

Habilitation à Diriger des Recherches
Information Flows through Networks:
Models and Algorithms

Laurent Massoulié

April 20, 2010

Jury members:

François Baccelli (INRIA)

Pierre Fraigniaud (CNRS)

Frank Kelly (Cambridge University)

Peter Key (Microsoft Research)

Jean-Yves Leboudec (Ecole Polytechnique Fédérale de Lausanne)

James Roberts (INRIA)

R. Srikant (University of Illinois at Urbana-Champaign)

I acknowledge with great pleasure the influence that François Baccelli, Pierre Fraigniaud, Frank Kelly, Peter Key, Jean-Yves Leboudec, Jim Roberts and R. Srikant have had on my personal research, through their own work and generous guidance. I am greatly honoured that they accepted to sit on this jury, and take this opportunity to express my gratitude.

LM

Contents

1	Introduction	5
2	Rate control	7
2.1	Network model and objectives	7
2.2	Fixed window control and buffers as dual variables	8
2.3	Differential Equation models – Stability and Delays	9
2.4	Multipath transfers: combinations in series and parallel	11
3	Flow-level models of Internet rate control	15
3.1	Basic models and stability	15
3.2	Instability	17
3.3	queueing networks and explicit formulas	18
3.4	Insensitivity and large deviations	19
3.5	Concluding remarks	20
4	Overlays and random graphs	21
4.1	Self-scaling graph growth	21
4.2	Graph rebalancing and Metropolis algorithms	22
4.3	Concluding remarks	23
5	Network Epidemics	25
5.1	SIS Epidemics: impact of topology	25
5.1.1	Spectral radius and fast extinction	26
5.1.2	Isoperimetric constants and long survival	26
5.1.3	Concluding remarks	27
5.2	Simple gossip with competing rumours	27
5.3	Concluding remarks	28
6	P2P live streaming	29
6.1	Rate optimality	29
6.1.1	Edge constraints	29
6.1.2	Node constraints	30
6.2	Delay performance	31
6.2.1	Uniform capacities	31
6.2.2	Heterogeneous capacities	32
6.3	Concluding remarks	33
7	Perspectives	35

Chapter 1

Introduction

As its title suggests, the common theme in the topics selected for this thesis is that of information flows in communication networks. The focus is more specifically on “wired networks”, composed of a collection of distinct physical links, where physical transmission along one link does not interfere with transmission along other links (in contrast to “wireless” communication). The topics are organised in a bottom-up fashion, starting from the lower layers of the network protocol stack and going upwards.

Chapters 2 and 3 can be roughly mapped to the transport layer. In particular, Chapter 2 introduces models of rate control motivated by the design of transport protocols for the Internet. It highlights two results. The first is an analysis of “fixed window” congestion control, featuring a formal interpretation of buffer levels as dual variables of an optimisation problem. This illustrates how simple distributed rules may lead to a global optimisation. Such a connection, while well established in statistical physics, is perhaps more surprising in the present context of data networks. The second result of Chapter 2 concerns stability of rate control rules in the presence of delays. Appealing sufficient conditions for stability are provided, which have a particular, decoupled form, making them readily usable in a distributed setting. Chapter 2 finally touches upon the subject of rate control for general multipath flows, which could be deployed at the application layer, on top of one of the overlay topologies discussed in Chapter 4.

Chapter 3 then surveys results on performance of so-called flow-level models of rate control. In such models, individual information flows are initialised over time, and terminated as soon as a corresponding piece of information is successfully transferred. One first result in this chapter is that, for sufficiently “fair” rate control, the stability region of the network is the largest possible. In other words, fair rate allocations ensure efficient use of networks for document transfers. Another set of results is of a negative nature: for “unfair” allocations, this optimality may be lost. This is shown in particular when prioritisation among classes is introduced. It is also shown when transfers proceed along multiple paths, in the absence of synchronisation among the rate controllers for the distinct paths used by individual transfers. Finally, performance evaluations, either exact or asymptotically accurate, depending on the congestion controller in use, are described.

Chapter 4 deliberately moves above the transport layer, going to the application layer. It describes simple strategies for building and adjusting “overlays”, that are logical graphs connecting network users. With objectives of robustness in mind, the first strategy addresses “graph growth”, i.e. incorporation of incoming nodes into the existing overlay. It succeeds in automatically tuning the average degree in the graph to the logarithm of the number of nodes, a desirable property for ensuring robustness. A second strategy aims at rewiring an existing graph. It is an instance of

the Metropolis-Hastings sampling strategy, shown to be implementable in a distributed fashion when the “graph energy” to be minimised has a specific form. Robustness guarantees of the random graphs resulting from specific energy functions are described.

Chapter 5 deals with two types of epidemic processes on graphs. Such graphs could be overlays constructed as in Chapter 4, and “infection” in such processes is to be interpreted as ownership of information. One process aims to maintain information alive in a network of forgetful nodes; such forgetfulness might just be how memory management of network-attached nodes has been designed. Alternatively, it might result from node failures, in failure-prone environments such as sensor networks. Sufficient conditions for epidemics to either last long or die fast are established, based on simple graph topology descriptors. The second epidemic process considered in this chapter is a “pure growth” process where only infections occur. It provides tight characterisations of the epidemic outreach, to be leveraged in the next chapter.

Chapter 6 focuses on peer-to-peer applications, and more specifically live streaming, to be deployed at the application layer on top of an overlay. It describes strategies for selection of which data to transmit to what neighbour, as well as associated performance results. In particular, rate-optimality is established under several models of network capacity constraints. Joint rate- and delay-optimality is established for a particular scheme, the so-called “random peer – latest useful chunk” strategy, leveraging the results of the previous chapter, in networks with homogeneous peer capacities. Finally, lower bounds on the delay in heterogeneous scenarios are described.

Perspectives are envisioned in Chapter 7. Ongoing activity on algorithmic design for peer-to-peer video-on-demand applications are mentioned, as well as open questions on design of stable schemes for peer-to-peer live streaming when “network coding” is in use.

Chapter 2

Rate control

2.1 Network model and objectives

Rate control, also known as congestion control, is fundamental to the Internet operation. In the current Internet it is essentially performed by TCP, the Transmission Control Protocol, and regulates the speed at which requested data is sent to each requestor. To fix ideas, one can view the network as a set of links $\ell \in \mathcal{L}$, each with capacity C_ℓ . Candidate information transfers can then be thought of as proceeding along a given sequence of such links, from the originating data server to the requesting data receiver.

While data flows through the Internet in discrete units of packets, a useful abstraction consists in representing the status of a particular data transfer by a single number, the net rate at which information is accrued at one particular receiver. Thus, if there are n_r current data transfers making use of the subset r of network links, and if each such data transfer proceeds at some rate x_r , the natural capacity constraints read

$$\sum_{r:\ell \in r} n_r x_r \leq C_\ell, \ell \in \mathcal{L}. \quad (2.1)$$

The set of equations (2.1) characterizes what is often referred to as the *schedulable region* of the network: the rate vector $\{n_r x_r\}$ belongs to the schedulable region Λ if and only if (2.1) holds.

Early research on rate control (late eighties to late nineties) was mostly concerned with finding algorithms that would let such rates x_r evolve towards a pre-defined vector of target values x_r^* . The target generally considered was the vector of so-called *max-min fair* rate allocations, defined axiomatically. Namely, it should maximise the smallest rate, and then the second smallest rate, etc. [1].

A first alternative was proposed by F. Kelly [19] under the form of *proportional fairness*, defined through another set of axioms. Further generalizations were proposed in Kelly, Maullo and Tan [20] under the form of utility maximizing allocations, namely the target rates x_r^* should achieve the optimum in the maximization problem

$$\text{Maximise } \sum_r n_r U_r(x_r) \quad (2.2)$$

$$\text{Over } x_r \geq 0 \quad (2.3)$$

$$\text{Such that } \{n_r x_r\} \in \Lambda. \quad (2.4)$$

The utility functions U_r in (2.2) are typically assumed non-decreasing and concave. The above

problem is an exact analogue of the welfare maximisation problem, central in the field of micro-economics.

The so-called proportionally fair allocation corresponds to the particular choice where $U_r(x) = \log(x)$ for all r in the above equation.

2.2 Fixed window control and buffers as dual variables

A classical rate control method consists in so-called *fixed window control*: each type r -transfer is assigned a window parameter w_r , which specifies the amount of data that the data sender is willing to pour into the network before such data is acknowledged by the receiver. This requires the receiver to send back acknowledgements to the sender.

A first contribution to rate control algorithms made in [31] consisted in a characterisation of the equilibrium rate allocations x_r^* that would result from such fixed window rate control algorithms. We reproduce part of the argument below.

Assume that for each type r flow, and each link $\ell \in r$ it traverses, a corresponding amount $B_{\ell,r}$ of data is buffered at the ingress of link ℓ . Assume further FIFO (First-in First-out) queueing, and homogeneous interleaving of all buffered contents for distinct r , and finally, let τ_r be the round-trip time it takes for data to flow from the source to the destination and then be acknowledged back at the source, for type r flows.

Characterisation of the equilibrium rates under such assumptions proceeds along the following lines. For some type r flow, the amount of information sent and not yet acknowledged, w_r , verifies

$$w_r = \tau_r x_r + \sum_{\ell \in r} B_{\ell,r}. \quad (2.5)$$

Furthermore, FIFO queueing entails that for each link ℓ , introducing the rate through ℓ , that is $x^\ell := \sum_{r:\ell \in r} n_r x_r$ and the total amount of data buffered at ℓ , $B_\ell = \sum_{r:\ell \in r} n_r B_{\ell,r}$, the following holds:

$$\begin{aligned} x^\ell &\leq C_\ell, \\ x^\ell < C_\ell &\Rightarrow B_\ell = 0, \\ B_\ell > 0 &\Rightarrow x_r \equiv C_\ell \frac{B_{\ell,r}}{B_\ell}, \quad r \ni \ell. \end{aligned} \quad (2.6)$$

Consider then the optimization problem (2.2–2.4), with utility function

$$U_r(x) := w_r \log(x_r) - \tau_r x_r. \quad (2.7)$$

Then the Karush-Kuhn-Tucker (KKT) conditions entail that optimal vectors $\{x_r\}$ are characterised by the existence of non-negative multipliers μ_ℓ such that

$$\begin{aligned} \frac{w_r}{x_r} - \tau_r &= \sum_{\ell \in r} \mu_\ell, \\ \mu_\ell (C_\ell - x^\ell) &= 0, \quad \ell \in \mathcal{L}. \end{aligned} \quad (2.8)$$

Letting $\mu_\ell = B_\ell/C_\ell$, Equation (2.6) gives $B_{\ell,r} = x_r \mu_\ell$. Plugging this expression in (2.5), one retrieves the KKT conditions (2.8), hence the identification of the stationary points (2.5–2.6) with the rates maximising (2.2–2.4) for the utility function (2.7).

A couple of remarks are in order. This result extends naturally to non-FIFO queues. For instance, per flow queueing with a ‘‘Serve the Longest Queue’’ policy yields quadratic utility functions. More generally, given some utility functions U_r , per-flow scheduling policies based on queue lengths can be invented to achieve the corresponding welfare maximisation.

This result provides a sharp illustration that backlog variables (here, the buffer sizes B_ℓ) can correspond to dual variables of the corresponding capacity constraints in a precise sense. Finally,

while the above argument says nothing about convergence of the dynamics of fixed window control to the fixed points discussed above, in the case of zero round trip times, recent work by Walton [48] shows that for some stochastic model of fixed window control (a particular closed multiclass queuing network), such convergence holds in the limit where packet sizes become small.

2.3 Differential Equation models – Stability and Delays

TCP does implement window-based congestion control, however the window there is not kept fixed, but is instead continuously updated. Specifically, it keeps increasing until data loss occurs, at which time the window is reduced. The corresponding dynamics are very complex, and to get insight into their behaviour, dynamical models based on differential equations have proven extremely useful. Such models have been introduced in [20], together with the utility maximisation objective (2.2–2.4). One particular model introduced there, coined as the “willingness to pay” rate control strategy, amounted to let rate x_r evolve according to

$$\frac{d}{dt}x_r(t) = \kappa_r \left[w_r - x_r(t) \sum_{\ell \in r} p_\ell \left(\sum_{s \ni r} n_s x_s(t) \right) \right]. \quad (2.9)$$

In the above, κ_r is a gain parameter, controlling the speed at which updates occur, w_r is an additive increase term, and the function $x \rightarrow p_\ell(x)$ is meant to represent the rate at which losses occur at link ℓ when it is handling traffic at rate x . Another interpretation of the function $x \rightarrow p_\ell(x)$, more in line with micro-economics, is as the per-packet *price* when the link serves traffic at rate x .

A heuristic interpretation of this equation is as follows. Upon each acknowledgement of a packet being received, rate is increased by some amount $\kappa_r w_r / x_r$, while upon acknowledgement of a packet being lost, rate is decreased by some amount $\kappa_r (1 - w_r / x_r)$.

As is readily shown, the dynamics (2.9) have trajectories along which the welfare function

$$\mathcal{W}(x) := \sum_r w_r \log(x_r) - \sum_{\ell \in \mathcal{L}} \int_0^{x_\ell} p_\ell(y) dy$$

is non-decreasing. It is moreover strictly increasing unless the rates x_r have reached a maximum of \mathcal{W} (local maxima are ruled out when $x \rightarrow p_\ell(x)$ is non-decreasing for all ℓ , for then function \mathcal{W} is strictly concave). Maximisation of \mathcal{W} can be thought of as an approximation to the optimisation problem (2.2–2.4) in which the hard capacity constraints (2.4) have been relaxed via the introduction of penalty functions p_ℓ .

Similar models have been used to interpret the rate allocation obtained by TCP as performing some utility maximisation for suitable utility functions [25]. However, practice contradicts the prediction of such models, namely smooth convergence and stabilisation to an equilibrium point. Apart from the fact that window updates in TCP always lead to substantial changes (preventing rest at equilibrium), another potential explanation investigated in [13] is that delays in feedback upon which updates are made might lead to over-reaction and hence forced oscillations.

To investigate such matters, simple modifications of Equation (2.9) consist in incorporating delays in the right-hand side. Denote by $D_{r\ell}$ the time for information to flow from data source r to link ℓ , and $D_{\ell r}$ the time for information to flow from link ℓ to the destination of flow r and then be fed back to the source of r . Noting as before τ_r the round-trip delay for type r traffic, for any link ℓ traversed by r one has the relation

$$\tau_r = D_{r\ell} + D_{\ell r}. \quad (2.10)$$

A natural modification of (2.9) would then be to replace $x_s(t)$ by $x_s(t - D_{s\ell} - D_{\ell r})$ in the right-hand side.

For given delay parameters $D_{r\ell}$, $D_{\ell r}$, upon reducing the gains κ_r by some factor N and then speeding up time by the same factor N , one arrives at the same equations (2.9), with delays divided by N . Hence, for small enough gains, the effect of delays should be diminished, and convergence to equilibrium should be ensured. The question raised in [13] was then whether suitable gains κ_r could be determined, for which indeed convergence holds in there.

To be of practical use, the gain values κ_r leading to non-oscillatory convergence should further be calculable from information directly available to the source of type r traffic.

To make progress on this set of issues, given the complexity of non-linear delay-differential equations, a first step consists in trying to establish non-oscillatory convergence in the close vicinity of the equilibrium, focusing on a linearised version of the original delay-differential equations in the vicinity of the equilibrium.

The linearised delay-differential equations for the perturbations $y_r(t) := x_r(t) - x_r^*$ around the equilibria x_r^* take the following form:

$$\frac{d}{dt}y_r(t) = -\kappa_r \left[y_r(t - \tau_r)\bar{p}_r + \bar{x}_r \sum_{\ell \in r} \sum_{s \ni \ell} \bar{p}'_\ell n_s y_s(t - D_{s\ell} - D_{\ell r}) \right], \quad (2.11)$$

where \bar{p}_r and \bar{p}'_ℓ are defined as

$$\bar{p}_r := \sum_{\ell \in r} p_\ell \left(\sum_{s \ni \ell} n_s x_s^* \right), \quad \bar{p}'_\ell := p'_\ell \left(\sum_{s \ni \ell} n_s x_s^* \right).$$

Then one result proven in [27] is that the delay-differential system (2.11) is stable provided the following conditions hold:

$$\kappa_r \tau_r \left(\bar{p}_r + \sum_{\ell \in r} \bar{p}'_\ell \sum_{s \ni \ell} n_s x_s^* \right) < 1. \quad (2.12)$$

The appeal of such conditions is that these can be met by tuning the gain κ_r to quantities related to the corresponding network route r , namely the round-trip delay τ_r , the equilibrium “price” \bar{p}_r , and the aggregate over the links ℓ in route r of the rate - price sensitivity products $\bar{p}'_\ell \sum_{s \ni \ell} n_s x_s^*$. All such quantities can be estimated by incrementing dedicated fields in packet headers as they traverse links by the suitable characteristic at that link.

The proof of that result developed in [27] relied on the following line of argument. Stability is known to be equivalent to the absence of roots z of a particular characteristic equation $F(z) = 0$ in the right-half of the complex plane. This equation $F(z) = 0$ characterises the exponents z of exponential solutions $y(t) = e^{zt}y(0)$ of the delay-differential equation (2.12).

Moreover, the locus of such roots varies continuously with the delay parameters τ_r , and for zero delays, roots are necessarily located in the left half of the complex plane. Thus it is enough to show that for any delays τ_r verifying Condition (2.12), the characteristic equation admits no purely imaginary roots $z = i\omega$, $\omega \in \mathbb{R}$. This is established by contradiction, using elementary algebraic manipulations.

This result was the first to provide such “distributed” conditions on control gains for stability of delay-differential systems such as (2.11) in the case of arbitrary delay parameters τ_r . In the process of obtaining it, simple conditions for stability of general delay-differential systems were obtained. One particular such result is the following

Theorem 2.1. *The system*

$$\frac{d}{dt}x_r(t) = - \sum_s M_{rs} x_s(t - \tau_r) \quad (2.13)$$

is stable, provided the matrix M is Hermitian positive definite, and the spectral radius $\rho(\tau M)$ of the matrix $\tau M := (\tau_r M_{rs})_{r,s}$ verifies

$$\rho(\tau M) < 1. \quad (2.14)$$

This result was further extended to establish that stability still holds with the right-hand side of (2.14) set to $\pi/2$ rather than 1. This strengthening of the above result, obtained by Glenn Vinnicombe [45], is essentially optimal. The results developed in [27] and perfected in [45] have been extensively used in subsequent work on design of stable congestion controllers based on models of delay-differential systems.

2.4 Multipath transfers: combinations in series and parallel

So far we have considered only “single path” data transfers, motivated by the problem of Internet congestion control as performed by the transport layer. There is a strong interest in developing efficient rate control algorithms leveraging not just one network path, but several. The most studied extension consists in leveraging, between the source and destination of data for a particular traffic type r several paths $p \in \mathcal{P}(r)$. For instance, Internet users could have dual connectivity via two service providers, in which case it is natural to try and accrue data along each of the corresponding Internet paths. This corresponds to *parallel* multiple paths.

It is also possible to combine network paths in *series*: a source node s may send data to a destination node d via some intermediary node i , through a combination of paths r_1 connecting s to i and r_2 connecting i to d . Indeed, splitting of network paths in this manner is advantageous for achieving shorter delays in control loops, and is used in the case of mobile networks.

What comes next should now be obvious: let us combine network paths both in series and in parallel! In practice, this could be achieved in the following context. Given a collection of network users i belonging to some group V , they may set up logical connections among themselves, which is formally captured by a collection E of edges $e = (i, j) \in E$. The construction of such logical graphs, usually called overlays, will be discussed in Chapter 4. In [23] we proposed mechanisms for performing rate allocation in this context, so as to perform utility maximisation as in (2.2–2.4), which we describe now.

Given a set of flows f , assume that for each flow f , with source $s(f)$ and destination $d(f)$, a collection of paths $\pi \in \Pi(f)$ can be used, where each path p is identified by its source and destination. A path π with source node $s(\pi) = i$ and destination node $d(\pi) = j$ is also denoted (ij) .

We denote by x_{ij}^f the rate sent for flow f along path (ij) , and by W_i^f the amount of data buffered at i for flow f , with the convention that

$$W_{s(f)}^f = W_{d(f)}^f = 0.$$

In this context, the utility maximization problem that we would like rates x_{ij}^f to solve is the

following:

$$\text{maximize } \sum_f U_f(x^f) - \Gamma(y) \quad (2.15)$$

$$\text{over } x^f \geq 0, x_{ij}^f \geq 0, (ij) \in \Pi(f) \quad (2.16)$$

$$\text{under } x^f = \sum_{j:(s(f)j) \in \Pi(f)} x_{s(f)j}^f, \quad (2.17)$$

$$i \notin \{s(f), d(f)\} \Rightarrow \sum_j x_{ij}^f = \sum_j x_{ji}^f, \quad (2.18)$$

$$y_{ij} = \sum_f x_{ij}^f. \quad (2.19)$$

The penultimate constraint specifies that the path rates x_{ij}^f satisfy flow conservation at intermediate, relay nodes, i.e. they define a *flow* from $s(f)$ to $d(f)$. Also, the term $\Gamma(y)$ in the first equation (2.15) represents the overall network cost, which could consist in a sum of penalty functions associated with individual network links, as in the previous relaxation.

We then consider the following scheme: for each path $(ij) \in \Pi(f)$, we set

$$\dot{x}_{ij}^f = \kappa_{ij}^f \left[\mathbf{1}_{i=s(f)} U_f' \left(\sum_{k:(ik) \in \Pi(f)} x_{ik}^f \right) - p_{ij} + \phi(W_i^f) - \phi(W_j^f) \right] \quad (2.20)$$

where ϕ is some continuous, strictly increasing function, such that $\phi(0) = 0$. In the above, κ_{ij}^f is a positive gain parameter, and p_{ij} denotes the marginal cost of sending along path (ij) , that is

$$p_{ij} = \frac{\partial}{\partial y_{ij}} \Gamma(y) \quad (2.21)$$

where y is the vector of path rates and Γ is the network cost function.

Note that the adaptation rule (2.20) could require some x_{ij}^f to remain positive while there is no data to forward from node i to node j at that particular time. We address this issue as follows. The actual sending rate for flow f along edge (ij) is denoted y_{ij}^f , and defined as

$$y_{ij}^f = \beta_i^f x_{ij}^f, \quad (2.22)$$

where the adjustment variable β_i^f is such that

$$\beta_i^f \in [0, 1], W_i^f > 0 \Rightarrow \beta_i^f = 1, W_i^f = 0 \Rightarrow \beta_i^f = \min \left(1, \frac{\sum_j y_{ji}^f}{\sum_j x_{ij}^f} \right). \quad (2.23)$$

Thus, the vector y of path rates used in the definition of marginal costs in (2.21) reads

$$y_{ij} = \sum_f y_{ij}^f. \quad (2.24)$$

Finally, for each i , we have

$$\dot{W}_i^f = \sum_{j:(ji) \in \Pi(f)} y_{ji}^f - \sum_{j:(ij) \in \Pi(f)} y_{ij}^f. \quad (2.25)$$

One technical point has been overlooked in the above description. The quantities x_{ij}^f should remain non-negative, a property that is not guaranteed for solutions of the ODE's (2.20–2.25). This can be addressed by adding to the right-hand side of (2.20) a time-dependent, non-negative term u_{ij}^f such that $u_{ij}^f = 0$ if $x_{ij}^f > 0$.

It is established in [23] that the stationary points of (2.20–2.25) are solutions of (2.15–2.19).

We reproduce the argument here. First, setting $W_i^f, i \notin \{s(f), d(f)\}$, to zero gives

$$\begin{aligned} \sum_j y_{ji}^f &= \sum_j x_{ij}^f, \\ W_i^f > 0 &\Rightarrow y_{ij}^f = x_{ij}^f. \end{aligned} \quad (2.26)$$

Setting x_{ij}^f to zero yields

$$\begin{aligned} \mathbf{1}_{i=s(f)} U_f'(x^f) + \phi(W_i^f) - p_{ij} - \phi(W_j^f) &\leq 0, \\ x_{ij}^f > 0 &\Rightarrow \mathbf{1}_{i=s(f)} U_f'(x^f) + \phi(W_i^f) - p_{ij} - \phi(W_j^f) = 0. \end{aligned} \quad (2.27)$$

By (2.26), nodes i at which $\sum_j y_{ji}^f < \sum_j x_{ij}^f$ are such that $W_i^f = 0$. In view of (2.27), at such nodes i , it holds that either $x_{ij}^f = 0$, or $W_i^f = p_{ij} = 0$ for all outgoing paths $(ij) \in \Pi(f)$.

Note that necessarily, the equilibrium quantities y_{ij}^f define a flow from $s(f)$ to $d(f)$. By (2.27), at paths $(ij), i \neq s(f)$, at which x_{ij}^f , and hence y_{ij}^f is positive, it holds that $\phi(W_i^f) = \phi(W_j^f) + p_{ij}$. Thus, at each concatenation of paths $(s(f)i_1), (i_1i_2), \dots, i_md(f)$ along which all corresponding rates x_{ij}^f are positive, then by the previous argument it holds that

$$U_f'(x^f) = p_{s(f)i_1} + p_{i_1i_2} + \dots + p_{i_md(f)}.$$

Furthermore, by the second part of (2.27), any concatenation of paths $(s(f)i_1), \dots, (i_md(f))$ along which some rate x_{ij}^f equals zero is such that

$$U_f'(x^f) \leq p_{s(f)i_1} + p_{i_1i_2} + \dots + p_{i_md(f)}.$$

These last two properties coincide with the KKT optimality condition for the welfare maximization problem (2.15–2.19). Thus, any equilibrium state for (2.20–2.25) provides a solution to this maximization problem.

We believe that a stronger result holds, namely that the above dynamical system converges asymptotically to solutions of this maximization problem. Lyapunov function techniques described in the work of Voice [46], and the books of Srikant [41] and Georgiadis, Neely and Tassiulas [12] could be adopted to the present framework. Indeed, the algorithm (2.20) is a type of back-pressure algorithm, where transformed delays, $\phi(W_i^f)$ are used to summarize or communicate downstream prices. The function ϕ allows scaling of the data queues, although the choice of ϕ has implications for the stability and the choice of gain parameter in the presence of delayed feedback. The main difference with classical backpressure algorithms studied in [12] lies in the fact that in the above schemes, distinct flows have rate allocations coupled indirectly via the underlying price signals p_{ij} , while in the classical approach of cross-layer control, a schedule is constructed in a different manner, via the solution of a maximum weight matching problem.

Chapter 3

Flow-level models of Internet rate control

A huge amount of work has dealt with the design of rate control algorithms for driving rates x_r towards intended targets x_r^* . In comparison, fairly little work has been devoted to identify suitable targets x_r^* . This is one primary purpose of the flow-level perspective taken in this chapter. A second objective is to determine network dimensioning rules.

The flow-level modeling approach considers the dynamics of data transfer initiations and completions. Thus, in contrast to the previous chapter, the number n_r of ongoing type r transfers is no longer fixed. Several kinds of data transfers can be considered. For most of the discussion we will focus on document transfers for which the volume to be transmitted is given, and the time may vary according to the rate at which transfer proceeds. Another important category is that of so-called real time transfers (e.g. phone conversations) whose duration is exogeneous to the network operation, and whose quality varies with the obtained rate. Pointers to modeling and analysis of rate control for such transfers are provided at the end of the chapter.

3.1 Basic models and stability

Assume that some rate control objective is given, which defines rate values $x_r(n) = x_r^*$ to be targeted by rate control, given the number of ongoing transfers $n = (n_r)$. The basic model of flow-level performance we consider is the following. Assume new type r transfers are initiated at the instants of a Poisson process with rate ν_r , and require a random volume of service that admits an exponential distribution with rate μ_r . Assume further to be given an ideal rate control algorithm, which achieves the target rate allocations $x_r(n)$ instantaneously after each change in the vector of flow counts n . One might interpret this assumption as one of time scale separation. This would be accurate if rate control succeeds in driving rates towards the targets $x_r(n)$, and does so on a much faster time scale than that at which new flows arrive or leave.

This modeling approach has first appeared in [36], where the so-called max-min fairness criterion was considered. It provides a Markov process for describing the evolution of flow counts, with non-zero transition rates

$$\begin{aligned} q(n, n + e_r) &= \nu_r, \\ q(n, n - e_r) &= \mu_r x_r(n), \end{aligned} \tag{3.1}$$

where e_r is the r -th unit vector. One would then like to analyse this process to answer questions

regarding the resulting performance of file transfers. For instance, what is the average transfer time in steady state? A prerequisite is to determine under which conditions this Markov process is ergodic.

Recall the notion of network schedulable region Λ , which is the set of rate vectors that are feasible given the network communication resources. Under the previous model of types r being associated with a fixed set of network links ℓ , each with capacity C_ℓ , this reads:

$$\Lambda = \left\{ \lambda_r : \sum_{r \ni \ell} \lambda_r \leq C_\ell \right\}.$$

A key result on stability of flow-level models was proven in [2]. It focuses on rate allocations specified by the utility maximisation principle (2.2–2.4), for utility functions given by

$$U_r(x) = w_r \frac{x^{1-\alpha}}{1-\alpha},$$

for some positive weights w_r and exponent α . It also covers the case where $\alpha = 1$, for which the definition of $U_r(x)$ is modified to $w_r \log(x)$. This family of utility functions defines the so-called *weighted α -fairness*, introduced by Mo and Walrand [34].

The stability result of [2] states that the above Markov process is stable (i.e. ergodic) when the vector ρ of loads $\rho_r := \nu_r/\mu_r$ belongs to the interior $\overset{\circ}{\Lambda}$ of the schedulable region Λ . Since it happens that this is also a necessary condition for stability, this result entails that rate allocations according to weighted α -fairness maximise the stability region.

The proof of this result relies on the general technique of *fluid limits* of Rybko and Stolyar [39] and Dai [6], which requires proving convergence to zero in finite time of so-called fluid dynamical systems, obtained from the original Markov process by joint rescaling of time and space.

For the Markov process as specified by (3.1), the corresponding fluid dynamical system is specified by

$$\frac{d}{dt} n_r(t) = \nu_r - n_r x_r(n(t)). \quad (3.2)$$

Convergence to zero in finite time is then established by exhibiting a suitable Lyapunov function. In the present case, it is given by

$$L(n) := \sum_{r \in \mathcal{R}} \frac{1}{\mu_r} \int_0^{n_r} U'_r(\rho_r/x) dx. \quad (3.3)$$

The proof that it decreases along the trajectories of (3.2) proceeds by noting that for any n , the function $F(t) := \sum_r n_r U_r(tx_r(n) + (1-t)\rho_r/n_r)$ is concave, maximal at $t = 1$ by definition of the welfare-optimising rates $x_r(n)$, and hence the derivative of F at $t = 0$ must be non-negative. However this derivative reads

$$\frac{d}{dt} F(0) = \sum_r n_r U'_r(\rho_r/n_r)(x_r(n) - \rho_r/n_r),$$

and the latter is easily seen to coincide with the negation of the time derivative of $t \rightarrow L(n(t))$.

This argument for proving stability has been extended and re-used in a number of papers. Originally constructed for the utility functions used to define α -fairness, it has been adapted to general concave utility functions (as given here) by Ye [49]. Further generalisations are given in [22].

A few more remarks are in order. This stability result is valid for arbitrary schedulable regions Λ , provided they are non-increasing and convex. An interesting question is whether the same

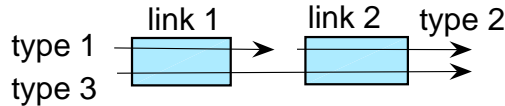


Figure 3.1: Example of a two-link network supporting three types of users

stability result would hold if the vector $(n_r x_r)$ of rates per class were constrained to lie in some arbitrary set Λ' , upon setting Λ equal to the convex hull of Λ' . As pointed out by R. Srikant (personal communication), a result in this direction has recently been established by S. Stolyar [42].

3.2 Instability

The previous result states that weighted α -fair rate allocations ensure maximal stability. To better appreciate the result, it is useful to note that stability is compromised under alternative rate allocation strategies. The first example is provided by priority disciplines. Consider the network in Picture 3.1 consisting of two links of unit capacity, and three transfer types $r = 1, 2, 3$.

Then the load vector ρ belongs to the interior of the capacity region if and only if $\rho_3 < \max(1 - \rho_1, 1 - \rho_2)$. Consider then the rate allocation policy which sets x_3 to zero whenever n_1 or n_2 are non-zero, in which case type 1 and type 2 flows fully utilise link 1 and link 2 respectively, and sets x_3 to $1/n_3$ if both n_1 and n_2 equal zero. In other words, priority is given to flows of types 1,2 over flows of type 3.

Then, as shown in [2], stability holds under this allocation if and only if $\rho_3 < (1 - \rho_1) \times (1 - \rho_2)$, a condition strictly more stringent than the previous one. This establishes that some amount of fairness might be required in order to ensure maximal stability.

A second example comes from multipath transfers. With multipath, a given transfer is allowed to proceed along several network paths. We model this by introducing for each flow type r a corresponding set \mathcal{P}_r of paths $p \in \mathcal{P}_r$ that can be used to send data. Then type r flows would proceed at rate x_r written as

$$x_r = \sum_{p \in \mathcal{P}_r} x_p, \quad (3.4)$$

where x_p is the per-flow rate used along path p , and hence subject to the constraints

$$\sum_{p \ni \ell} n_{r(p)} x_p \leq C_\ell, \quad \ell \in \mathcal{L}, \quad (3.5)$$

where $r(p)$ is the flow type using path p . The utility maximising allocation $x_r(n)$ is naturally generalised to this case as the allocation maximising the sum of utilities (2.2) over the non-negative variables x_r, x_p , under the constraints (3.4–3.5).

Various rate control schemes have been developed to solve this optimisation problem, which we refer to as the *coordinated* multipath objective. In particular, differential equation models have been proposed by Kelly and Voice [17], and shown to converge to the desired allocations.

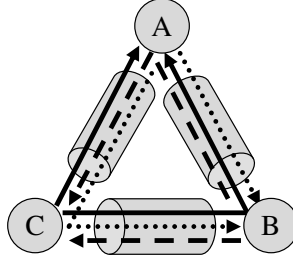


Figure 3.2: Example network where the use of parallel uncoordinated connections is inefficient

In this context, the corresponding schedulable region is defined as

$$\Lambda = \{\lambda_r \geq 0 : \exists \lambda_p \geq 0 \text{ such that } \sum_{p \in \mathcal{P}_r} \lambda_p = \lambda_r, \sum_{p \ni \ell} \lambda_p \leq C_\ell\}. \quad (3.6)$$

The previous ergodicity result still applies here: provided the load vector ρ belongs to the interior of Λ , stability of the Markov process holds.

However, instead of developing new congestion controllers to regulate rates x_p along each individual path $p \in \mathcal{P}_r$ in a coordinated fashion, a lazier approach would consist in letting simpler rate control mechanisms regulate such rates individually without further coordination.

The corresponding, uncoordinated approach, would produce path rates x_p solving a different optimisation problem, namely

$$\text{Maximise } \sum_p n_{r(p)} U_p(x_p) \quad (3.7)$$

under constraints (3.5), where n_p is the number of flows using path p .

One would thus like to know whether coordination brings any benefit. The following toy example shows that it does. The triangle network in Figure 3.2

has 3 traffic types, each offered a direct path through one link and an indirect path through the two other links. Assuming the offered loads for all three types are equal to a single parameter ρ , under coordinated rate control, stability holds under the condition $\rho < 1$. This would in fact be the same condition if the alternative two-hop paths were not used at all.

However, under uncoordinated rate control, with path utility functions $U_p(x) \equiv \frac{x^{1-\alpha}}{1-\alpha}$, the stability condition now reads:

$$\rho < \frac{1 + 2^{-1/\alpha}}{1 + 2^{1-1/\alpha}}.$$

Specialising further, for $\alpha = 2$, which is a reasonable model for the allocations performed by the dominant rate control in today's Internet (TCP Reno's; see Kunnyur and Srikant [25]) yields a stability condition of $\rho < 1/\sqrt{2} \approx 0.71$.

Thus, even by only looking at stability properties, one can make the following points. Both fairness (as opposed to discrimination, in the form of priorities) and coordination (among multiple paths, as opposed to relying on current rate controllers for each individual path, without further coordination) are necessary for optimal stability.

It is generally difficult to obtain closed-form expressions for performance indices of interest in relation to the previous model. We mention a few results available.

3.3 queueing networks and explicit formulas

First, there is a simple model which yields explicit performance formulas for arbitrary network topologies, assuming single paths are used per traffic type. To describe it, imagine the following. Each new connection implements window-control, with a window of one packet of data. A packet is sent on its route through the network, and is served at each link according to the so-called Processor-Sharing policy. When it has gone through all links of its route, a new packet is immediately generated at the origin of the route. This proceeds until the last packet of the file to be transmitted has crossed all links, at which point transfer is completed.

This simple strategy leads to a model that is entirely tractable, being a special case of a network of so-called *symmetric queues*, as per the terminology of Kelly [15]. Taking packet sizes to be 1, and letting σ_r be the average number of packets needed for type r transfers, ν_r still being the rate at which such transfers are initiated, one has the following performance formulas.

In steady state, the number X_ℓ of packets under service by link ℓ follows a geometric distribution with parameter $\sum_{r \ni \ell} \rho_r / C_\ell$, where $\rho_r = \nu_r \sigma_r$, the variables X_ℓ being mutually independent. Furthermore, conditionally on X_ℓ , the composition of the queue at link ℓ follows a multinomial distribution with parameters $\rho_r / \sum_{s \ni \ell} \rho_s$, $r \ni \ell$.

Thus, the average number of ongoing type r transfers, \bar{N}_r , is given at once by:

$$\bar{N}_r = \nu_r \sum_{\ell \in r} \frac{\sigma_r}{C_\ell - \sum_{s \ni r} \rho_s}.$$

As a result, Little's law yields that the average sojourn time S_r is precisely

$$S_r = \sigma_r \sum_{\ell \in r} \frac{1}{C_\ell - \sum_{s \ni r} \rho_s}. \quad (3.8)$$

This line of argument was proposed by the author shortly after the publication of [31], inspired by the treatment of fixed window control performed in there. Several subsequent developments have been made: Bonald and Proutière [4] characterised explicitly the allocation that results if one freezes the number of “spinning packets” and measures the long term transmission rate thereby achieved, in terms of closed queueing networks. Such closed queueing networks have been further analysed by Walton [48] to establish convergence properties, and characterisation of large deviations, when the number of spinning packets is made large.

One extension of such queueing networks to address performance at flow level when multiple paths can be used, has been proposed in [16], based on the characterisation of the schedulable region for multipath in terms of generalised cut constraints. This again leads to pleasingly simple formulas such as (3.8), where summation is over generalised constraints impacting type r transfers, and each term is further weighted by some factor appearing in the expression of the corresponding constraint.

3.4 Insensitivity and large deviations

The previous allocation, corresponding to spinning networks, has the appealing property of leading to an *insensitive* model, that is the steady state distribution depends on the service size distribution only through its mean σ_r . One consequence of insensitivity is that stability depends only on the load ρ_r per traffic and not on the higher moments of service time distributions.

This is appealing because under such insensitivity, dimensioning strategies could then be constructed on the sole basis of average loads ρ_r , which can be more robustly estimated than

finer statistics including higher moments of job size distribution. This has motivated Bonald and Proutière [4] to search for rate allocations that would ensure such insensitivity, and led them to invent the so-called balanced fairness allocation, which enjoys some optimality properties among insensitive allocations. The only drawback of balanced fairness is the absence of simple distributed rate control algorithms capable of achieving the corresponding bandwidth allocation.

It should be noted however that, for special classes of networks with grid topologies, proportional fairness and balanced fairness coincide. Closed form expressions for stationary distributions and average transfer times are available in such cases (see [30] for the simplest case of a line network, and [2] for a two-dimensional grid; extension to hypergrids of arbitrary dimension has been made by Frank Kelly—personal communication).

This therefore suggests that proportional fairness might be an attractive allocation strategy, combining the existence of practical schemes to achieve it with approximate insensitivity properties.

Progress towards showing approximate insensitivity properties of proportional fairness have been made in [28]. In particular, a new characterization of proportional fairness is provided there, exploiting convex duality. More precisely, the proportionally fair allocations $x_r^{PF}(n)$ are identified as:

$$n_r x_r^{PF}(n) = \exp\left(\frac{\partial F(n)}{\partial n_r}\right), \quad (3.9)$$

where $F(n)$ is defined as

$$F(n) = \sup_{z \in \mathbb{R}^{\mathcal{R}}: (\log(z_r))_{r \in \Lambda} \in \Lambda} \sum_{r \in \mathcal{R}} n_r z_r. \quad (3.10)$$

This characterization is further used to show that under proportional fairness, maximal stability holds with arbitrary phase-type distributions of job sizes. In other words, stability properties are insensitive to the job size distribution. This is established by identifying that the function

$$J(n) := F(n) - \sum_{r \in \mathcal{R}} n_r \log(\rho_r) \quad (3.11)$$

is a Lyapunov function for the dynamics of the system at fluid scale. We sketch the argument in the case of exponential file size distributions. Write

$$\begin{aligned} \frac{d}{dt} J(n(t)) &= \sum_{r \in \mathcal{R}} \left[\frac{\partial}{\partial n_r} F(n) - \log(\rho_r) \right] \left[\nu_r - \mu_r \exp\left(\frac{\partial F(n)}{\partial n_r}\right) \right] \\ &= \sum_{r \in \mathcal{R}} \mu_r \left[\frac{\partial}{\partial n_r} F(n) - \log(\rho_r) \right] \left[\rho_r - \exp\left(\frac{\partial F(n)}{\partial n_r}\right) \right]. \end{aligned}$$

It is then apparent from this last expression that each term in the summation is non-positive. The argument in the case of phase-type distributions is significantly more involved.

It is further shown in [28] that the function J as defined in (3.11) is the *rate function* associated with the large deviations of the vector of flow counts (n_r) in steady state, under both balanced fairness, and a modification of proportional fairness.

3.5 Concluding remarks

Apart from proportional fairness and insensitive allocations, the only other allocation for which stability is known to be insensitive to job size distribution is the max-min fair allocation. This was established by Bramson [5]. Analysis of performance under a heavy traffic scaling has been performed by Kelly and Williams [18]. Conjectures on steady state behaviour of proportional fairness under large deviations scaling and under heavy traffic scaling are described respectively in [28] and [16]. Discussion of performance models combining file transfers and real time transfers was initiated in [24] and further developed in [22].

Chapter 4

Overlays and random graphs

Overlays are logical graphs between network users, which each maintain active communication with a limited set of neighbours. Such overlays can be used to support a variety of applications, including Peer-to-Peer content sharing as will be considered in later chapters. The most basic requirement of such overlays is connectivity. It is for instance necessary if any given user is to send information destined to all other users. Such connectivity should furthermore be maintained even under unreliable communication between nodes. Thus, if a message originating at one user is forwarded along each overlay edge, but is dropped with some positive probability π along any edge, we would like the overlay to retain connectivity after removal of “failed” edges.

To create such overlays, one can resort to randomised strategies. As a reference, consider the simplest random graph model, namely the Erdős-Rényi random graph $G(n, p)$, consisting of n nodes, where each potential edge is present with probability p and this independently across edges. Then, as shown by Erdős and Rényi in [8], this graph is connected with high probability if $np - \log(n) \gg 1$, and is disconnected with high probability if $\log(n) - np \gg 1$. This suggests that random graphs with average degree np satisfying $np = c \log(n)$, for some constant $c > 1$, will be able to retain connectivity with high probability under edge failures with failure probability $\pi < 1 - \frac{1}{c}$.

It therefore suggests that a logarithmic scaling of the node degrees is adequate to maintain connectivity guarantees in the presence of such failures.

4.1 Self-scaling graph growth

One contribution done in [21] was to propose mechanisms for incorporating new nodes into an overlay that automatically yield such logarithmic degrees, with no explicit estimation of the total group size n . The basic idea is as follows. A newly arrived node selects at random one of the n existing nodes. It then asks this particular node its current degree, say D_n . Finally, it requires that $D(n)/2 + c$ new edges be created, pointing towards itself (using randomised rounding if $D(n)$ is odd or c is not an integer).

Ignoring randomness associated with the selection of the contacted node, one can develop intuition into why this results in a logarithmic node degree via the following argument. The total number of edges E_n at the step where n nodes have been incorporated satisfies the recursion

$$E_{n+1} = E_n + D_n/2 + c = E_n + \frac{1}{n}E_n + c,$$

where we used the fact that $nD_n = 2E_n$. Equivalently this reads

$$D_{n+1} = D_n + \frac{2c}{n+1},$$

so that for large n one indeed has $D_n \approx 2c \log(n)$. A more rigorous derivation is provided in [21], controlling the impact of randomness in this recursion via martingale arguments.

4.2 Graph rebalancing and Metropolis algorithms

The previously described procedure for growing overlay graphs does not guarantee by itself any global properties of the graph, it only addresses suitable tuning of the average node degree. It is however important to ensure more global properties, such as for instance the absence of sparse cuts. Indeed, sparse cuts entail poor resilience of connectivity to failures, may degrade the ability to stream data at high rates, and finally it entails small isoperimetric constants, which is bad to ensure efficiency of gossiping algorithms (see the SIS model of next chapter) and speed at which random walks mix.

We thus proposed in [29] a simple mechanism for rebalancing graphs, which works as follows.

Any node i , at the instants of a Poisson process with unit rate, picks one of its current neighbours uniformly at random, there being d_i such neighbours. Say neighbour j is selected. Then at node i 's request, node j picks uniformly at random one of its neighbours that is not node i ; there are $d_j - 1$ such neighbours where d_j is the degree of node j . Say node k is selected.

Then the following rewiring is considered: replace the edge (ij) by the edge (ik) (no rewiring is to be performed if edge (ik) is already present). Clearly, this would preserve both connectivity and the number of edges. Denoting by g the original graph, by g' the resulting graph, the proposed rewiring is accepted with probability $a(g, g')$ which we shall specify shortly.

The ultimate goal is to ensure that the overlay graph is eventually a random graph selected among connected n -node e -edge graphs according to a probability distribution

$$\pi(g) = \frac{1}{Z} \exp(-\mathcal{E}(g)), \quad (4.1)$$

for some suitable energy function \mathcal{E} . Note that transition from g to g' is attempted at rate $q(g, g') := 1/(d_i(d_j - 1))$, while the rate at which the reverse transition from g' to g is attempted reads $q(g', g) = 1/(d'_i(d'_k - 1))$, where d'_ℓ refers to the degree of node ℓ in graph g' . Clearly, $d'_i = d_i$ and $d'_k = d_k + 1$, so that $q(g', g) = 1/(d_i d_k)$.

We use an instance of the so-called Metropolis-Hastings algorithm in order to obtain a graph-valued Markov chain with stationary distribution given by (4.1). More precisely, we set the transition acceptance probabilities as

$$a(g, g') = \min \left(1, \frac{\pi(g')q(g', g)}{\pi(g)q(g, g')} \right) = \min \left(1, \frac{d_k}{d_j - 1} e^{-\mathcal{E}(g') + \mathcal{E}(g)} \right). \quad (4.2)$$

We now specialise our choice of energy function \mathcal{E} by letting $\mathcal{E}(g) = \sum_{i=1}^n f(d_i)$, where f is a suitable function of individual node degrees.

The interesting property of the above acceptance probability for this choice of energy function is that it can be computed locally at node i , based only on the degrees d_j, d_k of the two nearby nodes j, k . Indeed, the energy difference $\mathcal{E}(g') - \mathcal{E}(g)$ reads in that case $f(d_j - 1) - f(d_j) + f(d_k + 1) - f(d_k)$, which depends only on local characteristics d_j, d_k .

An obvious instance of this mechanism consists in setting $f(d) \equiv 0$, in which case the stationary distribution π is uniform over graphs $g \in \mathcal{G}$. We may however try to create random

graphs with more tightly balanced node degrees by using a suitable function f . One choice we considered consisted in taking $f(d) = \gamma d^2$, for some fixed coefficient $\gamma > 0$.

An interesting property is that graphs sampled from the corresponding distribution have their degrees sharply concentrated around their mean. More precisely, as shown in [10], when the number of edges e equals $(c/2)n \log(n)$ for a fixed constant c , then for large n , with high probability, all degrees are within $O(\sqrt{\log(n)})$ of $c \log(n)$. This is in turn shown to imply that random graphs drawn from this distribution remain connected with high probability under a link failure probability π , so long as $\pi < e^{-1/c}$. This is to be compared to the corresponding condition for Erdős-Rényi graphs with the same average degree $c \log(n)$, for which connectivity holds only if $\pi < 1 - 1/c$. Since $1 - 1/c < e^{-1/c}$, this entails that the above degree-balanced graphs are more resilient to failures than Erdős-Rényi graphs, for given node degrees.

4.3 Concluding remarks

Several extensions of the previous graph-rewiring strategy just described which lead to explicit Gibbs-like stationary distributions can be envisioned. For instance, in [21], the energy function $\mathcal{E}(g)$ was in fact taken to be of the form

$$\mathcal{E}(g) = \sum_i f(d_i) + \sum_{i \sim j} x_{ij}$$

where the second sum captures link costs x_{ij} for all links (i, j) present in the current graph g . It is easy to see that the Metropolis prescription of the acceptance probability can again be computed locally.

Another type of modification concerns the candidate rewiring. If for instance one needs to rewire graphs under the constraint that the node degrees remain fixed, then the following procedure can be envisioned. Node i contacts a neighbour j who contacts a neighbour $k \neq i$ as previously, but in addition node k contacts a further neighbour $\ell \neq i, j$. Then the proposed rewiring consists in replacing the two edges $(ij), (k\ell)$ by the two edges $(ik), (j\ell)$. We call this the N to Z transition (to see why, draw a picture). It preserves node degrees, and energy changes $\mathcal{E}(g') - \mathcal{E}(g)$ resulting from the rewiring can again be computed locally, provided the energy function $\mathcal{E}(g)$ is a sum of contributions of “local configurations” of the graph g (so far we considered the most local configuration, the node degree, and the simplest configuration involving two nodes, namely the edge cost).

Chapter 5

Network Epidemics

Given a group of users organised in a graph structure, a simple-minded approach for disseminating a piece of information to the group consists in “gossiping”: each user propagates the information to randomly selected neighbours. Several versions of this paradigm can be imagined, depending on the objective.

In the present chapter we will consider essentially two scenarios. In the first scenario, the objective is to enable the information to remain in the system for a long time without necessarily being kept for a long time at any one place. Thus the purpose is for the epidemics to ensure reliable storage of data based on unreliable individual components. We will then identify topological properties of the graph which influence the epidemics’ behaviour.

In the second scenario, new epidemics (or rumours) are launched one after another, and an infecting node will preferentially spread the “latest gossip”. The question we address concerns the outreach of any particular epidemics, i.e. does it reach a sizeable fraction of nodes before being completely superseded by more recent epidemics. Applications to Peer-to-Peer content dissemination will then be described in the next chapter.

5.1 SIS Epidemics: impact of topology

The epidemic model we consider is the so-called Susceptible-Infective-Susceptible model, where nodes can be re-infected after having recovered from infection. Specifically, we consider an undirected graph $G = (V, E)$. Each individual node $v \in V$, if infected (which we note $X_v = 1$), recovers at some rate δ (its state variable X_v is then reset to $X_v = 0$). Moreover, while infected, each node v attempts to infect each of its graph neighbours w at some rate β . Thus the system state $X = (X_v)_{v \in V}$ evolves as a Markov jump process on $\{0, 1\}^V$, with non-zero transition rates

$$\begin{aligned} q^{SIS}(X, X + e_v) &= \beta(1 - X_v) \sum_{w \in V} A_{vw} X_w, \\ q^{SIS}(X, X - e_v) &= \delta X_v, \end{aligned}$$

where A denotes the adjacency matrix of the graph G . In [11], we establish the following upper bound on the time to extinction of the epidemics:

5.1.1 Spectral radius and fast extinction

Denoting by ρ the spectral radius of matrix A , assuming that $\rho\beta < \delta$, the average time to extinction T_{ext} verifies

$$T_{ext} \leq \frac{1 + \log(|V|)}{\delta - \beta\rho}. \quad (5.1)$$

Thus, the spectral radius provides a topological descriptor that enables to identify parameter ranges where the epidemics die out after some time that scales with the *logarithm* of the group size $|V|$. The proof essentially relies on a coupling argument. The SIS epidemics can be constructed jointly with a so-called Branching Random Walk X^{BRW} , that is a Markov process on \mathbb{N}^V , with non-zero transition rates given by

$$\begin{aligned} q^{BRW}(X, X + e_v) &= \beta \sum_{w \in V} A_{vw} X_w, \\ q_v^{BRW} &= \delta X_v, \end{aligned}$$

so that $X_v^{SIS}(t) \leq X_v^{BRW}(t)$ for all nodes v and all times t with probability 1. Now, the time to extinction T_{ext}^{BRW} of the process X^{brw} is readily bounded by the inequality $\mathbb{P}(T^{BRW})_{ext} > t) \leq \sum_{v \in V} \mathbb{E}X_v^{brw}$. The result is then established by relying on the explicit formula for the expectation $\mathbb{E}X^{BRW}(t)$ as $\exp(t(\beta A - \delta I)X^{BRW}(0))$, where I denotes the identity matrix. The latter formula follows from linearity of the transition rates q^{BRW} .

5.1.2 Isoperimetric constants and long survival

We use another topology descriptor to express conditions for long survival of the SIS epidemics. More precisely, let η_m be defined as

$$\eta_m = \inf_{S \subset V, |S| \leq m} |E(S, \bar{S})|, \quad (5.2)$$

where $E(A, B)$ denotes the set of edges with one end-point in A and one endpoint in B . Again in [11], we show the following. If for some $r < 1$, one has

$$\beta\eta_m \geq \frac{\delta}{r}, \quad (5.3)$$

then the average time to extinction T_{ext} verifies

$$T_{ext} \geq \frac{1-r}{\exp(1)\delta} \frac{1-O(r^m)}{2m} \left(\frac{1}{r}\right)^{m-1}. \quad (5.4)$$

The interpretation of the previous formula is as follows: if Inequality (5.3) holds with m a sizable fraction, say $\epsilon|V|$, of the group size $|V|$, for fixed r and ϵ , then the epidemics will survive for a time that is exponential in the group size $|V|$. Thus in the case of large groups, for all practical purposes the epidemics will not extinguish, and the piece of information thus maintained in the system will remain available over long time scales.

The proof of this lower bound on extinction time is again via a coupling construction. One exhibits a one-dimensional Markov jump process Y on $\{0, \dots, m\}$, with non-zero transition rates

$$\begin{aligned} q^{GR}(y, y+1) &= \frac{\delta y}{r}, \quad y < m, \\ q^{GR}(y, y-1) &= \delta y, \quad y \leq m, \end{aligned}$$

where GR stands for ‘‘Gambler’s Ruin’’. The coupling ensures that for all times t , the two processes satisfy $\sum_v X_v^{SIS}(t) \geq Y(t)$. Finally, the time to extinction of the lower-bounding process Y is analysed by exploiting its relation to the classical Gambler’s ruin problem: after each departure from state m , process Y will return to this state with probability $\frac{1-r^{m-1}}{1-r^m}$.

5.1.3 Concluding remarks

The idea of using SIS epidemics to maintain some information is due to Anantharam and Wagner [47]. In [11], it is shown that for a number of graph topologies of interest, including Erdős-Rényi random graphs, Power-law random graphs, and hypercubes, the upper and lower bounds just described on the time to extinction are very close. Thus for such topologies a threshold behaviour holds. Anantharam and Wagner focused on finite two-dimensional grids, for which they showed a threshold behaviour: for infectivity β below some threshold, extinction occurs in logarithmic time, and for β above the same threshold, exponentially long survival prevails. The same SIS model has been extensively studied by Liggett [26] on infinite grids and infinite trees. For such infinite graphs, different questions arise: existence of multiple stationary distributions, and probability of ultimate extinction. In the case of infinite trees, two thresholds occur: for β above the first threshold, with positive probability the epidemics do not die out, but may eventually survive infinitely far from the tree root; for β above a second, distinct threshold, the epidemics can survive and revisit each node, as there are several stationary distributions.

Some open questions on these models concern the time that a node spends in the infected state, prior to extinction. This could be related to the quasi-stationary distributions of the process (limiting distribution conditioned on survival), and might be useful in capturing the “memory” cost of maintaining information based on such schemes.

5.2 Simple gossip with competing rumours

We now turn to a classical process extensively studied by Frieze and Grimmett [9] and Pittel [38], with a particular twist. The process studied in [9] and [38] is as follows. Initially one user (the content source) among a group of N users holds one content item. At discrete time instants $t = 0, 1, \dots$, each user holding the content item picks a target user uniformly at random from the whole population, and replicates the content item at the chosen target. The main results of [9] and [38] concern the time till the whole population is reached.

Here we assume a slightly different situation. In each time slot, the content source launches a new rumour. Nodes propagate rumours based on a uniform random selection of targets, but they always transmit the “latest gossip” among those that they have. It turns out that this technique enables each individual rumour to spread to a sizeable fraction of the population.

Informally, the following two results were established respectively in [40] and [3]:

Theorem 5.1. *fix some arbitrary $\epsilon > 0$. Then with high probability, a particular node receives a particular rumour within some time $(1 + \epsilon) \log_2(N)$ (respectively, within time $\log_2(N)$) after that rumour was launched, with probability $(1 - \epsilon)(1 - \exp(-1))$ (respectively, with probability $1 - \exp(-1/10)$), independently across nodes and rumours.*

Note that $\log_2(N)$ is a lower bound on the time required for a rumour to reach the whole collection of nodes, no matter how much control information is available at nodes, and this even without the adversarial presence of competing rumours assumed here. Thus, this result says that each rumour reaches a fraction of $(1 - \epsilon)(1 - \exp(-1))$ (respectively, $1 - \exp(-1/10)$) of users in optimal time (up to a proportionality factor $(1 + \epsilon)$ for the first version). The fraction reached can be made arbitrarily close to $1 - \exp(-1)$, that is approximately 63%.

This suggests a mechanism for disseminating a stream of information to interested receivers. By letting a source node perform erasure coding over an original stream of data, itself having data rate strictly below $1 - \exp(-1)$, and turning it with additional redundancy into a stream of erasure-coded data with rate of 1 packet per time unit, the above result implies that users

will be able to recover an arbitrarily large fraction of the original data with a delay of order $\log_2(N)(1 + \epsilon)$, plus some constant overhead.

This argument is further detailed in [3]. The main idea in the proof, appearing in [40], consists in establishing a tight control on the number of nodes reached in time t in the single rumour process considered in [38]. This process behaves in an essentially deterministic manner: for times $t \leq \log_2(N)$, the number of nodes infected within time t is given by $2^t(1 - o(1))$ with high probability. Now the key trick lies in the observation that the number of nodes which at time t hold the rumour originated at time 0, while not holding any of the subsequent rumours originated at times $1, 2, \dots, t$ can be written as $Y(t) - \tilde{Y}(t-1)$ where the two processes Y, \tilde{Y} behave as the single epidemic process of [38]. Thus, the number of attempts to spread the epidemics originated at time 0 over the time interval $\{1, \dots, \log_2(N)\}$ reads:

$$N_{\text{attempts}} = \sum_{t=1}^{\log_2(N)} Y(t) - \tilde{Y}(t-1) \approx \sum_{t=1}^{\log_2(N)} 2^t - 2^{t-1} = N - 1.$$

Since all such attempts are done towards targets selected independently, uniformly at random, the number of reached nodes is equal to the number of urns containing at least one ball in a balls-and-bins experiment where $N - 1$ balls are thrown into N bins. The latter is well known to approach $N(1 - \exp(-1))$, which concludes the sketch of the proof.

5.3 Concluding remarks

Variations on the Grimmett-Frieze-Pittel model have been investigated in the paper by Karp, Schenker, Vocking et al [14]. In the latter, lower bounds are established on the time to disseminate the rumour to the full population of N users, assuming that only random uniform selections of users are feasible (the “random telephone call” model). It is shown in particular in [14] that dissemination is achieved with the smallest number of transmissions by having a first phase in which each node picks a random target, and pushes the rumour (if it has it) onto the target, whereas in the second phase, the contacting node pulls the rumour from the contacted one. This has the effect of ensuring successful dissemination with order $N \log(\log(N))$ message transmissions. However, as shown in [40], in the presence of a stream of objects to be transferred, only order N transmissions per item is required. This is achieved by the so-called push-pull mechanism, whereby in odd slots nodes push the latest message they have to a random target, whereas in even slots they pull the earliest message they are missing from a random contact.

Chapter 6

P2P live streaming

The defining property of Peer-to-Peer content delivery is that receivers also contribute to the dissemination of content, by storing and subsequently serving parts of the content considered. The first P2P content delivery application historically and in terms of transferred volumes is file sharing. We will rather focus on another application, namely live streaming, in which a stream of content is generated at a content source, and is to be transmitted with negligible delay to the group of receivers.

We address issues of delay and rate efficiency in the context of live streaming.

6.1 Rate optimality

Several bandwidth constraints can be envisioned. We shall consider two distinct models here.

6.1.1 Edge constraints

Here we assume that all peers correspond to nodes of a graph $G = (V, E)$, with one distinguished node s being the content source. We further assume that to each edge $(i, j) \in E$ is associated a bandwidth capacity c_{ij} . Finally, the source node s receives fresh content at some rate λ . The objective is then for fresh content to reach each node $i \in V$ in limited time.

Clearly, a necessary condition for feasibility is that for any receiver node $i \in V - \{s\}$, the network capacities suffice to let data flow from s to i at rate λ . By the Ford-Fulkerson theorem, this is equivalent to requiring that for all such i , and each cut S separating s from i , that is each subset S such that $s \in S$, $i \notin S$, the cut capacity $c(S, \bar{S})$ defined as

$$c(S, \bar{S}) := \sum_{u \in S, v \in \bar{S}} c_{uv}$$

verifies $c(S, \bar{S}) \geq \lambda$. In this context it has been shown by Edmonds [7] that this condition is not only necessary but also sufficient. Indeed, it guarantees the existence of a collection \mathcal{T} of spanning trees $t \in \mathcal{T}$ rooted at s , and corresponding capacities $\lambda(t)$, such that $\sum_{t \in \mathcal{T}} \lambda(t) = \lambda$, and that such trees can be “packed” in the available edge capacities, that is:

$$c_{ij} \geq \sum_{t \in \mathcal{T}} \lambda(t) \mathbf{1}_{(ij) \in t}.$$

Thus, it is possible to relay information to all nodes at rate λ : it suffices to split the fresh content stream of rate λ into substreams associated with each tree $t \in \mathcal{T}$, ensuring that the

stream associated with t has rate $\lambda(t)$; then each node $u \in V$ only needs to forward to each of its neighbours v the streams corresponding to all trees t such that edge (uv) appears in t . This will clearly result in successful delivery of the original stream to all receivers.

The above approach is problematic since it requires construction of trees t , and maintenance of corresponding capacities $\lambda(t)$, while it would be more appealing to successfully deliver data to all receivers while avoiding any such construction. We now describe one such approach, first introducing some more details on the system model we consider.

We assume that data is in the form of packets, fresh packets appearing at the source s at the instants of a Poisson process with rate λ . Furthermore, when node i starts to send one packet to node j , the transfer will complete after an exponentially distributed random variable with rate c_{ij} .

The strategy we propose for data transmission then consists, for each edge (ij) , for node i to choose which packet to send to j uniformly at random among the packets that node i has and that node j does not yet have. We call this the ‘‘Random Useful’’ policy. The system state can be represented as follows. For each collection of nodes $S \subset V$ and each collection of edges $F \subset E$, let $X_{S,F}$ denote the number of packets already present at all nodes i in S and currently under transfer along edges $e \in F$.

It is then easily seen that the $X_{S,F}$ evolve as a Markov jump process. The main result on this model established in [33] is as follows.

Theorem 6.1. *Under the assumption*

$$\forall S \subset V \text{ such that } s \in S \text{ and } \overline{S} \neq \emptyset, \lambda < c(S, \overline{S}), \quad (6.1)$$

the Markov process $(X_{S,F})$ is ergodic. There thus exists a stationary regime under which every packet reaches all nodes in a random finite time after appearing at the source s .

The proof relies on the technique of fluid limits again, and on the identification of a suitable Lyapunov function that decreases along such fluid limits. At the fluid scale, only the numbers X_S of packets replicated precisely at the nodes $i \in S$ and not actively transferred along any edge remain. The suitable Lyapunov function is then given by

$$L(X) = \sup_{S \subset V} \beta_S \sum_{T \subset S} X_T,$$

where the β_S are suitably chosen positive coefficients.

Note that the above proof does not rely on Edmonds’ theorem, but nevertheless implies the conclusion of Edmonds’ theorem. Thus the proof of the above theorem given in [33] provides an alternative to the classical proof of Edmonds’ theorem.

6.1.2 Node constraints

We now move to a distinct set of bandwidth constraints, attached with nodes rather than edges. We further assume that each node can send data to every other node. The capacity of a node $i \in V$ is denoted by c_i . Now, an obvious necessary condition for feasibility of streaming at rate λ is as follows:

$$\lambda \leq \max(c_s, \frac{1}{|V| - 1} \sum_{i \in V} c_i). \quad (6.2)$$

In the present context, we propose the so-called ‘‘Most-Deprived Peer, Random Useful’’ packet forwarding strategy, described as follows. Each node u determines the set of peers v to which it can provide the largest amount of data that it has and they don’t yet have. Among this set

of most deprived peers, it chooses one arbitrarily, and then provides it with a packet chosen uniformly at random among those that it has and that the receiver does not currently have. Then, as shown in [33], this strategy leads a stable system, whenever the condition (6.2) holds with strict inequality. Again, the proof follows by an analysis of the fluid dynamics, the adequate Lyapunov function in the present case being given by

$$L(X) = \sum_S X_S(|V| - |S|).$$

It is readily interpreted as the pending work in the system: each packet replicated at all nodes in S still needs to be replicated at all nodes outside of S , and there are $|V| - |S|$ many such nodes.

Extensions to scenarios with multiple concurrent commodities, as well as with nodes choosing most deprived neighbors among a small subset of candidates, periodically updated, have been developed in [32]. They show that maximal stability, or in other words efficient use of bandwidth resources, still holds in these scenarios. For multiple commodities, the strategy is modified by measuring deprivation among candidate targets as the sum of the deprivation level for each commodity.

An open question concerns the following variant of the above model. Assume that a fixed set of directed edges is given, describing which node is allowed to send to whom. Then the question is: does the “Most Deprived Random Useful” strategy succeed in relaying the data stream to all receivers, when the injection rate λ can be accommodated by the node capacity resources?

6.2 Delay performance

6.2.1 Uniform capacities

Very little is known about delay performance. There is one particular scenario though for which the optimal delay is known, and schemes achieving it have been identified. It assumes slotted time, all nodes having a capacity of one packet per time unit. In that case, as already discussed in the previous chapter, at least $\log_2(N)$ slots are required before a packet reaches all nodes. Furthermore, clever scheduling strategies have been identified by Mundinger, Weber and Weiss [35] achieving dissemination of all packets within this delay, assuming arrival of one new packet per time slot at the source.

We now describe a result obtained in [3], building on the results of [40], which shows that comparable delays can be obtained using simpler distributed randomised schemes. More precisely, the mechanism we consider requires each node to pick in each time slot one target node uniformly at random. It then determines, among the packets that it has and the target does not have, the one with the latest time stamp. We call this policy the “Random Peer Latest Useful Packet” strategy. Our main result in [3] is then the following (loosely stated):

Theorem 6.2. *Let $\lambda \in (0, 1)$ denote the probability that a new packet is created at the source in any given time slot. Then for any constant $x > 0$ any given peer receives a fraction $1 - 1/x$ of the source packets in time no larger than $\log_2(N) + O(x)$.*

The constant in the term $O(x)$ depends on the injection rate λ . The interpretation is as follows: diffusion at rates arbitrarily close to the optimal rate (that is rate 1, which is precluded here by the assumption $\lambda < 1$) can be achieved with delays of optimal order $\log_2(N)$, if one is willing to accept a small fraction of packet loss.

The proof relies on the second statement in Theorem 5.1. Oversimplifying, Theorem 5.1 implies that a given peer is contacted sufficiently often by peers having packets with age larger

than $\log_2(N)$ and of interest to the receiving peer under consideration, and not having more recent packets, so that these contacting peers provide such missing packets, according to the Latest Useful packet selection strategy. The details are somewhat involved though.

6.2.2 Heterogeneous capacities

A lower bound on the average delay when peers have arbitrary heterogeneous uplink bandwidths was provided in [37]. We provide a version of the argument for a discrete time model.

We assume here that node i can send c_i packets in parallel in any given time slot. We also assume that we have n receiver nodes labeled by $i = 1, \dots, n$ in addition to a source node labeled by s . Let then $G(t)$ represent the largest number of packets that nodes can send by time t , assuming that at time 0 only the source node has any packet. Assume without loss of generality that the uplink bandwidths c_1, \dots, c_n are sorted in decreasing order: $c_1 \geq \dots \geq c_n$.

The function G can be determined in an iterative manner, introducing the functions G_i which correspond to the same uplink bandwidths c_s, c_1, \dots, c_i , but for which the uplink bandwidths c_j for $j > i$ have been set to zero. One then has the following relations:

$$\begin{aligned} G_0(t) &= c_s t, \\ G_i(t) &= G_{i-1}(t) + \max(0, c_i t - \tau_i), \end{aligned}$$

where $\tau_i = \inf\{t > 0 : G_{i-1}(t) \geq i\}$. We then let $G(t) = G_n(t)$.

Our lower bound on the delay performance takes the following form:

Proposition 6.1. *Given a profile of uplink bandwidths, let G be the corresponding function as defined above. Assume that the source node receives periodically, every δ slots, $\delta \in \mathbb{N}$, one packet to disseminate to the set of receivers. Then for any feasible scheme, the largest delay between the time a packet appears at the source and the time all nodes have received cannot be smaller than the lower bound D specified by*

$$D = \min\{d \geq 0 : G(d) - G(d - \delta) \geq n\}. \quad (6.3)$$

The proof of the lower bound (6.3) goes as follows. Assume there is a scheme that achieves successful delivery of all packets to all users with a delay no larger than $D_0 = D - 1$.

Consider successive packets $p = 0, 1, \dots, k$ generated at times $0, \delta, \dots, k\delta$. By time D_0 , by definition of G , no more than $G(D_0)$ copies of such packets have been disseminated. By assumption, n copies of packet 0 have been disseminated. Thus, at most $G(D_0) - n$ packets with labels 1 or higher have been spread. By definition of D , one has

$$G(D_0) - n \leq G(D_0 - \delta) - 1. \quad (6.4)$$

Simple reasoning guarantees that, if there are $G(D_0 - \delta) - x$ copies of particular packets spread within some time $D_0 - \delta$, then there will be at most $G(D_0) - x$ copies of such packets spread within time D_0 .

This entails that the number of copies of packets of labels 1 or higher disseminated by time $D_0 + \delta$ is then at most $G(D_0) - 1$. There are thus at most $G(D_0) - n - 1$ copies of packets with labels 2 or higher disseminated by time $D_0 + \delta$, assuming that n copies of packet 1 have been sent.

Using (6.4), it follows that there are at most $G(D_0 - \delta) - 2$ copies of packets with labels 2 or higher by time $D_0 + \delta$. This in turn implies that there are at most $G(D_0) - 2$ copies of such packets spread by time $D_0 + 2\delta$.

Iterating this argument, it follows that there are at most $G(D_0) - k$ copies of packets with labels k or more spread by time $D_0 + k\delta$. One eventually reaches a contradiction for k larger than $G(D_0)$.

One should note that it is not known how tight this lower bound is. In the homogeneous case $c_i \equiv c_s = 1$, and assuming $n = 2^I$, one has

$$G(t) = \begin{cases} 2^t & \text{if } t \leq I, \\ n + (n + 1)(t - I) & \text{otherwise.} \end{cases}$$

Thus under the rate feasibility condition $(n + 1) \geq n/\delta$, it can be readily checked that the bound provided by the proposition lies in the interval $[\log_2(n), \log_2(n) + 1]$. At this stage we do not know if the bound is tight for arbitrary heterogeneous profiles.

Relying on Little's formula, one can also show that for any scheme, the *average* delay \bar{D} is lower-bounded as follows:

$$\bar{D} \geq D - \sum_{t=D-\delta}^D G(t).$$

6.3 Concluding remarks

Besides the issue of characterising and achieving the best possible delays, another question of interest for P2P live streaming concerns the minimisation of network costs. In [44], a method for regulating the sending rate from user i to user j based on the difference between the *deprivation* of j relative to i , and the marginal cost of increasing the sending rate from i to j , is proposed. It is shown to have desirable properties when used in combination with so-called Random Linear Coding, whereby linear combinations of packets are sent rather than original data packets. In particular, fluid limits of system behaviour are shown to admit a cost-optimal configuration as a fixed point. However a complete analysis of this technique remains elusive.

Chapter 7

Perspectives

The art of rate control has now achieved some level of maturity, and the differential equation models evoked in Chapter 2, originated in [20], have certainly helped in providing a conceptual framework to reason about them. In particular, subsequent work on delay stability [13], [27], [45] has informed the question of how aggressively rates can be adjusted.

The flow-level viewpoint developed in Chapter 3 has also been instrumental in gaining some understanding about which target allocations rate controllers should pursue, and in giving guidance for network dimensioning. There are still loose ends requiring further thought: multipath rate control is currently the subject of debate at the IETF, and the ultimate multipath rate controller is still to be found. Also, a generalisation of multipath to combinations of paths in series and parallel, as sketched in Chapter 2.3, is made possible by the organisation of network users in a logical overlay. However, the model described in Chapter 2.3 is only a first step; more experimentation and modeling are needed to properly tune candidate protocols.

In parallel, theoretical questions are raised by the mathematical analysis of flow-level performance models. In particular, conjectures are formulated in [16] on the Large Deviations and the Heavy Traffic behaviour of network performance under Proportionally Fair sharing. One approach that we currently investigate amounts to develop a theory of reversibility at the Large Deviations level, which could be of more general interest. In addition, stability proofs so far have relied on the search for Lyapunov functions adequately capturing the system dynamics. There might nevertheless exist more general conditions to establish stability while bypassing explicit identification of Lyapunov functions.

The potential of Peer-to-Peer content sharing as a complement to traditional server-based content access is real. Its development raises several interesting questions. The so-called Video-on-Demand application is the more challenging. Conceptually, it differs from Live Streaming by the fact that users are not accessing the same content at the same time, hence there is no tight synchronisation as in Live Streaming. As a result, the problem is no longer one of setting up efficient flows among synchronised users, but one of orchestrating the storage and bandwidth resources of peers so that they behave as a distributed server with the best possible schedulable region, in relation with the expected demand facing the whole system.

Models have been developed in [43] to capture the performance of the system in terms of either waiting time or rejection probability, based on content placement strategies that do not discriminate between contents. We are currently working on adaptive strategies for content management with the aim to maximise the overall system ability to serve requests.

This latter problem departs from the more classical problem of ensuring transmission of requested data, provided the load on the system belongs to its schedulable region, itself being

fixed. In the Video-on-Demand problem, the schedulable region depends on both the bandwidth resources of peers, and on the content they store (this is sometimes coined as the “content bottleneck”). Interestingly, simple content management strategies seem to drive the system schedulable region to some efficient configuration, and are the topic of ongoing work.

Finally, we note that most of the problems described in the present thesis concern determination of control actions at network locations, based on local input and minimal feedback from the rest of the system. Most solutions presented exploit specific structure: for instance, it is feasible to measure the marginal cost $\sum_{\ell \in r} p_\ell$ of increasing rate along a network route r in the rate control application, as each packet sent along route r traverses each link $\ell \in r$, and can thus be suitably affected by that link, the overall impact being fed back to the origin of the route through an acknowledgement.

This suggests as a direction for future research the identification of which structures lead to simple distributed control schemes for optimisation of system-wide objective functions, depending also on the structure of the objective function itself. An instance of this general problem concerns rate control for peer-to-peer live streaming, when networking coding is used for transmissions (as opposed to store-and-forward operations only), the objective being minimisation of some overall network cost. Candidate control strategies have been recently proposed in [44], and partial stability results have been established. The complete picture in this case still remains elusive, and is the topic of ongoing research.

Bibliography

- [1] D. Bersekas and R. Gallager. *Data Networks*. Prentice Hall, 1992.
- [2] T. Bonald and L. Massoulié. Impact of fairness on internet performance. In *Proceedings of ACM Sigmetrics*, 2001.
- [3] T. Bonald, L. Massoulié, F. Mathieu, D. Perino, and A. Twigg. Epidemic live streaming: optimal performance trade-offs. In *Proceedings of ACM Sigmetrics*, 2008.
- [4] T. Bonald and A. Proutière. Insensitive bandwidth sharing in data networks. *Queueing Systems*, 44, 2003.
- [5] M. Bramson. Network stability under max-min fair bandwidth sharing. *to appear in Annals of Applied Probability*.
- [6] J. Dai. On positive harris recurrence of multiclass queueing networks: a unified approach via fluid limit models. *Annals of Applied Probability*, 5(1):49–77, 1993.
- [7] J. Edmonds. Edge-disjoint branchings. *Combinatorial Algorithms*, pages 21–31, 1972.
- [8] P. Erdős and A. Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.
- [9] A. Frieze and G. Grimmett. The shortest-path problem for graphs with random arc-lengths. *Discrete Appl. Math.*, 10:57–77, 1985.
- [10] A. Ganesh and L. Massoulié. Failure resilience in balanced overlay networks. In *41th Allerton Conference on Communication, Control and Computing*, 2003.
- [11] A. Ganesh, L. Massoulié, and D. Towsley. The effect of network topology on the spread of epidemics. In *Proceedings of IEEE Infocom*, 2005.
- [12] L. Georgiadis, M. Neely, and L. Tassiulas. *Resource Allocation and Cross-Layer Control in Wireless Networks*. Now Publishers, 2006.
- [13] R. Johari and D. Tan. End-to-end congestion control for the internet: delays and stability. *IEEE/ACM Transactions on Networking*, 9(6):818–832, 2001.
- [14] R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking. Randomized rumor spreading. In *Proceedings of the 41st IEEE Annual Symposium on Foundations of Computer Science*, 2000.
- [15] F. Kelly. *Reversibility and Stochastic Networks*. Wiley, 1979.

- [16] F. Kelly, L. Massoulié, and N. Walton. Resource pooling in congested networks: proportional fairness and product form. *Queueing Systems*, 63(1–4):165–194, 2009.
- [17] F. Kelly and T. Voice. Stability of end-to-end algorithms for joint routing and rate control. *Computer Communication Review*, 35(2):5–12, 2005.
- [18] F. Kelly and R. Williams. Fluid model for a network operating under a fair bandwidth-sharing policy. *Annals of Applied Probability*, 14:1055–1083, 2004.
- [19] F. P. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 1997.
- [20] F. P. Kelly, A. K. Maulloo, and D. Tan. Rate control for communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 1998.
- [21] A. Kermarrec, A. Ganesh, and L. Massoulié. Scamp: Peer-to-peer lightweight membership service for large-scale group communication. In *Proceedings of the 3rd International Workshop on Networked Group Communication*, 2001.
- [22] P. Key and L. Massoulié. Fluid models of integrated traffic and multipath routing. *Queueing systems*, 53(1–2):85–98, 2006.
- [23] P. Key and L. Massoulié. Control of communication networks: welfare maximization and multipath transfers. *Philosophical Transactions of the Royal Society S*, 2008.
- [24] P. Key, L. Massoulié, A. Bain, and F. Kelly. Fair internet traffic integration: network flow models and analysis. *Annales des Télécommunications*, 59:1338–1352, 2004.
- [25] S. Kunnyur and R. Srikant. End-to-end congestion control: utility functions, random losses and ecn marks. *IEEE/ACM Transactions on Networking*, 7(5):689–702, 2003.
- [26] T. Liggett. *Stochastic interacting systems: contact, voter and exclusion processes*. Springer, 1999.
- [27] L. Massoulié. Stability of distributed congestion control with heterogeneous feedback delays. *IEEE Transactions on Automatic Control*, 47(6):895–902, 2002.
- [28] L. Massoulié. Structural properties of proportional fairness: stability and insensitivity. *Annals of Applied Probability*, 17(3):809–839, 2007.
- [29] L. Massoulié, A. Kermarrec, and A. Ganesh. Network awareness and failure resilience in self-organising overlay networks. In *Symposium on Reliable and Distributed Systems (SRDS)*, 2003.
- [30] L. Massoulié and J. Roberts. Bandwidth sharing and admission control for elastic traffic. In *ITC specialists seminar*, 1998.
- [31] L. Massoulié and J. Roberts. Bandwidth sharing: objectives and algorithms. *IEEE/ACM Trans. Netw.*, 10(3):320–328, 2002.
- [32] L. Massoulié and A. Twigg. Rate-optimal schemes for peer-to-peer live streaming. *Journal of Performance Analysis*, 2008.
- [33] L. Massoulié, A. Twigg, C. Gkantsidis, and P. Rodriguez. Randomized decentralised broadcasting algorithms. In *Proceedings of IEEE Infocom*, 2007.

- [34] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, 2000.
- [35] J. Mundinger, R. Weber, and G. Weiss. Optimal scheduling of peer-to-peer file dissemination. *Journal of Scheduling*, 2007.
- [36] E. Oubagha, L. Massoulié, and A. Simonian. Delay analysis of a credit-based control for abr transfer, 1997.
- [37] F. Picconi and L. Massoulié. Is there a future for mesh-based live video streaming? In *Proceedings of IEEE P2P conference*, 2008.
- [38] B. Pittel. On spreading a rumour. *SIAM Journal of Applied Mathematics*, 47(1), 1987.
- [39] A. Rybko and A. Stolyar. Ergodicity of stochastic processes describing the operation of open queueing networks. *Probl. Peredachi Inf.*, 28(3):199–220, 1992.
- [40] S. Sanghavi, B. Hajek, and L. Massoulié. Gossiping with multiple messages. *IEEE Transactions on Information Theory*, 2007.
- [41] R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, 2003.
- [42] S. Stolyar. Dynamic distributed scheduling in random access networks. *Journal of Applied Probability*, 45(2):297–313, 2008.
- [43] K. Suh, C. Diot, J. Kurose, L. Massoulié, C. Neumann, D. Towsley, and M. Varvello. Push-to-peer video-on-demand system: Design and evaluation. *IEEE JSAC special issue on Peer-to-Peer and Multimedia*, 2007.
- [44] D. Tomozei and L. Massoulié. Flow control for cost-efficient peer-to-peer streaming. In *Proceedings of IEEE Infocom*, 2010.
- [45] G. Vinnicombe. On the stability of end-to-end congestion control for the internet, 2001.
- [46] T. Voice. *Stability of congestion control algorithms with multi-path routing and linear stochastic modelling of congestion control*. Ph.D. thesis, Cambridge University, 2006.
- [47] A. Wagner and V. Anantharam. Wireless sensor network design via interacting particles. In *Proceedings of the 40th Annual Allerton Conference on Communication, Control, and Computing*, pages 1554–1563, 2002.
- [48] N. S. Walton. Proportional fairness and its relationship to multi-class queueing networks. *Annals of Applied Probability*, 19(6):2301–2333, 2009.
- [49] H. Ye. Stability of data networks under an optimization-based bandwidth allocation. *IEEE Transactions on Automatic Control*, 48(7):1238–1242, 2003.