iterative algorithms, so computation on the original variables is preferred.

We can convert updates of the form

$$\tilde{x}_f\left(k+1\right) = \tilde{x}_f\left(k\right) - \alpha \partial_{\tilde{x}_f} L(\tilde{\mathbf{x}}\left(k\right), \lambda\left(k\right))$$

into

$$x_f(k+1) = x_f\left(k\right) \cdot e^{-\alpha \partial_{\tilde{x}_f} L(\mathbf{x}(k), \lambda(k))}$$

by taking exponential form from the first equation. This can be done for both variables $\tilde{x}_f$ and $\tilde{\xi}_f$, allowing to work in the range $[0, \infty)$. Algorithm 4.3 shows these steps. We note, that problem **P1** with utility function as defined in equation 3.9 is no longer convex for the choice of $\beta = 1$.

Algorithm 4.3 Lagrangian Method with 802.11 constraints

1. Initialize all vectors $\mathbf{x}(0)$ and $\lambda(0)$, and $k = 0$
2. Update all variables

$$\begin{aligned}
\mathbf{x}(k+1) &= \left[\mathbf{x}(k) \cdot e^{-\alpha \partial_{\tilde{x}} L(\mathbf{x}(k),\xi(k),\lambda(k))}\right]^{+} \\
\xi(k+1) &= \left[\xi(k) \cdot e^{-\alpha \partial_{\tilde{\xi}} L(\mathbf{x}(k),\xi(k),\lambda(k))}\right]^{+} \\
\lambda(k+1) &= \left[\lambda(k) + \alpha \partial_{\lambda} L(\mathbf{x}(k), \xi(k), \lambda(k))\right]^{+}
\end{aligned}$$

where

$$\partial_{\tilde{x}_f} L(\mathbf{x},\xi,\lambda) = -x_f \cdot U'\left(\sum_{f\in b} x_f\right) + x_f \cdot \sum_{i\in r_f}\left(\gamma + \frac{\lambda_i}{\sum_{f\in F_i} x_f}\right) \qquad \forall f \in \mathcal{F}$$

$$\partial_{\tilde{\xi}_i} L(\mathbf{x},\xi,\lambda) = \frac{1}{\chi_l} \cdot \sum_{k\in N_l} \lambda_k \left(\prod_{j\neq i}(1+\xi_j)\,\xi_i + \left(\frac{T_{succ,i}}{T_{coll}} - 1\right)\xi_i\right) - \lambda_i \quad \forall i \in N_l, l \in L$$

$$\partial_{\lambda} L(\mathbf{x},\xi,\lambda) = \left[\mathbf{A}^T\mathbf{x} - \mathbf{c}\right]_{\lambda}^{+}$$

$$\chi_l(\xi) = \frac{\sigma}{T_{coll}} - 1 + \prod_{i\in N_l}(1+\xi_i) + \sum_{i\in N_l}\left(\frac{T_{succ,i}}{T_{coll}} - 1\right)\xi_i \qquad l \in L$$

3. Update $k \leftarrow k+1$.
4. Repeat step 2 and 3 until convergence.

4.3 Proximal Gradient Methods

Given the structure of our optimization problem, we can also make use of proximal gradient methods, which include tools normally used to solve sparse problems. In this section, we will introduce these methods, and analyse the induced sparsity via the proximity operators for the norms of interest.

4.3.1 Framework formulation

We define the proximity operator as

$$\hat{x} = prox_{f_2}(u) := \arg\min_{x\in R^n} \frac{1}{2}\|u - x\|_2^2 + f_2(x) \tag{4.4}$$

where $f_2(x)$ represents the norm constraint in the network flow problem. The proximity operator was first introduced by Moreau in the 1960s [27], and proximal gradient methods were derived by Combettes and Wajs in [11]. For an introduction we also recommend [5, 6].

The proximal gradient method solves a general kind of problems formulated as

$$\min_{\mathbf{x}\in\mathbb{R}^n} f_1(x) + f_2(x) \tag{4.5}$$

where $f_1(x)$ is a continuously differentiable function in a finite Euclidean space $\mathbb{R}^n$ (we will also require convexity), and where $f_2(x)$ is a proper closed and convex function which is assumed subdifferentiable over its domain.

The well known gradient algorithm does not solve problem 4.5, where we require new techniques to analyse it if we do not want to use subgradient methods as explained in Section 4.2. The gradient algorithm is based on an iterative procedure where the points are produced by taking a small step along a steepest descent direction, namely $x_1(k+1) = x_1(k) - \alpha\nabla_x f_1(x)$ with a suitable step size α. For problem 4.5, we can formulate an approximate objective function that reasonably approaches the overall function around a point, and move along a gradient direction that minimizes the differentiable objective. Thinking of a quadratic model we can expand $f_1(\mathbf{x})$ with a Taylor expression around point $\mathbf{y}$ and get

$$Q(\mathbf{x},\mathbf{y}) = f_1(\mathbf{y}) + (\mathbf{x}-\mathbf{y})^T \nabla_x f(\mathbf{y}) + \frac{1}{2\alpha}\|\mathbf{x}-\mathbf{y}\|^2 + f_2(\mathbf{x})$$

which can be equivalently formulated as

$$Q(\mathbf{x},\mathbf{y}) = \frac{1}{2\alpha}\|\mathbf{x} - (\mathbf{y} - \alpha\nabla f_1(\mathbf{y}))\| - \frac{\alpha}{2}\|\nabla f_1(\mathbf{y})\| + f_1(\mathbf{y}) + f_2(\mathbf{x})$$

Minimizing the previous expression for $\mathbf{x}$ and disregarding constant terms,

we get

$$\min_{\mathbf{x}} \frac{1}{2\alpha} \|\mathbf{x} - (\mathbf{y} - \alpha \nabla f_1(\mathbf{y}))\| + f_2(\mathbf{x}) = prox_{\alpha f_2}(\mathbf{y} - \alpha \nabla f_1(\mathbf{y})) \tag{4.6}$$

making use of the proximity operator as defined in equation 4.4. Relating this formulation with the popular technique known as Majorization-Minimization (MM) will allow us to enunciate the proximal gradient method. MM algorithm constructs an approximate model to the objective function that satisfies

$$\begin{aligned} Q(\mathbf{x}, \mathbf{x}) &= F(\mathbf{x}) && \forall\, x \\ Q(\mathbf{x}, \mathbf{y}) &\geq F(\mathbf{x}) && \forall\, x, y \end{aligned}$$

Geometrically, this means that $Q(\mathbf{x}, \mathbf{y})$ lies above $F(\mathbf{x})$ and is tangent to it in $\mathbf{x}$. This scheme implies that if

$$\mathbf{x}(k+1) = \arg\min_{\mathbf{x} \in \mathbb{R}^n} Q(\mathbf{x}, \mathbf{x}(k))$$

$$Q(\mathbf{x}(k+1), \mathbf{x}(k)) \leq Q(\mathbf{x}, \mathbf{x}(k)) \quad \forall \mathbf{x}$$

and it follows

$$F(\mathbf{x}(k+1)) \leq Q(\mathbf{x}(k+1), \mathbf{x}(k)) \leq Q(\mathbf{x}(k), \mathbf{x}(k)) = F(\mathbf{x}(k)) \quad for\, k \geq 1 \tag{4.7}$$

for some objective function $F(\mathbf{x})$. See Figure 4.3.

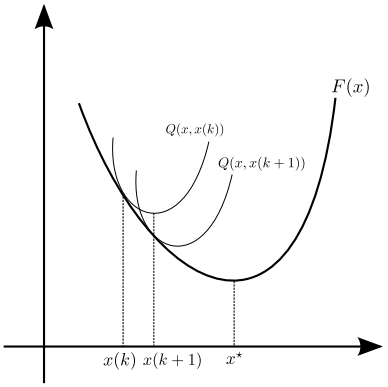


Figure 4.3: MM-method

Finally, we can introduce the prox-grad map from equation 4.6 as the update

$$\mathbf{x}(k+1) = prox_{\alpha f_2}(\mathbf{x}(k) - \alpha_k \nabla f_1(\mathbf{x}(k))) \tag{4.8}$$

which is a majorizer of 4.5 if $\alpha_k \leq \frac{1}{L}$, where L is a Lipschitz constant of f_1. We can see the non-expansive property of 4.8 because of equation 4.7. Note, that the local Lipschitz constant implies

$$f(\mathbf{x}) \leq f(\mathbf{y}) + (\mathbf{x}-\mathbf{y})^T \nabla f(\mathbf{y}) + \frac{L}{2}\|\mathbf{x}-\mathbf{y}\|^2$$

which guarantees the property.

Convergence to the optimal point of 4.5 is guaranteed if the problem is strongly convex. See [6] for a derivation of the proof. Otherwise it may only converge to a fixed-point.

4.3.2 Network Flow Problem

We can apply the previous algorithm of proximal gradient methods to solve problem **P1**. We propose Algorithm 4.4 that uses proximal operators to find the optimal value. Note that due to the feasible region, a projection would be required after computing the proximal operator. To avoid this approach, we solve the Lagrangian problem, that simplifies the exposition.

Algorithm 4.4 Proximal Gradient Method in the Network Flow Problem with energy constraints

1. Initialize vectors $\mathbf{x}(0)$, $\lambda(0)$, and $k = 0$
2. Compute
$$\mathbf{x}(k+1) = prox_{\alpha f_2}(\mathbf{x}(k) + \alpha \nabla f_1(\mathbf{x}(k)))$$
where $\nabla f_1(\mathbf{x}) = U'\left(\sum_{f \in b} x_f\right) - \sum_{l \in r_f} \lambda_l$ and $f_2(\mathbf{x}) = \gamma \sum_{l \in L} w_l \|\mathbf{x}_{F_l}\|_p$.
3. Calculate
$$\lambda(k+1) = [\lambda(k) + \alpha \partial_\lambda L(\mathbf{x}(k), \lambda(k))]^+$$
where $\partial_\lambda L(\mathbf{x}, \lambda) = \left[\mathbf{A}^T\mathbf{x} - \mathbf{c}\right]_\lambda^+$.
4. Update $k \leftarrow k + 1$.
5. Repeat step 2, 3 and 4 until convergence.

We note that the proximity operator is applied to a group-norm of the form $l_p/l_1 - norm$, which does not have a closed form for every p. In the next section we present an analysis of the $p = 1$ and $p = \infty$ cases, which can be solved efficiently.

4.4 Sparsity

In this section we study problem **P1** and analyse the induced sparsity by means of the proximal operator.

4.4.1 Proximity operator of the $l_1 - norm$

We analyse the case $p = 1$ where we use the $l_1 - norm$ as $\|\mathbf{x}\|_1 := \sum_{f \in F_l} |x_f|$ for every link l. Then, the proximity operator for each component of $\mathbf{x}$ becomes

$$prox_{\alpha_k f_2}\left(x_f(k) + \alpha \frac{\partial}{\partial x_f} f_1(x_f(k))\right) \quad \forall f \in \mathbf{x}$$

where $f_2 = \sum_{l \in r_f} \lambda_l$, and x_f represents the corresponding flow rate, and r_f is the set of links (the route) of flow f.

The solution to the previous operator is the well known soft-thresholding

operator, defined as

$$prox_{\mu}\left(u\right) = soft\left(u,\mu\right) := \begin{cases} u-\mu & if\ u>0\, and\, u>\mu \\ 0 & if\ \left|u\right| \leq \mu \\ u+\mu & if\ u<0\, and\, u<\mu \end{cases}$$

Note, that since every flow rate is positive, the third case does not apply to our problem.

Proof. We solve the operator for one variable when we have $x\left(k+1\right) = prox_{\mu\left|x\right|}\left(u\right) = \arg\min_{x\in\mathbb{R}} \frac{1}{2}\left\|u-x\right\|_2^2 - \mu\left|x\right|$. Taking subderivatives in x and making it equal to zero we get$-\left(u-x\right) - \mu\partial_x\left|x\right| = 0$. Therefore,

$$if\ x>0 \Rightarrow x = u-\mu$$

$$if\ x<0 \Rightarrow x = u-\mu$$

$$if\ x=0 \Rightarrow u+\mu\partial_x\left|x\right| = 0$$

and the solution follows. □

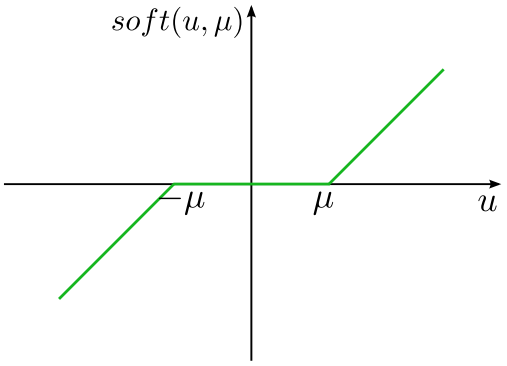


Figure 4.4: Soft-thresholding operator

The induced sparsity comes from the fact that for the range of values $\left|u\right| \leq \mu$ the proximity operator yields $x = 0$. In our Network Flow problem, the

proximity operator for flow f is calculated over $\mu = \sum_{l \in r_f} \gamma$, so we can guarantee that flows where $x_f^\star \leq \mu$ will imply $x_f^\star = 0$. Therefore, in our proposal of **P1** with multipath flows per user we search for solutions that minimize the energy consumption among the network exploiting sparsity.

We note also, that the soft-thresholding solution presents a shrinkage over the variables when $x_f \neq 0$, due to the induced linear cost on the objective function. We will explain this effect in our simulations on Chapter 5.

4.4.2 Proximity operator of the $l_1/l_\infty - norm$

The motivation for using the group $l_1/l_\infty - norm$ is due to the properties it presents when exploiting the graph structure of the network. We define the norm for the network as $\sum_{l \in L} \|\mathbf{x}_{F_l}\|_\infty$, where L represents the set of links and x_{F_l} is the vector of flow rates going through link l. By applying a linear cost per link we induce sparsity towards the links, but not towards the flows within the link, as the $l_\infty - norm$ only induces equality among the components. Intuitively, flows that go through one link are charged the same fee (cost proportional to the maximum of the flows in the link) and pay a unique cost on the objective function, so every flow tends to share links and avoid using connections exclusively. Simulations in Chapter 5 will show this behaviour, and also its limitations.

The Network Flow problem is presented as follows,

$$\begin{aligned} &\min \sum_{b \in \mathcal{B}} U\left(\sum_{f \in b} x_f\right) + \gamma \sum_{l \in L} w_l \|\mathbf{x}_{F_l}\|_\infty \\ &s.t.\, \mathbf{A}^T\mathbf{x} - \mathbf{c} \leq 0 \end{aligned} \tag{4.9}$$

where, the added penalty induces a $l_1 - norm$ regularization among groups, and flows belonging to the same group are affected by the $l_\infty - norm$. We note that solving this problem with the Proximal Gradient method explained in Section 4.3 requires the solution of a convex optimization on every iteration. The solution to the proximity operator for group norms does not have an explicit

formula, which makes the computation generally inefficient. However, this can be different for the $l_1/l_\infty - norm$, where a simpler problem can be solved. We will first show the derivation on how to solve the problem, and then the algorithm itself.

The Lagrangian formulation for problem 4.9 is

$$L = \sum_{b\in\mathcal{B}} U\left(\sum_{f\in b} x_f\right) - \gamma \sum_{l\in L} w_l \left\|\mathbf{x}_{F_l}\right\|_\infty - \lambda^T\left(\mathbf{A}^T\mathbf{x} - \mathbf{c}\right)$$

where, assuming stability of the dynamical system, we can iterate over the system 4.10 to solve the problem. Namely,

$$\begin{aligned} \mathbf{x}(k+1) &= prox_{\alpha f_2}\left(\mathbf{x}(k) - \alpha\nabla_x f_1(\mathbf{x}(k))\right) \\ \lambda(k+1) &= \lambda(k) + \alpha\nabla_\lambda L(\mathbf{x}(k), \lambda(k)) \end{aligned} \tag{4.10}$$

where we have substituted $f_1(\mathbf{x}) = \sum_{b\in\mathcal{B}} U\left(\sum_{f\in b} x_f\right) - \lambda^T\left(\mathbf{A}^T\mathbf{x} - \mathbf{c}\right)$ and $f_2(\mathbf{x}) = \gamma\sum_l w_l \left\|\mathbf{x}_{F_l}\right\|_\infty$ to simplify the notation.

The proximity operator is defined as

$$\hat{\mathbf{x}} = prox_{f_2}(u) := \arg\min_{\mathbf{x}\in R^n} \frac{1}{2}\|\mathbf{u} - \mathbf{x}\|_2^2 + f_2(\mathbf{x}) \tag{4.11}$$

but because the variables overlap among the groups, the solution of the proximal problem is not immediate and does not have a direct form. However, we can solve it via a dual formulation proposed in [24] and use their proposed lemma,

Lemma 1. *Given* $\mathbf{u}$ *in* $\mathbb{R}^p$*, the problem*

$$\begin{aligned} &\min_{\xi\in\mathbb{R}^{p x L}} \frac{1}{2}\left\|\mathbf{u} - \sum_{l\in L}\xi^l\right\|_2^2 \\ &s.t.\ \left\|\xi^l\right\|_1 \le \gamma w_l \qquad l\in L \\ &\xi_j^l = 0 \qquad if\ j\notin l \end{aligned} \tag{4.12}$$

where $\xi = \left(\xi^l\right)_{l \in L}$ *is in* $\mathbb{R}^{pxL}$*, and* ξ_j^l *denotes the* j^{th} *coordinate of the vector* ξ^l*. Then, every solution* $\xi^* = \left(\xi^{*l}\right)_{l \in L}$ *satisfies* $\hat{\mathbf{x}} = \mathbf{u} - \sum_l \xi^{*l}$*, where* $\hat{\mathbf{x}}$ *is solution of 4.11.*

In our problem, p is the dimension of $\mathbf{x}$ and refers to the total number of flows in the network, and $l \in L$ the different links in the network. Using this result in our problem, we can again apply Lagrange duality

$$L_{dual} = \frac{1}{2} \left\| \mathbf{u} - \sum_{l \in L} \xi^l \right\|_2^2 + \sum_{l \in L} \nu_l \left(\left\| \xi^l \right\|_1 - \gamma w_l \right)$$

where ν_l are the Lagrangian variables, and get the following dynamical system

$$\begin{aligned} \frac{\partial L_{dual}}{\partial \xi_j^l} &= -\left(u_j - \sum_{l \in r_j} \xi_j^l \right) + \nu_l && \forall l, j \\ \frac{\partial L_{dual}}{\partial \nu_l} &= \left\| \xi^l \right\|_1 - \gamma w_l && \forall l \end{aligned}$$

to solve the previous formulation.

The users compute only local variables ξ_j^l and routers use only information available on the links, so this algorithm could in principle be implemented in a distributed fashion. However, the inner loop that solves the dual formulation has to converge before updating the Lagrangian variables. Because of the instability if we update both ν and λ simultaneously, the algorithm presents some drawbacks that make it unsuitable for practical use. In that sense, it may be easier to use the subgradient methods proposed in Section 4.2.

Finally, Algorithm 4.5 shows the required steps.

Algorithm 4.5 Network Flow Problem with $l_1/l_\infty - norm$

1. Initialize vectors $\mathbf{x}(0)$, $\lambda(0)$.
2. Calculate $\mathbf{u} = \mathbf{x}^k - \nabla_x f_1(\mathbf{x}^k)$ where $f_1(\mathbf{x}) = \sum_{b\in\mathcal{B}} U\left(\sum_{f\in b} x_f\right) - \lambda^T\left(\mathbf{A}^T\mathbf{x} - \mathbf{c}\right)$.
3. Solve (in a distributed fashion) the Lagrangian $L_{dual} = \frac{1}{2}\left\|\mathbf{u} - \sum_{l\in L}\xi^l\right\|_2^2 + \sum_{l\in L}\nu_l\left(\left\|\xi^l\right\|_1 - \gamma w_l\right)$ using the difference equations

$$\begin{aligned}\xi^l(k+1) &= \xi^l(k) - \alpha\frac{\partial L(k)}{\partial \xi_j^l}\\ \nu_l(k+1) &= \nu_l(k) + \alpha\frac{\partial L}{\partial \nu_l}\end{aligned}$$

where

$$\begin{aligned}\frac{\partial L_{dual}}{\partial \xi_j^l} &= -\left(u_j - \sum_{l\in r_j}\xi_j^l\right) + \nu_l \forall l, j\\ \frac{\partial L_{dual}}{\partial \nu_l} &= \left\|\xi^l\right\|_1 - \gamma w_l \forall l\end{aligned}$$

4. Update
$$\lambda(k+1) = \lambda(k) + \alpha\left(\mathbf{A}^T\mathbf{x} - \mathbf{c}\right)$$
5. Update $k \leftarrow k+1$
6. Repeat steps 2-5 until convergence

Chapter 5

Simulation Results

In this chapter we present simulation results using the algorithms from the previous chapter, and analyse their solutions. We will focus on simple examples that highlight qualitative properties.

5.1 Multi-hop TDMA Wireless

In this section we show the behaviour of some simple examples and the effects that penalties produce in the optimal solution. These examples are representative for applications that adapt their data rate depending on availability, such as TCP, maximizing the transmission rates under channel capacity.

Note, that the optimal solutions in some of these examples might not be unique, as balancing load among user flows can still produce the same value in the objective function.

We will focus on the log utility function, corresponding to proportional fairness, as this is widely considered in network flow problems. We will analyse examples observing the effect of different penalties in the energy usage. To simplify the examples, we will assume that all links consume the same energy, so $w_l = 1$, and capacity of links will also be equal $c_l = 1$ for all $l \in L$.

5.1.1 Multihop Scenario: 1 User, 2 Routes

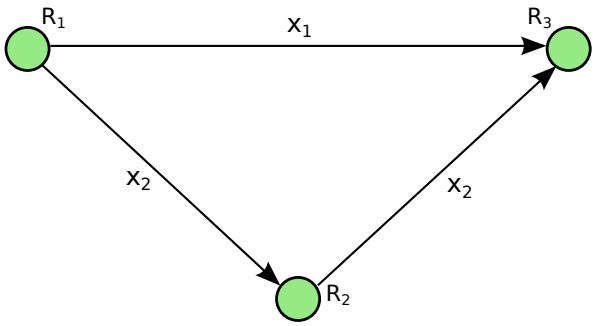


Figure 5.1: 1 User, 3 nodes

We analyse the effects of adding energy constraints on network routes in the simple case of only one user. This user has two routes, one that goes from router R1 to R3 using one link, and a second one that uses two links and goes through R2. We apply the l_1-norm penalty to the original utility maximization problem and get

$$\min_{\mathbf{x}} -\log(x_1 + x_2) + \gamma x_1 + 2\gamma x_2$$
$$s.t.\ \mathbf{A}^T\mathbf{x} \leq \mathbf{1}$$

Solving this problem for different values of λ, we can observe that the energy penalty introduced provides a solution that favours the shortest communication paths. Additionally, the effect of the l_1-norm on the flows produces a shrinkage on the transmission rates, as expected from computing a soft-thresholding. This is shown in Table 5.1 where the longest route is shut down and all traffic is transmitted through the shortest. Namely, links L1 and L2 are shut off when increasing the cost. On the third column we show how much capacity is saved when applying the penalty, which would allow energy savings as explained in Chapter 2.

γ	Rate	Links OFF	Underused C
$\gamma = 0$	$x = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$	None	0
$\gamma = 0.5$	$x = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$	L1,L2	66%
$\gamma = 1$	$x = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$	L1,L2	66%
$\gamma = 10$	$x = \begin{bmatrix} 0.1 & 0 \end{bmatrix}^T$	L1,L2	97%

Table 5.1: 1 User, 2 Routes

5.1.2 Multihop scenario: 2 Users, 5 links

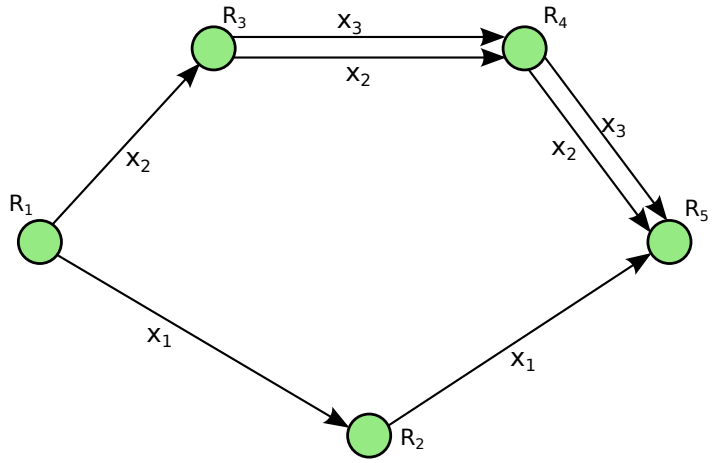


Figure 5.2: 2 Users, 5 nodes

In this case we analyse the effect of using different norm penalties in the network links. For instance, using the $l_1 - norm$ is intended when the network is capable of saving energy from underused links, and we can observe that a shortest path solution is optimal when constraints allow it. However, when the network is not capable of switching off elements when underusing capacity, then a $l_1/l_\infty - norm$ puts together flows under the same links, allowing to completely switch off other nodes.

In this example we have two users, one transmitting from R1 to R5, and another transmitting from R3 to R5. The first user has two possible routes x_1 and x_2 belonging to the same bundle, while the second user only has flow x_3.

When using the $l_1 - norm$ penalty, the problem to optimize is

$$\min_{\mathbf{x}} -\log(x_1 + x_2) - \log(x_3) + 2\gamma x_1 + 3\gamma x_2 + 2\gamma x_3$$
$$s.t.\ \mathbf{A}^T\mathbf{x} \leq \mathbf{1}$$

and when using the $l_1/l_\infty - norm$ penalty is

$$\min_{\mathbf{x}} -\log(x_1 + x_2) - \log(x_3) + 2\gamma x_1 + \gamma x_2 + 2\gamma \left\| (x_2, x_3)^T \right\|_\infty$$
$$s.t.\ \mathbf{A}^T\mathbf{x} \leq \mathbf{1}$$

Solutions to these problems are shown in Table 5.2. In the problem with the first penalty, we observe flow x_1 is preferred over x_2 for the first user as it has a shorter path, and only one link can be completely shut off. However, when using the $l_1/l_\infty - norm$ penalty for sufficiently large γ, the flows group under the same links, allowing to completely switch off the route of x_1. Specifically, links L1 and L2 can completely be shut off, versus only L3 in the first case.

We can still observe shrinkage in the rates due to the linear cost of the penalties.

l_1	Rate	Links OFF	Underused C
$\gamma = 0$	$x = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$	None	20%
$\gamma = 0.5$	$x = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$	None	20%
$\gamma = 1$	$x = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}^T$	L3	60%
$\gamma = 10$	$x = \begin{bmatrix} \frac{1}{20} & 0 & \frac{1}{20} \end{bmatrix}^T$	L3	96%

l_1/l_∞	Rate	Links OFF	Underused C
$\gamma = 0$	$x = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$	None	20%
$\gamma = 0.5$	$x = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} & \frac{2}{3} \end{bmatrix}^T$	None	26%
$\gamma = 1$	$x = \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}^T$	L1,L2	60%
$\gamma = 10$	$x = \begin{bmatrix} 0 & \frac{2}{30} & \frac{2}{30} \end{bmatrix}^T$	L1,L2	95%

Table 5.2: 2 Users, 5 links

5.2 Femtocell Networks

5.2.1 Femtocell scenario: 3 Users, 2 Femtocells, 1 Macrocell

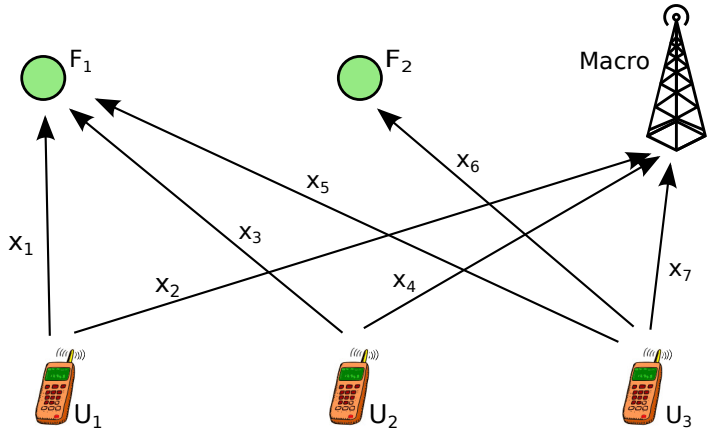


Figure 5.3: 3 Users, 2 Femtos, 1 Macro

In this example we consider a scenario where users can connect to several femtocell routers and/or the macrocell, and we analyse how traffic is distributed to minimise energy for different traffic demands. Note, that in this scenario every route consists of only one hop, so the $l_1 - norm$ penalty that encourages a shortest path route does not group traffic to unique femtocells, and it would present multiple optimal solutions. For this reason, we will only analyse results when using the $l_1/l_\infty - norm$. Nonetheless, using the $l_1 - norm$ penalty would still have a shrinkage effect on the flow magnitudes, which can still be exploited to use some energy scheme proposed in Chapter 2. The optimization problem is then

$$\begin{aligned}
&\min_{\mathbf{x}} -\log(x_1 + x_2) - \log(x_3 + x_4) - \log(x_5 + x_6 + x_7) \\
&\quad + \gamma \left\| (x_1, x_3, x_5)^T \right\|_\infty + \gamma x_6 + \lambda w_{macro} \left\| (x_2, x_4, x_7)^T \right\|_\infty \\
&s.t.\ \mathbf{Ax} \leq \mathbf{1}
\end{aligned}$$

where we have included a parameter w_{macro} to represent a different cost when using this link. This cost can be bigger than one, when we would like to encourage user terminals to connect to femtocells to increase the device's autonomy, or smaller than one when in low traffic hours we would like to switch off the femtocell routers and save more energy. Both considerations can be taken into account depending on the purpose of the network provider.

In Table 5.3 we analyse the flow behaviour under different γ, specifically for the case $w_{macro} = 5$. We observe that without adding further constraints all link capacity is used and proportional fairness allocates every user the same rate.

When the penalty is increased to $\gamma = 1$ terminals only connect to the femtocells, and the macrocell does not serve traffic from these users. Note in this case rates are no longer equal, as user U3 receives more traffic than U1 and U2. This happens because U3 does not have to share the connection with other users, as its connection has less demand. Finally, when $\gamma = 10$, all traffic is served by femtocell F1, and F2 can be switched off.

This example shows some versatility when controlling parameter γ, allowing automatic control of femtocell availability.

γ	Rate	Cell OFF	Underused C
$\gamma = 0$	$x = \left[\frac{1}{2} \quad \frac{1}{2} \quad \frac{1}{2} \quad \frac{1}{2} \quad 0 \quad 1 \quad 0 \right]^T$	None	0%
$\gamma = 1$	$x = \left[\frac{1}{2} \quad 0 \quad \frac{1}{2} \quad 0 \quad 0 \quad 1 \quad 0 \right]^T$	Macro	33%
$\gamma = 10$	$x = \left[\frac{3}{10} \quad 0 \quad \frac{3}{10} \quad 0 \quad \frac{3}{10} \quad 0 \quad 0 \right]^T$	Macro & F2	66%

Table 5.3: 3 Users, 2 Femtocells, 1 Macrocell

5.2.2 Femtocell scenario: 3 Users, 4 Femtocells

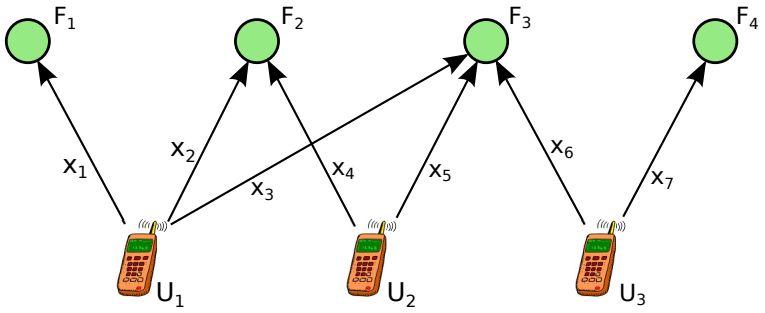


Figure 5.4: 3 Users, 4 Femtos

Here we consider an example where we analyse how the sets of users spread among femtocells under different constraints, without considering a macrocell. The optimisation problem is:

$$\begin{aligned}\min_{\mathbf{x}} & -\log\left(\sum_{i=1}^{3} x_i\right) - \log\left(\sum_{i=4}^{5} x_i\right) - \log\left(\sum_{i=6}^{7} x_i\right) \\ & + \gamma x_{1+} \gamma \left\|(x_2, x_4)^T\right\|_\infty + \gamma \left\|(x_3, x_5, x_6)^T\right\|_\infty + \gamma x_7 \\ s.t. & \mathbf{A}^T\mathbf{x} \leq \mathbf{1}\end{aligned}$$

Table 5.4 presents solutions to this problem for various parameter values from which we can see again the effects of shrinkage when increasing γ and also how flows group together for higher values. However, note that the distribution of user flow rates is not unique, and vectors in which users transmit to different femtocells simultaneously but that preserve their sum rate, would also be optimal.

γ	Rate	Cell OFF	Underused C
$\gamma = 0$	$x = \begin{bmatrix} 1 & \frac{1}{3} & 0 & \frac{2}{3} & \frac{2}{3} & \frac{1}{3} & 1 \end{bmatrix}^T$	None	0%
$\gamma = 1$	$x = \begin{bmatrix} \frac{1}{6} & \frac{1}{2} & \frac{1}{3} & \frac{1}{2} & \frac{1}{3} & \frac{1}{3} & \frac{2}{3} \end{bmatrix}^T$	None	29%
$\gamma = 10$	$x = \begin{bmatrix} 0 & 0 & \frac{3}{10} & 0 & \frac{3}{10} & \frac{3}{10} & 0 \end{bmatrix}^T$	F1,F2, F3	77%

Table 5.4: 3 Users, 4 Femtocells

5.3 802.11 Wireless Mesh Networks

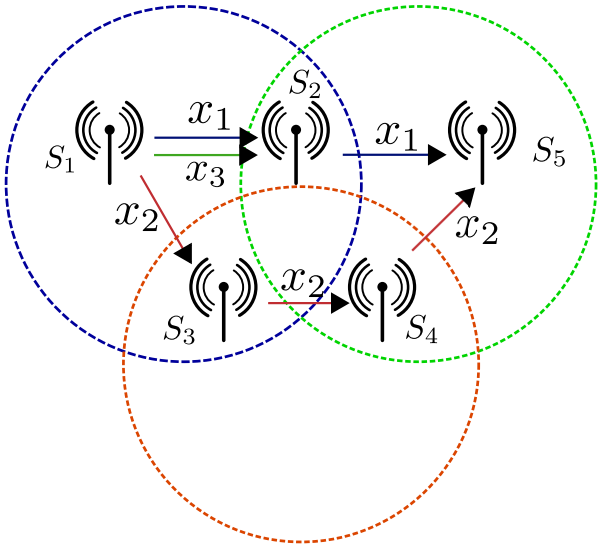


Figure 5.5: WLAN Network, 2 Users, 3 flows

In this section, we consider the example shown in Figure 5.5 with 802.11 network constraints from Subsection 3.1.1 and the following utility function:

$$U(x) := \begin{cases} \frac{x^{1-\beta}-1}{1-\beta} & x \geq 0,\ \beta > 1 \\ \log(x) & x \geq 0,\ \beta = 1 \end{cases}$$

The network constraints from this problem are log-convex and so we require a change of variables to obtain the solution. The formulation of the optimization

problem is

$$\begin{aligned}
&\min -U\left(e^{\tilde{x}_1}+e^{\tilde{x}_2}\right)-U\left(e^{\tilde{x}_3}\right)+2\gamma\tilde{x}_1+3\gamma\tilde{x}_2+\gamma\tilde{x}_3\\
&s.t.\log\left(e^{\tilde{x}_1}+e^{\tilde{x}_2}\right)-\tilde{\xi}_2+\log\chi_2-\log\frac{d}{T_{coll}}\leq 0\\
&\log\left(e^{\tilde{x}_1}+e^{\tilde{x}_2}\right)-\tilde{\xi}_4+\log\chi_2-\log\frac{d}{T_{coll}}\leq 0\\
&\tilde{\xi}_1=\tilde{\xi}_3=\infty\Longleftrightarrow\tau_1=\tau_3=1
\end{aligned}$$

where we have made the change of variables $\tilde{x}_i=\log x_i$ is the logarithm of the flow rate, $\tilde{\xi}_i=\log\xi_i$ is the logarithm of the normalized probability of transmission τ_i, and $\chi=\frac{\sigma}{T_{coll}}-1+\prod_{j\in N}(1+\xi_j)+\sum_{j\in N}\left(\frac{T_{succ,j}}{T_{coll}}-1\right)\xi_j$ is an expression dependent of all stations in the cell. For the specific example in Figure 5.5, stations s_1 and s_3 cannot transmit packets that may collide, as they are alone in their cells, and therefore, their probability of transmission is maximal.

$\beta=1.1$			
γ	Rates	Probabilities	User rates ($\begin{bmatrix} x_1+x_2, & x_3 \end{bmatrix}$)
$\gamma=0$	$x=\begin{bmatrix} 2.51 & 2.36 & 5.11 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.43 & 1 & 0.41 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 4.87 & 5.11 \end{bmatrix}^T$
$\gamma=0.1$	$x=\begin{bmatrix} 2.95 & 0.6 & 6.43 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.43 & 1 & 0.31 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 3.55.24 & 6.43 \end{bmatrix}^T$
$\gamma=1$	$x=\begin{bmatrix} 0.38 & 0.19 & 1.03 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.35 & 1 & 0.34 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 0.57 & 1.03 \end{bmatrix}^T$
$\gamma=10$	$x=\begin{bmatrix} 0.06 & 0.03 & 0.15 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.34 & 1 & 0.34 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 0.9 & 0.15 \end{bmatrix}^T$

$\beta=2$			
γ	Rates	Probabilities	User rates ($\begin{bmatrix} x_1+x_2, & x_3 \end{bmatrix}$)
$\gamma=0$	$x=\begin{bmatrix} 2.44 & 2.32 & 5.24 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.39 & 1 & 0.38 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 4.76 & 5.24 \end{bmatrix}^T$
$\gamma=0.1$	$x=\begin{bmatrix} 1.4 & 0.7 & 3.1 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.35 & 1 & 0.35 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 2.1 & 3.1 \end{bmatrix}^T$
$\gamma=1$	$x=\begin{bmatrix} 0.46 & 0.24 & 1 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.35 & 1 & 0.34 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 0.7 & 1 \end{bmatrix}^T$
$\gamma=10$	$x=\begin{bmatrix} 0.14 & 0.07 & 0.32 \end{bmatrix}^T$	$\xi=\begin{bmatrix} 1 & 0.34 & 1 & 0.34 \end{bmatrix}^T$	$x_s=\begin{bmatrix} 0.21 & 0.32 \end{bmatrix}^T$

Table 5.5: WLAN, 2 Users, 5 nodes

We select parameters $T_{succ}=T_{coll}=\sigma=1$ms, $d=1000$ bits and rates are measured in Mbits/s. In addition, we use utility with parameter $\beta\neq 1$, which is convex, and note that by making β close to 1 we can expect to obtain a similar

solution to the proportional fair allocation.

From Table 5.5, we observe first that the effects of proportional fairness are less present as β grows, moving towards max-min fairness as explained in [26]. Second, we observe that for $\gamma = 0$ flows x_1 and x_2 are similar, but as γ grows, flow x_2 is reduced to about half of x_1. This is due to the effects on the energy constraints, as the route from x_2 is longer.

Chapter 6

Conclusions

Wireless networks are expected to grow and develop increasingly rapidly in the near future due to increasing data traffic demands. Consequently, the impact of wireless network energy consumption is also expected to come to the fore, with efficient management of energy resources becoming more necessary both for environmental and economical reasons. At present times, wireless networks are deployed to give full service for the whole day even if resources are underused at certain hours, but more energy efficient solutions are possible.

In this context, the algorithms proposed here select traffic routes that can reduce energy consumption in the whole network in a fair manner while maximising network utility for users. The energy savings come from grouping data flows together and liberating resources, so that the network can switch off interfaces and power amplifiers, send basestations to sleep, or simply reduce the speed of links. This measures have the potential to greatly reduce energy consumption.

We formulate an optimization problem that maximizes user utilities while taking into account the energy costs of the transmission. We analyse an energy model for several types of wireless networks, as well as capacity constraints for different technologies. Dropping the constant term of the energy model in the optimization objective allows for a convex formulation and a distributed

solution, for which we prove convergence.

Future work includes dealing with non-convex formulations of the problem which include the constant term from the energy model, although this can be expected to significantly increase the problem difficulty. Additionally, dealing with real time variations, and delays between deciding to switch off devices and actually doing it, can become a matter of future study.

Appendix A

Convergence of Lagrangian Method

We give here a proof of convergence for the alternating variable update proposed in Chapter 3.

Problem

Given a problem of the form

$$\begin{aligned} &\min f(x) \\ &s.t.\, g_l(x) \leq 0 \quad l = 1, \ldots, m \end{aligned} \tag{A.1}$$

where $f(x)$ and $g_i(x)$ are convex in a compact set X, we assume it has a finite optimum and Slater's condition is satisfied (there is a vector x^s so that constraints are strictly feasible $g_l(x^s) < 0, \forall l$). Then, under these conditions the problem can be studied through the Lagrangian dual problem and strong duality holds. The Lagrangian has the form

$$L(x, \lambda) = f(x) + \sum_{l=1}^{m} \lambda_l g_l(x) \tag{A.2}$$

where we do not assume differentiability for functions $f(x)$ and $g_l(x)$, but

we require existence of subgradients (indicated with ∂_x and ∂_λ) and uniform boundedness for every $x \in X$, and $\lambda \geq 0$ ($\lambda \in \Lambda$) where,

$$C \geq \sup \{\|\partial_x L\| \, | \partial_x L \in \partial_x f(x)\},$$
$$D \geq \sup \{\|\partial_\lambda L\| \, | \partial_\lambda L \in \cup_{l=1}^{m} \partial_x g_l(x)\}$$

To solve the Lagrangian dual problem we propose a discrete system of difference equations (A.3)-(A.4)

$$x_i(k+1) = x_i(k) - \alpha \partial_{x_i} L(x(k), \lambda(k)) \tag{A.3}$$
$$\lambda_i(k+1) = \lambda_i(k) + \alpha \partial_{\lambda_i} L(x(k), \lambda(k)) \tag{A.4}$$

and will analyse uniform and asymptotic stability in the Lyapunov sense [19, 32]. We will establish that a sufficient condition for the system to be asymptotically stable is that functions $f(x)$ and $g_l(x)$ from (A.2) are radially unbounded, meaning $\|x\| \to \infty \Rightarrow |f(x)| \to \infty$. Note, we do not require problem (A.1) to be strictly convex, as in previous works of Arrow et al. [3], or Feijer and Paganini [13].

We propose function $V(k)$ as a Lyapunov candidate for the previous system

$$V(k) = \frac{1}{2\alpha} (x(k) - \overline{x})^T (x(k) - \overline{x}) + \frac{1}{2\alpha} \left(\lambda(k) - \overline{\lambda}\right)^T \left(\lambda(k) - \overline{\lambda}\right)$$

where $\left(\overline{x}, \overline{\lambda}\right)$ is a saddle point of the Lagrangian from equation (A.2).

Theorem 2. *Under the previous formulation, the function $V(k)$ is a strict generalized Lyapunov function for the discrete dynamical system formed by equations (A.3)-(A.4), and the variables $(x(k), \lambda(k))$ converge asymptotically in the Lyapunov sense to a ball around the saddle point of the Lagrangian given by*

$$\Omega = \left\{x, \lambda | L(x, \lambda) \leq L\left(\overline{x}, \overline{\lambda}\right) + \frac{\alpha}{2}\left(C^2 + D^2\right), \, x \in X, \, \lambda \geq 0\right\}$$

Proof. We have

$$\begin{aligned} V(k+1) &= \frac{1}{2\alpha}\left(x(k)-\overline{x}-\alpha\partial_x L(x(k),\lambda(k))\right)^T\left(x(k)-\overline{x}-\alpha\partial_x L(x(k),\lambda(k))\right) \\ &+ \frac{1}{2\alpha}\left(\lambda(k)-\overline{\lambda}+\alpha\partial_\lambda L(x(k),\lambda(k))\right)^T\left(\lambda(k)-\overline{\lambda}+\alpha\partial_\lambda L(x(k),\lambda(k))\right) \\ &= V(k)+\Delta(k) \end{aligned}$$

where

$$\Delta(k) = -\left(x(k)-\overline{x}\right)^T\partial_x L(x(k),\lambda(k)) + \left(\lambda(k)-\overline{\lambda}\right)^T\partial_\lambda L(x(k),\lambda(k)) + \frac{\alpha}{2}\epsilon$$

and

$$\epsilon = \left\|\partial_x L(x(k),\lambda(k))\right\|^2 + \left\|\partial_\lambda L(x(k),\lambda(k))\right\|^2$$

We need to show that $\Delta(k)\leq 0$ for $(x(k)-\overline{x})$, $(\lambda(k)-\lambda)$ sufficiently large (alternatively, α sufficiently small). From the definition of subgradients, we know $L(\bar{x},\lambda)$ is convex for fixed λ, and $L(x,\bar{\lambda})$ is concave for fixed x. Therefore,

$$\begin{aligned} L(\bar{x},\lambda) - L(x,\lambda) &\geq (\overline{x}-x)\,\partial_x L(x,\lambda) \\ L(x,\bar{\lambda}) - L(x,\lambda) &\leq \left(\overline{\lambda}-\lambda\right)\partial_\lambda L(x,\lambda) \end{aligned}$$

Changing the sign and summing we get the relation

$$L(\bar{x},\lambda) - L(x,\bar{\lambda}) \geq -\left(x-\overline{x}\right)^T\partial_x L(x,\lambda) + \left(\lambda-\overline{\lambda}\right)^T\partial_\lambda L(x,\lambda)$$

and particularizing for $x(k)$ and $\lambda(k)$,

$$\Delta(k) \leq L(\bar{x},\lambda(k)) - L(x(k),\bar{\lambda}) + \frac{\alpha}{2}\epsilon \tag{A.5}$$

We need to prove that $L(\bar{x},\lambda(k)) - L(x(k),\bar{\lambda}) \leq 0$ for all $x(k)$ and $\lambda(k)$,

but we know

$$L(\bar{x},\lambda) = f(\overline{x}) + \lambda g(\overline{x}) \leq f(\overline{x})$$
$$L(x,\bar{\lambda}) = f(x) + \overline{\lambda} g(x) \geq f(\overline{x})$$

so it follows.

With relation (A.5) we can now establish the following statements

1. Function $V(k)$ is decreasing along the trajectories of local flow $\mathcal{F} = (K, X, \Lambda, \varphi)$, where $k \in K$, $x \in X$, $\lambda \in \Lambda$, and φ is a state transition function given by the dynamics of equations (A.3)-(A.4), so that

$$\varphi(k_0, x_0, \lambda_0) \subset X, \Lambda \Rightarrow V(k, \varphi(k, x, \lambda)) \leq V(k_0, x_0, \lambda_0)$$

2. Additionally, the local flow $\mathcal{F} = (K, X, \Lambda, \varphi)$ is strictly $\mathcal{F}$-decreasing away from Ω, if it satisfies for every $\epsilon > 0$

$$\varphi(k_0, x_0, \lambda_0) \subset (X, \Lambda) \setminus B(\Omega, \epsilon) \Rightarrow V(k, \varphi(k, x, \lambda)) \leq V(k_0, x_0, \lambda_0) - \gamma(k - k_0)$$

for some function $\gamma(\cdot) = \gamma(\cdot; \epsilon) : \mathbb{R}_+ \to \mathbb{R}_+$, satisfying $\lim_{\tau \to \infty} \gamma(\tau) = \infty$.

A sufficient condition for the Lagrangian to satisfy the second statement, is that functions $f(x)$ and $g_l(x)$ from (A.2) are radially unbounded. Specifically, given a ball $B_\epsilon = \{x, \lambda | d(x, \lambda; \Omega) \leq \epsilon\}$, $L(\bar{x}, \lambda(k)) - L(x(k), \bar{\lambda}) \leq -\frac{\alpha}{2}\epsilon + \gamma(k, \epsilon)$, where $\gamma(k, \epsilon) > 0$ due to radially unbounded functions (closed sublevel sets). This implies that

$$\begin{aligned} V(k+1) &\leq V(k) + \gamma_k(\epsilon) \\ V(k+n) &\leq V(k) + n\gamma_{\min}(\epsilon) \end{aligned}$$

where $\gamma_{\min}(\epsilon) = \min_{k \in [k, k+n]} \gamma(k, \epsilon)$. Because V satisfies the previous two statements for every $k \in K$, it is then a strict generalized Lyapunov function

for $\mathcal{F}$ at Ω on (K, X, Λ).

Finally, we invoke the Stability Theorem from [19], page 223, and because V is a generalized Lyapunov function, we can establish the following results

1. Ω is $\mathcal{F} - invariant$ and stable at any time $k_0 \in K$.

2. V is upper bounded by $V(x_0, \lambda_0)$, and therefore Ω is uniformly stable.

3. V is a strict generalized Lyapunov which is upper bounded, and therefore Ω is uniformly asymptotically stable.

4. Under condition 3, there are lower and upper bound by $W_1(x, \lambda) = W_2(x, \lambda) = V(x(k), \lambda(k))$ that fulfil $W_1(x, \lambda) \leq V(k, x(k), \lambda(k)) \leq W_2(x, \lambda)$ for all $k \in K$, $x \in X$, $\lambda \in \Lambda$ and W_1, W_2 satisfy statements (1)-(2). Therefore, $x \in X$, $\lambda \in \Lambda$ form the basin of attraction of Ω.

□

Bibliography

[1] M. Andrews, A.F. Anta, L. Zhang, and Wenbo Zhao. Routing and scheduling for energy and delay minimization in the powerdown model. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –5, March 2010.

[2] M. Andrews, A.F. Anta, L. Zhang, and Wenbo Zhao. Routing for energy minimization in the speed scaling model. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –9, March 2010.

[3] K.J. Arrow, L. Hurwicz, and H. Uzawa. *Studies in Linear and Non-Linear Programming*. Stanford Mathematical Studies in the Social Sciences. Stanford University Press, 1958.

[4] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Arxiv preprint arXiv:1108.0775*, 2011.

[5] Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski. Optimization with sparsity-inducing penalties. *CoRR*, abs/1108.0775, 2011.

[6] A. Beck and M. Teboulle. Gradient-based algorithms with applications to signal recovery problems. *Convex Optimization in Signal Processing and Communications, ed. DP Palomar and YC Eldar*, pages 42–88, 2010.

[7] D.P. Bertsekas, A. Nedić, and A.E. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific optimization and computation series. Athena Scientific, 2003.

[8] D.P. Bertsekas and J.N. Tsitsiklis. Parallel and distributed computation. 1989.

[9] S.P. Boyd and L. Vandenberghe. *Convex Optimization*. Berichte über verteilte messysteme. Cambridge University Press, 2004.

[10] L. Chiaraviglio, D. Ciullo, M. Meo, and M.A. Marsan. Energy-efficient management of umts access networks. In *Teletraffic Congress, 2009. ITC 21 2009. 21st International*, pages 1 –8, Sept. 2009.

[11] P.L. Combettes, V.R. Wajs, et al. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200, 2006.

[12] A. Conte, A. Feki, L. Chiaraviglio, D. Ciullo, M. Meo, and M.A. Marsan. Cell wilting and blossoming for energy efficiency. *Wireless Communications, IEEE*, 18(5):50 –57, October 2011.

[13] D. Feijer and F. Paganini. Stability of primal-dual gradient dynamics and applications to network optimization. *Automatica*, 46(12):1974–1981, 2010.

[14] P. Frenger, P. Moberg, J. Malmodin, Y. Jading, and I. Godor. Reducing energy consumption in lte with cell dtx. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, May 2011.

[15] F. Giroire, D. Mazauric, J. Moulierac, and B. Onfroy. Minimizing routing energy consumption: from theoretical to practical results. In *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, pages 252–259. IEEE, 2010.

[16] K. Gomez, R. Riggio, T. Rasheed, and F. Granelli. Analysing the energy consumption behaviour of WiFi networks. In *Online Conference on Green Communications (GreenCom), 2011 IEEE*, pages 98 –104, Sept. 2011.

[17] Karina Gomez, Roberto Riggio, Tinku Rasheed, Daniele Miorandi, and Fabrizio Granelli. Energino: A hardware and software solution for energy consumption monitoring. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, pages 311 –317, May 2012.

[18] D. Halperin, B. Greenstein, A. Sheth, and D. Wetherall. Demystifying 802.11 n power consumption. *Proc. HotPower*, 2010.

[19] D. Hinrichsen and A.J. Pritchard. *Mathematical Systems Theory I: Modelling, State Space Analysis, Stability and Robustness*. Texts in Applied Mathematics. Springer, 2011.

[20] F.P. Kelly, A.K. Maulloo, and D.K.H. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research society*, 49(3):237–252, 1998.

[21] D. Leith, V. Subramanian, and K. Duffy. Log-convexity of rate region in 802.11 e wlans. *Communications Letters, IEEE*, 14(1):57–59, 2010.

[22] D.J. Leith and V.G. Subramanian. Utility fairness in 802.11-based wireless mesh networks. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pages 961–968. IEEE, 2010.

[23] X. Lin and N.B. Shroff. Utility maximization for communication networks with multipath routing. *Automatic Control, IEEE Transactions on*, 51(5):766–781, 2006.

[24] Julien Mairal, Rodolphe Jenatton, Guillaume Obozinski, and Francis Bach. Network flow algorithms for structured sparsity. *CoRR*, abs/1008.5209, 2010.

[25] M.A. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo. Optimal energy savings in cellular access networks. In *IEEE International Conference on Communications Workshops*, pages 1–5, 2009.

[26] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *Networking, IEEE/ACM Transactions on*, 8(5):556 –567, oct 2000.

[27] J.J. Moreau. Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93(2):273–299, 1965.

[28] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, pages 323–336, 2008.

[29] Zhisheng Niu, Yiqun Wu, Jie Gong, and Zexi Yang. Cell zooming for cost-efficient green cellular networks. *Communications Magazine, IEEE*, 48(11):74 –79, november 2010.

[30] Eunsung Oh and B. Krishnamachari. Energy savings through dynamic base station switching in cellular wireless access networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1 –5, Dec. 2010.

[31] Roberto Riggio and Douglas J. Leith. A measurement-based model of energy consumption in femtocells. In *Proc. Wireless Days, Dublin*, 2012.

[32] J.J.E. Slotine and W. Li. *Applied nonlinear control*. Prentice Hall, 1991.

[33] W.H. Wang, M. Palaniswami, and S.H. Low. Optimal flow control and routing in multi-path networks. *Performance Evaluation*, 52(2):119 – 132, 2003.