

Prepared for Ofcom under MC 316

A Study of Traffic Management Detection Methods & Tools

Predictable Network Solutions Limited

www.pnsol.com

June 2015



Contents

1. Introduction	11
1.1. Centrality of communications	11
1.2. Computation, communication	11
1.2.1. Circuits and packets	12
1.2.2. Resource sharing	13
1.3. Connectivity and performance	14
1.4. Traffic Management	15
1.4.1. Traffic management detection	15
1.4.2. Traffic management in the UK	16
1.5. Previous BEREC and Ofcom work	16
1.5.1. BEREC reports	16
1.5.2. Notes on previous Ofcom studies	18
1.6. Summary	18
2. TM detection	19
2.1. Introduction	19
2.2. Traffic management	19
2.3. TM Detection Techniques	20
2.3.1. NetPolice	20
2.3.1.1. Aim	20
2.3.1.2. Framing the aim	20
2.3.1.3. Implementation	21
2.3.1.4. TM techniques detected	22
2.3.1.5. Discussion	22
2.3.2. NANO	23
2.3.3. DiffProbe	25
2.3.4. Glasnost	28
2.3.5. ShaperProbe	31
2.3.6. ChkDiff	33
2.3.7. Network Tomography	35
3. TMD in operational context	38
3.1. Introduction	38
3.2. Review of TMD	38
3.2.1. Technical aspects of flow differentiation	39
3.2.2. Underlying assumptions made in TMD techniques	39
3.2.3. Comparison of main approaches	40
3.3. TMD in a UK context	40
3.3.1. Offered-load-based differentiation	42
3.3.2. Association-based differentiation	42
3.3.3. Cost of the detection process	43
3.3.4. TM detection techniques as proxy for user experience impairment	43
4. Concs. & recommendations	45
4.1. Conclusions	45
4.2. Recommendations	47

Bibliography	49
A. ICT performance	55
A.1. Translocation	55
A.1.1. Mutual interference in network traffic	55
A.2. Application outcomes	56
A.2.1. Application performance depends only on ΔQ	57
A.2.2. How ΔQ accrues across the network	57
A.3. Summary	60
B. TM Methods and ΔQ	61
B.1. PBSM and $\Delta Q _V$	61
B.1.1. FIFO	62
B.1.1.1. Ingress behaviour	62
B.1.1.2. Egress behaviour	62
B.1.1.3. Discussion	62
B.1.1.4. Fairness with respect to ΔQ	63
B.2. Load Correlation	64
B.3. TM trading space	66
B.3.1. Overall delay and loss trading	66
B.3.1.1. Component-centric view	66
B.3.1.2. Translocation-centric view	67
B.3.2. Location-based trading	67
B.4. Other approaches	68
B.4.1. Prerequisites for deployment of differential treatment	69
B.4.2. Priority queueing	69
B.4.3. Bandwidth sharing	70
B.4.4. Rate shaping	73
B.4.5. Rate policing	73
B.5. Further factors	74
B.6. Static/dynamic allocation	75
B.7. Heterogeneous delivery	75
C. UK Internet	77
C.1. UK network boundaries	77
C.1.1. Non-Wireline ISP provision	78
C.2. How ΔQ accrues in the UK	83
C.2.1. Specialised services	84
C.3. UK Domain interfaces	85
C.3.1. Potential points of TM application	85
C.4. Summary	86
D. Analysis of BT SINS	88
D.1. Bandwidth caveats	91
E. Additional Literature	92

List of Figures

1.1. Potential TM points in the UK broadband infrastructure (wireline)	17
2.1. Detecting various types of differentiation with end-host based probing	21
2.2. NANO architecture	23
2.3. DiffProbe architecture	26
2.4. Delay distributions due to strict priority and WFQ scheduling (simulated)	27
2.5. The Glasnost system	29
2.6. Glasnost flow emulation	30
2.7. ShaperProbe method	32
2.8. ShaperProbe sample output	32
2.9. Chkdifff architecture	34
A.1. The network is a tree of multiplexors	56
A.2. Impact of ΔQ on application performance	57
A.3. An end-to-end path through a network (from A.1b)	58
A.4. ΔQ and its components	59
B.1. Example of one way delay between two points connected to UK internet	65
B.2. Differential delay in a two-precedence-class system (with shared buffer)	70
B.3. Bandwidth sharing viewed as a collection of FIFOs	71
C.1. Representation of the administrative and management boundaries in UK broadband provision (wireline)	79
C.2. UK ISPs in wider context	81
C.3. Administrative and management boundaries in UK broadband provision (non-wireline)	82
C.4. Idealised end-to-end path for typical UK consumer	83
C.5. Potential TM points in the UK broadband infrastructure (wireline)	87
E.1. Citation relationship between relevant papers	93

Nomenclature

3G Third generation mobile cellular.

ΔQ See Quality Attenuation.

ADSL Asymmetric DSL.

Applet Small program dynamically downloaded from a webpage and executed locally in a constrained environment.

AS Autonomous System.

Asymmetric In the context of UK internet provision, this means that the linkspeed to the end user is higher than the linkspeed from them.

ATM Asynchronous Transfer Mode.

BRAS Broadband Remote Access Server.

CDN Content Distribution Network.

CSP Communication Service Provider.

CT Computerised Tomography.

DDOS Distributed Denial of Service.

Discrimination In this document the definition used is that of choosing between two or more alternatives.

DOCSIS Data Over Cable Service Interface Specification.

DPI Deep Packet Inspection.

DSL Digital Subscriber Line.

FCFS First-Come-First-Served.

FIFO First-In-First-Out.

GGSN Gateway GPRS Support Node.

ICMP Internet Control Message Protocol.

internet (adj) of, or pertaining to, The Internet.

Internet, the The global aggregation of packet-based networks whose endpoints are reachable using a unique Internet Protocol address.

IP Internet Protocol.

ISP Internet Service Provider.

Java VM Java Virtual Machine.

L2TP Layer Two Tunneling Protocol.

LAN Local Area Network.

Layer 2	The layer in the internet protocol stack responsible for media access control, flow control and error checking.
Layer 3	The layer in the internet protocol stack responsible for packet forwarding including routing through intermediate routers.
LTE	Long Term Evolution - fourth generation mobile cellular.
MPLS	Multi-Protocol Label Switching.
MT	Mobile Terminal.
OS	Operating System.
P2P	Peer to Peer.
PASTA	Poisson Arrivals See Time Averages.
PBSM	Packet-Based Statistical Multiplexing.
PDH	Plesiochronous Digital Hierarchy.
PDU	Protocol Data Unit: the composite of the protocol headers and the service data unit (SDU).
PGW	Packet Data Network Gateway.
PRO	Predictable Region of Operation.
QoE	Quality of Experience.
Quality Attenuation	The statistical impairment that a stream of packets experiences along a network path.
RNC	Radio Network Controller.
RTT	Round Trip Time.
SDH	Synchronous Digital Hierarchy.
SDN	Software Defined Networking.
SDU	Service Data Unit.
SGSN	Serving GPRS Support Node.
SGW	Service Gateway.
SIN	Supplier Information Note.
SLA	Service Level Agreement.
Stationarity	The degree to which the characteristics of something (for example Quality Attenuation) are constant in time.
TCP	Transmission Control Protocol.
TDM	Time-Division Multiplexing.
TM	Traffic Management.
TOS	Type of Service.
TTL	Time to live - the number of router hops that a packet can transit before being discarded.
UDP	User Datagram Protocol.

VDSL Very-high-bit-rate DSL.

VLAN Virtual LAN - a method for limiting association in a LAN.

VoD Video on Demand.

VoIP Voice over IP.

WFQ Weighted Fair Queuing.

WRED Weighted Random Early Detection.

Executive Summary of the Research Report

As the demand for broadband capacity by a range of end application services increases, a greater focus is being placed on the use of traffic management (TM) to help meet this increasing demand. Given this, Ofcom commissioned work to further understand the availability of techniques and tools that may be used to detect the presence of TM in broadband networks.

Practical TM detection methods have the potential to form a useful part of the regulatory toolkit for helping the telecommunications market deliver online content and services that meet consumer expectations. In principle, they could potentially help in the following ways:

Increasing transparency for consumers: providing consumers with more information on the application of traffic management and its likely effect on the quality of experience (QoE) of accessing different online services;

Increased visibility for the regulator: the ability to verify operator claims on the employed TM practices within their networks; and

Increased benefits for online service providers: Enabling content and application providers to better deliver their services over broadband networks, by providing more information on the potential effects of TMs and on their products and services.

This report provides the outcome from a literature review of the different techniques that could be used to detect the use of TM.

The report provides a comparative analysis of the identified methods and tools, for example, in terms of:

- Their efficacy in detecting and quantifying the presence of TM in a given network;
- The impact on the network and the consumer in terms generated traffic volume, quality of experience, etc; and
- The need for a given tool or methodology to be integrated within, or executed outside, a given ISP's network infrastructure.

Finally, the report also sets out the key attributes that any future effective TM detection method should meet.

To this end, the report first reviews key papers that cover the most recent, most cited and most deployed techniques for detecting differential management of user traffic. These principally aim to provide end-users with tools that give some indication of whether discrimination is being applied to their own broadband connection. Commercial organisations such as content providers appear to have taken relatively little interest in the commercialisation of TM detection.

While their business is dependent on suitable bounds on the network performance along the path from their servers to their end-users, traffic management (differential or otherwise) is only one of many factors affecting this.

Next, the report further considers the operational behaviours and scalability of these detection approaches, and their potential application and impact in an operational context (i.e. by actors other than individual end-users). Relevant technical developments, models of practical TM measures, and details of the UK context are presented in the appendices.

In terms of key attributes that a future TM detection should meet, we suggest the following:

1. Identify who is responsible for the TM, i.e. where along the complex digital delivery chain it is applied;
2. Be reliable, minimising false positives and false negatives; and

3. Be scalable to deliver comprehensive coverage of potential TM locations, without excessive deployment cost or adverse effect on network performance.

The studied TM detection techniques have been mostly developed in North America, where the market structure differs from that of the UK. Where there is a single integrated supplier, as is typical in North America, establishing that discrimination is occurring somewhere on the path to the end-user is broadly sufficient to identify responsibility. However, in the UK, delivery of connectivity and performance to the wider Internet is split across a series of management domains (scopes of control) and administrative domains (scopes of responsibility). This makes it harder to identify that domain in which differential management is occurring. The survey of the open literature identified a set of key papers that describe TM detection methods.

These are:

NetPolice which aims to detect content- and routing-based differentiations in backbone (as opposed to access) ISPs. It does this by selecting paths between different access ISPs that share a common backbone ISP, and using ICMP to detect packet loss locations.

NANO which aims to detect whether an ISP causes performance degradation for a service when compared to performance for the same service through other ISPs. It does this by collecting observations of both packet-level performance data and local conditions and by applying stratification and correlation to infer causality.

DiffProbe which aims to detect whether an access ISP is deploying certain differential TM techniques to discriminate against some of its customers' flows. It does this by comparing the delays and packet losses experienced by two flows when the access link is saturated.

Glasnost which aims to determine whether an individual user's traffic is being differentiated on the basis of application. It does this by comparing the successive maximum throughputs experienced by two flows.

ShaperProbe which tries to establish whether a token bucket shaper is being applied to a user's traffic. It does this by sending increasing bursts of maximum-sized packets, looking for a point at which the packet rate measured at the receiver drops off.

Chkdiff which tries to discern whether traffic is being differentiated on the basis of application. Rather than testing for the presence of a particular TM method, this approach simply asks whether any differentiation is observable, using the performance of the whole of the user's traffic as the baseline.

These techniques are largely successful in their own terms, in that they can detect the presence of particular kinds of differential traffic management operating along the traffic path from an individual user to the Internet. Further work would be needed to independently confirm their reliability claims.

However, none of the currently available techniques meet the desired key attributes of a TM detection system. This is because:

1. Some attempt to establish where TM is occurring along the path examined, but only at the IP layer, which will only localise TM performed at user-visible Layer 3 routers; in the UK context there may not be any such between the user and the ISP. This localisation also relies on a highly restricted router resource, which would limit the scale at which such techniques could be deployed.
2. They aim only to detect the presence of differential TM within the broadband connection of a particular end user.
3. Those that are currently in active deployment generate significant volumes of traffic, which may make them unsuitable for large-scale use.

A key constraint of most of the currently available tools is that they focus on detecting a particular application of a particular TM technique. Even in combination they do not cover

all of the potential TM approaches that could be applied. Only NANO and Chkdif may be sufficiently general to overcome this problem.

A further difficulty arises because of the need to attain a broader understanding of what the various actors in the UK digital supply chain may or may not be doing from a TM perspective and how these activities interact. This would require a deeper analysis of the results of many measurements, potentially along the lines of network tomography. This requires further research, and so we must conclude that there is no tool or combination of tools currently available that is suitable for practical use.

In our view, further work is required to develop a broader framework for evaluating network performance, within the context of the inevitable trade-offs that must be made within a finite system. This framework should encompass two aspects:

- A way of identifying the network performance requirements for different applications. The process should be unbiased, objective, verifiable and adaptable to new applications as they appear; and
- A way of measuring network performance that can be reliably related to application needs. This measurement system would need to deal with the fragmented nature of the end-to-end inter-connected delivery chain by reliably locating performance impairments. Any such approach would have to avoid unreasonable loads on the network.

Together these could determine whether a particular network service was fit-for-purpose for different applications; some novel approaches outlined in the report have the potential to do this, in particular developing the ideas of network tomography. This uses the ‘performance’ of packets traversing a given network to infer details about the underlying network, its performance; and potentially the presence and location of TM.

Network tomography requires further work to establish whether it could become a practical tool (other topics for further study are outlined in the recommendations of the report). TM detection could then become a way to fill in any gaps in the overall framework outlined above.

1. Introduction

1.1. Centrality of communications

“The Internet is increasingly central to the lives of citizens, consumers and industry. It is a platform for the free and open exchange of information, views and opinions; it is a major and transformative medium for business and e-commerce, and increasingly a mechanism to deliver public services efficiently. As such it provides access to a growing range of content, applications and services which are available over fixed and wireless networks.” [1]

While BEREC defines the Internet as “... the public electronic communications network of networks that use the Internet Protocol for communication with endpoints reachable, directly or indirectly, via a globally unique Internet address”, in common usage it is shorthand for an ever-expanding collection of computing devices, communicating using a variety of protocols across networks that themselves increasingly rely on embedded computing functions.

In order to deliver a useful service, both the computing and communication elements of this system must perform within certain parameters, though the complexity of defining what those parameters should be seems daunting. The current delivery of internet services largely separates responsibility for the computing component (typically shared between the end-user and some service provider) from that for the communications (delivered by various ‘tiers’ of Internet / Communications Service Providers). The end-to-end digital supply chain is complex and heterogeneous, and the demands placed upon it change so rapidly that some sectors of the industry find themselves “running to stand still”; at the same time, network-enabled functions pervade ever deeper into everyday life. If the promise of “the Internet of Things” is fulfilled this process will go much further still.

1.2. Computation, communication and ICT

Fifty years ago the boundary between ‘communication’ and ‘computation’ was relatively clear. Communication took place over circuits constructed on a mainly analogue basis, with the analogue/digital conversion occurring at the network edges. Computation occurred in a limited number of very specialised locations, containing mainframes (or, later, minicomputers). Even though those computers consisted of many components that exchanged data (processors, memory, disk drives), these exchanges were not in the same conceptual category as ‘communications’. The dominant mode of use was that the edges transferred data (punch card or line-printer images, characters to/from terminals) via communication links to the central location. The computation was centralised; the edges processed and communicated data, the central computer dealt with the information that was contained within that data.

Today, communication involves extensive use of computation, and ICT functions are no longer centralised. The analogue parts of communication have been relegated to a minor role, with even signal construction/extraction and error detection/correction being done digitally. Communication is now intimately tied to computational processes, and computation (of the kind previously only seen in mainframes, etc.) is occurring in myriad locations. The conceptual separation that existed in the mainframe-dominated world has disappeared.

The new dominant model of ICT is that of interacting and collaborating elements that are physically distributed: web services rely on web browsers to render (and interpret scripts within) the content, which is (often dynamically) constructed on remote web servers; video-on-demand relies on rendering in the device to interpret the content served through a CDN or

from a server; cloud services, VoIP, Teleconferencing (both voice and video), etc. all rely on outcomes that involve interaction between communication and computation (often not just at the endpoints¹).

As computation has been distributed, the requirement to ‘pass data’ has also been distributed - memory and processing may be half a continent apart, disk drives half the world away. This shift has also ‘distributed’ other aspects from the computational world to the new communications world, in particular the statistically multiplexed use of resources and its associated scheduling issues. The understanding, management and economic consequences of these issues are no longer confined within the mainframe, but pervade the whole ICT delivery chain.

The distinction between computing and communications has become so blurred that one major class of ‘communications’ service - that of mobile telephony and data - is perhaps better viewed as a the operation of a massive distributed supercomputer. The ability of a mobile network to deliver voice or data is the direct result of a distributed set of connected computational actions; the network elements² are all interacting with each other to facilitate the movement of information.

Such movement of ‘voice content’ and/or ‘data content’ is far removed from the concept of ‘communication’ from 50 years ago. It is no longer about the transmission of bits (or bytes) between fixed locations over dedicated circuits, it is about the *translocation of units of information*. In the mobile network case these ‘units’ may be voice conversation segments or data packets for application use, the translocation being the consequence of interactions between computational processes embedded in network elements.

At the heart of this process is the statistical sharing of both the raw computation and the point-to-point communication capacity.

1.2.1. Circuits and packets

The underlying communications support for ICT has also changed radically in the last 50 years. The dominant communications paradigm is no longer one of bits/bytes flowing along a fixed ‘circuit’ (be that analogue or TDM) like “beads on a string”. Today’s networks are packet/frame³ based: complete information units are split into smaller pieces, copies of which are ‘translocated’ to the next location. Note that the information does not actually *move*, it simply becomes available at different locations⁴. This translocation is the result of a sequence of interactions between computational processes at the sending and receiving locations. This is repeated many times along the network path until the pieces of data reach the final computational process⁵ that will reassemble them and interpret the information.

Each of these ‘store-and-forward’ steps involves some form of buffering/queueing. Every queue has associated with it two computational processes, one to place information items in the queue (the receiving action, ingress, of a translocation), the other to take items out (the sending action, egress, of a translocation). This occurs at all layers of the network/distributed application, and each of these buffers/queues is a place where statistical multiplexing occurs, and thus where contention for the common resource (communication or computation) takes place.

Statistical multiplexing is at the core of the current ICT evolution. Using it effectively is key to amortising capital and operational costs, as this permits costs to drop as the number of customers increases⁶. This makes it economic for broadband networks to deliver ‘always on’

¹E.g. combining audio streams in a teleconference is another computational process.

²I.e handsets, cell towers, regional network controllers, telephone network interconnects, interface points with the general Internet, etc.

³Typically using Ethernet and/or IP.

⁴At most network layers original information units are discarded some time after the remote copy is created.

⁵Always accepting that this is not a perfect process and that there are many reasons why it may get ‘lost’.

⁶Note that this is not new: the telegraph was a message-based statistically-multiplexed system in which people took the roles now performed by network elements, such as serialisation and deserialisation, routing, and even traffic management.

connectivity⁷, and an ensemble of shared servers to provide ‘always available’ services.

1.2.2. Theoretical foundations of resource sharing

While distributed computing has advanced tremendously over the last several decades in a practical sense⁸, there has been comparatively little attention given to its theoretical foundations since the 1960s. Few ‘hands-on’ practitioners worry about this, on the basis that ‘theory is no substitute for experience’. However, given the extent and speed of change in this industry, there is always a danger that continuing to apply previously successful techniques will eventually have unexpected negative consequences. Such hazards cannot be properly assessed without a consistent theoretical framework, and their potential consequences grow as society becomes increasingly dependent on interconnected computational processes.

To understand network ‘traffic management’ we must first understand the fundamental nature of network traffic, and indeed of networks themselves. This understanding is built upon three well-established theoretical pillars:

1. A theory of computation, started by Turing, that assumes that information is immediately available for each computational step;
2. A theory of communication, developed by Shannon, that assumes that data is directly transmitted from one point to another over a dedicated channel [2];
3. A theory of communicating processes, developed by Milner, Hoare and others, that assumes that communication between processes is always perfect.

While all of these have been enormously successful, and continue to be central to many aspects of ICT, they are not sufficient to deal with the inextricably woven fabric of computation and communication described in §1.2.1 above, that is loosely referred to as ‘the Internet’. The first two theoretical pillars are focused on local issues, whereas the key problem today is to deliver good outcomes on a large scale from a highly distributed system. This inevitably requires some degree of compromise, if only to bring deployments to an acceptable cost point. Statistical sharing - the principle that makes ‘always on’ mass connectivity economically feasible - is also the key cause of variability in delivered service quality. This is because an individual shared resource can only process one thing at a time, so others that arrive have to wait⁹. This is the aspect of communications that is missing from the third pillar.

Distributed computation necessarily involves transferring information generated by one computational process to another, located elsewhere. We call this function ‘translocation’, and the set of components that performs it is ‘the network’. Instantaneous and completely loss-less translocation is physically impossible, thus all translocation experiences some ‘impairment’ relative to this ideal. Typical audio impairments that can affect a telephone call (such as noise, distortion and echo) are familiar; for the telephone call to be fit for purpose, all of these must be sufficiently small. Analogously, we introduce a new term, called ‘quality attenuation’ and written ‘ ΔQ ’, which is a statistical measure of the impairment of the translocation of a stream of packets when crossing a network. This impairment must be sufficiently bounded for an application to deliver fit-for-purpose outcomes¹⁰; moreover, the layering of network protocols isolates the application from any other aspect of the packet transport. This is such an important point it is worth repeating: the great achievement of network and protocol design has been to hide completely all the complexities of transmission over different media, routing

⁷Note, however, that it provides only the *semblance* of a circuit, since in commodity broadband there is no dedication of any specific portion (either in space or time) of the common resources to individual customers.

⁸Driven by advances in processing power and transmission capacity combined with remarkable ingenuity in the development of protocols and applications.

⁹Or, in extremis, be discarded.

¹⁰Just as a telephone call might fail for reasons that are beyond the control of the telephone company, such as excessive background noise or a respondent with hearing difficulties, applications may fail to deliver fit-for-purpose outcomes for reasons that are beyond the control of the network, e.g. lack of local memory, or insufficient computing capacity. Such considerations are out of scope here.

decisions, fragmentation and so forth, and leave the application with only one thing to worry about with respect to the network: the impairment that its packet streams experience, ΔQ . ΔQ is amenable to rigorous mathematical treatment¹¹, and so provides a starting point for the missing theoretical foundations of large-scale distributed computation.

For the purposes of this report, a key point is that ΔQ has two sources:

1. Structural aspects of the network, such as distance, topology and point-to-point bit-rate;
2. Statistical aspects of the network, due to the sharing of resources (including the effects of load).

Separating these two components makes the impact of traffic management easier to understand, as it is concerned only with the sharing of resources. As stated above, sharing resources necessarily involves some degree of compromise, which can be expressed as quality impairment. Traffic management controls how the quality impairment is allocated; and since quality impairment is always present and always distributed somehow or other, traffic management is always present.

1.3. Networks: connectivity and performance

A communications network creates two distinct things:

connectivity the ability of one computational process to interact with another even at a remote location;

performance the manner in which it reacts or fulfils its intended purpose, which is the translocation of units of information between the communicating processes.

Any limitation on connectivity (or more technically the formation of the associations) is typically either under the control of the end-user (e.g. using firewalls) or follows from due legal process (e.g. where the Courts require ISPs to bar access to certain sites).

For a distributed application to function at all, appropriate connectivity must be provided; however, for it to function well, appropriate performance (which is characterised by ΔQ) is also essential¹².

Performance, however, has many aspects that act as a limit. Geographical distance defines the minimum delay. Communication technology sets limits on the time to send a packet and the total transmission capacity. Statistical sharing of resources limits the capacity available to any one stream of packets. The design, technology and deployment of a communications network - its structure - sets the parameters for a best-case (minimum delay, minimal loss) performance at a given capacity. This is what the network 'supplies', and this supply is then shared between all the users and uses of the network. Sharing can only reduce the performance and/or the capacity for any individual application/service.

Communications networks are expensive, and so the ubiquity of affordable access is only possible through dynamic sharing of the collective set of communication resources. It is a truism that such dynamically shared networks deliver the best performance only to their very first customers; the gradual decrease in performance for individual users as user numbers increase is a natural consequence of dynamic resource sharing in PBSM.

To give a practical example of what this sharing means, for a single consumer to watch an iPlayer programme successfully, typically there must be 15 to 20 other consumers (on the same ISP) who are not using the network at all in any one instant of the programme's duration¹³.

Traffic management (the allocation of quality impairment) is at the heart of this sharing process. It works in one of two ways: it either shares out access to the performance (its

¹¹This is discussed in more detail in Appendix A.

¹²This is discussed in more detail in Appendix §A.2.

¹³It doesn't have to be the same 15 to 20 users, it can be a dynamically changing set; note also that it is not just the aggregate capacity that is shared, but the ability to deliver data within time constraints.

delay, its loss and its capacity); or it limits demand on the supply (thus reducing the effects of sharing elsewhere).

1.4. Traffic Management

Clearly a balance needs to be struck between TM techniques applied to improve services to end-users and TM that (either intentionally or otherwise) degrades services unnecessarily. As stated in [3], “The question is not whether traffic management is acceptable in principle, but whether particular approaches to traffic management cause concern.”

Statistical sharing of resources inevitably involves a tradeoff: the more heavily a resource is used, the more likely it is to be in use when required. Buffering is needed to allow for arrivals to occur when the resource is busy. This creates contention for two things, the ability to be admitted into the buffer (ingress) and the ability to leave the buffer (egress). Whether the first is achieved determines loss, and the time taken to achieve the second determines delay; together these create the variable component of quality attenuation. Traffic management mechanisms vary in the way they control these two issues. In Appendix B, we discuss the TM techniques that are widely deployed, and their impact on network performance. One key application of TM is to keep services within their ‘predictable region of operation’ (PRO); this is particularly important for system services (such as routing updates or keep-alives on a L2TP tunnel) whose failure might mean that all the connections between an ISP and its customers are dropped.

It is important to distinguish between TM that is ‘differential’ (in that it treats some packet flows differently from others) from that which is not, which is far more common (for example rate limiting of a traffic aggregate¹⁴). Differential TM may be intra-user (treating some flows for a particular user differently to others) or inter-user (treating traffic of some users differently from that of others, for example due to different service packages).

TM may be ‘accidental’ (the emergent consequences of default resource sharing behaviour) or ‘intentional’ (configuration of resource sharing mechanisms to achieve some specific outcome). The use of intentional TM to maintain essential services may be uncontroversial, but its application to manage the tension between cost-effectiveness and service quality is not. Because quality attenuation is conserved (as discussed in more detail in §B.3), reducing it for some packet flows inevitably means increasing it for others, to which some users may object. Traffic Management Detection sets out to discover whether such differential treatment is occurring.

1.4.1. Traffic management detection

The purpose of this report is to increase the understanding of the methods and tools available, to understand the art of the possible in the area of TM detection and evaluation. First of all, we must ask: what is the purpose of such detection? It is important to distinguish between testing for the operational effect of an intention and inferring an intention from an observed outcome. The latter is logically impossible, because observing a correlation between two events is not sufficient to prove that one causes the other, and, even if an outcome is definitely caused by e.g. some specific configuration, this does not prove a deliberate intention, as the result might be accidental. The former is possible, but must start from an assumption about the intention; TM detection, by its nature, falls into this category. Secondly, we can ask: what criteria should any TMD methods and tools satisfy? At a minimum, we suggest, in addition to ‘detecting’ TM, any method should:

1. Identify the location of application of TM along the digital delivery chain;
2. Be reliable, minimising false positives and false negatives;

¹⁴Note that, just because TM is not differential does not guarantee that it will be ‘fair’ to all packet flows, as discussed in §B.1.1.4.

3. Be scalable to deliver comprehensive coverage of potential TM locations, without excessive deployment cost or adverse effect on network performance¹⁵.

In §2, we review and compare the most up-to-date techniques in the literature for performing TM detection, and discuss the operational context of such detection in §3.

1.4.2. Traffic management in the UK

A consumer of internet access services (whether domestic or commercial) has to have a connection to some infrastructure, which in the UK is quite diverse in both structure and technology. In Appendix C, we explore the particular characteristics of network provision in the UK, and the implications of this for TM and TM detection. An important aspect is that the delivery of connectivity and performance to the wider Internet is split across different entities, some internal and some external. These form a series of management domains (scopes of control) and administrative domains (scopes of responsibility). Boundaries between these domains are points where TM might be applied; some of them are points where TM *must* be applied to keep services within their PRO. These are illustrated for the UK wireline context in Figure 1.1 on the facing page. Note especially the different coloured arrows that distinguish the level of aggregation at which TM might be applied.

It is important to consider what ‘positive detection’ of traffic management would mean in a UK context. Knowing that there may be traffic management occurring somewhere along the path between an end-user and the Internet does not identify which management / administrative domain it occurs in, which could be:

- before the ISP (even outside the UK);
- within the ISP;
- after the ISP;
- in a local network (depending on router settings).

Thus it is a challenge simply to determine whether the ‘cause’ is within the UK regulatory context. Even ‘locating’ the point at which intentional TM seems to be occurring still leaves the question of whose management domain this is in (and whose administrative domain *that* is in), which may not be straightforward to answer.

1.5. Previous BEREC and Ofcom work

1.5.1. BEREC reports

BEREC has published a variety of reports related to this topic. In general their approach is to look at:

1. Performance of internet access as a whole and its degradation;
2. Performance of individual applications and their degradation.

BEREC’s 2012 report [4] makes the important point that “A precondition for a competitive and transparent market is that end users are fully aware of the actual terms of the services offered. They therefore need appropriate means or tools to monitor the Internet access services, enabling them to know the quality of their services and also to detect potential degradations.” This is a positive and helpful contribution, but it leaves open the question of what parameters should be used to specify the services offered to assure that they are suitable for delivering fit-for-purpose application outcomes. Again this leads to the question of what

¹⁵By its nature, the intention behind any TM applied is unknowable; only the effects of TM are observable. It may be worth noting that, due to this, the best way to ensure end-users receive treatment in line with expectations may be two-fold: to contract to objective and meaningful performance measurements; and to have means to verify that these contracts are met.

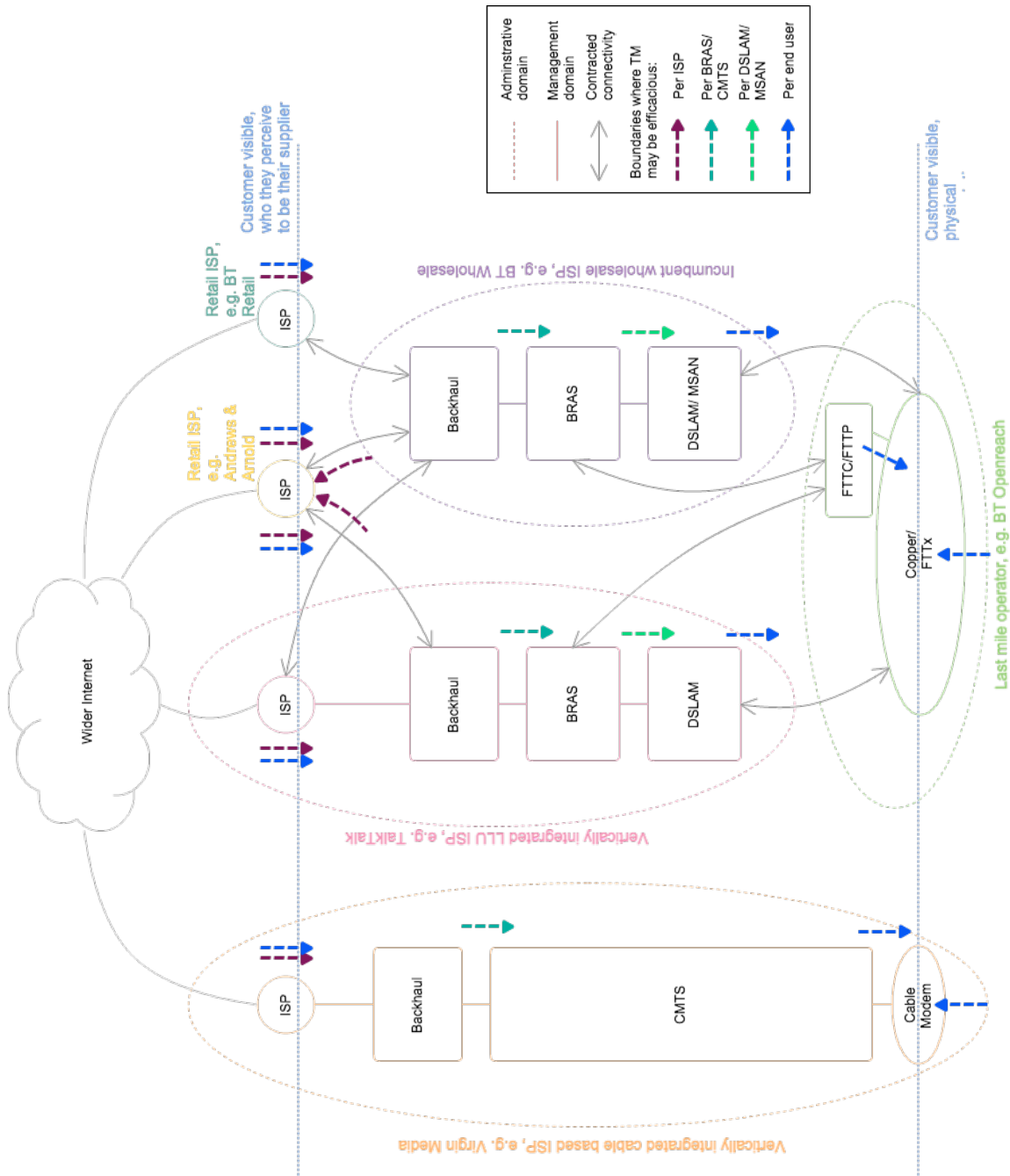


Figure 1.1.: Potential TM points in the UK broadband infrastructure (wireline)

the typical fluctuations of such parameters are during normal operation, so as to distinguish these from the effects of traffic management.

BEREC's most recent report on this topic [5] uses an approach which equates 'equality of treatment' with delivering equality of outcomes. As discussed in §B.2, this assumption does not always hold. It further states (in Section 4.1 of [5]) that delivering good 'scores' against averages of standardised measures will be sufficient to guarantee good outcomes. As discussed in Appendix A, this assumption may also not hold.

However, the BEREC report does identify a number of requirements for quality monitoring systems, but does not explicitly specify that they should be directly relatable to application outcomes. While the report only identifies a small amount of application-specific degradation, it concludes that wide-scale monitoring is desirable. This may be an important recommendation, but may lead to large expenditure and mis-steps without a greater understanding in the industry in general of the relationship between measured performance and fit-for-purpose outcomes.

1.5.2. Notes on previous Ofcom studies

The most recent study on this topic commissioned by Ofcom [6] is very thorough, but misses crucial points:

- There are implicit assumptions that customer QoE is determined primarily by bandwidth and that additional measures such as prioritisation will have predictable effects.
- There is a further assumption that typical measurements of additional parameters, such as average latency, can be reliably related to QoE for 'latency sensitive' applications.

However this 2011 study makes a distinction between 'decision basis' and 'intervention', which is useful, as is the observation that traffic management can vary from user to user depending on their contractual situation and usage history. It also points out that flow identification and marking is generally done at Layer 3, while rate limiting/shaping may be applied at Layer 2; and that traffic management is typically applied in order to deliver better QoE for the majority of users. The comments in section 6 of [6] regarding the difficulty of observing traffic management represent a starting point for this report. However we note that the suggestion in section 8 of [6], that real-time indicators should be provided of whether various services can be supported, can only be realised if they are based on appropriate measures and models (as discussed in §A.2 below) not on proxies such as bandwidth or latency.

1.6. Summary

Communications have changed a great deal in the last half-century, particularly in the shift from dedicated circuits to statistically-shared resources, which has made global connectivity widely affordable. The consequences of this shift are still being worked out, in particular understanding what it would be reasonable for users to expect. As more people and services come to depend on this fundamentally rivalrous resource, the issues of experience, application outcome, consistency and differential treatment (intentional or otherwise) are becoming increasingly important. These factors impact the effectiveness of the delivered service for any individual user's needs-of-the-moment, and hence the value that it has for them.

The stakes are increasing and thus so are the pressures to apply intentional traffic management (if only to mitigate the emergent effects of implicit and unintentional traffic management). Having tools to confirm that the delivered operational characteristics are as intended, and to raise appropriate questions when the intention and the observed outcomes are at odds, will be an important part of the regulator's toolset.

2. Traffic management detection and analysis

2.1. Introduction

In statistically-multiplexed networks such as the Internet, it is inevitable that there will be periods in which various resources are overloaded. At such times, some packets will have to be delayed, and some may need to be discarded. Any mechanism that applies a policy as to which packets will receive which treatment¹ can be called ‘traffic management’. The main focus of interest, however, is on ISP-configured policies that ‘discriminate’ against particular traffic flows. This interest is, as far as it is possible to tell, almost entirely academic. While it might be expected that commercial organisations whose business depends on delivering some form of good experience across the Internet would be interested in this topic, on careful consideration this expectation is misguided. These organisations are dependent on suitable bounds on the quality attenuation, on the path from their servers to the end-users², which is a function of much more than TM policies applied by an ISP. While some ISPs may enable better performance for the application in question than others, exactly why this is the case is of secondary concern³.

2.2. Traffic management

Transferring information generated by one computational process to another, located elsewhere, is called ‘translocation’, and the set of components that performs it is ‘the network’. Instantaneous and completely loss-less translocation is physically impossible, thus all translocation experiences some ‘impairment’ relative to this ideal.

Translocating information as packets that share network resources permits a tremendous degree of flexibility and allows resources to be used more efficiently compared to dedicated circuits. In packet-based networks, multiplexing is a real-time ‘game of chance’; because the state of the network when a packet is inserted is unknowable, exactly what will happen to each packet is uncertain. The result of this ‘game’ is that the onward translocation of each packet to the next element along the path may be delayed, or may not occur at all (the packet may be ‘lost’). This is a source of impairment that is statistical in nature.

The odds of this multiplexing game are affected by several factors, of which load is one. In these ‘games’, when one packet is discarded another is not, and when one is delayed more another is delayed less, i.e. this is a zero-sum game in which quality impairment (loss and delay) is conserved.

As discussed in Appendix B, ‘traffic management’ is applied to the translocation of information through these networks, and its effect is to alter the odds of the multiplexing game and hence the delivered quality attenuation (ΔQ). This ΔQ is the way in which the network impacts the performance of an application⁴.

¹Even FIFO queuing is a policy, and as discussed in §B.1.1, not one that can be assumed to always deliver good outcomes.

²This is discussed in Appendix A.

³Although, where this is the case, commercial organisations may want to measure and publicise this to promote their product.

⁴This is discussed in Appendix A.

Most traffic management detection approaches implicitly use application performance to infer aspects of ΔQ , and thereby draw conclusions regarding the nature of the traffic management; a doubly-indirect process.

2.3. Techniques for detecting traffic management

A variety of approaches have been proposed for detecting whether any form of differential traffic management is being applied at some point in the delivery chain (typically by ISPs). The key papers used in this study are [7, 8, 9, 10, 11], which are collectively the most recent, most cited and most deployed techniques, as revealed by a diligent study of scholarly sources (discussed further in Appendix E). These are described in more detail below. Most aim to provide end-users with a tool that gives some indication of whether such intra-user discrimination is being applied to their own connection. A thorough discussion of the constraints this imposes on the testing process can be found in [8], where Dischinger et al. assert that:

1. Because most users are not technically adept, the interface must be simple and intuitive;
2. We cannot require users to install new software or perform administrative tasks;
3. Because many users have little patience, the system must complete its measurements quickly;
4. To incentivise users to use the system in the first place, the system should display per-user results immediately after completing the measurements.

Since information is translocated between components of an application as sequences of packets, any discrimination must be on the basis of attributes of those packet sequences. Most approaches actively inject traffic whose packets differ in one specific respect from reference packets⁵ and then seek to measure differences in throughput, loss or delay. These approaches are criticised in [9] on the grounds that ISPs might learn to recognise probing packets generated by such tests and avoid giving them discriminatory treatment⁶.

There is a further body of relevant literature, outlined in Appendix E. Few papers appear to have been published in this field in the last two or three years.

2.3.1. NetPolice

This tool was developed at the University of Michigan in 2009, by Ying Zhang and Zhuoqing Morley Mao of the University of Michigan and Ming Zhang of Microsoft Research [10].

2.3.1.1. Aim

This system, called NetPolice, aims to detect content- and routing-based differentiations in backbone (as opposed to access) ISPs. This is mainly to inform large users, such as content providers, rather than individual end-users.

2.3.1.2. Framing the aim

NetPolice focuses on detecting traffic differentiation occurring in backbone ISPs that results in packet loss. Since backbone ISPs connect only to other ISPs, not to end-users, this can only be done by measuring loss between end-hosts connected to *access* ISPs. By selecting paths between different access ISPs that share a common backbone ISP (a technique that is conceptually similar to the network tomography approach discussed in §2.3.7) measurements can be inferred for the common backbone ISP. ISPs are distinguished on the basis of their ‘autonomous system’ (AS) number.

⁵These reference packets may be passively observed as in [12] or actively generated such as in [8, 10].

⁶It is to be noted that this would only become likely if such methods came to be used widely, which so far none have.

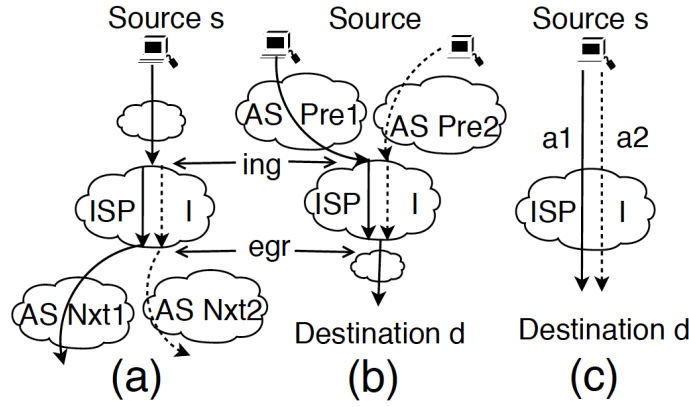


Figure 2.1.: Detecting various types of differentiation with end-host based probing
Reproduced from [10]

Key challenges included selecting an appropriate set of probing destinations to get a sufficient coverage of paths through backbone ISPs⁷ and ensuring the robustness of detection results to measurement noise. The system was deployed on the PlanetLab platform and used to study 18 large ISPs spanning 3 continents over 10 weeks in 2008.

2.3.1.3. Implementation

NetPolice exchanges traffic between end-hosts, selected so that paths between them have appropriate degrees of difference and commonality, and measures loss rates in order to detect differentiation. To measure the loss rate along a particular subsection of the end-to-end path, NetPolice sends probe packets with pre-computed TTL values that will trigger ICMP ‘time exceeded’ responses⁸, unless the packet is lost. As packet loss may occur in either direction, large probe packets are used to ensure the measured loss is mostly due to forward path loss, on the assumption that large probe packets are more likely to be dropped than small ICMP response packets on the reverse path. Subtracting the measured loss rate of the sub-path to the ingress of a particular AS from that of the egress from it provides the loss rate of the internal path. Figure 2.1 illustrates how NetPolice uses measurements from end systems to identify differentiation in ISP *I*. In Figure 2.1(a), an end host probes two paths sharing the same ingress and egress within ISP *I*, but diverging into two distinct next-hop ASes after the egress. By comparing the loss performance of the two paths, NetPolice determines whether ISP *I* treats traffic differently based on the next-hop ASes. Similarly, Figure 2.1(b) shows how NetPolice detects differentiation based on previous-hop ASes. In Figure 2.1(c), an end-host probes a path that traverses the same ingress and egress of ISP *I* to the same destination.

To detect content-based differentiation, the tool measures loss rates of paths using different application traffic. Five representative applications were used: HTTP; BitTorrent; SMTP; PPLive; and VoIP. HTTP was used as the baseline to compare performance with other applications, on the assumption that it would receive neither preferential nor prejudicial treatment. The remaining four applications were selected based on a prior expectation that they may be treated differently by backbone ISPs. Packet content from real application traces was used, with all packets padded to the same (large) size, and their sending rate restricted to avoid ICMP rate-limiting constraints⁹. NetPolice detects differentiation by observing the differ-

⁷Choosing the optimal set of hosts to exchange traffic in order to probe a particular sub-path is an instance of the set covering/packing problem, a classic question in combinatorics, computer science and complexity theory. See https://en.wikipedia.org/wiki/Set_packing, which also includes some discussion of useful heuristics.

⁸Although an ICMP response may be forwarded on a slow path, this will not affect the loss measurement provided the packet is not dropped.

⁹Intermediate routers limit the rate of ICMP requests they will respond to.

ences in average loss rates measured along the same backbone ISP path using different types of probe traffic.

The issue of network load induced by probing is addressed by means of “collaborative probing”. This consists of selecting end-host pairs whose connecting paths traverse the sub-paths of interest. The selection is made so that these sub-paths are probed sufficiently often (by traffic between different pairs of hosts) whilst ensuring that the probing traffic is spread out over different access ISPs.

Differences due to varying network load (rather than ‘deliberate’ differentiation) were addressed by:

1. taking repeated measurements;
2. assuming even distribution of “random noise”¹⁰;
3. applying multivariate statistical tests to the measurements to compare the distributions of baseline and selected application traffic.

2.3.1.4. TM techniques detected

Only traffic management that induces packet loss can be detected¹¹. Since the rate of each probing flow is low, this must be applied to a traffic aggregate (i.e. an aggregated flow of packets from many users sharing some common attribute). Thus rate policing of aggregate traffic based on port number, packet contents and/or source/destination AS is the only mechanism detected.

2.3.1.5. Discussion

In the paper it is assumed that inaccuracy of loss rate measurements is likely to be caused by three main factors:

1. overloaded probers;
2. ICMP rate limiting at routers; and
3. loss on the reverse path.

Little evidence is produced to justify these assumptions other than a partial validation of single-ended loss-rate measurements against a subset of double-ended measurements (i.e. loss rate measured at the remote host), by plotting the corresponding CDFs and showing that they are broadly similar. There is also a correlation of the results with TOS values returned in the ICMP response packets, presumably added by ISP ingress routers.

Since packets are padded to the same (large) size, and their sending rate restricted to avoid ICMP rate limiting constraints, the packet streams are not representative of real application traces.

Note that routers typically limit their ICMP response rate (on some aggregate basis), in order to ensure that other critical router functions remain within their PRO. Thus, it would seem that consistent application of this technique would require a single point of control to coordinate the packet streams in order to avoid exceeding this rate at any router being probed. Also the possibility that routers may have this function disabled altogether must be considered.

This technique is restricted to detecting TM performed by Tier 1 ISPs. Therefore it appears to have limited applicability for ISPs with multiple geographically diverse subnetworks within the same AS.

There is a fundamental difficulty with ensuring that the selection of end hosts is optimal and that all sub-paths will be probed, particularly in the presence of dynamic routing.

¹⁰The paper’s authors’ term for the effects of congestion.

¹¹In ΔQ terms, what is actually being measured is an approximation to that part of $\Delta Q|_V$ whose packets are never delivered or whose delays are beyond a cut-off, in this case the duration of the test, since it is impossible to distinguish packet loss from very large delay by observation.

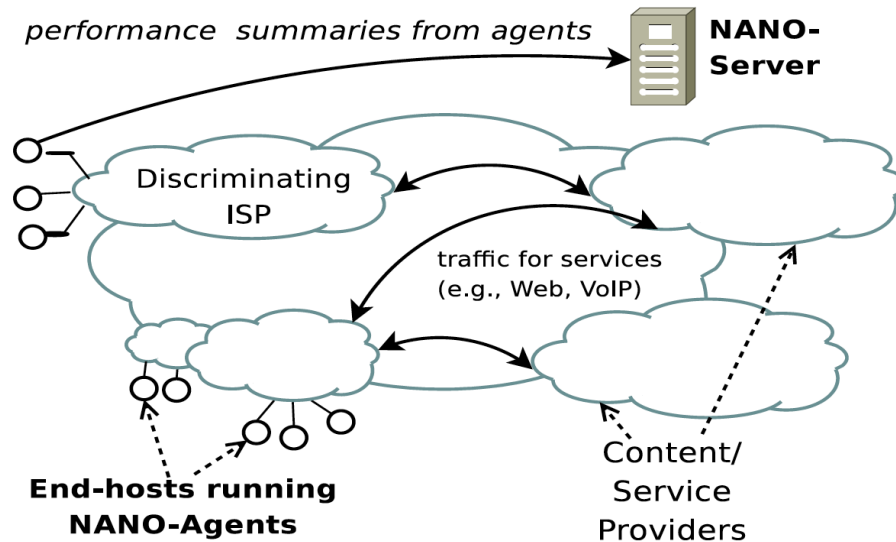


Figure 2.2.: NANO architecture
Reproduced from [9]

2.3.2. NANO

Detecting Network Neutrality Violations with Causal Inference, here referred to by the name of its technique NANO, is a 2009 paper by Mukarram Bin Tariq, Murtaza Motiwala, Nick Feamster and Mostafa Ammar at the Georgia Institute of Technology [9].

Aim

The aim is to detect whether an ISP causes performance degradation for a service when compared to performance for the same service through other ISPs.

Framing the aim

A service is an “atomic unit” of discrimination (e.g. a group of users or a network-based application). ‘Discrimination’ is an ISP policy to treat traffic for some subset of services differently such that it causes degradation in performance for the service. An ISP is considered to ‘cause’ degradation in performance for some service if a causal relation can be established between the ISP and the observed degradation. For example, an ISP may discriminate against traffic, such that performance for its service degrades, on the basis of application (e.g. Web search); domain; or type of media (e.g. video or audio). In causal analyses, “X causes Y” means that a change in the value of X (the “treatment variable”) should cause a change in value of Y (the “outcome variable”). A “confounding variable” is one that correlates both with the treatment variable in question (i.e. the ISP) and the outcome variable (i.e. the performance).

NANO is a passive method that collects observations of both packet-level performance data and local conditions (e.g. CPU load, OS, connection type). To distinguish discrimination from other causes of degradation (e.g. overload, misconfiguration, failure), NANO establishes a causal relationship between an ISP and observed performance by adjusting for confounding factors that would lead to an erroneous conclusion. To detect discrimination the tool must identify the ISP (as opposed to any other possible factor) as the underlying cause of discrimination.

Implementation

NANO agents deployed at participating clients across the Internet collect packet-level performance data for selected services (to estimate the throughput and latency that the packets experience for a TCP flow) and report this information to centralised servers, as shown in Figure 2.2. Confounding factors are enumerated and evaluated for each measurement. The values of confounding factors (e.g. local CPU load) are stratified¹². Stratification consists of placing values into ‘buckets’ (strata) sufficiently narrow that such values can be considered essentially equal, while also being wide enough that a large enough sample of measurements can be accumulated. Measurements are combined with those whose confounding factors fall into the same strata, and statistical techniques drawn from clinical trial analysis are used to suggest causal relationships. Stratification requires enumerating all of the confounding variables, as leaving any one variable unaccounted for makes the results invalid. NANO considers three groups of such confounding variables: client-based, such as the choice of web-browser, operating system, etc.; network-based, such as the location of the client or ISP relative to the location of the servers; and time-based, i.e. time of day.

Discussion

NANO captures specific protocol interactions related to TCP, measuring the interaction of the network with the performance of an application. This is mediated by the behaviour of the sending and receiving TCP stacks. As such, it does not measure delay and loss directly, but rather the combined effects of both the bi-directional data transport and the remote server. From a ΔQ perspective (discussed in more detail in Appendix A), the measurements are of an application outcome (throughput achieved over a TCP connection), which is highly dependent on $\Delta Q_{|G,S}$, as well as on $\Delta Q_{|V}$, the component that is affected by TM.

The technique has significant advantages that come with passive data collection such as: protection from preferential treatment for probe traffic; an absence of resource saturation caused by testing; and no impact on user data caps (where applicable), other than server upload (which is not deemed significant). A disadvantage of being entirely passive, however, is that data gathering depends on usage profiles of participating users.

Collecting data on local conditions helps to isolate some confounding factors. While the statistical basis for the work and the use of stratification as a technique within which to do comparative testing is well-established, it has also been criticised, e.g. in [13]. The paper asserts that NANO can isolate discrimination without knowing the ISP’s policy, as long as values are known for the confounding factors. It further asserts that these confounding factors are “not difficult to enumerate using domain knowledge”, an assertion that may need both further investigation and justification that is not provided in the paper itself. While this technique has had successful test deployments (using a combination of Emulab and PlanetLab), this proof-of-concept run does not seem to provide an adequate basis for the assumptions made with respect to the possible set of confounding factors. There appears to be an implicit assumption that the only difference between one ISP and another is the TM that they perform. At one point the idea of “network peculiarities” is mentioned as something on which performance might depend, but if, for instance, the technology used in one network (e.g. cable) gave a different set of performance criteria to another (e.g. 3G) it is unclear whether or not this would be seen as discrimination¹³.

Nano has the advantage of adding only minimal traffic to the network (only that required to report the results to the central server), but it does not seem to provide any way to establish where in the digital supply chain any discrimination is taking place, unless it were possible to observe packets at intermediate points. Combining the sophisticated statistical approach here with some variant of the network tomography ideas discussed in §2.3.7 might produce a

¹²http://en.wikipedia.org/wiki/Stratified_sampling

¹³Clarifying this would require laboratory-based study.

powerful and scalable tool, although the computational cost of performing the analysis would need to be investigated.

2.3.3. DiffProbe

DiffProbe was developed by P. Kanuparth and C. Dovrolis at the Georgia Institute of Technology in 2010 [12].

Aim

The objective of this paper was to detect whether an access ISP is deploying mechanisms such as priority scheduling, variations of WFQ¹⁴, or WRED¹⁵ to discriminate against some of its customers' flows. DiffProbe aims to detect if the ISP is using delay discrimination, loss discrimination, or both.

Framing the aim

The basic idea in DiffProbe is to compare the delays and packet losses experienced by two flows: an Application flow A and a Probing flow P . The tool sends (and then receives) these two flows through the network concurrently, and then compares their statistical delay and loss characteristics. Discrimination is detected when the two flows experience a statistically significant difference in queueing delay and/or loss rate. The A flow can be generated by an actual application or it can be an application packet trace that the tool replays. It represents traffic that the user suspects their ISP may be discriminating against (e.g. BitTorrent or Skype). The P traffic is a synthetic flow that is created by DiffProbe under two constraints: firstly, if there is no discrimination, it should experience the same network performance as the A flow; secondly it should be classified by the ISP differently from the A flow.

Implementation

DiffProbe is implemented as an automated tool, written in C and tested on Linux platforms, comprising two endpoints: the client (CLI, run by the user), and the server (SRV). It operates in two phases: in the first phase, CLI sends timestamped probing streams to SRV, and SRV collects the one-way delay time series¹⁶ of A and P flows; in the second phase, the roles of CLI and SRV are reversed.

DiffProbe generates the A flow using traces from Skype and Vonage¹⁷. Various aspects of the A flow are randomised (port, payload, packet size and rate) to generate the P flow.

Two techniques are used to minimise the rate of false positives, i.e. to ensure that the two flows see similar network performance when the ISP does *not* perform discrimination. The first of these is to consider only those P packets that have been sent close in time with a corresponding A packet¹⁸. Secondly, when a P packet is sent shortly after an A packet, it is generated such that it has the same size as that A packet. This ensures that the network transmission delays of the (A , P) packet pairs considered are similar. This is illustrated in Figure 2.3.

¹⁴WFQ is a form of bandwidth sharing, described in §B.4.3.

¹⁵WRED is a form of policing and shaping, as discussed in §B.4.4 and §B.4.5, in which packets are discarded with some probability when the queue is in states other than full.

¹⁶The term 'time series' as used in this paper means the end-to-end delays of a flow, after subtracting the minimum observed measurement from the raw end-to-end delay measurements. The presence of a clock offset does not influence these measurements as the focus is on relative, not absolute delays.

¹⁷This is presumably based on an expectation that these particular applications may be discriminated against.

¹⁸This should mean that, even if the P flow includes many more packets than the A flow, with different sizes and inter-arrival intervals, only (A , P) packet pairs that have 'sampled' the network at about the same time are considered.

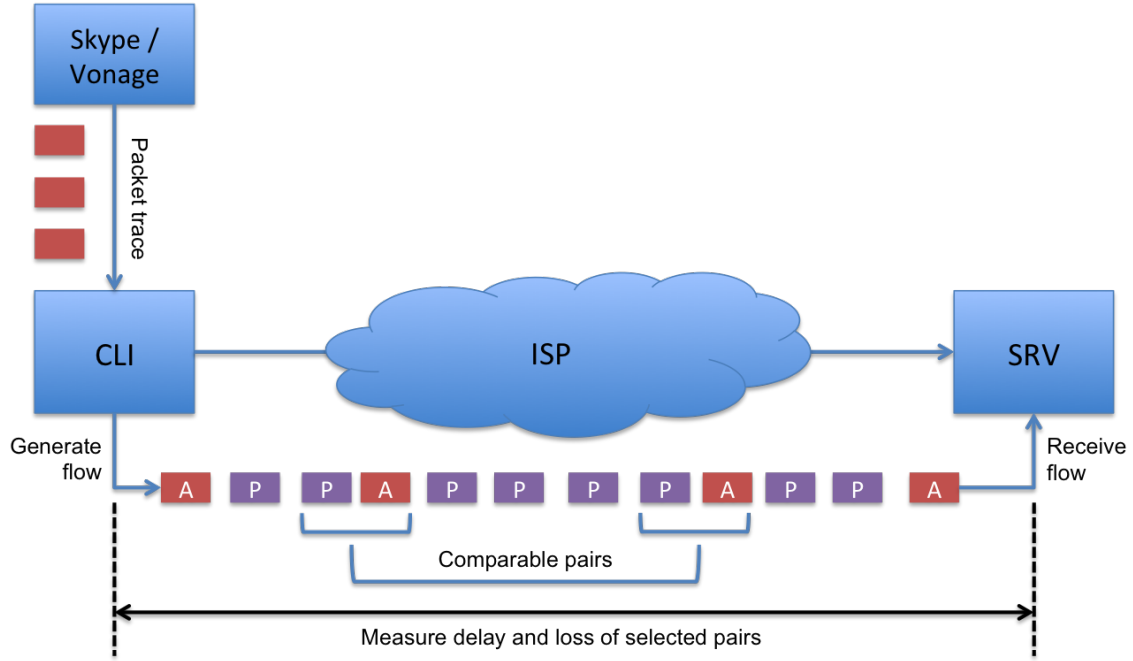


Figure 2.3.: DiffProbe architecture

In order to increase the chances that a queue will form inside the ISP, causing the supposed discriminatory mechanism to be applied, the rate of the P flow is increased to close to the rate of the access link (whose capacity is estimated in a previous phase¹⁹). If no significant difference²⁰ is detected between the delays during an interval with a typical load and one with an increased load, the measurement is discarded (on the grounds that no discrimination has been triggered).

Discrimination is detected by comparing the delay distributions of the (A, P) pairs, taking account of the fact that many packets experience a delay that is dominated by propagation and transmission times²¹. If the delay distributions are statistically equivalent, then a null result is returned. Otherwise they are compared to see if one is consistently and significantly larger than the other.

Loss discrimination is also measured, by comparing the proportion of lost packets in the two flows. In order to apply the chosen significance test, the high-load period is extended until at least 10 packets are lost from each of the flows.

TM methods detected

Discrimination due to strict priority queuing is distinguished from that due to WFQ on the basis of the delay distribution of the ‘favoured’ packets (see Figure 2.4, reproduced from the paper). This approach detects both delay-affecting TM (such as Priority Queuing, discussed in §B.4.2, and bandwidth sharing, discussed in §B.4.3) and loss-affecting TM, such as WRED¹⁵.

Discussion

This paper considers both delay and loss discrimination, but unfortunately treats delay and loss as entirely separate phenomena (whereas they are always linked through the two degrees of

¹⁹This is done by: sending K packet trains of L packets, each of size S ; at the receiver, measuring the dispersion D for each train (the extent to which packets have become separated in their passage across the network); estimating the path capacity as: $C = (L-1)S/D$; finally, taking the median of the K trains [14].

²⁰The differential factor for this decision was chosen empirically.

²¹In terms of ΔQ , the process in Footnote 16 can be seen as an estimation of the unidirectional $\Delta Q_{|G}$. The statistical test used here appears to have been chosen mitigate the effects of $\Delta Q_{|S}$, which manifests here as a (unwanted) correlation between packet size and delay.

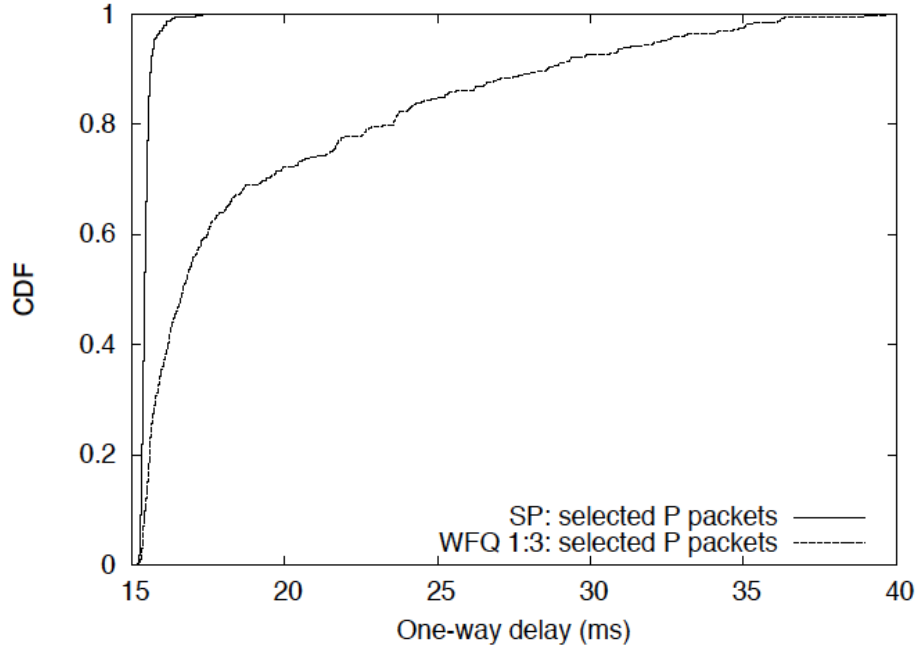


Figure 2.4.: Delay distributions due to strict priority and WFQ scheduling (simulated)
Reproduced from [12]

freedom that all queueing systems inherently have). By considering only differential delays²², $\Delta Q|_G$ is effectively separated from the other components of ΔQ . However, it appears that $\Delta Q|_S$ is not fully considered²³ and the authors do not exploit the fact that $\Delta Q|_V$ can be extracted from the full ΔQ . This leads to the use of a complex statistical test in order to cope with delay distributions having a large cluster of measurements around $\Delta Q|_{G,S}$.

This approach tries to avoid the (common) overly-strong stationarity assumption (that packets sent at different times will see essentially similar quality attenuation) by selecting packet pairs for comparison. However, this requires care to avoid the edge effect of the loss process due to tail drop²⁴ (or other buffer exhaustion, see §B.1.1.4). There is no apparent evidence that such care has been taken in this case; in particular the fact that the selected (A, P) packet pairs always have the P packet second may introduce bias²⁵.

There is an assumption in the paper that any differential treatment will only be manifest when a particular network element is reaching resource saturation²⁶. To bring this about, the offered load of the P traffic is increased until it reaches the (previously determined) constricting rate. In a typical UK broadband deployment, this method would likely only detect differential treatment on the access link. In the upstream direction this would be in the CPE device (under the nominal control of the end user themselves); and in the downstream direction would typically be under the control of the wholesale management domain²⁷. If the *retail* ISP was engaging in such discrimination²⁸, it would be applied to the traffic aggregate whose load this test would be unlikely to influence to any significant degree.

²²There appears to be no consideration of clock drift between the client and server during the duration of the test.

²³By measuring only limiting performance of a fixed size stream of UDP packets, there is an implied assumption that there is a linear relationship between packet size and service time. It also seems to be assumed that TCP packets will experience identical treatment.

²⁴As this is not a continuous process, but a discrete one, it can have a large effect on the relative application outcome.

²⁵To investigate this further would require laboratory experiments.

²⁶The authors say “we are not interested in such low load conditions because there is no effective discrimination in such cases”.

²⁷Whose configuration would be independent of the particular ISP serving the end-user.

²⁸Some UK retail ISP’s Ts&Cs reserve the right to differentially treat certain classes of traffic during “periods of abnormal load”, in order to maintain key services within their PRO.

The loss discrimination test requires an arbitrarily long duration since it cannot complete until 10 packets have been lost in each stream.

There seems to be a contradiction between the decision to focus on VoIP applications and the approach for inducing discrimination by loading the network, which is not the normal behaviour of such applications; indeed an ISP could easily classify such traffic as part of a DDOS attack.

It is acknowledged that some appearances of discrimination are due to routing changes and that this needs to be accounted for; such accounting does not seem to have been disclosed in the paper.

There does not appear to be a bulk deployment of this measurement approach, nor does it appear to be in active development. The paper's authors went on to create ShaperProbe (§ 2.3.5 on page 31) which is available on M-Lab, but this only measures throughput and its limitation, not delay and loss characteristics.

This technique seems unable to distinguish TM applied at different points on the path between the client and the server.

2.3.4. Glasnost

Glasnost is the work of M. Dischinger, M. Marcon, S. Guha, K. P. Gummadi, R. Mahajan and S. Saroiu at both the MPI-SWS (Max Planck Institute for Software Systems) and Microsoft Research in 2010 [8].

Aim

The aim of Glasnost is to enable users to detect if they are subject to traffic differentiation. The question that Glasnost tries to answer is whether an individual user's traffic is being differentiated on the basis of application, in order to make any differentiation along their paths transparent to them. This project particularly aims to reach a mass of non-technical users, while providing reliable results to each individual.

Framing the aim

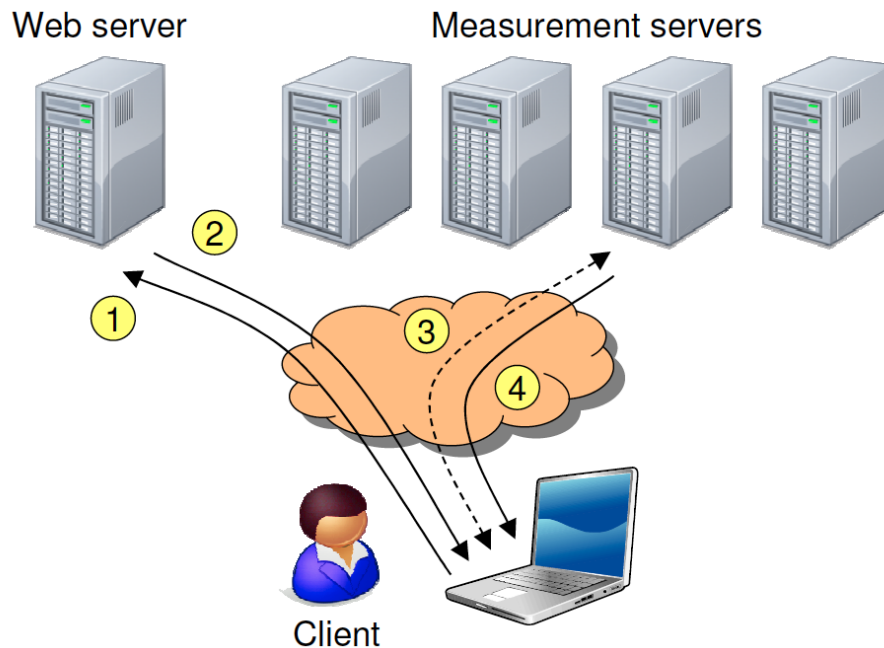
Glasnost detects the presence of differentiation based on its impact on application performance. It does this by determining whether flows exhibit different behaviour by application even when other potential variables are kept constant. The key assumptions are:

1. ISPs distinguish traffic flows on the basis of certain packet characteristics, in particular port number or packet contents;
2. ISPs may treat these distinguished flows to and/or from an individual user differently;
3. Such differential treatment can be detected by its impact on application performance;
4. Confounding factors²⁹ can be controlled or are sufficiently transient that a sequence of repeated tests will eliminate them, while not being so transient that they have an impact on one flow but not on the other;
5. Users may not have administrative privileges on the computers they use and are unable/unwilling to engage with technical issues.

The approach is to generate a pair of flows that are identical in all respects except one; this one respect is chosen as it is expected to trigger differentiation along the path. This is illustrated in Figure 2.6. Comparing the performance³⁰ of these flows is the means to determine whether differentiation is indeed present.

²⁹Such factors include the user's operating system, especially its networking stack and its configuration, and other traffic, either from the user or other sources.

³⁰In principle, various performance measures could be used, but in the current implementation, the only parameter measured is throughput of TCP flows.



(1) The client contacts the Glasnost webpage. (2) The webpage returns the address of a measurement server. (3) The client connects to the measurement server and loads a Java applet. The applet then starts to emulate a sequence of flows. (4) After the test is done, the collected data is analysed and a results page is displayed to the client.

Figure 2.5.: The Glasnost system
Reproduced from [8]

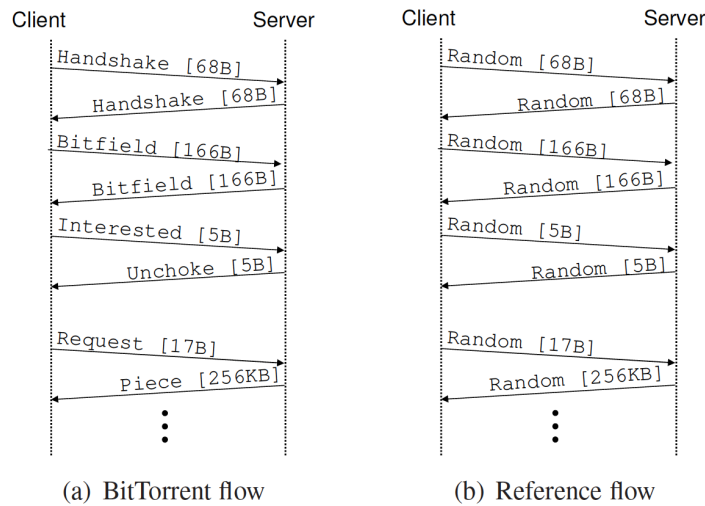
Implementation

The current implementation of Glasnost detects traffic differentiation that is triggered by transport protocol headers (i.e. port numbers) or packet payload. The tool works using a Java applet downloaded from a webpage. This acts as a client that opens a TCP session to communicate with a Glasnost server, as illustrated in Figure 2.5. This client/server service then runs pairs of emulated application flows back-to-back to detect throughput differentiation between them. In each pair the first uses the port number or packet payload that may be being differentiated against; the second uses random data intended to have all the same characteristics except that being tested for (e.g. non-standard port number and random packet contents, as illustrated in Figure 2.6). Upstream and downstream tests are “bundled” to make the tests complete faster and the tests are repeated several times to address the confounding factor of “noise” due to cross-traffic³¹. Experimental investigations on throughput led to a classification of cross-traffic as being one of the following:

- Consistently low;
- Mostly low;
- Highly variable;
- Mostly high.

Measurements that suggest cross-traffic is ‘highly variable’ or ‘mostly high’ are discarded.

³¹This means traffic contending in the multiplexing tree to the sink, as discussed in §A.1.



A pair of flows used in Glasnost tests. The two flows are identical in all aspects other than their packet payloads, which allows detection of differentiation that targets flows based on their packet contents.

Figure 2.6.: Glasnost flow emulation
Reproduced from [8]

Detectable TM techniques

TM techniques detectable by Glasnost would be those that impact the throughput of a TCP session for certain flows to/from a particular user. Thus techniques such as bandwidth sharing or prioritisation *between* users will not seem to be detectable. Rate-limiting of specific types of traffic should be detectable provided the limit is less than other constraints, such as the rate of the access link. If rate-limiting is being applied to a traffic aggregate (e.g. the total amount of P2P traffic rather than that of any particular user), then it will only be detectable if the aggregate rate exceeds the limit (i.e. it is dependent on the actions of other users of the network). Rate limiting that is applied only when the network is heavily loaded may not be detectable due to the rejection of measurements when cross-traffic is high or highly variable.

Discussion

While this method is capable of detecting differentiation against a single application by a single method, it seems to lack a coherent analysis of potential confounding factors. These are aggregated as “noise”, which is dealt with by performing repeated tests³². The paper includes a discussion of false results (both positive and negative), quantified by an empirical method. However, claims for the robustness of the results are based on empirical analysis of a relatively small data set, and the assessment appears to be affected by assumptions and axiomatic beliefs (enumerated in Framing the aim above).

Significant emphasis is placed on the advantages of an active measurement approach, and the benefits of using emulated rather than actual applications. However this is likely to be an unfaithful reproduction of real application behaviour, as the timing of the application packet stream is not reproduced. Moreover, using TCP throughput measurements adds variability to the tests, due to the interaction of the Java VM with the specific OS TCP stack; thus two users connected to the same network endpoint could report different results. The paper makes

³²The paper points out that limitations are imposed by end-user attention span, with the result that the length and number of iterations of the tests was reduced, which may compromise the statistical significance of the results.

strong claims of generality for this approach, while admitting that substantial compromises had to be made for the sake of user-friendliness. For example, in section 5.3 of the paper it is mentioned that new, shorter tests were implemented to increase test completion rates and combat problems caused by user impatience³³. As part of this the tests for upstream and downstream directions were “bundled”. It is unclear what is meant by this, but if it means that both upstream and downstream tests are carried out at the same time or with overlap, self-contention could add a confounding factor, in particular the interaction of TCP ‘acks’ and bulk elastic data flow behaviour.

While it is claimed that “Glasnost detects the presence of differentiation based on its impact on application performance”, it appears the only type of application performance that is measured is achievable TCP throughput. This is relevant if the application in question is BitTorrent, but not if it has real-time characteristics, e.g. an interactive web session or VoIP. The Glasnost design also tries to create an adaptable system that can be configured for novel management methods. This is laudable and a logical step but, given the potential variety of TM policies that might be applied, detecting all of them from a single end-point may swiftly prove to be infeasible. The construction of the detector itself and its apparent reliance on limited aspects of an application’s performance seem to make the system’s ability to generally distinguish differentiation questionable.

This technique appears unable to distinguish TM applied at different points on the path between the client and the server.

2.3.5. ShaperProbe

ShaperProbe was developed by P. Kanuparth and C. Dovrolis at the Georgia Institute of Technology in 2011 [7].

Aim

The question that ShaperProbe tries to answer is whether a token bucket shaper (as described in § B.4.4 on page 73) is being applied to a user’s traffic. This is intended to be an active measurement service that can scale to thousands of users per day, addressing challenges of accuracy, usability and non-intrusiveness.

Framing the aim

ShaperProbe tries to address this aim by asking whether a shaper kicks in once a certain (unknown) data transfer rate is reached. It first estimates the link rate, then sends bursts³⁴ of maximum-sized packets at a series of rising data rates (up to just below the estimated limiting rate). It looks for the point where the packet rate measured at the receiver drops off, by counting arrivals in a given interval (this is illustrated in Figure 2.7). If the delivered rate drops to a lower rate after a period of time, the presence of a token-bucket traffic shaper on the path is declared, and its token generation rate and bucket depth estimated, based on the amount of data sent before the rate dropped and the asymptotic rate.

Measured values are adjusted to smooth the rate-response curve. To minimise intrusiveness, probing is terminated early when either shaping is detected or packets are lost.

Implementation

The technique is to first use short UDP packet trains to get an estimate for the limiting link rate³⁵. This is done by sending short trains of back-to-back maximum-sized packets

³³The number of tests for each combination of port pairs was reduced to one. The remaining tests take 6 minutes.

³⁴These bursts have constant spacing between their constituent packets.

³⁵This seems to assume that these packet trains are short enough not to be affected by shaping themselves.

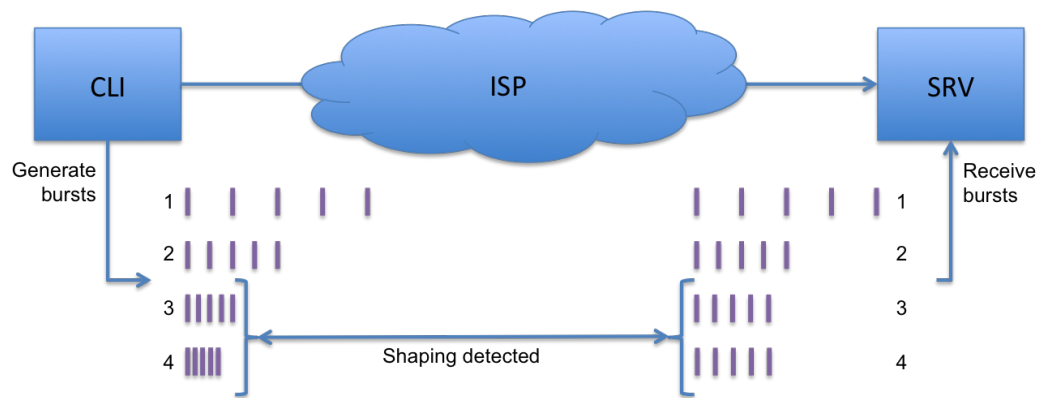


Figure 2.7.: ShaperProbe method

DiffProbe release. January 2012.
Shaper Detection Module.

Connected to server 4.71.254.149.

Estimating capacity:
Upstream: 2976 Kbps.
Downstream: 96214 Kbps.

The measurement will last for about 3.0 minutes. Please wait.
Checking for traffic shapers:

Upstream: No shaper detected.
Median received rate: 2912 Kbps.

Downstream: No shaper detected.
Median received rate: 59957 Kbps.

For more information, visit: <http://www.cc.gatech.edu/~partha/diffprobe>

Figure 2.8.: ShaperProbe sample output

and observing their arrival times³⁶. The spacing of these packets at the receiver should be constant, given that packet sizes are constant in the offered load. However, the packet arrivals can be affected by experiencing non-empty queues. To deal with this, standard nonparametric rank statistics are applied to derive a “robust estimator” (note that this may differ from the allocated capacity - see Figure 2.8).

The total burst length and the threshold rate ratio for detection were chosen empirically, using a small sample, to maximise the detection rate (this is described in the Technical Report [15]).

The ShaperProbe client is a download-and-click user-space binary (no superuser privileges or installation needed) for 32/64-bit Windows, Linux, and OS X; a plugin is also available for the Vuze BitTorrent client. The non-UI logic is about 6000 lines of open-source code.

An example output from running the tool from a UK cable-connected endpoint is shown in Figure 2.8; note that this appears to seriously overestimate the allocated downstream rate of 60Mb/s (as advertised by the ISP and recorded by SamKnows).

The tool is deployed on M-Lab, which hosts the servers, and the tests reported in the paper were performed on a number of ISPs between 2009 and 2011.

³⁶As previously discussed in footnote 19 on page 26.

Detectable TM techniques

Token bucket shapers with a sufficient bucket size should be detected but those which kick in very quickly may not be seen. False positive results could be caused by coupled behaviour, for example a large file download by another user of the same shared last-mile segment (e.g. cable segment), which would result in a drop in the received rate by the tool. Since results are discarded if any loss occurs, policers will not be detected.

Discussion

There is some analysis of the robustness of the results, using case studies where the ISPs had declared their shaping policies, but the vulnerability to ‘cross traffic’ (i.e. contention along the path between client and server) is unclear.

There are classes of traffic conformance algorithms that would seem to be undetectable using this approach, such as those proposed and used in ATM traffic management [16], and those in use in BRASs in UK networks³⁷. Shaping, as detected here, is only likely to be deployed in systems that statistically share last-mile access capacity, as discussed in § B.6 on page 75. The paper reports a false positive rate of 6.4%, but then claims a rate of less than 5% without apparent further justification.

This technique seems unable to distinguish TM applied at different points on the path between the client and the server.

2.3.6. ChkDiff

Chkdiff is a 2012 work of Riccardo Ravaoli and Guillaume Urvoy-Keller, of l’Université Nice Sophia Antipolis, and Chadi Barakat of INRIA [11].

Aim

The question that Chkdiff tries to answer is whether traffic is being differentiated on the basis of application. It attempts to do this in a way that is not specific to the application or to the discrimination mechanisms in use. Rather than testing for the presence of a particular TM method, this approach simply asks whether any differentiation is observable.

Framing the aim

In order to answer this question, this approach tries to observe user traffic in such a way as to detect whether specific flows have different performance characteristics when compared to the user’s traffic as a whole. The key design principles are:

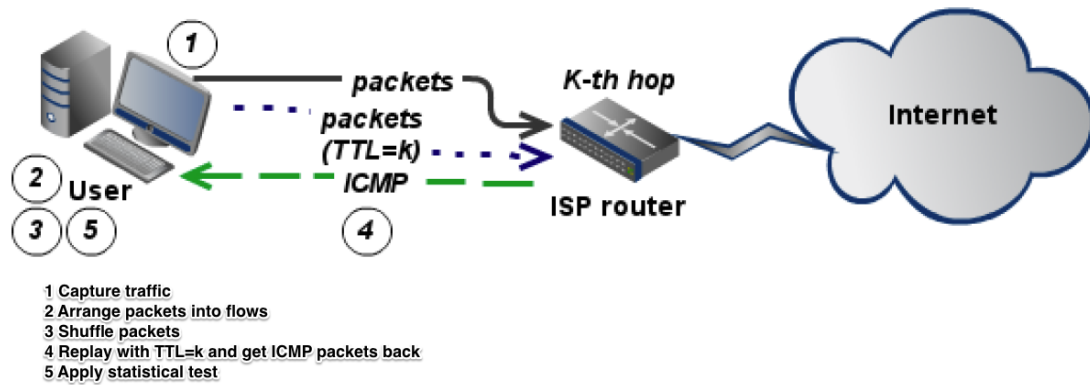
1. Use only user-generated traffic;
2. Leave user traffic unchanged;
3. Use the performance of the whole of the user’s traffic as the performance baseline.

Implementation

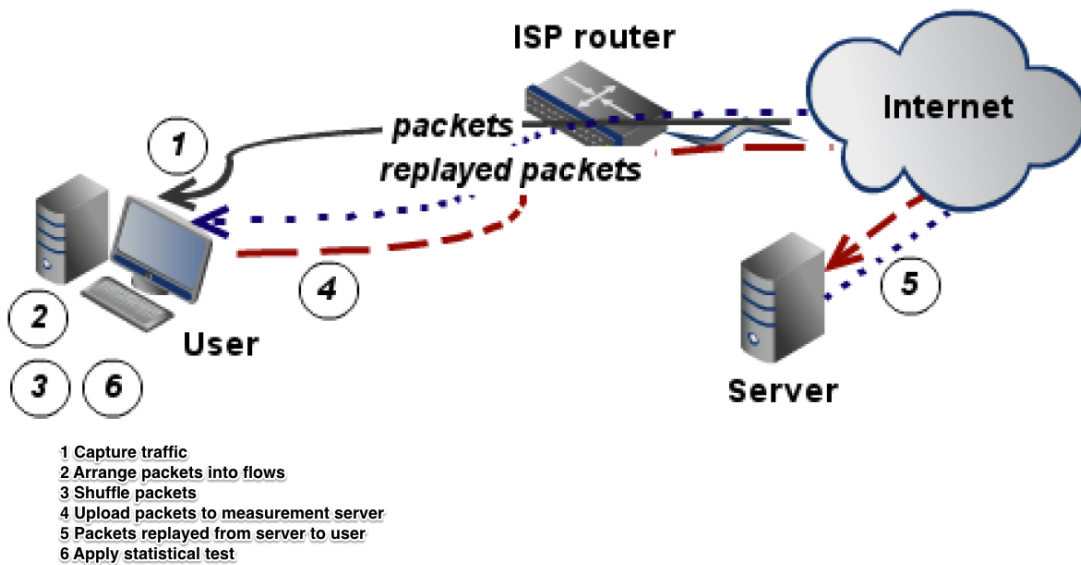
The process is represented in Figure 2.9 (note that the downstream component has not been implemented). The metric used in the upstream direction is the round-trip time (RTT) between the user and a selected router on their access ISP; the number of hops to the router is selected by modifying the TTL field. The process is:

1. Capture user traffic for a fixed time-window of a few minutes;

³⁷Fully clarifying the range of applicability and limitations of this technique would require laboratory investigation.



(a) Upstream



(b) Downstream

Figure 2.9.: Chkdiff architecture
 Reproduced from [11]

2. Classify the traffic into flows using the packet header information;
3. Generate a test by repeatedly picking packets from different flows at random, weighted by the overall volume of each flow;
4. Focus the measurement by setting the value of the TTL fields of the packets;
5. Apply a statistical test, by fitting delay histograms to a Dirichlet distribution.

User-generated packet traces are replayed with modified TTL fields, and the time to receive the ICMP response is measured³⁸. Different flows are mixed by taking Bernoulli samples in order to invoke the PASTA property³⁹, and the results are compared for different flows on the basis of the distribution of response times (using histograms).

A downstream test is proposed using a similar system, in which arriving packets are captured at the client, and then uploaded to a server for replay. This has not been implemented.

Detectable TM techniques

This very general method would be able to detect delay differentiation between different flows, e.g. due to priority queuing or WFQ applied on a per-application or network host basis. However, it would be unable to detect differentiation on an individual end-user basis, since it relies on the aggregate performance of the user's traffic as a baseline. Thus, any differentiation that affects the user's traffic as a whole (e.g. a token bucket shaper as discussed in § B.4.4 on page 73) would not be able to be detected. Since packet loss is not measured, techniques that affect loss such as WRED could not be detected.

Discussion

By measuring the distribution of round-trip delays, this approach is very close to measuring differential ΔQ , so the aim of "application and differentiation technique agnosticism" is sound. Extending the method to include measuring loss, as proposed, would make their measure correspond more closely to ΔQ , except that it measures round-trip instead of one-way delays. By measuring delays to intermediate points, this approach laudably aims to localise rather than merely detect differentiation. The principal disadvantage of this method appears to be that it relies on the fidelity of the intermediate routers' ICMP response to the packet expiry. Generating ICMP responses is not a priority for routers, and so the response time is highly load-dependent; also the rate limitation on ICMP responses may have an impact on the scalability of the technique.

Applying this technique in the downstream direction would require a server to replay spoofed packets. This has not been implemented.

False positives and negatives do not seem to be well addressed in the paper, but Chkdif was only in early development when it was written.

Overall this is a promising approach, and it is a pity that it does not seem to have been developed beyond a laboratory prototype.

2.3.7. Network Tomography

Network tomography is a body of work that takes a multi-point observational approach to measuring network performance [17, 18, 19].

Aim

Network tomography uses the 'performance' of packets traversing a network much as radiologic tomography uses the 'performance' of X-rays passing through the body. X-ray intensity is

³⁸Note that this is the same technique used by NetPolice [10], discussed in § 2.3.1 on page 20.

³⁹This means the results are robust against transient and phase-related effects.

modulated by the tissues passed through; packet performance is modulated by the path traversed. Using multiple ingress and egress points on the periphery of the network means this is seen as analogous to a CT scan of a body, in that distinct internal features become visible by combining multiple measurements. A recent paper by Zhang [20] explores the use of this approach for the detection of differential treatment of traffic.

Framing the aim

The approach is to start with a description of the network's connectivity at a link/path level, expressed as an adjacency matrix \mathbf{A} . This is combined with a vector of external observations \vec{y} , to infer a vector \vec{x} of the link/path properties by solving the following system of equations:

$$\vec{y} = \mathbf{A} \cdot \vec{x}$$

In principle, if more than enough observations are available, the system can be solved using only a subset of them. The insight relevant to TM detection is that if different subsets of observations yield *different* results for any particular internal link/path, this could indicate the presence of some differential treatment⁴⁰. By selecting the subsets of observations in different ways, insights might be gained as to the factors that trigger differential treatment. Useful subsets might be aspects of the path and/or association data (addressing, content), packet contents, etc..

Implementation

These papers have been written in the context of mathematical ‘thought experiments’, and where validation has been performed this has been done as simulations. No deployable tool has yet been produced.

Discussion

There appear to be several underlying assumptions. Firstly, this approach explicitly requires knowledge of the structure of the network at a link/path level, which may be hard to discover. It also seems to assume that the routing and link structure of the network is constant for the set of observations, which may not be the case given the dynamic nature of routing protocols. Secondly, there is an important requirement on the mathematical structure of the performance measure in order to validly solve the equations⁴¹. This means that the type of values that can be solved for do not seem to correspond to realistic performance measures⁴². In particular, ΔQ (discussed in §A.2.1) is not a simple scalar⁴³, so the particular solution process proposed in this body of literature could not be directly applied to it.

However, combined with an appropriate performance measure⁴⁴, this approach does represent a potential way forward for detecting TM effects. The tomographic approach supports not only detecting whether discrimination is performed on the basis of application or originator, but also the evaluation of differential service between customers. It could provide a scalable means of assessing whether classes of users were actually receiving the service that

⁴⁰Zang et al express this as the system being “unsolvable”; they appear to be making the assumption that a “neutral network” will form a system of equations that are solvable, even if they are massively over-specified.

⁴¹In order to solve a system of equations, the values have to have a particular set of mathematical properties (such as those that hold for real numbers). Typically they must form a ‘field’ (see http://en.wikipedia.org/wiki/Linear_equation_over_a_ring) in order to form \mathbf{A}^{-1} (the inverse of \mathbf{A}) so that $\mathbf{A}^{-1} \cdot \vec{y} = \mathbf{A}^{-1} \cdot \mathbf{A} \cdot \vec{x} = \vec{x}$ can be calculated.

⁴²Adding average delays is not meaningful, nor is adding up ‘congestion’, for example.

⁴³Mathematically, ΔQ is akin to a cancellative monoid, http://en.wikipedia.org/wiki/Cancellative_semigroup.

⁴⁴Using a solution approach that is mathematically appropriate to such a performance measure.

they expected (for example whether ‘premium’ customers receive a markedly different service from ‘standard’ ones). Thus conformance to marketing claims and T&Cs may be able to be independently assessed.

The power of this approach is that it does not focus on a single metric of interest, e.g. throughput, but takes a general observational approach (much like NANO and Chkdif, with which it might usefully be combined). Also it does not, by its nature, entail stressing the network infrastructure⁴⁵. It could be done in an entirely passive way or make use of only low bandwidth test streams. All of these factors mean it could be deployed on a large scale. However, considerable further research would be required to develop a practical methodology; encouragingly, this is one area in which research seems to be ongoing.

⁴⁵The approach taken by Glasnost and ShaperProbe is to by drive a path to saturation so that any differential treatments come into play and hence become measurable.

3. Traffic Management detection in an operational context

3.1. Introduction

In Chapter 2, various approaches to detecting the presence of differential traffic management were discussed. Most of these approaches are designed for sporadic use by individual end-users. In this chapter, the focus is on the operational behaviours and scalability of these detection approaches and their potential application and impact in an operational context (i.e. by actors other than individual end-users).

3.2. Review of TM detection techniques

It is inherently impossible to detect directly the specific application of differential treatment (other than by inspecting the configuration of network elements). Even when there is such an intention, it may not have any effect, depending on the particular circumstances of load, etc.. Thus the techniques listed in Table 3.1 do not directly detect traffic management, but rather attempt to infer its presence through structured observations. They look for differences in specific aspects of translocation performance, either directly by measuring delay or loss (though none measures both together) or indirectly by measuring the operational performance of TCP bulk transport.

Traffic Management detection literature, as surveyed in §2.3, typically starts from the assumption that discrimination is occurring and that the task is to detect it. Such presumed discrimination falls into one (or both) of two broad categories:

1. Restriction on the freedom of association - the ability to have access to a particular service, to a particular location (e.g. server) or from a particular location (e.g. client)¹. This restriction can take one of several forms: e.g. port blocking, intercepting protocol behaviour to insert resets, or hijacking domain name resolution. Identification of the association can be done on the basis of the addressing in the packets², their ingress/egress ASNs and/or contents (i.e. using DPI);
2. Taking deliberate actions that impact the *performance* of some set of associations³ identified as above. For example, limiting the transported load of traffic identified as P2P.

The approaches are structured to detect performance differences, typically measured end-to-end. They then aim to infer that these differences are caused by application of discriminatory queueing and scheduling somewhere along the path. This inference hinges on several factors:

- The nature of “discrimination”. To discriminate, two steps are needed: firstly, a classification or choice needs to be made to distinguish packets belonging to one flow from those belonging to others; secondly, a difference needs to be applied in the treatment of the packet exchanges making up such flows. How this choice can be made is discussed in § 3.2.1 on the facing page;

¹Firewalls are an expression of this freedom to associate, in particular the freedom to *not* associate.

²An example would be discarding all packets to or from a particular set of addresses when responding to a DDOS attack.

³This is done by increasing the ΔQ of the corresponding translocation.

- The underlying assumptions being made in the construction of the detection approach; these are discussed in § 3.2.2;
- The likely efficacy of such approaches in an adversarial context. Some of the aspects of this are explored from a “game” perspective in § 3.3 on the following page. Various forms of discriminatory practice can be envisaged that would not be detected by any of the techniques discussed in §2.3.

3.2.1. Technical aspects of flow differentiation

Packet flow discrimination can be done by classifying packets based on addressing information⁴, the pattern of offered load, or a combination thereof. Note that devices have access to more ‘address’ information than just the IP source and destination contained within the packet itself. This can be explicit⁵ or derived⁶: explicitly derived from the packet header⁷, or based on an analysis of the SDU⁸. The pattern of offered load can be measured using a token-based scheme⁹ or historical information (such as volume used over some previous period).

Only after classification has occurred can a particular queueing/scheduling choice be applied. From that choice, differential behaviour of the end-to-end packet flows can emerge (i.e. differential delivered ΔQ). That, in turn, can lead to differential protocol performance and application outcomes.

3.2.2. Underlying assumptions made in TMD techniques

The general assumption made in most TMD approaches is that TM is the cause of differentiation in service. This is a narrow approach that does not seek to understand the factors influencing the performance of applications and protocols, but rather aims to ‘prove’ the hypothesis that ‘the ISP’ is restricting the delivered service to some degree. This is done by trying to disprove the ‘null hypothesis’ that no differentiation is taking place. Thus TMD techniques typically fall into the general category of statistical hypothesis testing¹⁰. Such testing depends on being able to conclude that any differences in the resulting outcome can be unambiguously attributed to a constructed distinction between a ‘test’ and a ‘control’. It is important to show that such differences are not due to some other ‘confounding’ factor that would result in false positive/negative results. In the absence of a comprehensive model of the factors affecting performance, the methodology is to control as many potential confounding factors as possible, and deal with others by means of statistics¹¹.

There are many possible confounding factors that seem to have not been taken fully into account by any of the approaches. One such factor is the inherent variability in the performance of PBSM, which leads a number of techniques to discard measurements when there is ‘noise’ due to contention (i.e. for which $\Delta Q_{|V}$ is too large). However, as discussed in Appendix B, it is precisely in the allocation of $\Delta Q_{|V}$ that the effects of TM are manifest. Thus many approaches to TMD deliberately ignore the circumstances in which TM is most likely to be active. Another implicit assumption is that occasional tests from self-elected end hosts can

⁴Note that classification on the basis of addressing information is effectively reverse-engineering the end-point association, endeavouring to identify some aspect of the ‘parties’ involved - such as application, provider and customer.

⁵This can be based on the VLAN, some virtual router function, or the physical port of reception/transmission.

⁶Derived information includes the originating/terminating/next-hop AS number.

⁷One example of this could be port numbers in the transport layer header.

⁸This is typically done by deep-packet inspection. Note that this becomes more difficult when packet contents are encrypted or otherwise modified, e.g. by compression.

⁹This is as described in Appendix B.4.4, where arrivals reduce the token pool that is being filled at a set rate; when the pool empties the stream is treated differently.

¹⁰http://en.wikipedia.org/wiki/Statistical_hypothesis_testing

¹¹This can easily lead to assuming that correlation implies causation.

be expected to detect reliably differential traffic management. This would only be the case if such TM were applied uniformly.

A further assumption is that the underlying end-to-end performance (in the absence of any deliberate differentiation) is the same for the ‘test’ and ‘control’ experiment streams¹². The effect of this is minimised when the packets for the two streams are interleaved.

Some techniques assume that ICMP responses from intermediate routers can be relied upon. However, ICMP was not intended to provide accurate performance data, and responses to pings or TTL exhaustion are entirely at the mercy of the processing load of the targeted router and its application of ICMP rate limiting.

In order to create repeatable tests, captured or emulated traces are often used¹³, generally of TCP sessions. This implicitly assumes that actual application/protocol behaviour is not important. So, while TMD techniques are attempting to compare application outcomes (in particular protocol performance), some do so only by comparing differential treatment of TCP behaviour, which leads to information fidelity loss¹⁴. Furthermore, the protocol peer has specific implementation and parameter settings that may differ by application, and there may be other unknown factors such as loading and performance issues (e.g. power saving by the end device).

3.2.3. Comparison of main approaches

We classify the most interesting approaches by the following criteria:

Readiness Level To what extent the technique is available to be exploited;

Active or passive Whether the approach actively injects test packets or passively observes the existing traffic flow; if active, whether it relies on saturating the constraining link of the end-to-end path and an estimate of the traffic volume generated;

Detect based on What measured property of selected flows is used to detect discrimination;

TM types Which TM techniques the approach is designed to detect;

Target TM locations Where in the end-to-end path TM is being looked for;

Measurement duration How long an individual test may take;

Test traffic volume Estimated volume of traffic generated per test; note that this will in many cases depend on the sync rate of the end-user’s line¹⁵.

Supply Chain Localisation Ability to localise TM in a heterogenous digital supply chain.

Table 3.1 compares the different approaches on these criteria.

3.3. Likely efficacy of TMD in a UK context

Even where some correlation could be detected, the UK market (see Appendix C) is such that there often would not be a single administrative/management domain to which the discrimination can be attributed, as shown in Figure 1.1. The authors agree with the authors of [20] that detection of the location where traffic management is being deployed is as important as the detection of its existence. A clear issue-isolation process is required for any operational framework.

TM detection techniques have been mostly developed in North America, where the market structure differs from that of the UK. Where there is a single integrated supplier, as is typical

¹²This is to say that $\Delta Q^{A \leftrightarrow Z}$ is stationary over the period of measurement.

¹³With the exception of NANO that collects protocol data; this has the issue that it may leak privacy-related information, such as which servers were contacted.

¹⁴An example of this, and the consequences of it, can be found in [21].

¹⁵For example, a 10Mb/s DSL line delivers approximately 1MB/s of user-level data. Thus saturating such a link for one minute will consume 60MB.

Paper	Readiness Level	Active or passive	Detect based on	TM types	Target TM locations	Test duration	Test traffic volume per test	Supply chain localisation
NetPolice [10]	Deployed on PlanetLab during research	Active	Differential loss by AS number	Rate limiting	Tier 1 ISPs	2 hours	One ICMP packet/s per element tested	ISP exchange points only
NANO [9]	Deployed on PlanetLab and Emulab during research	Passive	TCP throughput and latency by association/addressing	Various	Local ISP	Unknown	2.5kb/s per end-user for reported results	None
DiffProbe [12]	NS trials - then deprecated	Active Saturating	Differential delay distributions and differential loss by association/addressing	Queueing and prioritisation	Whole path	15s minimum; many repetitions	Unbounded: 10s link saturation per test	None
Glasnost [8]	Deployed at scale (MLab)	Active Saturating	Differential throughput by association/addressing	All affecting elastic throughput	Whole path	6 minutes	6 minutes of saturation per test	None
Shaper Probe [7]	Deployed at scale (MLab)	Active Saturating	Throughput variation over time per end-user	Rate limiting	Whole path	2-3 minutes	Variable: up to c. 1GB	None
ChkDiff [11]	Lab trials only	Mixed	Distribution of RTTs to intermediate router by association/addressing	All delay affecting	All	c. 10 minutes?	Unknown	User-visible Layer 3 routers
Network Tomography	Only tested in simulation	Either	Performance measures over multiple paths by association/addressing	All (depending on performance metric)	All	unknown	Unquantified but low	Good

Table 3.1.: Taxonomy of Traffic Management Detection Approaches.

in North America, establishing that discrimination is occurring somewhere on the path to the end-user is broadly sufficient to identify who is responsible, but when there are multiple administrative domains involved, as in the UK, the situation is more complex.

3.3.1. Offered-load-based differentiation

Differential service on the basis of offered load has been part of the contractual relationship at network boundaries since the inception of PBSM (e.g. ATM used this as the major basis of service differentiation). Control of the offered load by means of rate limiting is an essential element needed for stable operation of PBSM, and it is present at multiple locations¹⁶. There is extensive use of such limiting at management/administrative boundaries to manage both bills and costs.

Detection of the most limiting network egress point is feasible, e.g. ShaperProbe, though this technique does make the implicit assumption that network contention effects (which could create false results) are absent.

Detection of the presence of such rate/pattern limiting can be done at the receiving end point with a single-point measurement process¹⁷, and could deliver measurements for each direction separately. As with all single-point measurement processes, there is no spatial isolation. i.e. it is not possible to say where along the path the limiting occurred. In this case, in order to apply a high load, traffic must be sent to a remote host, i.e. along an entire end-to-end path¹⁸. Without intermediate measurement points (i.e. multi-point measurement) there is no way to isolate which section of the path induces the most stringent limitation.

Several major UK network providers make these limits available, either in their commercial T&Cs (in the terms of “up to”) or in their technical interfaces (i.e. ADSL sync rates and BRAS limiters). As each of these measures is an upper bound, which only apply when there are no other data transport quality impairment effects.

3.3.2. Association-based differentiation

Some differentiation may depend on the association, i.e. exactly what the communicating entities are (e.g. an end-host at a particular IP address - the user - communicating with a server in a particular domain, or using a particular protocol). All the TM detection techniques that were found are single-point measures of a composite effect, typically involving multiple administrative/management domains, two directions of flow and some computational element.

Epistemologically the best that such techniques can do is to detect some differential treatment of the traffic flows that will result in a different observed distribution of delay and loss for that composite set of effects. They may do this directly, either by passive observation (as by NANO, §2.3.2), or by active measurement (as by NetPolice, §2.3.1, and DiffProbe, §2.3.3), or indirectly by measuring the effects on the performance outcomes of an application (as by Glasnost, §2.3.4). NetPolice’s inability to detect TM applied to individual users would make it of limited use for the detection of differential TM. Its key feature of distinguishing between differentiation applied by backbone ISPs can probably be addressed more systematically by using a variant of network tomography (discussed in §2.3.7).

The majority of approaches endeavour to “prove” that application-based differentiation is occurring on traffic to/from a particular end user. In contrast, network tomography-based approaches would use a more general strategy that may be a better fit for use for the detection of differential TM. Additionally, such approaches would have benefits in terms of scalability and localisation.

¹⁶Given that every network interface is, in effect, a rate limiter, rate limiting could be said to be everywhere.

¹⁷This means observing any particular flow at a single point in its journey. There may be multiple measurement locations, but each of them is a single point measure. This means that all the techniques discussed here have no spatial localisation.

¹⁸Techniques to ‘probe’ intermediate routers using ICMP responses are inherently rate-limited.

The reviewed techniques may detect the existence of differential traffic treatment, but not pinpoint its location (with the exception of network tomography-type approaches); nor are they reliably able to assure the absence of such treatment due to the sporadic nature of the tests and the effect of confounding factors. Localisation might be addressed by mandating the installation of measurement points at suitable administrative boundaries, rather than relying entirely on measurements performed from the edge of the network.

3.3.3. Cost of the detection process

A common misconception is that additional load ‘costs nothing’, however wide-scale use of the saturating active methods could place a significant load on the network as a whole. For example, a single test on a 60Mbit/s connection taking several minutes, represents the load of several hundred average broadband users over that period. Although the assumption is that network traffic has no marginal cost, anecdotal evidence suggests that test traffic can be a significant factor driving capacity upgrades [22]. NANO does not have this issue (it is passive) and network tomography approaches could use either passive or low data rate active analysis¹⁹.

3.3.4. TM detection techniques as proxy for user experience impairment

Glasnost and ShaperProbe are the only techniques that appear widely deployed (using M-Lab²⁰), and both are focused on bandwidth “impairment”. ShaperProbe does this at the uni-directional packet flow level: it is about capping the “up to” speed and does not aim to detect differential treatment based on association, only offered load. Glasnost does this at the bi-directional application outcome level; although the Glasnost paper implies that it can emulate (via synthetic behaviour) multiple applications, examination of the information available via M-Lab²¹ shows that this test approach is only suitable for bulk data transfers (transfers that try to saturate the path to the end user) whose time-to-complete is more than 10 seconds. Thus this is not a suitable proxy for many user interactions, which are either short-lived (getting email, interacting with Twitter or Facebook), or have different usage patterns, like video streaming (which may last a longer time). Typical video streaming (e.g. YouTube) is not a bulk data transfer, because it is not endeavouring to saturate the path, but rather aiming to ensure that the play-out buffer does not empty to maintain the continuity of the video delivery. Other types of video streaming such as DASH or iPlayer do use TCP (via HTTP) to download ‘chunks’ of content. However, in this case maximising the TCP peak transfer rate can have a negative impact on application performance, by downloading a chunk so quickly that the TCP connection closes down before the next chunk is started. Once again, the details of the application behaviour matter.

Scrutiny of the M-Lab data for 2013 does not generate great confidence in the reliability or efficacy of these methods: the data set is actually quite small, and, because tests require active participation by end-users, the sample is inherently biased.

The set of ways in which TM techniques that could be differentially/prejudicially applied is much greater than the set that the available tools could detect. The authors can imagine several ways in which, for example, Glasnost could be ‘gamed’²².

¹⁹There are distinct advantages to using low data rate active analysis. By exploiting the PASTA principle, as used by ChkDiff, the data rate could be very low - a few bits per second. The active data would not have any particular privacy issues in that it would not contain any information that can be tied back to the user’s activity, *except* for the induced delay and loss experienced.

²⁰M-Lab hosts are generally located in academic institutions, however, so would not be representative of a typical consumer experience.

²¹<http://broadband.mpi-sws.org/transparency/createtest.html>

²²The problem of applying a measure whose optimisation actually benefits the end-user is not dissimilar to the problem of creating a CPU benchmark that reflects real application performance; see for example <http://goo.gl/S6sZd7>.

The absence of an established baseline makes it impossible to detect discrimination on a per-user basis (or sub-set of users). Furthermore the absence of detected prejudicial treatment does not imply the received service is going to be fit for any intended purpose, such as video streaming, VoIP conversation or gaming.

4. Conclusions and recommendations

4.1. Conclusions

The success of packet-based statistically-multiplexed networks such as the Internet is dependent on sharing resources dynamically. This dynamic sharing is ubiquitous, occurring at every WiFi access point, mobile base station and switch/router port. Each of these multiplexing points allocates its resources in response to the instantaneous demand placed upon it, which can typically exceed the available supply. The result depends on the sharing mechanism employed, its configuration, and the pattern of the demand (as discussed in some detail in Appendix B). Whether the outcome is ‘biased’ or ‘fair’ depends on many factors, including:

- The nature or aspect of the resource being shared (e.g. ingress to versus egress from a buffer);
- The pattern of the demand;
- The configuration of the sharing mechanism; and
- The exact definition of ‘fairness’ (per packet? per flow? per application? per outcome? per user? etc.).

Insofar as the outcome depends on the configuration of the sharing mechanism, any configuration may be called ‘traffic management’ (TM). TM may be used to maintain the stability of network services by creating outcomes that are deliberately ‘unfair’. For example, it might be ‘fair’ for a temporary overload to cause equal packet loss and delay across all flows, but where some of those flows are essential to maintain the operation of the network such ‘fairness’ is undesirable. TM may also be used to select one form of ‘fairness’ over another, for example, to ensure that all users receive a similar level of service, even when some are applying much higher levels of demand than others.

The emergent effects of many multiplexing points joined in a network are complex; consequently so is the relationship between desired outcomes and actual behaviour¹. What ultimately matters to any application is the probability distribution of loss and delay in the delivery of its packets; this may be influenced by TM but not completely controlled by it. It is this delivered distribution² that determines user satisfaction; how this is achieved is of little concern to either end-users or their content and service suppliers - except when it is unsatisfactory. Poor performance may have many causes, including the overall network architecture and topology, capacity planning and in-life management. ‘Traffic Management’ is only part of the equation.

Presumably for this reason, traffic management detection (TMD) has been pursued almost entirely from an academic perspective³. Given the complexity of the relationship between desired outcomes and actual behaviour, inferring an intention from observed outcomes is effectively impossible. Rather than trying to address this general problem, most TMD starts from assumed intentions mediated by assumed particular TM techniques and then attempts to deduce whether or not certain observations are consistent with such assumptions. However, even positive results do not prove a deliberate intent to introduce bias; given the overall

¹Further, laboratory-based study would be required to elucidate this relationship further. It may be possible to quantify ‘typical’ behaviour, so that unusual circumstances meriting investigation, for example by TMD, can be detected.

²Which we refer to as ‘quality attenuation’ and designate ‘ ΔQ ’.

³Initial interest from M-Lab (supported by Google) has diminished in the last few years.

complexity of relating intentions to outcomes, demonstrating a differential outcome does not demonstrate an intent to produce that outcome.

Most research completed in this area (explored in Chapter 2) has been undertaken from the perspective of allocating responsibility for both quality of experience and use of traffic management in single, vertically-integrated suppliers. These approaches might not be suitable in the UK due to its heterogeneous broadband delivery structure, detailed in Appendix C; even if it could be shown that some users or applications were being differentially treated, there is (in most cases) no single administrative entity that can be shown to be responsible. Some approaches attempt to localise the TM by using responses from intermediate routers; apart from the potential inaccuracy of this method, any attempt at large-scale deployment risks hitting the limits imposed on such responses⁴.

Table 4.1 summarises table 3.1 with respect to the criteria set out in §1.4.1, using the legend that ‘✓’ means a requirement is met; a ‘✗’ means that it is not met; a ‘—’ means that it is partially met; and a ‘?’ means that there is insufficient evidence to reach a reliable conclusion. Reliability of the methods is essentially unknown because, while most of the papers make estimates of their technique’s reliability, there has been no independent and uniform confirmation of these claims.

Technique	Localisation	Reliability	Scalability
NetPolice	—	?	—
NANO	✗	?	✓
Diffprobe	✗	?	✗
Glasnost	✗	?	✗
ShaperProbe	✗	?	✗
ChkDiff	—	?	✓
Network Tomography	✓	?	✓

Table 4.1.: Comparison of techniques with criteria

None of the TMD methods studied satisfy all the key attributes that would make them suitable for effective practical use. In particular, those that are currently in active deployment generate significant volumes of traffic, which would risk damaging the QoE of other users if applied widely, and incur costs to the service providers of carrying this traffic; thus they may be unsuitable for large-scale use. The reliability of these tools would require further study, using a uniform test environment in which their performance could be objectively compared.

It is easy to envisage TM policies that would not be detectable by any of the methods analysed, and in any case, TMD techniques that test for specific configurations of specific TM mechanisms risk being rendered rapidly obsolete by new TM approaches and more sophisticated service provider policies⁵. The introduction of SDN, as discussed in [23], makes it likely that TM policies may be reconfigured on a timescale much shorter than any of the available tools can obtain statistically reliable results. It is not clear where the effort would come from to update TMD techniques or to develop new ones, particularly since the focus of academic interest appears to have moved elsewhere. Finally, these tools are limited in that they aim only to detect the presence of differential (intra-user) traffic management, as the detection of non-differential traffic management (inter-user or aggregate) was not their goal.

These tools are not sufficient to enable effective detection and location of TM application along a fragmented digital delivery chain such as that in the UK. Our conclusion is thus that no tool or combination of tools currently available is suitable for effective practical use.

⁴Indeed, service providers might well conclude that their routers were under attack and thus decide to disable such responses altogether.

⁵Only NANO and Chkdif may be sufficiently general to overcome this problem.

4.2. Recommendations

TMD sits within a wider context of ensuring that internet service provision satisfies suitable criteria of fitness-for-purpose, transparency and fairness. Confirming such properties is challenging because of the inherently statistical nature of packet-based networks, and is further complicated by the heterogeneity of the digital supply chain. The absence of differential traffic management does not, by itself, guarantee fairness, nor does fairness guarantee fitness-for-purpose. TMD is thus, at best, one component of an overall solution for measuring network service provision. However, it could be used to help establish transparency; for example, if TM policies to be used on end-user traffic were published, their implementation could be independently verified.

Another difficulty in measuring fairness and fitness-for-purpose of network service provision is the application-dependent relationship between network performance and application outcomes (discussed in Appendix A). This means that particular differences in performance may or may not matter to end-users, depending on the applications they are using. The choice of application also determines which aspects of the delivered performance are significant⁶. TMD thus risks highlighting aspects of service provision that are largely irrelevant, while overlooking others that could have a significant impact, depending on the applications in use. This is a subject for further study.

TMD needs to be considered in relation to a broader framework for evaluating network performance. This framework should encompass two aspects. The first would be application-specific demands, captured in a way that is unbiased, objective, verifiable and adaptable to new applications as they appear. This could be used to ascertain the demand profile of key network applications, which would give operators more visibility of what performance they should support, and OTT suppliers encouragement to produce “better” applications (imposing a lower demand on the network). The second would be a system of measurement for service delivery that could be unequivocally related to application needs. This would be necessary if one wished to know if a particular network service was fit-for-purpose with respect to an particular application. This measurement system would need to deal with the heterogeneous nature of the supply chain by reliably locating performance impairments whilst avoiding unreasonable loads on the network. Due to significant boundaries along the end-to-end path, responsibility could only be ascribed to commercial entities if these needs were met. A development of the tomographic approaches discussed in §2.3.7, combined with a generic network performance measure such as ΔQ (outlined in Appendix A) has the potential to do this. TMD could then become a way to fill in any gaps in this overall framework⁷.

Collection and publication of data within such a framework could have a transformative effect on the broadband market in the UK and beyond. Ofcom’s publication of performance tables has already significantly benefited the market situation. Further benefit may be gained by enhancing this with richer data relating to application needs and complete network performance (beyond bandwidth measures). Users could then be empowered to choose applications that were appropriate for their network service⁸. Conversely users could choose network services that were fit for the applications they want to use⁹; if there were any interest in selecting network services that additionally did or did not apply specific forms of TM, then TMD would have a role.

More work is needed to better manage the relationship between supply, demand and delivered quality. This should address the systemic issue of the lack of feedback on demand, either

⁶VoIP is more sensitive to delay while VoD is typically more sensitive to loss, for example.

⁷How much benefit there would be in checking conformance to criteria that have no significant impact on end-user application performance is debatable.

⁸For example, a user whose service was known to have significant variation in latency could choose the online gaming platform that was least sensitive to this.

⁹For example, a user interested in a streaming video service might prefer a service with sufficient throughput and stable translocation characteristics over one with much higher throughput but occasional variations that might cause playback glitches.

to consumers (encouraging them to time shift demand, making better use of spare capacity) or to application producers (to make applications more efficient). Consistency of supply can be addressed with an appropriate measurement framework, as discussed above. Finally, we recommend investigating how a “quality floor¹⁰” could be maintained, perhaps requiring short-timescale incentives¹¹ such as some form of Pigovian tax¹².

¹⁰I.e. a bound on the end-to-end quality attenuation.

¹¹This is needed because the timescales on which customers can switch are far too long compared with the timescales on which bad-actors could exploit them.

¹²http://en.wikipedia.org/wiki/Pigovian_tax

Bibliography

- [1] Ofcom Commercial Team. Consultancy framework mini competition: A study of traffic management detection methods and tools mc no: Mc/316. Restricted Tender, February 2014.
- [2] Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication*. Number ISBN 0-252-72548-4. Univ of Illinois Press, 1949.
- [3] Ofcom. *Ofcom's approach to net neutrality*, 2011.
- [4] Guidelines for Quality of Service in the scope of Net Neutrality. Technical Report BoR (12) 32, BEREC, May 2012.
- [5] Monitoring quality of internet access services in the context of net neutrality. Technical Report BoR (14) 24, BEREC, March 2014.
- [6] Jeremy Klein, Jonathan Freeman, Rob Morland, and Stuart Revell. Traffic management and quality of experience. Technical report, Ofcom/Technologia, April 2011.
- [7] Partha Kanuparth and Constantine Dovrolis. Shaperprobe: End-to-end detection of isp traffic shaping using active methods. pages 473–482, 2011. URL: <http://www.measurementlab.net/measurement-lab-tools#tool5>, doi:10.1145/2068816.2068860.
- [8] Marcel Dischinger, Massimiliano Marcon, Saikat Guha, P Krishna Gummadi, Ratul Mahajan, and Stefan Saroiu. Glasnost: Enabling end users to detect traffic differentiation. In *NSDI*, pages 405–418, 2010.
- [9] Mukarram Bin Tariq, Murtaza Motiwala, Nick Feamster, and Mostafa Ammar. Detecting network neutrality violations with causal inference [online]. 2009. URL: <http://noise-lab.net/projects/old-projects/nano/>.
- [10] Ying Zhang, Zhuoqing Morley Mao, and Ming Zhang. Detecting traffic differentiation in backbone isps with netpolice. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 103–115. ACM, 2009.
- [11] Riccardo Ravaioli, Chadi Barakat, and Guillaume Urvoy-Keller. Chkdif: Checking traffic differentiation at internet access. In *Proceedings of the 2012 ACM Conference on CoNEXT Student Workshop*, CoNEXT Student '12, pages 57–58, New York, NY, USA, 2012. ACM. URL: <http://doi.acm.org/10.1145/2413247.2413282>, doi:10.1145/2413247.2413282.
- [12] Partha Kanuparth and Constantine Dovrolis. Diffprobe: Detecting isp service discrimination. In *IEEE Conference on Computer Communications (INFOCOM)*, San Diego, CA, USA, 2010.
- [13] Kevin Arceneaux, Alan S. Gerber, and Donald P. Green. A cautionary note on the use of matching to estimate causal effects: An empirical example comparing matching estimates to an experimental benchmark. *Sociological Methods & Research*, 39(2):256–282, 2010.
- [14] C. Dovrolis, D. Moore, and P. Ramanathan. Packet Dispersion Techniques and Capacity Estimation. *IEEE/ACM Transactions on Networking*, 12(6):963–977, Dec 2004.
- [15] Partha Kanuparth and Constantine Dovrolis. End-to-end detection of isp traffic shaping using active and passive methods. Technical report, Technical Report, Georgia Tech, 2011. <http://www.cc.gatech.edu/~partha/shaperprobe-TR.pdf>, 2011.
- [16] Natalie Giroux and Sudhakar Ganti. *Quality of Service in ATM Networks*. Prentice Hall PTR, 1999.

- [17] Rui Castro, Mark Coates, Gang Liang, Robert Nowak, and Bin Yu. Network tomography: recent developments. *Statistical science*, pages 499–517, 2004. URL: <http://projecteuclid.org/euclid.ss/1110999312>, doi:doi:10.1214/088342304000000422.
- [18] Earl Lawrence, George Michailidis, Vijay Nair, and Bowei Xi. Network tomography: A review and recent developments. *Ann Arbor*, 1001:48109–1107, 2006.
- [19] Yiyi Huang, Nick Feamster, and Renata Teixeira. Practical issues with using network tomography for fault diagnosis. *ACM SIGCOMM Computer Communication Review*, 38(5):53–58, 2008.
- [20] Zhiyong Zhang, Ovidiu Sebastian Mara, and Katerina Argyraki. Network neutrality inference. In *Proceedings of the ACM SIGCOMM Conference*, 2014. URL: http://infoscience.epfl.ch/record/186414/files/neutralityInference_1.pdf.
- [21] Systems Research Lab. Apology: Broadband network management [online]. URL: http://systems.cs.colorado.edu/mediawiki/index.php/Broadband_Network_Management [cited 2014/05/05].
- [22] Anonymous. Private communication. commercially confidential, 2008.
- [23] Fujitsu. Carrier software defined networking (sdn). Technical report, OfCom, March 2014.
- [24] Razvan Beuran. *Mesure de la qualité dans les réseaux informatiques*. PhD thesis, Bucharest, Polytechnic Inst. and St. Etienne U., 2004.
- [25] Chris J Vowden and Laura Lafave. Analysis of composed M/D/1/K networks. In *UKPEW'01: proceedings of 17th annual UK performance engineering workshop*, 2001.
- [26] Aleksandar Kuzmanovic and Edward W Knightly. Low-rate tcp-targeted denial of service attacks: the shrew vs. the mice and elephants. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 75–86. ACM, 2003.
- [27] Keith Winstein and Hari Balakrishnan. Tcp ex machina: Computer-generated congestion control. *SIGCOMM Comput. Commun. Rev.*, 43(4):123–134, August 2013. URL: <http://doi.acm.org/10.1145/2534169.2486020>, doi:10.1145/2534169.2486020.
- [28] Leonard Kleinrock. A conservation law for a wide class of queueing disciplines. *Naval Research Logistics Quarterly*, 12(2):181–192, 1965.
- [29] Frank Kelly. Notes on effective bandwidth. *Stochastic networks: theory and applications*, pages 141–168, 1996.
- [30] A Arulambalam, Xiaoqiang Chen, and N. Ansari. Allocating fair rates for available bit rate service in atm networks. *Communications Magazine, IEEE*, 34(11):92–100, Nov 1996. doi:10.1109/35.544198.
- [31] J.W. Roberts. A survey on statistical bandwidth sharing. *Computer Networks*, 45(3):319 – 332, 2004. In Memory of Olga Casals. URL: <http://www.sciencedirect.com/science/article/pii/S1389128604000544>, doi:http://dx.doi.org/10.1016/j.comnet.2004.03.010.
- [32] Cisco Tech Notes. Comparing traffic policing and traffic shaping for bandwidth limiting. *Document ID*, 19645.
- [33] William Lehr, Steven Bauer, Mikko Heikkinen, and David Clark. Assessing broadband reliability: Measurement and policy challenges. In *Research Conference on Communications, Information and Internet Policy*, Arlington, VA, 2011.
- [34] Steven Bauer, David Clark, and William Lehr. Powerboost. In *Proceedings of the 2nd ACM SIGCOMM workshop on Home networks*, pages 7–12. ACM, 2011.
- [35] Marcel Dischinger, Andreas Haeberlen, Krishna P Gummadi, and Stefan Saroiu. Characterizing residential broadband networks. In *Internet Measurement Conference*, pages 43–56, 2007.

- [36] Myles Hollander and Douglas Wolfe. *A.(1973). Nonparametric Statistical Methods*. John Wiley and Sons, New York, 1979.
- [37] Karthik Lakshminarayanan and Venkata N Padmanabhan. Some findings on the network performance of broadband hosts. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 45–50. ACM, 2003.
- [38] Guohan Lu, Yan Chen, Stefan Birrer, Fabián E Bustamante, Chi Yin Cheung, and Xing Li. End-to-end inference of router packet forwarding priority. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 1784–1792. IEEE, 2007.
- [39] Ratul Mahajan, Ming Zhang, Lindsey Poole, and Vivek S Pai. Uncovering performance differences among backbone isps with netdiff. In *NSDI*, pages 205–218, 2008.
- [40] Mukarram Bin Tariq, Murtaza Motiwala, and Nick Feamster. Nano: Network access neutrality observatory. 2008.
- [41] George Varghese. *Network Algorithmics: an interdisciplinary approach to designing fast networked devices*. Morgan Kaufmann, 2005.
- [42] Udi Weinsberg, Augustin Soule, and Laurent Massoulie. Inferring traffic shaping and policy parameters using end host measurements. In *INFOCOM, 2011 Proceedings IEEE*, pages 151–155. IEEE, 2011.
- [43] Marcel Dischinger, Alan Mislove, Andreas Haeberlen, and Krishna P Gummadi. Detecting bittorrent blocking. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pages 3–8. ACM, 2008.
- [44] EFF “Test Your ISP” Project. URL: <https://www.eff.org/testyourisp>.
- [45] Nikolaos Laoutaris and Pablo Rodriguez. Good things come to those who (can) wait. In *Proc. of ACM HotNets*. Citeseer, 2008.
- [46] Vuze: Bad ISPs [online]. URL: http://wiki.vuze.com/w/Bad_ISPs [cited 2014/05/05].
- [47] M-Lab [online]. URL: <http://www.measurementlab.net> [cited 2014/05/05].
- [48] The ICSI Netalyzer [online]. URL: <http://netalyzer.icsi.berkeley.edu/> [cited 2014/05/05].
- [49] John Markoff. ‘neutrality’ is new challenge for internet pioneer [online]. September 2006. URL: http://www.nytimes.com/2006/09/27/technology/circuits/27neut.html?_r=1&oref=slogin [cited 2014/05/02].
- [50] Brad Stone. Comcast: We’re delaying, not blocking, BitTorrent traffic [online]. October 2007. URL: http://bits.blogs.nytimes.com/2007/10/22/comcast-were-delaying-not-blocking-bittorrent-traffic/?_php=true&_type=blogs&_r=0 [cited 2014/05/02].
- [51] The Associated Press. F.T.C. Urges Caution on Net Neutrality [online]. June 2007. URL: <http://www.nytimes.com/2007/06/28/technology/28net.html>.
- [52] The Associated Press. F.C.C. Chairman Favors Penalty on Comcast [online]. July 2008. URL: <http://www.nytimes.com/2008/07/11/technology/11fcc.html> [cited 2014/05/02].
- [53] Vern Paxson, Andrew K Adams, and Matt Mathis. Experiences with nimi. In *Applications and the Internet (SAINT) Workshops, 2002. Proceedings. 2002 Symposium on*, pages 108–118. IEEE, 2002.
- [54] Planet Lab [online]. URL: <http://www.planet-lab.org/> [cited 2014/05/05].
- [55] Neil Spring, David Wetherall, and Tom Anderson. Scriptroute: a public internet measurement facility. In *Proceedings of the 4th conference on USENIX Symposium on Internet Technologies and Systems-Volume 4*, pages 17–17. USENIX Association, 2003.
- [56] Velocix (Alcatel-Lucent) [online]. URL: <http://www.velocix.com/> [cited 2014/05/05].
- [57] Vuze network status monitor. Technical report. URL: http://plugins.vuze.com/plugin_details.php?plugin=aznetmon [cited 2014/05/05].

- [58] Ying Zhang, Z Morley Mao, and Ming Zhang. Ascertaining the reality of network neutrality violation in backbone isps. In *Proc. of ACM HotNets-VII Workshop*, 2008.
- [59] David Andersen, Hari Balakrishnan, Frans Kaashoek, and Robert Morris. Resilient overlay networks. Master's thesis, 2001.
- [60] Robert Beverly, Steven Bauer, and Arthur Berger. The internet is not a big truck: toward quantifying network neutrality. In *Passive and Active Network Measurement*, pages 135–144. Springer, 2007.
- [61] Canadian radio-television and telecommunications commission 2008-11-20 - #: 8646-c12-200815400 - public notice 2008-19 - review of the internet traffic management practices of internet service providers [online]. November 2008. URL: http://crtc.gc.ca/PartVII/eng/2008/8646/c12_200815400.htm.
- [62] Yu-Chung Cheng, Urs Hölzle, Neal Cardwell, Stefan Savage, and Geoffrey M Voelker. Monkey see, monkey do: A tool for tcp tracing and replaying. In *USENIX Annual Technical Conference, General Track*, pages 87–98. Boston, MA, USA, 2004.
- [63] COMCAST. Attachment b: Comcast corporation description of planned network management practices to be deployed following the termination of current practices [online]. 2008. URL: http://downloads.comcast.net/docs/Attachment_B_Future_Practices.pdf.
- [64] Weidong Cui, Marcus Peinado, Karl Chen, Helen J Wang, and Luis Irun-Briz. Tupni: Automatic reverse engineering of input formats. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 391–402. ACM, 2008.
- [65] The DIMES Project [online]. URL: <http://www.netdimes.org/>.
- [66] Nicholas P Jewell. *Statistics for epidemiology*. CRC Press, 2004.
- [67] Keynote homepage [online]. URL: <http://www.keynote.com/> [cited 2014/05/05].
- [68] Diane Lambert and Chuanhai Liu. Adaptive thresholds: Monitoring streams of network counts. *Journal of the American Statistical Association*, 101(473):78–88, 2006.
- [69] Harsha V Madhyastha, Tomas Isdal, Michael Piatek, Colin Dixon, Thomas Anderson, Arvind Krishnamurthy, and Arun Venkataramani. iPlane: An information plane for distributed services. In *Proceedings of the 7th symposium on Operating systems design and implementation*, pages 367–380. USENIX Association, 2006.
- [70] Matt Mathis, John Heffner, Peter O'Neil, and Pete Siemsen. Pathdiag: automated tcp diagnosis. In *Passive and Active Network Measurement*, pages 152–161. Springer, 2008.
- [71] Nate Anderson. Cox ready to throttle P2P, non “time sensitive” traffic [online]. January 2009. URL: <http://arstechnica.com/tech-policy/2009/01/cox-opens-up-throttle-for-p2p-non-time-sensitive-traffic/> [cited 29/04/2014].
- [72] Judea Pearl. *Causality: models, reasoning and inference*, volume 29. Cambridge Univ Press, 2000.
- [73] Charles Reis, Steven D Gribble, Tadayoshi Kohno, and Nicholas C Weaver. Detecting in-flight page changes with web tripwires. In *NSDI*, volume 8, pages 31–44, 2008.
- [74] Joel Sommers, Paul Barford, Nick Duffield, and Amos Ron. Accurate and efficient sla compliance monitoring. *ACM SIGCOMM Computer Communication Review*, 37(4):109–120, 2007.
- [75] Mukarram Tariq, Amgad Zeitoun, Vytautas Valancius, Nick Feamster, and Mostafa Ammar. Answering what-if deployment and configuration questions with wise. In *ACM SIGCOMM Computer Communication Review*, volume 38, pages 99–110. ACM, 2008.
- [76] Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer, 2004.
- [77] Andy C Bavier, Mic Bowman, Brent N Chun, David E Culler, Scott Karlin, Steve Muir, Larry L Peterson, Timothy Roscoe, Tammo Spalink, and Mike Wawrzoniak. Operating

- systems support for planetary-scale network services. In *NSDI*, volume 4, pages 19–19, 2004.
- [78] TelecomTV One. Its back to ‘pipes’ and ‘free rides’: Internet neutrality under attack (again) [online]. June 2009. URL: http://www.telecomtv.com/comspace_newsDetail.aspx?n=45072&id=e9381817-0593-417a-8639-c4c53e2a2a10 [cited 2014 04 29].
- [79] BT heavily throttling BBC, all video [online]. June 2009. URL: <http://fastnetnews.com/dslprime/42-d/1758-bt-heavily-throttling-bbc-all-video> [cited 29/04/2014].
- [80] Internet 2 Performance tools [online]. URL: <http://www.internet2.edu/products-services/performance-monitoring/performance-tools/> [cited 29/04/2014].
- [81] Ian Clarke. A distributed decentralised information storage and retrieval system. Master’s thesis, University of Edinburgh, 1999.
- [82] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters, osdi04: Sixth symposium on operating system design and implementation, san francisco, ca, december, 2004. *S. Dill, R. Kumar, K. McCurley, S. Rajagopalan, D. Sivakumar, ad A. Tomkins, Self-similarity in the Web, Proc VLDB*, 2004.
- [83] ED FELTEN. Three flavors of net neutrality [online]. December 2008. URL: <https://freedom-to-tinker.com/blog/felten/three-flavors-net-neutrality/> [cited 29/04/2014].
- [84] cPacket Networks Inc. Complete Packet Inspection on a Chip [online]. URL: <http://www.cpacket.com/> [cited 2014/05/05].
- [85] Paul Francis, Sugih Jamin, Vern Paxson, Lixia Zhang, Daniel F Gryniewicz, and Yixin Jin. An architecture for a global internet host distance estimation service. In *INFOCOM’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 210–217. IEEE, 1999.
- [86] Lixin Gao. On inferring autonomous system relationships in the internet. *IEEE/ACM Transactions on Networking (ToN)*, 9(6):733–745, 2001.
- [87] Vikrant S Kaulgud. Ip quality of service: Theory and best practices, 2004.
- [88] Stavros G Kolliopoulos and Neal E Young. Approximation algorithms for covering/packing integer programs. *Journal of Computer and System Sciences*, 71(4):495–505, 2005.
- [89] Arbor Networks [online]. URL: <http://www.arbornetworks.com/> [cited 2014/05/05].
- [90] Ratul Mahajan, Neil Spring, David Wetherall, and Tom Anderson. Inferring link weights using end-to-end measurements. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 231–236. ACM, 2002.
- [91] Ratul Mahajan, Neil Spring, David Wetherall, and Thomas Anderson. User-level internet path diagnosis. In *ACM SIGOPS Operating Systems Review*, volume 37, pages 106–119. ACM, 2003.
- [92] Andrew W Moore and Denis Zuev. Internet traffic classification using bayesian analysis techniques. In *ACM SIGMETRICS Performance Evaluation Review*, volume 33, pages 50–60. ACM, 2005.
- [93] Vern Paxson, Jamshid Mahdavi, Andrew Adams, and Matt Mathis. An architecture for large scale internet measurement. *Communications Magazine, IEEE*, 36(8):48–54, 1998.
- [94] Larry Peterson, Tom Anderson, David Culler, and Timothy Roscoe. A blueprint for introducing disruptive technology into the internet. *ACM SIGCOMM Computer Communication Review*, 33(1):59–64, 2003.
- [95] Jerome H Saltzer, David P Reed, and David D Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems (TOCS)*, 2(4):277–288, 1984.

- [96] Joel Sommers and Paul Barford. An active measurement system for shared environments. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 303–314. ACM, 2007.
- [97] Neil Spring, Ratul Mahajan, David Wetherall, and Thomas Anderson. Measuring isp topologies with rocketfuel. *Networking, IEEE/ACM Transactions on*, 12(1):2–16, 2004.
- [98] Liz Gannes. At&t continues to adjust tos to limit 3g video [online]. April 2009. URL: <http://newteevee.com/2009/04/29/att-continues-to-adjust-tos-to-limit-3g-video>. [cited 2014/05/05].
- [99] Neil Spring, David Wetherall, and Thomas Anderson. Reverse engineering the internet. *ACM SIGCOMM Computer Communication Review*, 34(1):3–8, 2004.
- [100] Maurice Kendall, Alan Stuart, J Keith Ord, and A OHagan. Kendalls advanced theory of statistics, volume 1: Distribution theory. *Arnold, sixth edition edition*, 1994.
- [101] M Kendall, A Stuart, KJ Ord, and S Arnold. Kendalls advanced theory of statistics: Volume 2a—classical inference and and the linear model (kendalls library of statistics). *A Hodder Arnold Publication*,, 1999.
- [102] John W Tukey. Bias and confidence in not-quite large samples. In *Annals of Mathematical Statistics*, volume 29, pages 614–614. Institute Mathematical Statistics, 1958.
- [103] Charles V Wright, Fabian Monrose, and Gerald M Masson. On inferring application protocol behaviors in encrypted network traffic. *The Journal of Machine Learning Research*, 7:2745–2769, 2006.
- [104] Aditya Akella, Srinivasan Seshan, and Anees Shaikh. An empirical evaluation of wide-area internet bottlenecks. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 101–114. ACM, 2003.
- [105] Brice Augustin, Timur Friedman, and Renata Teixeira. Measuring load-balanced paths in the internet. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 149–160. ACM, 2007.
- [106] Brice Augustin, Xavier Cuvellier, Benjamin Orgogozo, Fabien Viger, Timur Friedman, Matthieu Latapy, Clémence Magnien, and Renata Teixeira. Avoiding traceroute anomalies with paris traceroute. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 153–158. ACM, 2006.
- [107] Ioannis C Avramopoulos and Jennifer Rexford. Stealth probing: Efficient data-plane security for ip routing. In *USENIX Annual Technical Conference, General Track*, pages 267–272, 2006.
- [108] Cisco. Configuring port to application mapping [online]. URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800ca7c8.html [cited 2014/05/05].
- [109] Marta Carbone and Luigi Rizzo. Dummynet revisited. *ACM SIGCOMM Computer Communication Review*, 40(2):12–20, 2010.
- [110] Augustin Soule, Kavé Salamatia, Nina Taft, Richard Emilion, and Konstantina Papagiannaki. Flow classification by histograms: or how to go on safari in the internet. *ACM SIGMETRICS Performance Evaluation Review*, 32(1):49–60, 2004.

A. ICT and network performance

A.1. Translocation

Distributed computation necessarily involves transferring information generated by one computational process to another, located elsewhere. We call this function ‘translocation’, and the set of components that performs it is ‘the network’. Instantaneous and completely loss-less translocation is physically impossible; thus all translocation experiences some ‘impairment’ relative to this ideal.

Translocating information as packets that share network resources permits a tremendous degree of flexibility in how computational processes interact, and allows resources to be used more efficiently compared to dedicated circuits¹. In packet-based networks, multiplexing is a real-time ‘game of chance’; because the state of the network when a packet is inserted is unknowable, exactly what will happen to each packet becomes uncertain. At each multiplexing point, the ‘game of chance’ is played out between packets of the multiplexed flows. The result of this game is that the onward translocation of each packet to the next element along the path may be delayed, or may not occur at all (the packet may be ‘lost’). This is a source of impairment that is statistical in nature.

The odds of this multiplexing ‘game’ are affected by several factors, of which load is one. In these ‘games’, when one packet is discarded, another is not. Similarly, when one is delayed more, another is delayed less - i.e. this is a zero-sum game in which quality impairment (loss and delay) is conserved.

A.1.1. Mutual interference in network traffic

There is a common misconception that the complexity of networks comes from their inter-connectivity - the fact that they can form an arbitrary ‘graph’². However, given the use of routing protocols that select particular paths through this connectivity graph, the particular path of network elements traversed by the packets in a given flow³ is essentially fixed. The translocation characteristics of the flow are affected only by the other flows that share a common network element on that path, so the complexity of the problem is bounded. The process of sharing resources between flows that follow a common path is called multiplexing. For any particular end-to-end flow, the network is effectively a tree of multiplexers, as illustrated in Figure A.1.

In Figure A.1a, the different coloured lines indicate potential valid routes. Black lines are potential routes that have been ‘pruned’ by the operation of routing algorithms. The lines coloured in red, green and blue represent traffic flowing from sources to sinks, passing through multiplexers (‘Mux’). In practice, any network endpoint functions as both a source and sink, but, for understanding network traffic, it is essential to separate these two roles.

If we now focus on the traffic flowing towards any one sink, for example that flowing to Sink a

¹This is similar to the familiar benefits of sharing individual computing elements between a number of processes. However, processor sharing is better understood than network resource sharing. This is partly because packets share many and varied network elements, and partly because the number of packets exchanged between processes tends to far exceed the number of processes in a computing node. Thus the sharing of network resources is complex, and predicting its consequences seemingly intractable.

²[http://en.wikipedia.org/wiki/Graph_\(mathematics\)](http://en.wikipedia.org/wiki/Graph_(mathematics))

³Where a flow is the sequence of packets between a particular source and sink.

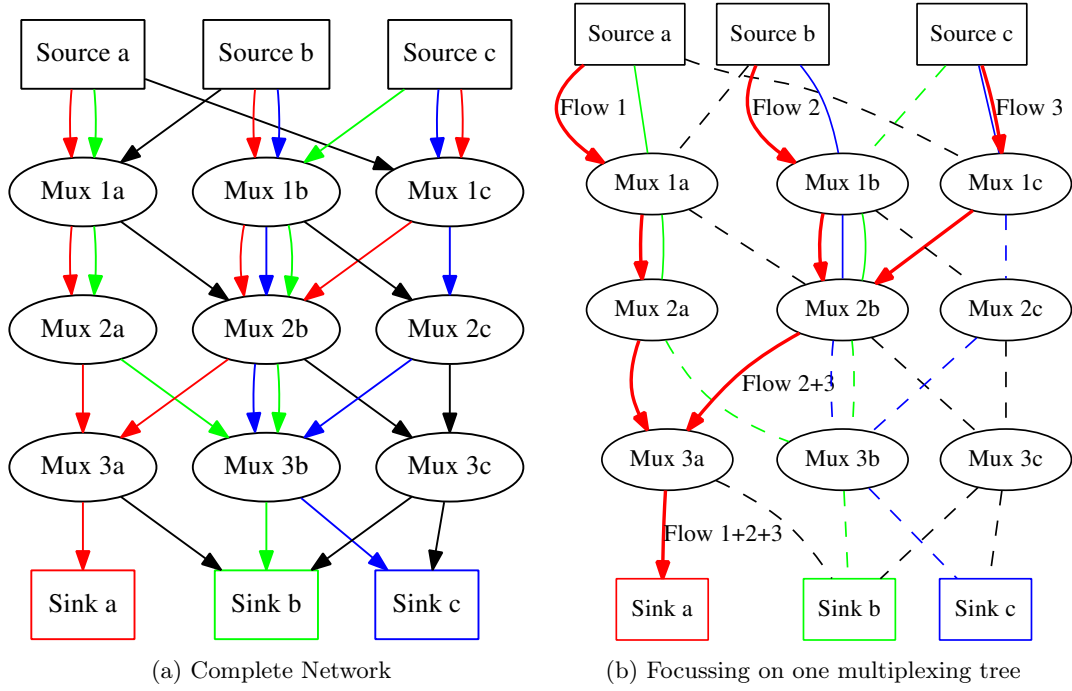


Figure A.1.: The network is a tree of multiplexors

(represented by the red lines in Figure A.1b), these flows share resources⁴ over portions of the path with other flows (represented by the solid green and blue lines). Note that it is only the common sub-paths that are sources of inter-stream impairment; the rest of the traffic in the network has no influence, as it is running over disjoint paths that do not share resources with the red flows (represented by dotted lines in the figure). Thus, when evaluating the impairment due to competition for resources (the statistical multiplexing) within any network, it is sufficient to consider the tree of multiplexors rooted at each sink.

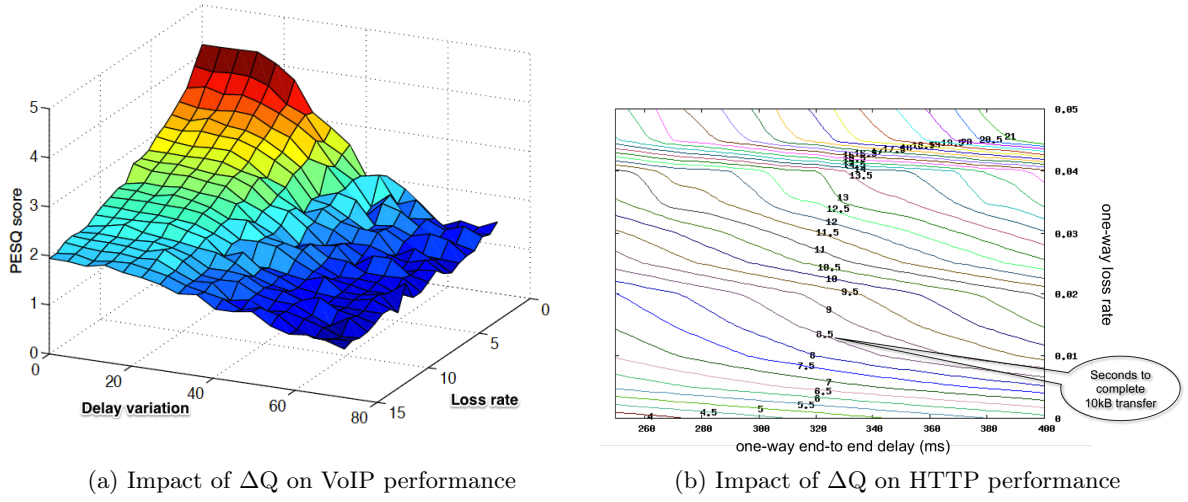
A.2. Network influence on application outcomes: ΔQ

Typical impairments that can affect an analogue telephone call (such as noise, distortion and echo) are familiar; for the telephone call to be fit-for-purpose, all of these must be sufficiently small. Analogously, we introduce a new term, called ‘quality attenuation’ and written ‘ ΔQ ’, which is a measure of the impairment of the translocation of a stream of packets when crossing a network. This impairment must be sufficiently bounded for an application to deliver fit-for-purpose outcomes⁵. For example, Figure A.2a (reproduced from [24]) shows the impact of delay variation and loss rate (both of which are aspects of ΔQ) on the audio quality of a G.711 VoIP call. Figure A.2b shows the impact of delay and loss rate on the 95th percentile time to complete a 10kB HTTP transfer, such as a small web page.

ΔQ captures the effects of the network’s structure, together with the the impairment due to statistical multiplexing (as discussed in §A.2.2 below). Thus ΔQ is an inherently statistical measure that can be thought of as the probability distribution of what might happen to a packet transmitted at a particular moment from source A to destination B , or the statistical properties of a stream of such packets.

⁴For example, the finite capacity to transmit data from each Mux to the next, and the finite capacity to buffer data for transmission at each egress point from a Mux.

⁵Just as a telephone call might fail for reasons that are beyond the control of the telephone company (such as excessive background noise or a broken handset), applications may fail to deliver fit-for-purpose outcomes for reasons that are beyond the control of the network (e.g. lack of local memory or insufficient computing capacity). Such considerations are out of scope here.

Figure A.2.: Impact of ΔQ on application performance

A.2.1. Application performance depends only on ΔQ

Applications depend on information to complete computations. To provide appropriately timely outcomes, delivery of this information needs to be done in a timely and correctly sequenced manner. If information takes too long to arrive (and/or too much of it is missing⁶) then the computations cannot proceed, and the application fails to deliver the requested service or to deliver an acceptable performance of that service.

Different components of a distributed application (e.g. a client and a server) exchange information as streams of packets. If those packets were all delivered instantaneously (i.e. if there were no impairment in the translocation), and the computational components performed correctly, the application would work. However, as discussed above, sending packets over distances using shared resources *inevitably* means there will be some delay and occasionally packets may be lost - this is ΔQ . Whether the application still delivers fit-for-purpose outcomes depends entirely on the extent of the quality impairment (the magnitude of ΔQ), and the application's sensitivity to it. The layering of network protocols isolates the application from any other aspect of the packet transport. This is such an important point it is worth repeating: the great achievement of network and protocol design has been to completely hide all the complexities of transmission over different media, routing decisions, fragmentation and so forth, and leave the application with only one thing to worry about with respect to the network: ΔQ .

'Bandwidth required' is a characteristic of the application load. If many of the packets the application offers are discarded, users would typically say that the 'available bandwidth' is too low; however, from the perspective of the application, the immediate problem is that ΔQ is too large. Indeed such packet loss might well occur for reasons other than the capacity limitation of the transmission links. If it is delay (rather than loss) that is too large, this may not be because of constraints of capacity, but rather of schedulability⁷ - i.e. issues of instantaneous, rather than average, loading⁸.

A.2.2. How ΔQ accrues across the network

Network structure (including the types, lengths and speeds of network links) affects ΔQ . To illustrate this, consider Figure A.3, which focuses on the path from Source_b → Sink_a from

⁶It may be thought that data 'corruption' could also occur, but the underlying data transport mechanisms have checksums that cause any such corruption to be treated as loss. Even though a data packet may be lost, the protocols recover (typically through retransmission, where needed), transforming such loss into delay.

⁷Where schedulability is the ability to sequence the instantaneous demand to meet requirements.

⁸Loss can also be caused by schedulability constraints, especially where applications produce large bursts of packets.

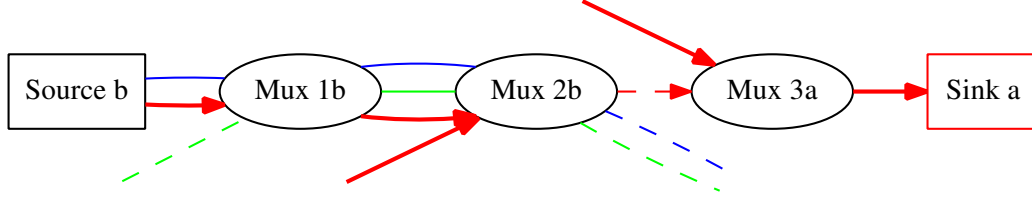


Figure A.3.: An end-to-end path through a network (from A.1b)

Figure A.1b. The overall end-to-end ΔQ , is the ‘sum’ of the ΔQ associated with each path⁹, i.e.:

$$\Delta Q^{\text{Source}_b \rightarrow \text{Sink}_a} = \Delta Q^{\text{Source}_b \rightarrow \text{Mux}_{1b}} \oplus \Delta Q^{\text{Mux}_{1b} \rightarrow \text{Mux}_{2b}} \oplus \dots \oplus \Delta Q^{\text{Mux}_{3a} \rightarrow \text{Sink}_a}$$

The overall ΔQ of flows following this path is dependent on several aspects, which can be split into two broad categories: *structural* and *variable*. Structural ΔQ captures properties such as the geographical distribution of the network elements (denoted $\Delta Q_{|G}$) and the extent to which bigger packets take longer to be transmitted¹⁰ (denoted $\Delta Q_{|S}$).

Figure A.4 illustrates the process of extracting ΔQ and its components from raw point-to-point delay data. If one measures delays for packets with a range of sizes and then plots these delays by packet size, a structure emerges. Structural components of ΔQ can be extracted, the remainder is the variable component.

Like the overall ΔQ , the individual elements can also be combined:

$$\begin{aligned} \Delta Q_{|G}^{\text{Source}_a \rightarrow \text{Sink}_b} &= \Delta Q_{|G}^{\text{Source}_a \rightarrow \text{Mux}_{1b}} \oplus \Delta Q_{|G}^{\text{Mux}_{1b} \rightarrow \text{Mux}_{2b}} \oplus \dots \oplus \Delta Q_{|G}^{\text{Mux}_{3a} \rightarrow \text{Sink}_a} \\ \Delta Q_{|S}^{\text{Source}_a \rightarrow \text{Sink}_b} &= \Delta Q_{|S}^{\text{Source}_a \rightarrow \text{Mux}_{1b}} \oplus \Delta Q_{|S}^{\text{Mux}_{1b} \rightarrow \text{Mux}_{2b}} \oplus \dots \oplus \Delta Q_{|S}^{\text{Mux}_{3a} \rightarrow \text{Sink}_a} \\ \Delta Q_{|G,S}^{\text{Source}_a \rightarrow \text{Sink}_b} &= \Delta Q_{|G,S}^{\text{Source}_a \rightarrow \text{Mux}_{1b}} \oplus \Delta Q_{|G,S}^{\text{Mux}_{1b} \rightarrow \text{Mux}_{2b}} \oplus \dots \oplus \Delta Q_{|G,S}^{\text{Mux}_{3a} \rightarrow \text{Sink}_a} \end{aligned}$$

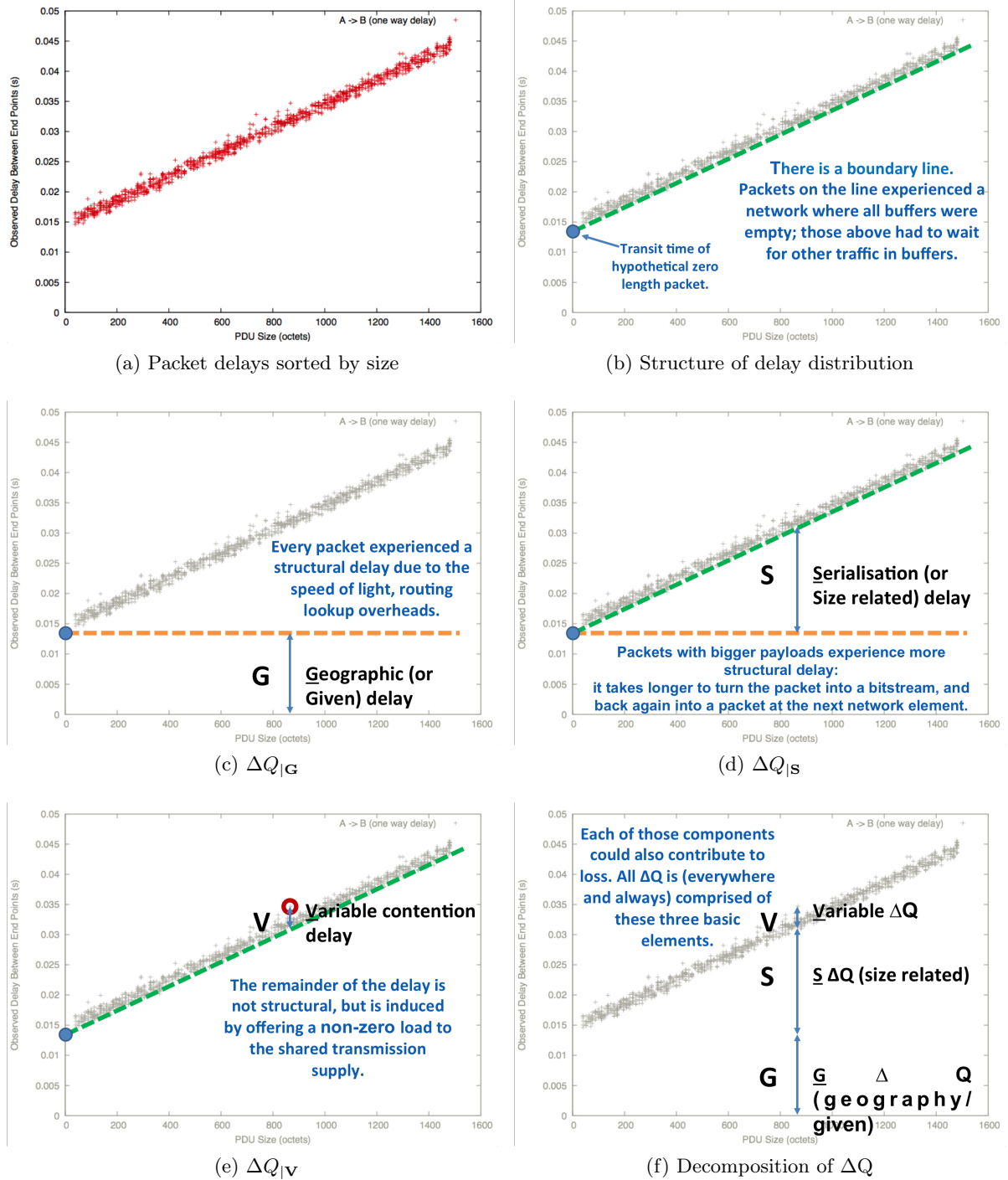
In addition to the $\Delta Q_{|G,S}$ (structural ΔQ) along a path, there is a variable component, denoted $\Delta Q_{|V}$. This component captures the effects of multiplexing resources (such as link capacity in wired networks, or local spectrum capacity in wireless access). In Figure A.3, multiplexing will occur at each of the nodes (Source_b, Mux_{1b}, Mux_{2b} and Mux_{3a}). This is where $\Delta Q_{|V}$ accrues, as a function of the load offered there, i.e. the set of packets requiring to be forwarded at a particular moment. Note that $\Delta Q_{|V}$ is related to the *total* offered load¹¹ and is a direct and unavoidable consequence of packet-based statistical multiplexing. In exchange for the efficiency gained by not dedicating resources to individual data flows (as circuit-based networking does), we must accept the possibility that more packets will arrive than can immediately be forwarded, so some must wait (or be lost). ΔQ is conserved (as discussed above). So, whatever mechanism is used to affect the $\Delta Q_{|V}$ of any flow at any point (say the blue flow as it egresses Mux_{1b} in Figure A.3), the *best* that can be achieved is that the overall $\Delta Q_{|V}$ (without regard to any particular flow) is not increased. The constraint that the sum of the $\Delta Q_{|V}$ for individual streams cannot be less than that for the aggregate flow is expressed in the equation:

$$\sum_{c \in \{\text{red, green, blue}\}} \Delta Q_{|V}^{\text{Mux}_{1b} \xrightarrow{c} \text{Mux}_{2b}} \geq \Delta Q_{|V}^{\text{Mux}_{1b} \rightarrow \text{Mux}_{2b}}$$

⁹We treat ‘ ΔQ ’ as a plural noun.

¹⁰This is more than the ‘speed’ of the network link, it incorporates the influence of transmission technology on the time taken to service packets of varying length.

¹¹Where the total offered load is the combined load of all flows passing through the shared node.

Figure A.4.: ΔQ and its components

The way in which the $\Delta Q_{|V}$ is distributed between different flows at a particular multiplexing point is the result of the queuing and scheduling mechanisms operating there. However, any such mechanisms are inherently subject to the above conservation constraint (this is a generalisation of the work in [25]). Thus, the overall $\Delta Q_{|V}$ that the red traffic experiences is:

$$\Delta Q_{|V}^{\text{Source}_a \xrightarrow{\text{red}} \text{Sink}_b} = \Delta Q_{|V}^{\text{Source}_a \xrightarrow{\text{red}} \text{Mux}_{1b}} \oplus \Delta Q_{|V}^{\text{Mux}_{1b} \xrightarrow{\text{red}} \text{Mux}_{2b}} \oplus \dots \oplus \Delta Q_{|V}^{\text{Mux}_{3a} \xrightarrow{\text{red}} \text{Sink}_a}$$

For a given end-user communicating with a given endpoint, the main network factor that influences the variation in their experience is the $\Delta Q_{|V}$ (in both directions) of the translocation along the path connecting them.

Each user experience of a particular application is affected by the presence of other resource-sharing traffic. This traffic acts as ‘pollution’ that, from the user’s point of view, potentially degrades their application’s performance. TM is one approach to addressing this problem, but it is again subject to the conservation law above - any ‘pollution’ can be ‘traded’ but never eliminated.

Trading occurs whenever resources are shared, whether this is explicitly acknowledged or not. In networks, such trading occurs at every network element and at every network port (i.e. every multiplexing point). If no action is taken, these trades are determined implicitly by the various mechanisms operating in each element, and are of an unstructured and disordered nature. They do not intrinsically provide fairness nor do they explicitly support the policy or aims of the network operators or designers. Managing this may appear to be an overwhelmingly complex problem¹², but mathematically-based approaches (such as the one outlined here) can contain that complexity and clarify the constraints on what is achievable. These can be used to inform a higher-level discussion of desirable outcomes, and can also enable the identification of any related hazards to the delivery of fit-for-purpose outcomes.

A.3. Summary

In this appendix, we have introduced the notion of translocation - the end-to-end transport of information units between computational processes. We have outlined the notion of ΔQ , a statistical measure that captures the performance of such translocation, in a way that is independent of the underlying network technology¹³. As a measure, ΔQ :

- accrues along the end-to-end path of each data flow;
- expresses the impact of the structural aspects of the network on translocation;
- can be directly related to the delivered QoE of applications;
- is conserved, in that having been ‘created’ it can not be ‘destroyed’ - although some aspects can be differentially shared;
- depends on load, thus incorporating the way in which ‘bandwidth’ is typically used to express requirements;
- captures the variability of translocation due to the statistical sharing of resources at multiplexing points.

We have shown how the apparent complexity of analysing interactions between multiple packet flows can be mitigated by focusing on the tree of multiplexors rooted at a particular sink. By combining this with the composability of ΔQ , the analysis of network performance interactions becomes tractable.

¹²From an ontological point of view, these systems are completely predictable (that is they would produce the same results given precisely the same starting conditions and inputs over time). The overall outcome can be highly dependent on seemingly minor aspects of the inputs; thus it is in their epistemology that the complexity lies.

¹³This holds whether the underlying network is wired, wireless, copper, fibre, 2G/3G/4G, satellite, etc..

B. Traffic Management methods and their impact on ΔQ

As discussed in § 1.2.1 on page 12, multiplexing in ICT systems is the statistical sharing of common resources, such as point-to-point transmission capacity. Buffering is needed to allow for arrivals to occur when the resource is busy. This creates contention for two things: the ability to be admitted into the buffer (ingress), and the ability to leave the buffer (egress). Whether the first is achieved determines loss, and the time taken to achieve the second determines delay; together these represent the mechanisms that create $\Delta Q_{|V}$, the variable component of quality attenuation¹. At every multiplexing point in a network a ‘game’ is being played out between different streams of packets. The term ‘Traffic Management’ is usually associated with the configurations of multiplexing points, as these determine the ‘odds’ of this game.

B.1. Packet-based multiplexing and $\Delta Q_{|V}$

In packet-based networks, each packet has a header that contains the information necessary to direct it towards its destination on a hop-by-hop basis (this is the function of routing). Each point along this hop-by-hop path acts as a multiplexor, processing complete packets². As packets can arrive when the ongoing transmission path is busy, buffering is needed³.

While the competition for network resources is typically viewed in terms of ‘bandwidth’, it is more useful to regard multiplexing as two competitions between packets; one to get into the buffer (ingress); and another to get out of it (egress). Queueing and scheduling techniques differ solely in their ingress and egress actions with respect to this buffer⁴. Viewing the operation in this way, it is clear that:

1. The failure to be admitted to the buffer, as part of the ingress behaviour, is a source of packet loss⁵;
2. The instantaneous occupancy of the buffer represents *the total accrued delay*; this total delay is independent of the order in which the packets are eventually serviced⁶;
3. The order in which packets are chosen, the egress behaviour, determines the delay that the individual packets experience.

In point 2 above, there is an assumption that the egress behaviour is work-conserving - i.e. packets will be sent whenever the buffer is non-empty. Most queueing and scheduling

¹While there are other ways in which the overall ΔQ can accrue, for example due to electrical noise in transmission and associated recovery, these are not the dominant factors for most broadband connections.

²I.e. when a packet is sent, a complete packet is sent; when a packet is discarded, a complete packet is discarded. While packet fragmentation is possible, for the purposes of this report it is an advanced topic.

³For the sake of completeness, we note that this is where TDM-based transmission fundamentally differs from PBSM. TDM’s design eliminates the need for buffering at intermediate routing points. Between entering and leaving a pure TDM network, packets will experience ‘perfect’ $\Delta Q_{|V}$, zero delay and no loss from multiplexing.

⁴We note that equipment may allocate separate buffer capacity to different purposes. This is an operational refinement that does not affect the total buffering being used. It is the total buffer use that we will consider here.

⁵While there are techniques in which existing packets can be ‘pushed-out’ by other arriving packets, they do not represent a fundamental change to the nature of the problem and so we will not consider them in this report.

⁶More accurately, the instantaneous occupancy represents an absolute lower bound on the overall delay.

techniques work this way, with the exception of rate limiting (§B.4.4), whose specific aim is to control the egress rate from the buffer⁷.

When examining the effects of a queueing and scheduling mechanism, there are two complementary viewpoints. The first is a component-centric view, considering the total $\Delta Q|_V$ being created by the component's operation; the second is a translocation-centric view, which focuses on the $\Delta Q|_V$ that the packets for an individual application (or end-user) experience. Application outcomes are not generally determined by the fate of any one particular packet, so the $\Delta Q|_V$ of interest is the *probability distribution* of the individual packet experiences. This includes the two extremes of \emptyset (perfect transmission without delay) and \perp (loss). The fine-grain behaviour of network protocols is sensitive to the pattern of the end-to-end $\Delta Q|_V$. Taking TCP/IP as a case in point, timeouts are calculated on recent round-trip times⁸, and the pattern of loss drives congestion avoidance.

B.1.1. FIFO

The most common queueing and scheduling approach, and hence the most common ‘traffic management’ technique, is a FIFO (first-in first-out) queue⁹.

B.1.1.1. Ingress behaviour

On arrival, a packet is admitted to the buffer if there a free slot. Packets arriving (from all sources) whilst the buffer is fully occupied are discarded (this is referred to as ‘tail-drop’). It should be noted that the destination system receives no direct indication of this loss, but must infer it from the non-arrival of an expected packet¹⁰.

B.1.1.2. Egress behaviour

Packets are chosen from the buffer in the order that they were admitted¹¹. The delay that each packet will experience is made of two components. The first is the time taken for the transmission link to become idle, i.e. to finish processing the packet currently being sent, if any. The second is the time for the packet in question to be chosen for transmission (the queueing time).

B.1.1.3. Discussion

When a packet arrives at a FIFO where both the buffer is empty and the transmission resource is idle, it will be forwarded immediately¹² without being discarded¹³. In this case there is no contention for the common resource, and the experienced $\Delta Q|_V$ is \emptyset - ‘perfection’, no delay or loss.

When a packet arrives at a FIFO whose buffer is full (which implies the transmission resource is non-idle) it will be discarded and never arrive at its intended destination¹⁴. This corresponds

⁷The use of buffers to de-jitter streams, such as in VoIP, has a similar non-work-conserving property.

⁸These RTTs are, in turn, dependent on the bi-directional $\Delta Q|_V^{A \leftrightarrow Z}$.

⁹This is also known as FCFS (first-come first-served).

¹⁰This is the role of sequence numbers and timeouts in protocols.

¹¹This is typically done by choosing the packet at the head of a queue. The queue in question is formed by placing each admitted packet at the back of the queue as they arrive during ingress processing.

¹²We are assuming that the FIFO is work-conserving, unless stated otherwise.

¹³From the point of view of an external observer, the leading edge of the packet will commence transmission at the time the trailing edge arrives ($\Delta Q|_S$ would come into play if, for example, the time was measured between arrival and departure of leading edges). Any difference between the end of the packet arriving and the packet being transmitted (such as time required to look up routing tables) would be a contributor to the $\Delta Q|_G$.

¹⁴This may seem a spurious distinction, however the difference is important. The non-arrival at the receiver within a time period of interest is an externally observable phenomenon, whereas the packet discard is an

to a ΔQ_V of \perp (mathematically called ‘bottom’). When a packet arrives at a FIFO whose buffer is not full but whose transmission link is not idle, it will experience a delay determined by the current state of the buffer. This delay is dependent on both the length of the queue on arrival and the residual service time for the packet being transmitted.

As discards occur when the buffer is full, it is interesting to ask the following questions:

1. Given that a buffer is full, how long can it remain full;
2. How many packets can arrive while the buffer is full?

The buffer remains full until the packet in transmission has been completely sent. The time taken to send this packet is dependent on the size of the packet (bounded by the technology and its maximum packet size) and the transmission rate of the egress link. For example, a 2Mbps ADSL connection¹⁵ takes 6.1ms to transmit a 1500 byte IP packet. In the same amount of time a 1Gbps Ethernet connection can transmit 495 such packets¹⁶.

The number of packets that can arrive in any period of time is dependent on the aggregate ingress rate to the device. When the multiplexing point is at the egress of a switch, the maximum ingress rate would be the sum of the individual ingress link rates. Thus, if the individual rates are the same¹⁷, the maximum number of packets that can arrive while the buffer is full is given by the number of ports on the switch.

Under the assumption of ‘random’ traffic arrivals, at low loading there is a high probability that a packet arriving will experience a ΔQ_V of \emptyset , and a very low probability of experiencing \perp . Hence traversing this particular hop is highly likely to increase only the overall end-to-end ΔQ by its contribution to $\Delta Q_{G,S}$. For this to hold, ‘randomness’, i.e. the independence of packet arrivals, is essential. Even in networks with very low average loads¹⁸ correlated loading patterns can generate significant ΔQ_V . These correlation issues are discussed in §B.2.

When the ingress rate approaches or equals the egress rate and the load is uncorrelated, FIFO has the interesting property that all possible states¹⁹ of the buffer become equally likely²⁰. For example, at 100% offered load, a FIFO with 100 buffers²¹ would deliver a link utilisation of 99%, a loss rate of 1%, and a uniform distribution of all the possible delays between 0 and 99 packet service times.

B.1.1.4. Fairness with respect to ΔQ

In data networks, and ICT in general, resource usage is often ‘rivalrous’²². The instantaneous state of a buffer can be seen as recording the recent history of that rivalry²³.

FIFO is often viewed as a ‘pure’ mechanism that treats traffic ‘fairly’. This sense of fairness may have arisen from a particular mathematical formulation of FIFO queueing²⁴. In practice, the distribution of ΔQ_V between competing translocation streams can be substantially biased by their individual arrival patterns²⁵. The authors have had experience of large network

internal event and thus not necessarily observable. One can be measured by an external third party, the other cannot.

¹⁵That is, one that would sync at around 2,208 kbps and transmit up to 5,208 ATM cells per second.

¹⁶A 1Gbps ethernet connection can carry 81,274 maximum ethernet frames per second - <http://goo.gl/xPY5g2>

¹⁷Here, the “individual rates” include those of the egress and all ingresses.

¹⁸This could be measured by, for example, link utilisation over five minute periods.

¹⁹These states would include \emptyset , \perp , and all values of delay in between.

²⁰Thus, the system is at maximum entropy.

²¹That is, one with 1 buffer for the packet in service and 99 queueing slots.

²²This means that use by one party prevents use by another. [http://en.wikipedia.org/wiki/Rivalry_\(economics\)](http://en.wikipedia.org/wiki/Rivalry_(economics))

²³Noting that the ‘memory’ of that history is wiped clean every time the buffer becomes empty.

²⁴There is a circumstance in which the arriving streams will experience the ‘same’ ΔQ_V , i.e. they will experience the same distribution of delay and the same rate of loss. This occurs when the service pattern is Markovian and all traffic sources are Poisson processes - i.e. the overall aggregate arrivals are Markovian.

²⁵This is particularly true for the distribution of loss, a phenomenon that has been exploited in the design of low rate denial-of-service attacks [26].

providers encountering issues stemming from this when increasing capacity in core parts of their systems²⁶.

The ΔQ_V of a single network element is not the only contributory factor to the overall end-to-end ΔQ , even where this is the only network element at which there is contention. The other aspects of ΔQ , the difference in ΔQ_G and ΔQ_S between two end-points, can substantially influence the delivered outcome of the same application at different locations²⁷. Thus the key question regarding fairness is: *with respect to what metric?* Fair distribution of ΔQ_V at a single contention point does not assure overall fairness in outcome, and may even hinder such a goal.

B.2. Load correlation, elastic protocols and Predictable Region of Operation (PRO)

In this report we view traffic management as the choice and configuration of queueing and scheduling within network elements, combined with their order and location²⁸.

As ΔQ_V is conserved, traffic management can differentially share it (see §B.3.1) and/or change in which network element it occurs (this is discussed in §B.3.2). Even in the case of the finite FIFO discussed above, there is a choice of how many buffers to configure; this biases the trade between delay and loss (as mentioned in §B.3.1).

Multiplexed resources are ones that match demand and supply over some timescale. In this case, the demand is the arrival (ingress) pattern and the supply is the departure (egress) pattern from the buffer for onward transmission²⁹. The instantaneous occupancy of the buffer is influenced by both the loading factor (the ratio of arrival rate to departure rate) and any correlation in the arrival pattern³⁰. For a given loading factor, the correlation between arrivals will have a substantial effect. In the Internet, a significant cause of such correlation is the operation of protocols that are ‘elastic’ (i.e. they endeavour to adapt their offered load to the apparent capacity constraints on the end-to-end path). TCP/IP is the most widespread example. Different choices of protocol behaviour at end-points have an influence on the delivered quality attenuation³¹ [27].

Correlated load causes ΔQ_V to vary, as shown occurring between two ISPs within the UK in Figure B.1. The issue for end-to-end service delivery is that excessive ΔQ_V can cause a network service to leave its *Predictable Region of Operation* (PRO). This arms the hazard that it will not perform ‘correctly’. The consequences of the hazard maturing are service dependent. For a video-on-demand service, it might mean a video artefact on the screen or a ‘buffering pause’. For an integral system service (such as routing updates or keep-alives on a L2TP tunnel), the consequence might be that all the connections between an ISP and its

²⁶When capacity was increased, longer and more dense back-to-back packet sequences formed. These sequences then generated burst loss in a downstream FIFO, with the overall effect of reducing the delivered QoE for some applications.

²⁷For example, the rate at which TCP/IP increases its window is a function of the overall round trip time ($\Delta Q_{G,S,V}^{A \leftrightarrow Z}$). This TCP/IP performance property has a great impact on the ‘time-to-first-frame’, which is an important QoE metric in video delivery.

²⁸We are focusing on those factors that affect a data translocation service between defined boundaries.

²⁹We are going to assume that the onward transmission is not, itself, a dynamically multiplexed resource (as would be the case if the transmission was being carried as an MPLS circuit or some other statistically multiplexed transmission such as LTE). This does not affect the general argument, and that situation still remains amenable to analysis, but full explanation is beyond the scope of this report.

³⁰A general discussion of the causes and effects of correlation is beyond the scope of this report. Interested readers can find more material in works on Large Deviations Theory (http://en.wikipedia.org/wiki/Large_deviations_theory) and texts on teletraffic engineering (http://en.wikipedia.org/wiki/Teletraffic_engineering). Correlations do not occur at the network translocation level only; correlation of load also occurs in the demand for service.

³¹Many approaches have been taken within the framework of TCP, where the key concern is the “avoidance of congestion”, not the delivery of consistent performance. See http://en.wikipedia.org/wiki/TCP_congestion-avoidance_algorithm.

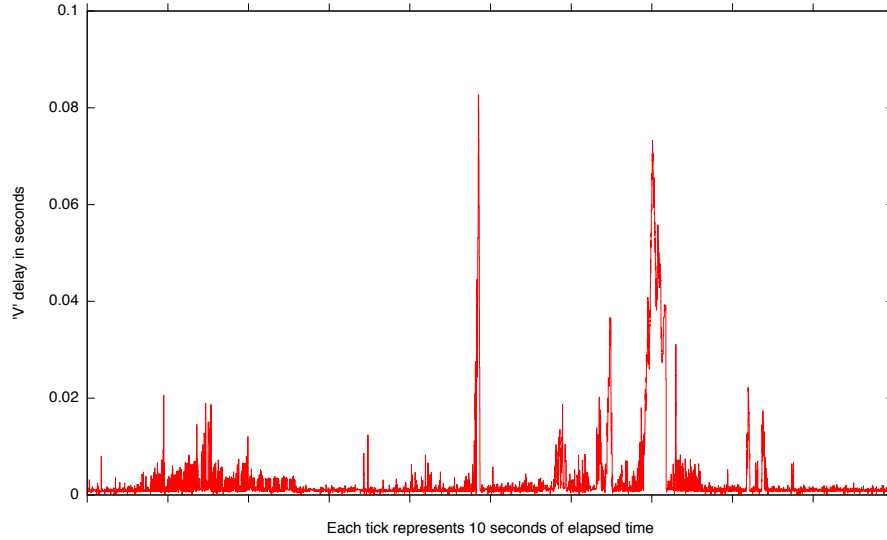


Figure B.1.: Example of one way delay between two points connected to UK internet

The figure shows a measure of the combined $\Delta Q_{|V}$ over time between a network element within ISP_a's core network and a network element within ISP_b's core network, across a UK internet exchange. The data rate applied was less than 3Mbps. There were no reported errors or performance issues along the path over the measurement period.

customers are dropped. This potential for operational ‘catastrophe’ is a key driver for traffic management³².

This risk of catastrophe is a consequence of the coupling of system stability with operational activity. It results from combining control plane and data plane traffic, a practice fundamental to the internet design philosophy. This $\Delta Q_{|V}$ -related issue, and the associated performance hazards, is inherent in the current use of PBSM. The fundamental distinction is between data bearers for which $\Delta Q_{|V}$ is \emptyset (e.g. PDH, SDH³³) and those for which it is not (e.g. MPLS, Carrier Ethernet³⁴).

Where (and hence within which management domains) quality attenuation accrues has changed over time due to the commercial evolution of large-scale broadband. This means that inter-user effects have become possible (as described in §A.1.1) and the PBSM supply chain can now influence how any resulting $\Delta Q_{|V}$ is distributed. As traditional telcos have taken on the delivery of broadband using PBSM, some control over the distribution of $\Delta Q_{|V}$ has left the telcos’ customers’ (i.e. ISPs) hands³⁵. This has two consequences:

1. The customer sees a $\Delta Q_{|V}$ that is no longer in direct relationship with their own pattern of load. In particular, a level of control over the PRO of their applications of interest has been removed;
2. The PBSM network operator has taken on the inter-end-user $\Delta Q_{|V}$ hazard, typically with little or no associated contractual risk. In particular, the hazard of $\Delta Q_{|V}$ causing the end-user’s application to leave its PRO is outside their contractual scope³⁶.

³²This is likely to become more important due to SDN and other developments, as discussed in a recent Ofcom report[23]. In section 4.6.3 (p49) Ofcom touches on issues of emergent fairness in traffic management.

³³In fact, this could be any resource where there is strong isolation between users - namely each user’s traffic patterns and usage don’t affect the $\Delta Q_{|V}$ for other users of the same resource. Examples of this are: different light wavelengths within the same fibre, unshared point-to-point wireless, and the use of SDH/TDM from end-to-end.

³⁴This is true even when such resources are allocated to peak.

³⁵This situation is in contrast to the days of dial-up modems, when all of the contention for resources occurred in the end-users’ premises or within the ISP’s own network.

³⁶SLAs are typically about long term (e.g. monthly) averages and ΔQ is about instantaneous properties. A

The current nature of the management and administrative domains in the UK, and their traffic management influences, is discussed in Appendix C.

B.3. Trading space available for traffic management

It is self-evident that if a packet is delayed whilst traversing a network it cannot be ‘undelayed’. Similarly, if a packet is discarded (lost) it cannot be ‘un-lost’³⁷. ΔQ is ‘conserved’³⁸, i.e. it can only increase. It cannot be ‘undone’; at best it can be differentially shared. The individual components (ΔQ_V , ΔQ_G and ΔQ_S) are also conserved in the same way. When considering TM, we focus on the ΔQ_V component.

At any point in time, the contents of a network element’s buffer would take a particular time to empty. This would be independent of the order in which the packets were serviced (i.e. the delay is conserved). The fact that the overall delay in a queuing system is independent of the choice of scheduling algorithm has been well known since the mid-1960’s [28]. It is of interest to note that this analysis assumed an infinite buffer - in such a case delays would then be unbounded. With a finite buffer, the overall delay is always bounded; however this bounding of delay is at the cost of sometimes discarding packets³⁹.

B.3.1. Overall delay and loss trading

The fact that quality attenuation is conserved has profound consequences for PBSM systems, influencing not only their design and deployment but also their underlying cost structures [29]. Traffic management can be used to ‘trade’ within the overall conservation constraints. This trading process can be viewed from two different perspectives: one focusing on the accrual of ΔQ_V at a component; one focusing on the effects on the overall translocation for a specific flow.

B.3.1.1. Component-centric view

Given that a finite buffer must discard some packets whenever its instantaneous load is too high, increasing its size will decrease the rate of loss (at the cost of increasing the maximum total delay). Similarly, if the experienced delay is deemed too high (for a given arrival pattern), reducing the number of buffers⁴⁰ will reduce the overall delay, with increased loss⁴¹. In data networks such trades may occur many times along an end-to-end path, at every multiplexing point (in particular, every switch and router), so the configuration of these network elements influences the resultant $\Delta Q_V^{A \leftrightarrow Z}$.

A way of reducing the overall ΔQ_V at a network element is to lower its loading factor (the ratio of the arrival rate to the departure rate). This can be done either by reducing the offered load or by increasing the service capacity. The latter is the common industry practice of “use more bandwidth” or “apply generous dimensioning”. This can be cost-effective; however its effectiveness is predicated on certain assumptions:

1. That arrivals are independent and ‘random’. This assumption is fragile for the reasons discussed in §B.2. The operation of elastic protocols means that increasing capacity does not generate as much performance headroom as might be expected.
2. That the increased capacity improves the statistical multiplexing gain, i.e. increases the number of active load sources required to saturate the constrained resource. The

Telco meeting an SLA does not mean that an application of interest will remain within its PRO.

³⁷The information in a packet can be resent, but this generates a new packet.

³⁸ ΔQ is thus similar to the concept of entropy in thermodynamics.

³⁹As loss is also quality attenuation, the overall ΔQ is still conserved.

⁴⁰Alternatively, packets already queued may be discarded.

⁴¹Such a trading space is a common property of all statistically shared resources.

market-driven trend to increase capacity in the last mile (narrowband \rightarrow broadband \rightarrow superfast) has reduced the number of active end-points required to saturate network resources along the path. The corollary is that the ability of one user to affect the QoE of neighbouring⁴² users has increased.

These factors have lead to a reduction in the effectiveness of capacity increases to maintain customer experience. In the absence of any economic incentive to temper the volume and pattern of demand over the short timescales on which QoE is most affected, an increasing focus on traffic management has emerged as an alternative solution.

B.3.1.2. Translocation-centric view

The telecommunications supply chain tends to take a component-centric view, e.g. upgrade planning tends to be done on the basis of how busy or ‘hot’ individual network elements are⁴³. However, the overall $\Delta Q|_V$ at a multiplexing point is determined by a combination of the total buffering, the ingress pattern and the egress rate. This is a more complex relationship than can be captured by, for instance, a 5-minute average of utilisation; in general, there is no lower bound of such utilisation that will guarantee a bound on $\Delta Q|_V$ ⁴⁴.

It is possible to ‘trade’ $\Delta Q|_V$, that is, the $\Delta Q|_V$ of a given translocation through a network element can be made different from the rest. This can be done:

- by modifying the ingress behaviour (to the buffer). That is giving the particular flow (or class of flows) preferential access to some or all of the buffers, which has the effect of reducing the loss rate experienced;
- by modifying the egress behaviour (from the buffer). That is preferentially servicing packets from the chosen flow (or class of flows), which has the effect of reducing the the delay experienced.

These ingress and egress treatments are driven by some notion of precedence, which itself can be based on:

- the association (information derived from the source or destination address or similar, e.g. protocol type);
- recent usage patterns (e.g. offered load rate);
- some notion of ‘share’ (which could be some weighting, like servicing several packets for a particular flow for every one for another flow).

It should be remembered that whether this differential treatment can deliver an upper bound on the quality attenuation ($\lceil \Delta Q|_V \rceil$) of a given flow will depend on the pattern of its offered load as well as properties of the total load⁴⁵. Also, such differential treatment has consequences for the other flows passing through this multiplexing point, since the overall $\Delta Q|_V$ is conserved.

B.3.2. Location-based trading

As has been discussed above, the $\Delta Q|_V$ that occurs at a component depends on the traffic pattern, so changing that pattern can reduce the overall $\Delta Q|_V$ that accrues at this point in the network. This occurs during traffic shaping and rate policing, which induce additional $\Delta Q|_V$

⁴²This is in the sense of §A.1.1.

⁴³This ‘temperature’ is typically some measure of average utilisation, such as a moving average of the 5 or 15-minute load. Note that this is a pure heuristic, since averaging averages does not have any coherent mathematical interpretation.

⁴⁴The inference does work the opposite way around: when $\Delta Q|_V$ is frequently exceeding some threshold, often this implies high utilisation.

⁴⁵For example $\lceil \Delta Q|_V \rceil$ can be shown to depend only on the number of streams when applying the policy of “allocation to peak” (where the individual offered loads are strongly policed - either by the physical characteristics of the interface/circuit or otherwise - and their peak, including any encapsulation overheads, cannot exceed the service capacity of the egress). In all other cases delivering a $\lceil \Delta Q|_V \rceil$ depends on schedulability constraints being able to be met.

at one point to change the arrival pattern at a subsequent point. Thus rate limiting/traffic shaping ‘moves’ where the $\Delta Q|_V$ (for that particular translocation stream) accrues.

From the point of view of application outcomes, such $\Delta Q|_V$ trading does not necessarily have a detrimental effect. The contour lines of ‘equal outcome’ in Figure A.2 in §A.2.1 show that there is scope for trading ΔQ , while maintaining application outcome and hence user experience.

Interfaces implicitly act as traffic shapers. Thus, the change from narrowband to broadband to superfast can be seen as the slackening of rate limiters. This effectively moves $\Delta Q|_V$ between locations, in particular between management domains.

B.4. Other queueing and scheduling approaches

As we have seen, there are a set of inherent performance properties that naturally arise out of the operation of PBSM networks. The simplest implementation of a broadband network (comprising first-in first-out, tail-drop queues served by fixed-rate dedicated circuits) still engages in ‘traffic management’, in that it shares out the $\Delta Q|_V$ that inevitably occurs.

The particular $\Delta Q|_V$ that streams experience at a multiplexing point is the result of the ‘game’ that is being played out there, for the ingress and egress of the buffer. FIFO represents one set of rules for that game, but there are others. The game is driven by the arrival patterns of the streams as they pass through the multiplexing point. Although in many games there are notions of ‘winner’ and ‘loser’, the measure of success for the statistical multiplexing game is more complex. Indeed, the notion of success, and the value of delivering performance bounds, is an area with which the industry is only beginning to engage.

Although success may be difficult to quantify, the notion of failure is more amenable to analysis. Application performance over broadband is a technically sophisticated topic, but at its highest level the objective is delivering the outcome required in a suitably bounded time. The technical aspects of this can be summed up as delivering a bound on ΔQ so that the application remains within its *predictable region of operation*.

The internet design philosophy is one in which control traffic (such as routing updates) and data traffic traverse the same paths using common infrastructure. Thus some of the services that need to be kept within their PRO are essential to the effective operation of the Internet as a co-operating system. Typically, such services are maintaining associations (routing information, tunnel/encapsulation keep-alive exchanges) or detecting their failure (to manage redundancy and resilience). Failure to meet the translocation constraints for these services arms an operational hazard that may have wide-ranging effects⁴⁶. Trying to avoid such hazards often drives the deployment of different traffic management approaches. This is an attempt to maintain suitable translocation quality for ‘key’ applications (the notion of what is ‘key’ being driven by other concerns).

Inevitably, in a relatively new and technical subject, thinking is often driven by analogy with other areas or experiences. The term ‘traffic’ naturally evokes other applications of that word, but the nature of network packet traffic means that management and mitigation strategies from other sectors may not apply. Significantly, it is not possible to have control information flow faster than the packets themselves, which restricts the applicability of control theory. This has implications for the potential efficacy of control loops, in particular congestion management.

⁴⁶For example, when congestion on a path delays router updates too much, routers may conclude that the path is no longer available and so update their tables, shifting traffic onto another path that then becomes congested, and so on. This contributes to ‘router flap’.

B.4.1. Prerequisites for deployment of differential treatment

In order to apply TM, e.g. to maintain key services within their PRO, it is essential to be able to distinguish different components of the traffic. This requires some form of classification. This classification is typically performed on association information (addressing information or packet marking), though it can also be based on recent offered load or on the packet contents (through use of DPI). It should be noted that a particular marking does not imply that a particular treatment will occur, as the mapping between marking and treatment is determined by the per-device configuration.

B.4.2. Priority queueing

Priority queueing operates by differentially servicing the egress from the buffer. Packet flows are assigned to particular treatment classes on the basis of some classification (as described above).

Ingress

On arrival, a packet is admitted to the buffer if there is a free slot, as with the tail-drop behaviour in the FIFO case in §B.1.1. There are two common variants of buffer management:

1. where the buffering is shared amongst all the queues;
2. where there is an allocation of buffering to each treatment class.

Thus the loss element of ΔQ_V can be influenced either by all traffic or by just a subset of that traffic. This choice determines the exact nature of the coupling that occurs between the streams, within the constraint of there being two degrees of freedom in all finite queueing systems⁴⁷.

Egress

The egress treatment (and hence the delay component of ΔQ_V) is determined by the relative precedence of treatment classes. Packets are serviced from the highest precedence treatment class first. Packets are serviced from lower precedence classes only when all higher precedence classes are empty. Within a treatment class, packets are serviced in order of arrival⁴⁸.

Discussion

The highest precedence traffic experiences lower mean delay and a lower delay variance than other traffic. It also gets the strongest isolation from other streams, with its delay being affected only by other traffic in the same class⁴⁹. Traffic from other precedence classes can potentially experience large perturbations in delay, depending on both the volume and arrival pattern of traffic in higher precedence classes. Where there is per-treatment-class buffer allocation, the collective arrival pattern of all higher precedence treatment classes can cause the buffers for lower precedence classes to fill. This has the effect of differentially allocating loss to the lower precedence classes. If the buffer is shared, the loss rate is the same for all treatment classes⁵⁰. This is illustrated in Figure B.2.

If the highest precedence treatment class arrival rate is not limited (either explicitly in the device, or implicitly by other design constraints or traffic management approaches), then the lower precedence treatment classes can experience an effective denial-of-service.

⁴⁷These two degrees of freedom are loss and delay.

⁴⁸This is typically implemented by placing arriving packets at the end of the corresponding treatment class queue, and by servicing non-empty queues in precedence order.

⁴⁹Traffic in lower precedence classes can affect higher precedence classes only if it is already being serviced. When this is the case, higher precedence traffic is delayed by any residual packet service time.

⁵⁰This assumes the conditions mentioned in §B.1.1.4 are met. That is, that arrivals are Markovian etc.

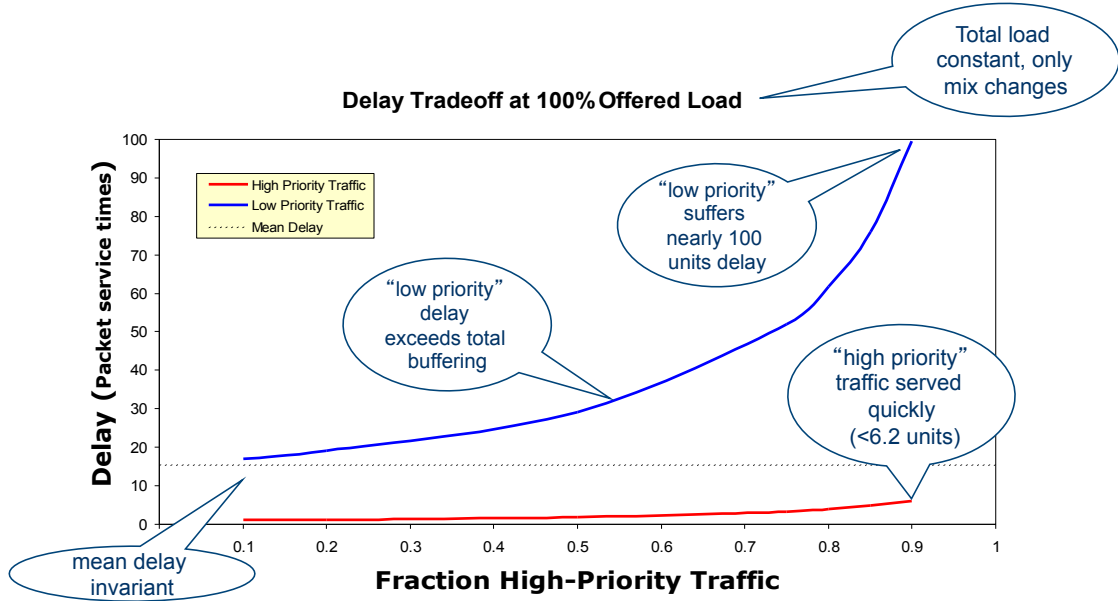


Figure B.2.: Differential delay in a two-precedence-class system (with shared buffer)

B.4.3. Bandwidth sharing

Bandwidth sharing endeavours to share egress capacity so as to deliver some minimum service capacity (‘bandwidth’) to a set of streams over a long period. It does not aim to deliver a particular bound on, or ordering between, the $\Delta Q_{|V}$ of any of these streams⁵¹. Differential treatment is expressed as the shares (which may be equal) that the (dynamic) set of competing streams receive (the comments on classification on the previous page apply). The design and discussion of queueing and scheduling of this type is underpinned by the fluid-flow approximation⁵² [30]. There are several approaches to implementing this approximation (such as round-robin, hierarchical token bucket shaping, etc.) [31], and substantial variation in the way that the resulting sharing can be configured within a particular implementation. The aim here is not to discuss these differences, but rather to describe the common effects of bandwidth sharing on the distribution of the $\Delta Q_{|V}$ inherently created in the multiplexing process.

Ingress

On arrival, a packet is admitted to the buffer if there is a free slot, as with the tail-drop behaviour in the FIFO case in §B.1.1. The total buffering can be shared, or can be reserved/limited on a per-stream basis.

Egress

In bandwidth sharing, the way that one stream’s $\Delta Q_{|V}$ is affected by the other streams depends on both their offered load and the number of streams that are active. We will consider examples (see Cases 1 and 2 below) of these two distinct couplings before discussing their composite $\Delta Q_{|V}$ effects and the consequences on example deployments.

A useful mental model is to consider bandwidth sharing as dividing the traffic into separate queues amongst which the egress service capacity is distributed in some fashion (i.e. as a collection of FIFOs with continuously varying egress service, see Figure B.3).

⁵¹Delivering more ‘bandwidth’ to a stream does not imply that its $\Delta Q_{|V}$ will be better; it depends on the balance of supply and demand.

⁵²The fluid-flow approximation treats packets as entities whose service can be ‘spread out’ over time. This fails to capture aspects of the discrete service time of packets.

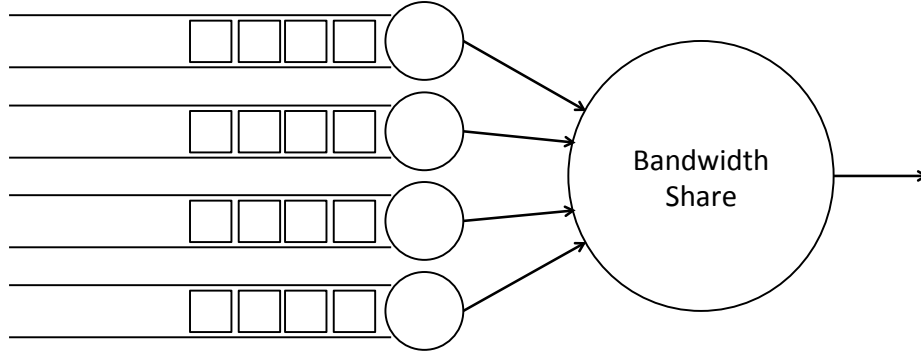


Figure B.3.: Bandwidth sharing viewed as a collection of FIFOs

Case 1: offered load variation and effects on $\Delta Q|_V$ stationarity Consider the situation where there are two such streams⁵³, A and B , each being assured $1/2$ of the egress capacity⁵⁴. Let us also assume that each has access to any spare capacity that is not being used by the other (this is a commonly used configuration).

Imagine the situation where stream B is idle, and stream A is using 60% of the overall capacity (which is 120% of its assured capacity). In this circumstance, queue A behaves as a FIFO under a 60% load, exhibiting all the properties discussed in §B.1.1. Now imagine that B becomes active and offers a 45% load (90% of its assured capacity). The system is now being offered a total load of 105%. Stream A is receiving 55% of the egress capacity to carry 60% of the load, and so it is operating at just over 109% loading. The result is that queue A fills and starts losing in excess of 8% of its packets⁵⁵, until the offered load is reduced. Thus, the $\Delta Q|_V$ experienced by stream A is coupled to both its offered load *and* the offered load of B . As a result, the overall ΔQ for stream A can vary quickly.

As was discussed in §B.2 there is no explicit feedback in PBSM. The network relies on the originators of A 's traffic reducing their load. The end-to-end principle means that the time to reduce the load at this single point is dependent on how quickly these remote systems can adjust their behaviour. This is related to the round-trip time these systems are experiencing, which includes the $\Delta Q|_V$ induced by the change in the offered load through queue B .

Thus bandwidth sharing induces variability in the distribution of $\Delta Q|_V$, which manifests as non-stationarity⁵⁶ for the constituent translocations passing through it.

Case 2: active streams and the effect on $\Delta Q|_V$ stationarity Consider the situation where there are a number of streams⁵⁷, each receiving the same share of the egress capacity (when all are busy). In this situation, the $\Delta Q|_V$ of any one stream is affected not only by the collective offered load of the others, but also by how many of them are active at any point in time.

Imagine that stream A 's load is low compared to its assigned share. Whenever it has just received service, it must wait to be serviced again until each of the other busy streams has received service. Thus the time between successive services for stream A is proportional to the number of other busy streams. So its $\Delta Q|_V$, in particular the delay distribution, is now

⁵³When using the mental model mentioned above, it is important to recognise the difference between a 'stream' and a 'queue'. A stream, in this context, is a packet-flow of information (from a particular source and/or to a particular destination). A queue is the logical arrangement of packets, from a stream, within a network element. In this example, the two queues also separate the treatment of the two flows, however in general several flows may share one queue.

⁵⁴When both queues are non-empty they are serviced so as to get a 50:50 share.

⁵⁵The exact proportion will depend on A 's arrival pattern and the distribution of B 's busy period.

⁵⁶Non-stationarity is the variation of $\Delta Q|_V$ over timescales of seconds or less.

⁵⁷Access network elements will often have several hundred configured streams (at least one per end customer). In core networks (where MPLS or Carrier-Ethernet services are sold) such streams can be used in significant numbers to offer bandwidth-based service levels.

dependent not only on the overall load, but on how many other streams are busy. This can vary in the short term (introducing non-stationarity), as already configured streams become busy. It can also vary over longer time scales, as the configured number of streams grows. This increases the potential non-stationarity.

Discussion

Here we consider some concrete examples of: how bandwidth sharing can be deployed; the magnitude of the effect on ΔQ_V ; and the possible consequences of the resulting variation in ΔQ_V .

A commercial response to the costs of delivering high-speed mobile broadband has been to create RAN sharing agreements. One of the ways in which costs are saved is by sharing backhaul connectivity. This sharing is typically contracted on a bandwidth basis. Picking up on Case 1 above, consider a 400Mbps Carrier Ethernet link to a shared MNO base station. This link is shared by two MNO partners, each having an assured 200Mbps, with access to the full 400Mbps in bursts. Under some general assumptions⁵⁸, in the initial scenario (stream A offering 60% load, while stream B's load is negligible) the ΔQ_V that A experiences is effectively \emptyset ⁵⁹. After B's offered load increases to 45% (which drives the system to 105% capacity) the ΔQ_V for A will rise, to a loss of over 8% with mean delays ranging between 2ms and 50ms⁶⁰. Although elastic network protocols (e.g. TCP) will adapt, this takes several end-to-end round trip times⁶¹.

This ΔQ_V , if evenly distributed amongst the sub-streams in A, would represent an operational hazard to both control protocols (where loss and delay stability are important) and voice streams (for which it is consuming a substantial portion of the end-to-end delay variation budget⁶²). This is an example of how variation of offered load generates variable ΔQ_V .

Referring to Case 2 above, as an example of how the number of active streams affects ΔQ_V , consider a multiplexing point in the downstream path to broadband end-users (e.g. a Head End or BRAS from Figure C.1 on page 79). As the number of active users increases, so does the time between service of their individual streams. Not only does this change in the distribution of ΔQ_V potentially influence the user experience (particularly affecting services like gaming and VoIP), if it persists it can also cause issues with timing protocols⁶³.

Unlike in Case 1 (where stream A could return to a stable ΔQ_V by reducing its load to a suitable level, however long that might take), achieving a stable ΔQ_V is now dependent on the the number of other active streams. Thus, no level of reduction in the volume of traffic for stream A will achieve the desired result. This is where some differential distribution of

⁵⁸These assumptions are: the fluid flow approximation; the Markovian arrival/service assumption; that each queue is treated as $M/M/1/K$; that the packets in question are full QinQ Ethernet packets of 1546 octets, including both the frame overheads and the inter-packet gap; and that the change in the egress capacity changes the Markovian service rate of the queue under consideration.

⁵⁹There is a slight dependence on the number of buffers, but the loss would be $\ll 10^{-7}$ pkts/sec and the mean delay would be around 28 μ s.

⁶⁰The loss and delay depend on the buffering available, for example: at 50 buffers the mean delay would be about 75% of the buffering capacity or 2.17×10^{-3} s; at 100 buffers, 88% (4.95×10^{-3} s); at 500 buffers 97.6% (27.4×10^{-3} s); and at 1000 buffers 98.8% (55.5×10^{-3} s). Note that the effective service rate has dropped. In the first scenario the *effective* packet service time (for the queuing calculations) was 30.9×10^{-6} , in the second scenario it becomes 56.2×10^{-6} s.

⁶¹The RTT could vary from 100ms to 500ms or more, depending on the location of the sources of the variable data load. Note that the congestion feedback signal (the discard of packets) does not occur until the queue is full, i.e. there is already - for the 1000 packet buffer case - 50ms of additional delay to the round trip.

⁶²ITU Y.1541 suggests that the one-way voice delay budget should be bounded at ≤ 150 ms mouth-to-ear, with 50ms due to delay variation. ETSI TS 103 210 V1.2.1 suggests that access networks should induce < 35 ms of "jitter".

⁶³Protocols such as NTP and IEEE1588 underpin the deployment of small cell technology over commodity broadband infrastructure. The response of such a cell to the sort of variation described is to conclude that the local clock has suffered a precision failure. This, in turn, can result in the device using an incorrect frequency at the radio interface or having to reset.

$\Delta Q|_V$ may be required for keeping key services (be they user or system orientated) within their PRO.

A common sharing model is to offer a guaranteed lower bound, together with a potentially achievable peak bandwidth for a particular stream (a ‘committed/peak’ model). In broadband the peak may be set: implicitly by the sync rate of the line (as in xDSL); explicitly (as in Cable); or by a combination of both (as in 3GPP). The lower bound is implicitly determined by the number of end-users connected to the last hop multiplexor. Between these two limits there is an even distribution of the capacity amongst the active end-users. Such scheduling arrangements can act as a bandwidth leveller; as the number of end-users (and their offered load) increases, those end-users that have the highest peaks (whether explicit or implicit) will start to experience a reduction in their rates *before* those with lower peak rates. This leads to the situation where those with the highest peak data rates actually experience the largest variation in delivered rate and $\Delta Q|_V$ non-stationarity (which many may consider counterintuitive).

B.4.4. Rate shaping

As has been discussed above, a key factor determining the amount of $\Delta Q|_V$ occurring at a contention point is the arrival pattern of the offered load. Rate shaping [32] is a mechanism used to mitigate the ‘worst’ patterns of such offered load. It is, in effect, a tail drop FIFO that is used to adjust the inter-packet gap and discard packets when the ingress rate exceeds the egress rate for a sufficient period of time.

Ingress

On arrival, a packet is admitted to the buffer if there a free slot, as with the tail-drop behaviour in the FIFO case in §B.1.1. Since the packet is dropped if the buffer is full, rate shaping also implicitly limits the maximum sustained rate.

Egress

Packets are sent in the order they are received. The average rate at which packets can depart is fixed. When the instantaneous arrival rate exceeds this fixed rate⁶⁴, the packets experience $\Delta Q|_V$.

Discussion

Resource sharing in PBSM is rivalrous, in that consumption by one user affects the service delivered to others. Rate shaping, by modifying the arrival pattern of the offered load, can help place some limits on these effects.

The limiting aspect can be used to enforce the demand side of peak-rate-based contracts. Such rate shaping can also be applied to specific sub-streams of user data. When such sub-streams are being offered a ‘higher quality’ translocation, it can be used to ensure that the offered load conforms to agreed limits.

B.4.5. Rate policing

Another approach to demand-side management is rate policing [32]. This approach does not introduce any $\Delta Q|_S$ or any delay part of $\Delta Q|_V$. Its function is to drop (or re-mark) packets that exceed a pre-configured rate, so it contributes to the loss component of $\Delta Q|_V$.

⁶⁴There are several implementation choices which trade the ΔQ of the rate limiter between $\Delta Q|_S$ and $\Delta Q|_V$, e.g. Token Bucket shaping. They can also permit bursts (where the packet rate exceeds the rate limit for a short time), which have the effect of moving the point at which $\Delta Q|_V$ accrues further downstream.

Ingress

The packet is discarded, or re-marked, if the time since the arrival of previous packets is too short. The time would be deemed to be too short if the short-term average packet arrival rate exceeds some preconfigured limit⁶⁵. There is no buffer, as Rate Policing is never performed at a transmission egress point.

Egress

Packets that are admitted are passed on immediately in the order of arrival.

Discussion

Rate policing and rate shaping (discussed in B.4.4 above) take a different approach from the methods previously discussed to sharing out the overall ΔQ_V . FIFO, priority queuing and bandwidth sharing are concerned with sharing ΔQ_V at a particular multiplexing point, whereas rate policing and rate shaping are concerned with how it is shared along a path. By modifying the arrival pattern (introducing additional ΔQ_V locally) they are changing the ΔQ_V that would occur downstream.

Rate policing (using re-marking) is currently deployed in UK broadband delivery (as outlined in BT Supplier Information Note 506, discussed in Appendix D). With the general increase in the variety of services using PBSM, it is likely that the use of rate policing and rate shaping techniques will increase. If some services were assured, this assurance would come at a cost of worse ΔQ for others, and/or increased costs on the provider to maintain the overall service level. This would mean that there would be an advantage for other traffic to masquerade as an assured service. This is a hazard already armed by sharing the last mile between different services (for example VoD and CDN distribution).

B.5. Factors influencing the further deployment of traffic management

It is in the very nature of statistical multiplexing that ΔQ_V will increase with the number of subscribers⁶⁶. From a ΔQ perspective, the trend in broadband provisioning has been a race between improving technology and increasing demands. Improving technology has created faster links (thereby decreasing ΔQ_S) and increased switching capacity (thus potentially reducing ΔQ_V). Rising demand, in overall and in individual peak quantity (due, in particular to higher access link speeds), increases ΔQ_V and its non-stationarity. This increasing non-stationarity is not only a challenge for ‘critical services’, whose PRO is essential for network stability, but may become an issue for some end-user services, depending on their sensitivity.

If there is a demand for more consistent delivery of end-user services whose PRO cannot be maintained by the current ‘best effort’ approach, the service paradigm may have to change. Attempting to deliver all traffic within the ΔQ bound of the most sensitive applications (allowing “the needs of the few to drive the costs of the many”) may prove to be commercially unsustainable. This could be a strong driver for an increase in the use of differential traffic management.

⁶⁵There are several approaches to implementing this; Token Bucket is one that is commonly available

⁶⁶This phenomenon was heavily exploited by users during the earlier dial-up internet days when the ‘smart thing’ was to hop from one ISP to the next as they launched.

B.6. Static and dynamic allocation of translocation resources

As has been seen, a key element of traffic management is the process of deciding whether, and in what order, to deliver service (be that ingress or egress). The outcome of these decisions is the distribution of $\Delta Q_{|V}$ across competing streams.

Although we have described the broad range of traffic management approaches that are available, there is another important factor that also needs to be covered. This is the difference between scheduling at a single point (that has been discussed so far) and the distributed scheduling that is required over a shared medium. In broadband deployments, such distributed scheduling is found mainly in the last mile. It is a key characteristic of: 3GPP-based systems (2G/3G/4G-LTE); 802.11 WiFi (and similar); the upstream (from the customer premise) of DOCSIS cable and Satellite systems⁶⁷; and some last-mile fibre systems, depending on the deployed variant and its configuration. Such systems deliver connectivity when the physical medium is shared, and are able dynamically to allocate resources in order to deliver high peak rates to individual users.

Where multiple ‘talkers’ share a common medium, this creates a *distributed contention domain*. In such a distributed contention domain, permitting any arbitrary end-point to talk at will gives neither a predictable service nor efficient use of the shared medium⁶⁸. The typical approach to improve the PRO of such systems is to use an arbiter, an entity that receives requests for service and grants access to a portion of the shared medium’s capacity⁶⁹. This has consequences for both the $\Delta Q_{|V}$ and the inherent efficiency of the aggregate system⁷⁰.

Consider how a conversation would start. Initially, a talker has to arrange for some of the shared resource⁷¹ to be allocated to it⁷² (thereby removing that capacity from the common pool). Not only does this take time, it also has to be achieved by use of un-arbitrated capacity⁷³. This allocation process has an associated $\Delta Q_{|V}$, that depends on the instantaneous demand on the un-arbitrated capacity from all the other talkers⁷⁴.

As a talker increases its demand, there is a time lag in being granted more capacity (if it is available) which contributes to $\Delta Q_{|V}$. When a talker reduces its demand, there is a corresponding time lag before that resource can be allocated to another talker, causing some inefficiency in the use of the shared medium⁷⁵.

B.7. Consistent performance and a heterogeneous delivery chain

The Internet comprises a heterogeneous delivery chain. It is a set of autonomous entities (telcos of various tiers, carriers, ISPs, etc.) that can be seen as collectively constructing connectivity, with emergent translocation performance. The equipment under the control

⁶⁷In the downstream direction these systems have visibility of the instantaneous offered load at their head-end, and therefore can apply the scheduling techniques discussed above.

⁶⁸For example, random talking (Aloha) access restricts the PRO to $< 18\%$ of capacity (under a Markovian arrival assumption). This rises to about 36% when access to the resource is slotted (as is used for grant requests in DOCSIS and 3GPP). See <http://en.wikipedia.org/wiki/ALOHA>.

⁶⁹In 3GPP this would be the RNC or the eNodeB, in 802.11 the access point, and in DOCSIS the CMTS.

⁷⁰The arbiter scheduling the upstream capacity typically has the same emergent behaviour as bandwidth sharing systems (described above). In particular, with a large number of active talkers, the interval between grants for service for any particular talker can become so large (and so variable) that higher-level protocols (such as TCP/IP) are pushed outside their PRO.

⁷¹This could be a time slot in DOCSIS or a portion of the code space in 3GPP.

⁷²By doing this, it is effectively setting its future $\Delta Q_{|G,S}$ for this hop for the period of the allocation.

⁷³This typically operates as described in footnote 68.

⁷⁴If the number of talkers becomes too large, the demand on the un-arbitrated grant/request capacity can exceed its PRO (as described in footnote 68), resulting in a failure to issue grants and a consequent denial of service.

⁷⁵The speed with which resources are allocated to and deallocated from individual talkers is thus a key factor determining the tradeoff between delivered ΔQ and the efficient use of the shared medium.

of a single entity (which represents an administrative domain) may be operated by several management domains. Thus, traffic being translocated end-to-end may well cross a plethora of management, administrative and even regulatory jurisdictions.

To keep a particular application within its PRO, the ‘sum’ of the ΔQ across all the multiplexing points traversed has to be sufficiently bounded and $\Delta Q|_V$ sufficiently stationary. Since ΔQ is conserved, if the $\Delta Q|_V$ accrued through even a few multiplexing points is large, the delivered delay and loss may cause the application to exceed its PRO. From the end-user’s perspective, the application will have ‘failed’.

When constructing connectivity the entities do not, typically, contract to any instantaneous performance guarantees⁷⁶. They may aspire to a given level of availability, or even some assurance of reachability, but performance is almost always provided on a purely ‘Best Efforts’ basis⁷⁷.

Detecting a lack of connectivity is a relatively clear-cut process, as there will be an identifiable location that is either reachable or not. The additive nature of performance impairment means that identifying the underlying location of a performance issue (let alone its root cause) is more difficult. The ΔQ that accrues along a section of the end-to-end path is indistinguishable from that which accrues along another - it is only their combined effect that can be observed at an end-point.

For any entity in the end-to-end delivery chain, maintaining consistent performance as load increases may be difficult to justify commercially. This is especially true given the absence of contractual performance guarantees, and the current inability of interested parties to pinpoint where ΔQ accrues. There are particular problems in access networks, which are commercially predicated on amortising capital costs over a large number of customers. This creates the natural tendency for the multiplexing points in the network to be run “hotter” over time, resulting in increasing $\Delta Q|_V$.

Along any end-to-end path several multiplexing points may be manifesting this trend of increasing $\Delta Q|_V$. If the end-user experiences an ‘application failure’ (due to translocation being outside the PRO), it could be simply due to the aggregated effects of these commercial trends, rather than due to any one entity’s traffic management policies.

⁷⁶Guarantees that might be offered would be in terms of averages, usually over long periods of time (e.g. a month). Such measures cannot be composed along a path, and entities do not take on the risks of their suppliers or onward connections, so these assurances are not ‘transitive’.

⁷⁷“Best Efforts” in internet network terms - http://en.wikipedia.org/wiki/Best-effort_delivery is at complete variance with the use of the term in UK commercial practice <http://dictionary.cambridge.org/dictionary/business-english/best-efforts>.

C. The Internet in the UK from a traffic management perspective

“Internet: A global computer network providing a variety of information and communication facilities, consisting of interconnected networks using standardised communication protocols.” - OED

Evidently, in the current commonly-understood definition of the Internet, computation and communication capabilities are mingled together. The term “internet” was originally used to refer solely to information exchange capability; now “internet” is used to refer to the facilities based upon this capability.

Here we are going to focus on how consumers are provided access to this global network from the premises onwards¹. This description is intended to highlight the major boundaries within, and distinctions between, different widespread forms of access provision. Despite being quite detailed, this is not a complete statement of the UK market, as there are additional delivery models (e.g. specific fibre-to-the-premise installations or point-to-point/multipoint wireless) that will not be covered here. There is also a significant simplification of the international picture, which is included for reference.

C.1. UK network administrative and management boundaries

A consumer of internet access services (whether domestic or commercial) has to have a connection to one of a variety of infrastructures. The UK market has a considerable diversity of structure and technology in terms of ISPs’ service provision. We will cover this in more detail later in this section, but for now let us highlight an aspect common to all, namely that the management of a user’s connection to the wider Internet is split across different entities, some internal and some external.

The UK ISP market is based around the construction and reselling of connectivity monopolies. For example, BT OpenReach may provide the physical communication path between a premise and the first active component (this being a natural monopoly). That monopoly is ‘sold’, on a per end-point basis, to either an LLU unbundler or BT Wholesale, who then have sole use of it. They take that physical circuit and ‘activate’ it (placing active components at each end), which creates point-to-point connectivity (monopolistically). That connectivity is then (in the case of BT Wholesale) sold on to the ISP², who has a monopoly over all the traffic sent and received by the end-user³.

The organisations that supply broadband tend to be large and so are split into multiple internal, semi-autonomous, management domains (silos). Each of these management domains is assigned key performance indicators that it then works to optimise⁴. This pattern is common to all of the services provided in the UK, the differing factor being whether (and how)

¹Local network conditions (LANs, router configurations, wireless interference) are not going to be considered.

²Other services can be multiplexed into that connectivity, but only within the same exclusivity constraints. These interactions of other services with the ‘normal’ ISP function represent a potential source of performance impairment, as is clear from the BT Supplier Information Notes (SINs: discussed in Appendix D). Specific quantitative investigation of this topic is outside the scope of this report.

³This is why end-users typically assume that the ISP has sole responsibility for the performance of their services.

⁴Unfortunately, it is a general feature of the engineering of complex systems that the combination of local optimisations rarely delivers a globally optimal result.

these management domains are split into different administrative domains. In Figure C.1 we see that vertically integrated ISPs, for example Virgin Media’s cable division, have different management domains that all fall under the administrative domain of one company. With ISPs like BT Retail, though the consumer deals with BT Retail their connection is managed at various points by the separate companies (and thus administrative domains): BT Retail, BT Wholesale, and BT OpenReach.

Each of the boundaries between these different companies’ administrative domains is subject to contracts for the supply of services. It should be noted that whether this series of bilateral agreements delivers anything that is enforceable end-to-end is an open question. So, in the example above, BT Retail has no direct contractual relationship with BT OpenReach, even though it is the latter who is responsible for the actual physical connection to the customer⁵. Figure C.1 illustrates the general administrative and management domain structure of the UK broadband market. What consumers consider to be ‘their ISP’ really represents an administrative domain (encompassing a collection of management domains), which typically provides their service using other administrative domains’ services.

Different customers of the same ISP (for example at different locations) can have their service delivered through entirely different collections of administrative and management arrangements. This is an ecosystem that contains a large diversity, not only in the paths that can be taken through the management and administrative domains, but also in the management, configuration and operational policies and practices thereof. Despite this diversity, any given end consumer is presented with only a small number of connection options, typically ADSL and possibly VDSL and/or cable. Changing ISP does not change the physical medium for the final access tail, except in the case of a cable or non-wireline provider.

From a performance analysis perspective, the difference in how the final access tail operates has a significant bearing. Broadly it can be dedicated connectivity (e.g. ADSL) or a contended medium (e.g. cable/3G/LTE), see §B.6. In both cases, the peak potential capacity is asymmetric. A brief comparison of the performance affecting properties of these different access tails can be found in Table C.1.

Figure C.2 provides a simple depiction of how UK network users connect to the wider Internet. The diagram follows upwards from Figures C.1 and C.3. For clarification, the ‘Tier 1 Service Providers’ are international higher-level CSPs who provide connectivity to other networks in other locations (the multitude of connections from them has not been shown).

C.1.1. Non-Wireline ISP provision

Although the following discussion is going to focus on wireline-based ISPs, we must also consider the wireless side of the UK telecoms industry, illustrated by Figure C.3 on page 82. The techniques and approach in this report can be applied to these networks, but the detailed analysis is out of scope for this study.

Satellite-based ISPs inevitably have a large $\Delta Q_{|g}$ associated with the path to/from the geosynchronous satellite⁶, which affects the efficiency of the upstream resource allocation (the grant of a time/frequency slot to transmit), as discussed in §B.6.

Mobile cellular radio networks have a complex set of resource management issues. These networks have multiple resource constraints and multiplexing points⁷, all of which are po-

⁵There is no reason to believe that, because a set of management domains are part of the same administrative domain, they will work ‘better’ together. Logically, these silos operate to maximise their incentives, which are set by their overarching administrative domain. Administrative domains set objectives (be they fiscal or of another type) and create contractual arrangements with other administrative domains. Management domains take those objectives (as input aspirations), and construct operational steps to meet them (architecting interconnects, planning/provisioning capacity, configuring equipment, etc.). None of this fosters an end-to-end view of service provision.

⁶MEO/LEO satellites have a lower $\Delta Q_{|g}$, but this varies with time, depending on the orbital position.

⁷These include the GGSN, SGSN, RNC, SGW, PGW, and the connectivity between them, as well as the (e)NodeB air interface.

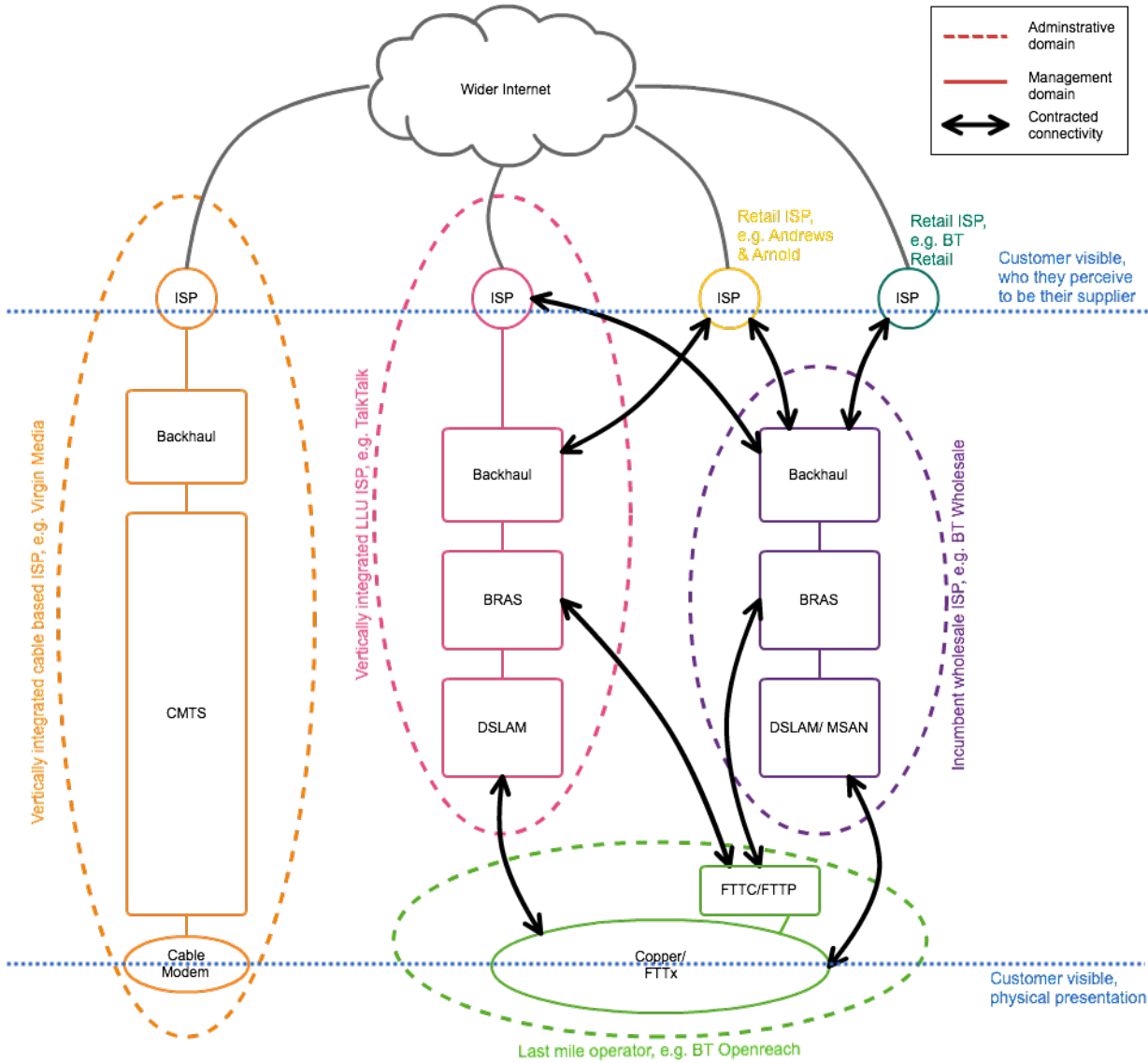


Figure C.1.: Representation of the administrative and management boundaries in UK broadband provision (wireline)

Connection Type	Capacity Delivery Strategy (to/from final active component)	Upstream Access Strategy	First/Last queueing component
ADSL	Dedicated time slots (capacity) in both directions. Capacity environmentally constrained.	Encode packet into cells, place in time slots.	DSLAM/MSAN
VDSL/FTTC	Dedicated time slots (capacity) in both directions. Capacity environmentally constrained.	Encode packet into cells/frames, place in time slots.	Street DSLAM
FTTP ^a	Dedicated capacity in the upstream (to OLT). Downstream overbooked.	CPE must shape. Allocation to peak.	OLT
FTTP ^b	Variable capacity in both directions. The medium is contended but has upstream coordination.	Request upstream capacity (time slots), when granted fill with packet(s).	OLT
Cable	Variable capacity in both directions. The medium is contended but has upstream coordination.	Request upstream capacity (time slots), when granted fill with packet(s).	CMTS
3G/LTE	Variable capacity in both directions. The medium is contended but has upstream coordination. Capacity is environmentally constrained.	Request upstream capacity (time slots), when granted fill with fragmented packets(s).	(e)NodeB
Satellite	Variable capacity in both directions. The medium is contended but has upstream coordination. Capacity is environmentally constrained.	Request upstream capacity (time slots), when granted fill with packet(s).	Downstream - base station. Upstream - VSAT.
WiFi ^c	Variable capacity in both directions. The medium is contended. Various environmental constraints.	Distributed coordination (detection of non-delivery leads to exponential backoff).	WiFi Access Point.

^aBT OpenReach-style deployment^bOther deployment approaches, included for comparison^cIncluded for comparison purposes only

Table C.1.: Comparison of access connection performance properties.

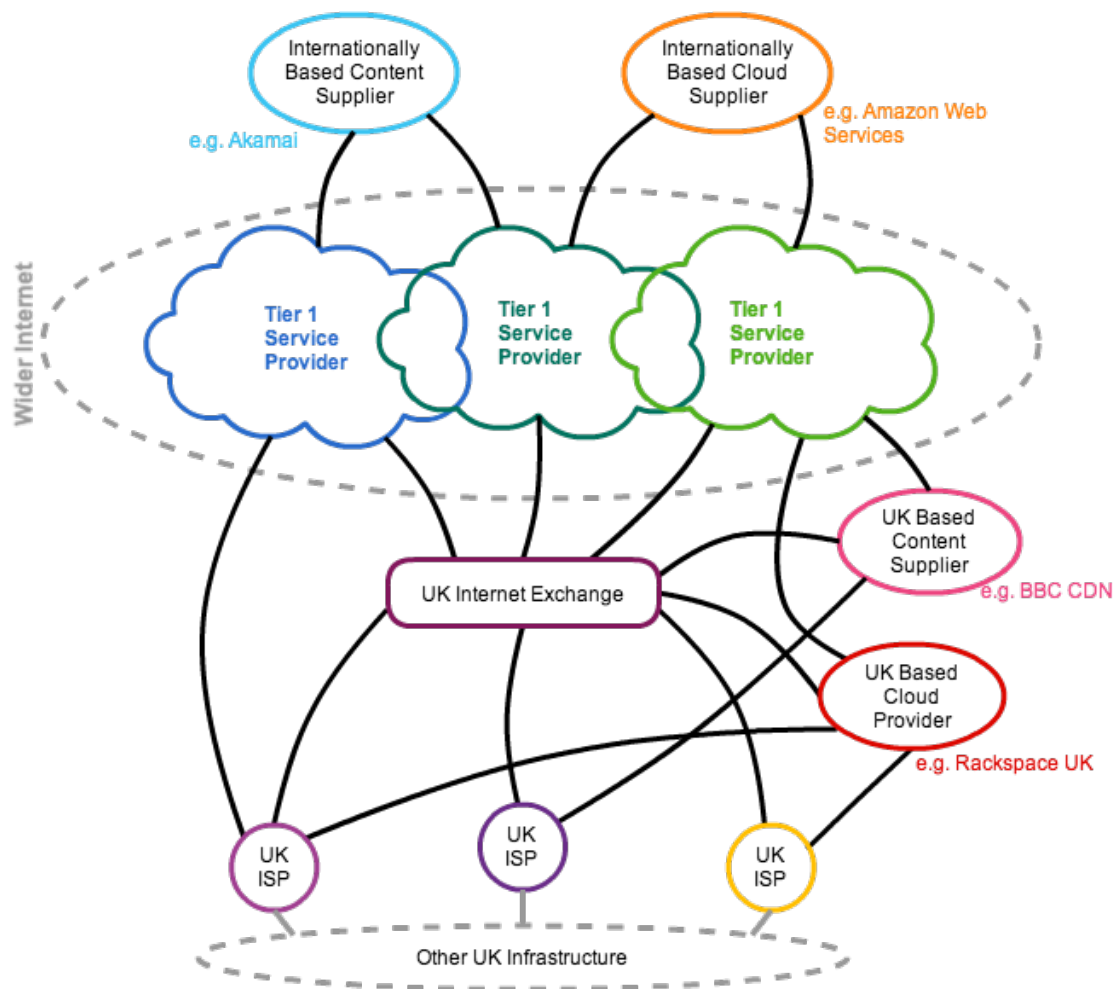


Figure C.2.: UK ISPs in wider context

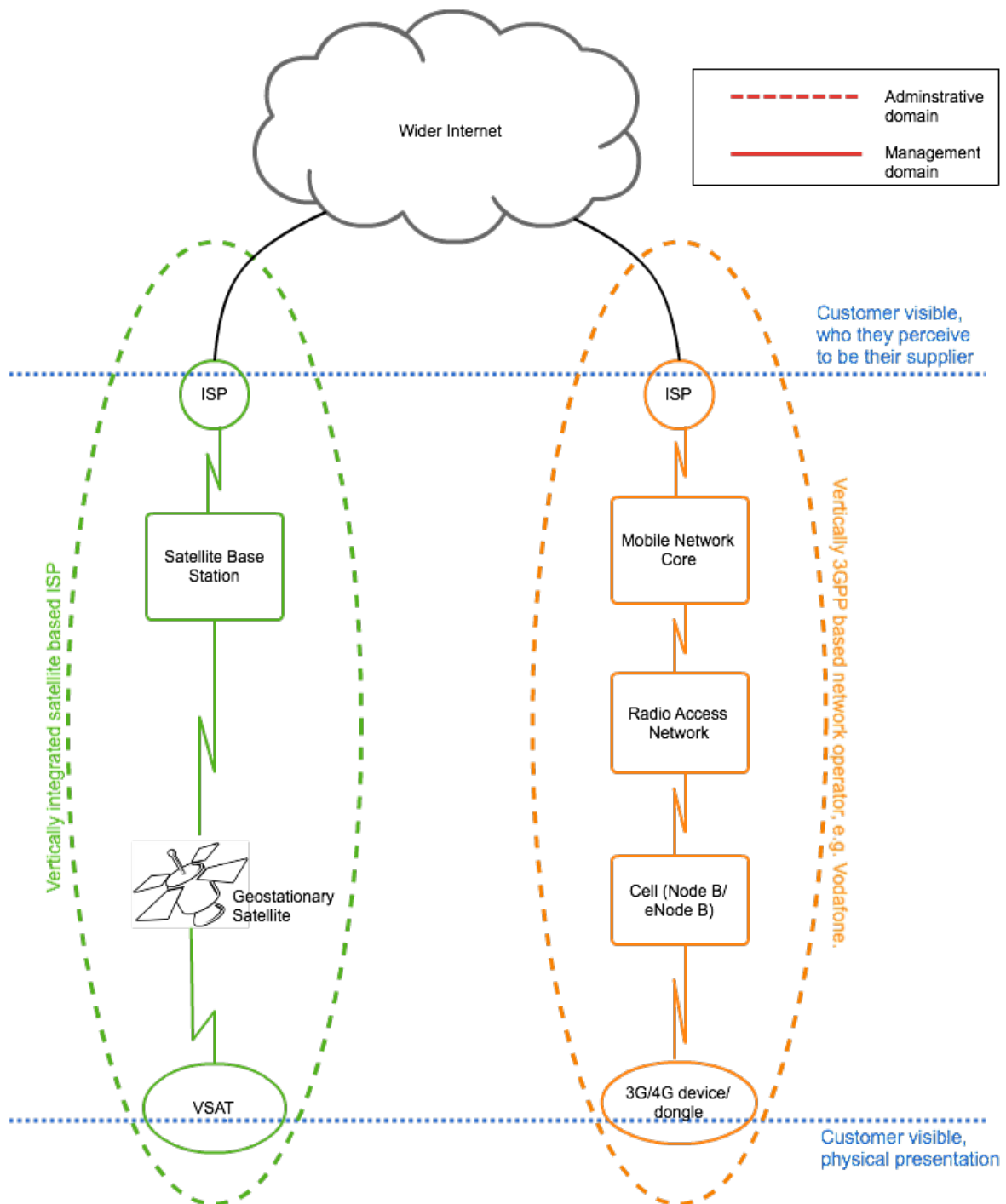


Figure C.3.: Administrative and management boundaries in UK broadband provision (non-wireline)

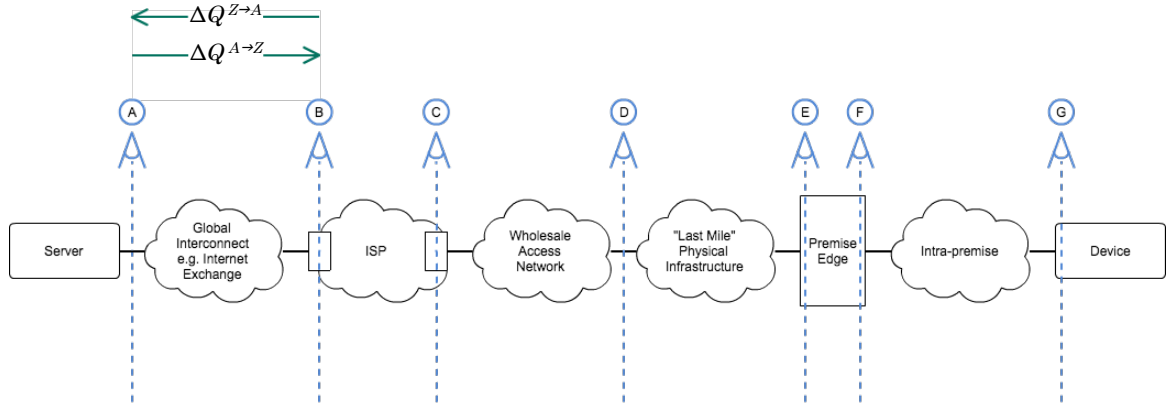


Figure C.4.: Idealised end-to-end path for typical UK consumer

tentially subject to contention and resource saturation. With the advent of LTE, the direct interaction of eNodeBs with one another (a role that was once uniquely assigned to RNCs) further complicates delivering consistent ΔQ whilst mobile.

There is a set of interlinked issues in the scheduling of the air interface resource, touched upon in §B.6. The mobile base station communicates with a number of associated Mobile Terminals (MTs). Each of the MTs may require a different fraction of the overall capacity to achieve a given throughput⁸. The load on the resource is not just dependent on the data being transferred, but also on how the relative MT location and other environmental conditions (including the utilisation of nearby cells) influences the code-space costs for carrying that data. The translocation costs are a function of all these factors.

C.2. How ΔQ accrues in the UK broadband context

As discussed in §A.2.1, applications produce outcomes by exchanging information between protocol peers (e.g. between client and server), and the only aspect of this translocation that affects an application's outcome is the ΔQ experienced by the corresponding traffic flows⁹. As has already been discussed, the $\Delta Q^{A \rightarrow Z}$ between two points 'A' and 'Z' is the 'sum' of the ΔQ s accrued along the path between them¹⁰. So, the structural component, $\Delta Q_{|G,S}^{A \rightarrow Z}$, is going to be determined by the actual path taken; while the variable component, $\Delta Q_{|V}^{A \rightarrow Z}$, is going to be determined by the contention at the individual hops. Each such multiplexing point is a location at which $\Delta Q_{|V}$ both forms and is distributed over the set of competing flows, as discussed in Appendix B. Since ΔQ is conserved (i.e. it cannot be undone), this means that the performance of an application is dependent on the composite effect of the journey of its traffic through different management and administrative domains.

Figure C.4 illustrates an idealised end-to-end path for a typical UK broadband connection. It identifies the major boundaries, some of which represent administrative domains (e.g. $C \leftrightarrow E$ - retail ISP, or $A \leftrightarrow B$ - an Internet exchange), and some management boundaries (e.g. $D \leftrightarrow E$ - DOCSIS headend in cable systems or xDSL in an LLU provider).

The overall experience delivered to the user (via a device) is constrained by the combination of the performance of the application and the performance of the translocation between the components thereof. The application is split, with a portion in the user device (at 'G') and

⁸The quantity of time/frequency code slots needed depends on the signal-to-noise ratio of the corresponding radio bearer.

⁹Note that QoE is solely based on *delivered* ΔQ , and that bounds on delivered ΔQ are not currently offered by any ISP (or administrative domain in the end-to-end path).

¹⁰Technically, this is the transitive closure of the appropriate convolution operation, i.e. $\Delta Q^{A \rightarrow Z} = \Delta Q^{A \rightarrow B} \oplus \Delta Q^{B \rightarrow C} \oplus \dots \oplus \Delta Q^{Y \rightarrow Z}$.

a portion in the server¹¹ (at ‘A’). The translocation that affects the application performance comprises two unidirectional ΔQ s: $\Delta Q^{A \rightarrow G}$ and $\Delta Q^{G \rightarrow A}$.

Although the delivery path may be composed of many different technologies, their only effect on the user experience is the way in which they affect the ΔQ over each (unidirectional) end-to-end path¹². As an example, consider the difference between a DOCSIS-based and an ADSL LLU-based ISP service. In ΔQ terms, these would substantially differ from each other only along the bi-directional path $D \leftrightarrow E$ (the last mile physical infrastructure). The DOCSIS service uses a distributed contention domain, which requires coordination (logically occurring at D) for traffic flowing in the $E \rightarrow D$ direction. This coordination is performed on-demand and takes a certain amount of time (as discussed in §B.6). Thus, while for the ADSL LLU, $\Delta Q_{V}^{E \rightarrow D}$ is effectively zero, for DOCSIS it can be several milliseconds. This difference may be irrelevant for bulk data transfer¹³, but significant for other applications such as interactive online gaming¹⁴. Even though the service time for a given packet ($\Delta Q_{S}^{E \rightarrow D}$) may be three times larger for ADSL than for DOCSIS (i.e. cable can provide higher uplink speeds), the $\Delta Q_{V}^{E \rightarrow D}$ on DOCSIS can be a factor of 5-10 times greater, thus dominating the ΔQ budget in this direction.

Given that the influence of ΔQ on QoE is over the entire end-to-end path, there is a further complication to address. Although Figure C.1 on page 79 describes the UK side, protocol peers are not just in the UK (as illustrated by Figure C.2 on page 81). An application cannot determine whether excessive ΔQ is accruing in the UK or elsewhere. Many CDNs (even international ones) have UK-based servers to reduce RTTs, but the performance of (and contention on) links around the world still has a bearing on UK user application outcomes.

C.2.1. Specialised services

A ‘specialised service’ is one that is delivered along a path that does not terminate in the general Internet. It is commercially attractive to run both specialised services and general Internet provision over a common infrastructure; this is especially true for the delivery of such services to consumers, where “last-mile” costs tend to dominate.

Given that the last mile typically has the greatest capacity limitation, there is a strong desire to benefit from statistical sharing. Static allocation of capacity to a specialised service would inevitably mean less available for general internet connectivity. Statistical sharing implies that there is a performance coupling between such services and general internet use. This coupling can have detrimental effects in both directions. Static allocation of capacity would mean that the consumer would experience the capacity penalty at all times, irrespective of whether the specialised service was in use or not.

Today’s Internet connection offerings are not sold with any lower bound on their effective performance. Specialised services are likely to be “value-added services” (i.e. those that attract a commercial premium). Many services that are currently offered as value-added (e.g. video conferencing) have stringent ΔQ requirements¹⁵.

Traffic management would be needed to create an appropriate level of performance isolation (and quality trading) required for reliable application outcomes. An early approximation of such TM is already present in the UK market¹⁶. Given that there is no contractual quality

¹¹Many application clients need to interact with a variety of different servers; this analysis applies to each interaction separately.

¹²This ΔQ will depend on the offered load at which traffic constraints bite; such constraints can be due to physical issues, such as maximum achievable data encoding rates, or due to explicit rate limitation.

¹³Bulk data transfer forms the basis for most consumer ‘speed test’ services.

¹⁴Interactive gaming represents a use-case for which small differences in ΔQ can have a large impact on the quality of the user experience. See, for example, <http://goo.gl/iqFCp1>.

¹⁵By this we mean that they have an upper bound on the ΔQ that they can tolerate while still delivering an effective service.

¹⁶E.g BT’s TV Connect service as described in SIN 511, see Appendix D. Other LLU providers will have had to make similar design and implementation choices; descriptions of their choices are not publicly available.

floor for the general Internet, the performance effect of the operation of such specialised services in the last mile is an area that may need investigation in further work.

C.3. Management domain interfaces in the UK

The UK is unusual in that, due to its market structure and regulation, several of the technical interfaces that would be internal in a vertically integrated ISP have become externalised. BT is required to publish “Supplier Information Notes” (SINs) that contain technical information needed to connect to the constituent services offered by BT OpenReach and BT Wholesale. They also make some qualitative (but not quantitative¹⁷) descriptions relating to traffic management. Thus it is possible to analyse such SINs to extract what statements (if any) they make regarding queuing/scheduling, traffic management, etc.. This is done in some detail in Appendix D.

The SINs themselves do not contain enough information to know the PROs of BT’s network elements (i.e. they lack the detailed technical parameters and behaviour descriptions needed to ensure reliable use of their services). However, they state (both explicitly and implicitly) that an ISP should use traffic management to avoid incorrect operation. In fact, they point out that failure to manage correctly certain control traffic could result in the loss of connection to one or even all of an ISP’s customers (as described in §B.2).

Such documents are interesting because they expose the interfaces between management domains and highlight the extent to which TM is essential to keep network systems within their PRO. They also reveal that existing specifications do not provide any means to predict or control the emergent ΔQ . They embody an implicit assumption that bandwidth is fungible, which is not the case in a PBSM context. Compounding this, services used in the delivery of consumer broadband rarely provide minimum bandwidth or delay guarantees, particularly in the upstream direction.

In conclusion, the analysis of the BT SINs shows the presence of translocation performance hazards. There is no reason to believe that other UK market providers do not have the same issues. These hazards are present both for network control traffic (affecting system stability) and end-user traffic (affecting application outcomes). Mitigating these hazards is one reason for the deployment of TM in the UK broadband infrastructure.

C.3.1. Potential points of TM application

While (as pointed out in Appendix B) every output port of a switch or router is a point at which packets may queue¹⁸, in practice most of these implement default FIFO behaviour. Points at which non-FIFO TM may be effectively applied correspond to points of ingress and egress between distinct management and administrative domains. Even without this, however, the limited rate of interfaces tends to shape traffic, resulting in the smoothing of bursts. At other points, where both the fan-in¹⁹ and total data rates are low, non-FIFO TM will deliver little benefit.

Let us consider an example of the path from an ISP to its customers via BT Wholesale. The use of non-FIFO TM makes sense²⁰ at the egress from the ISP to BT Wholesale (point C in Figure C.4). Assuming that this rate-limits the ingress streams to BT Wholesale, there is little to be gained by using non-FIFO TM in most of their network (the path C \rightarrow D in the

¹⁷BT does make more quantitative information available in commercially confidential ‘handbooks’. A flavour of the additional quantitative information can be found in an edition of the FTTC handbook that has become public <http://goo.gl/d6Y5q1>. Several other handbooks with “in commercial confidence” markings can be found on BT’s websites through a simple web search.

¹⁸Thus each such point is where some form of scheduling decision is made.

¹⁹Where the fan-in is the number of active input ports, or sources, destined for a particular output port, or destination, within a switch or router.

²⁰This is because: (a) the connection capacity is finite; and (b) charges are based on averaged peak bandwidth.

Figure would represent multiple paths in this case). However, as traffic from multiple ISPs is dispersed towards different BRASs (points D), there is a potential for correlation (due to demand from the corresponding sets of end-users in a particular geographical area). This creates a performance hazard; mitigating this would require non-FIFO TM to be applied by BT Wholesale.

Figure C.5 is an annotated version of Figure C.1 that illustrates locations where non-FIFO TM might be usefully applied in a UK wireline context²¹. Note especially the different coloured arrows that distinguish the level of aggregation at which TM might be applied.

It is important to consider what ‘positive detection’ of differential traffic management would mean in a UK context. Knowing that there may be differential management occurring somewhere along the path between an end-user and the internet does not identify which management / administrative domain it occurs in, which could be:

- before the ISP (even outside the UK);
- within the ISP;
- after the ISP;
- in a local network (depending on router settings).

Thus it is a challenge simply to determine whether the ‘cause’ is within the UK regulatory context.

C.4. Summary

In this Appendix, we have seen how the market structure of broadband in the UK creates quite complex delivery chains, particularly in the wireline case²². We have discussed how ΔQ (the observable performance characteristic of the delivery chain) accumulates along the path. We have explored the issue that effects due to remote parts of the network may not be distinguishable from those due to more local causes.

We have used the particular situation in the UK to analyse some aspects of management domain interfaces (explored in more detail in Appendix D), in particular the requirement for TM to be applied to keep sections of the network within their PRO. We have identified potential points at which non-FIFO TM might be usefully applied along the delivery chain, both for this reason and others.

An important point to reiterate is that, even if TMD could detect differential TM, the structure of the UK market (coupled with the current state of TMD research) prevents one from being able to ascribe responsibility to any single body²³.

²¹Establishing which forms of TM applied at these various points might be detectable would require a laboratory-based study, including an emulation of an appropriate subset of the UK broadband infrastructure.

²²Mobile and satellite providers remain, for the most part, vertically integrated.

²³As it stands, one could not even be sure such differential TM was being applied in the UK.

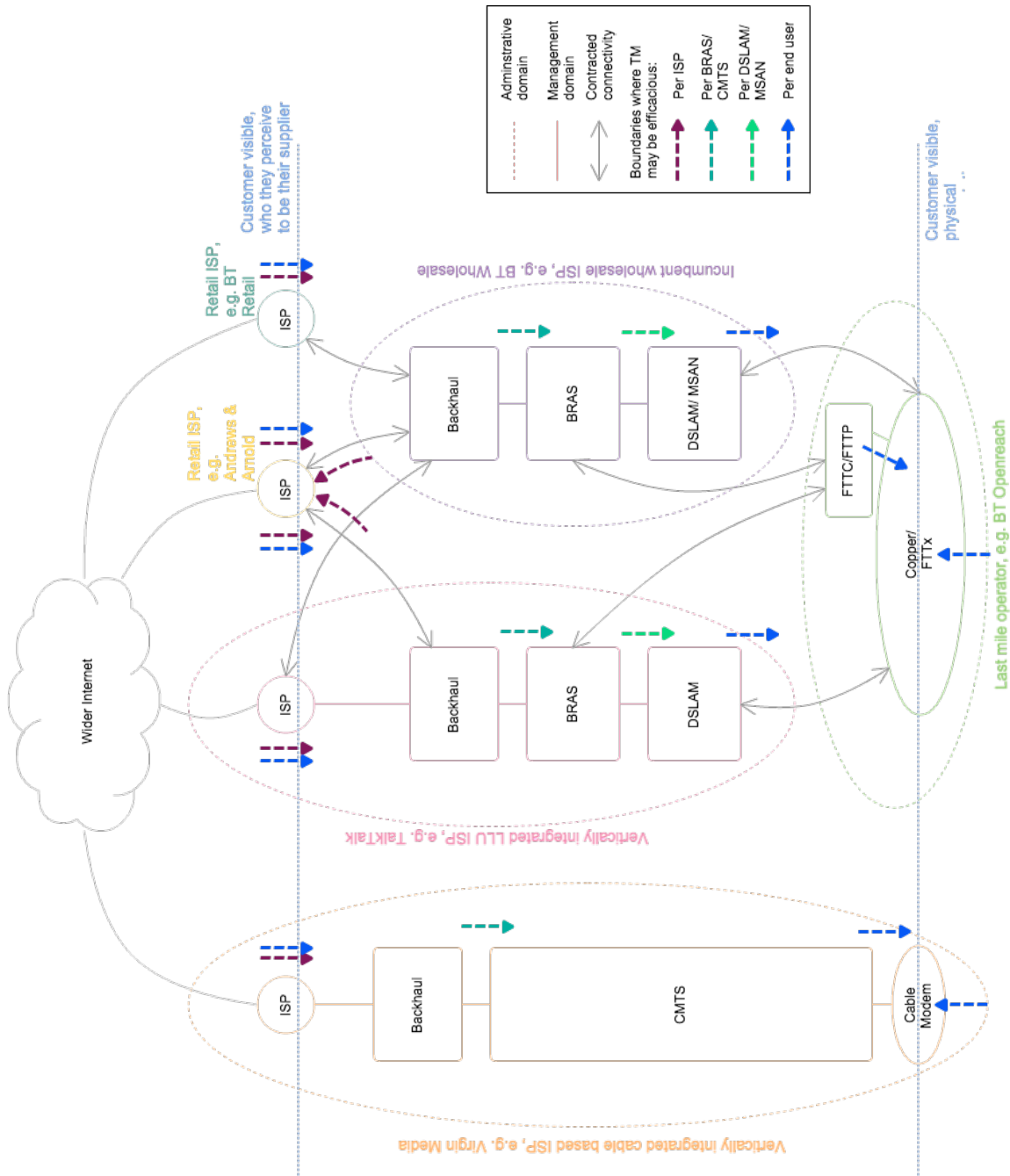


Figure C.5.: Potential TM points in the UK broadband infrastructure (wireline)

D. Analysis of BT SINS

The UK is unusual in that, due to its market structure, several of the interfaces that would be internal in a vertically integrated ISP have become externalised. Consequently, BT is required to publish “Supplier Information Notes” (SINs) that contain much of the technical information needed for other parties to connect to the constituent services offered by BT OpenReach and BT Wholesale. The SINs also make some qualitative (but not quantitative¹) descriptions relating to traffic management. Thus it is possible to analyse them to extract what statements (if any) they make regarding queuing/scheduling, traffic management, etc.. The currently available SINs have been analysed and the following extracted²:

SIN	Title	Version	Comments
472	BT Wholesale Broadband Connect (WBC) Products Service Description	472v2p6	There are interesting issues relating to the inter-path and inter-user effects of Content Connect ³ and TV connect ⁴ . The potential QoE effects resulting from these may be something bodies like Ofcom would be interested in, particularly as there is no discussion about quality isolation in this SIN.
498	Generic Ethernet Access Fibre to the Cabinet (GEA-FTTC) Service and Interface Description	498v5p1	In §2.1.5.1, BT’s interpretation of the priority code point of the ethernet frame is described. As a result of this interpretation, there are effectively only two drop-precedence levels in their system. These levels are used for both intra- and inter-consumer $\Delta Q _V$ management. The GEA product maintenance traffic “has priority” over the end-user traffic and the multicast offering is in the highest available priority level, both of these facts present a performance hazard. There is insufficient information provided within this SIN for it to be possible to know whether any observed ΔQ is within design limits. In addition, those design limits are not made fully clear in this SIN.

¹BT does make more quantitative information available in commercially confidential “handbooks”. A flavour of the additional quantitative information can be found in an edition of the FTTC handbook that has become public <http://goo.gl/d6Y5q1>. Several other handbooks with “in commercial confidence” markings can be found on BT’s websites through a simple web search.

²Note that all references to sections in this Appendix are to sections of the corresponding BT SIN, not sections of this document.

³Content Connect is BT’s CDN service.

⁴TV connect is BT’s CDN for TV streaming video.

SIN	Title	Version	Comments
385	IP Connect UK Service Description	385v2p8	Although there is reference to IP Precedence marking and DSCP marking there is no description of what methods might be used (this SIN contains less detail in this respect than SIN 498). There is, however, an acknowledgement of the effect of fragment size on delay (particularly for low-speed frame-relay-based access services). The capacity made available as SDU carriage is not articulated, only PDU costs are quoted (and even here there is still ambiguity).
471	BT Wholesale Broadband Managed Connect Shared Service Description	471v3p3	This SIN acknowledges that the bandwidth measurements used are taken at layer 2, and include all protocol overheads. It also states that this is the bandwidth billing metric. Policing occurs at 110% of purchased load. Inaccurate packet marking leads to loss. If the wholesale customer (e.g. an ISP) does not use TM to prevent this policing, there is a service stability hazard.
492	Ethernet Access Direct (EAD) inc. EAD Enable and Ethernet Access Direct Local Access Service & Interface Description	492v1p8	SyncE effects are outlined, thus giving some indication of resulting ΔQ effects. There is a focus on loss notification but no other TM consequences.
495	BT Wholesale Broadband Connect Fibre to the Cabinet Service	495v1p1	PPP is used to support BRAS profile (sync rate) information exchange. This requires significantly lower PPP/L2TP time-outs (<20s), increasing the overall baseline end-to-end cost of this service.
503	Generic Ethernet Access Multicast Service & Interface Description	503v1p2	This SIN asserts that multicast is carried in a separate VLAN (§3.1). Multicast is assigned a relatively high urgency – 'level 3' (§3.1.5). The CP has the responsibility to shape and manage the effects of the multicast traffic delivered on the rest of the traffic (even though its traffic pattern may not be visible) - §3.1.6 & §3.1.6.1. There is no policing (at present) on the FTTP – this may represent an inter-end-user performance coupling hazard. (§3.1.6.2).

SIN	Title	Version	Comments
506	Fibre to the Premises (FTTP) Generic Ethernet Access Service and Interface Description	506v1p2	In §2.1.5.1, the SIN describes the “prioritisation” mapping. This is a single marking that embodies both ‘urgency’ (the requirement for low latency) and ‘cherish’ (the requirement for low loss rate). When expressing urgency, 4 is the most urgent and 0 is the least. To achieve this 7, 6, and 5 are remarked to 4. When expressing cherish, there are only 2 levels - 0 or any other value. This two-level cherish marking (§2.1.5.1.1) is used to resolve inter-end-user resource contention. The wholesale customer needs to keep traffic with non- zero marking {7,6,5,4,3,2,1} within a “prioritised rate”. This rate is product dependent. The upstream is managed using strict priority queueing (§2.3.6), with 4 urgency classes {{6,7}, {4,5}, {2,3}, {0,1}}. This particular management only occurs within the CPE, the rest of the path has no explicit traffic management (§2.3.5).
509	BT Wholesale Broadband Connect (WBC) Fibre to the Premise (FTTP) Service & Interface Description	509v1p2	This SIN covers purely interface, and other non-performance effecting, issues.
511	BT Wholesale TV Connect (TVC) Service & Interface Description	511v1p5	This SIN is a technical description of the service interface. The only explicit performance information contained in it is that the stream is a MPEG-2 single program transport stream (§5.2) with video bit rates of 2.5Mbit/s, 3.0Mbit/s, 7.5Mbit/s and 10Mbit/s, and audio bit rates of 128kbit/s or 224 kbit/s. In each case these figures represent minima, the real frame cost (i.e contention for the common resource) will be higher. TVC is expected to be carried over the multicast service (§6.3). There is a loose performance guarantee for the BRAS/IGMP query – general queries are made every 125s with a maximum response time of 10s. Specific queries can made at a rate of one every 10s, with a response time of 8s (§6.3).
482	BT IPstream Connect Service Description	482v1p13	This SIN is a service description - it describes how ‘speed’ can be measured as part of the diagnostic measures (§5.1), and the management domain boundaries (§3.1).

SIN	Title	Version	Comments
485	BT IPstream Connect Office, BT IPstream Connect Home, BT IPstream Connect Max & BT IPstream Connect Max Premium Products Service Description and Interface Specification	485v1p2	This SIN describes services in terms of their available sync rates (§4.1). It is worth noting, however, that all of the products in this specification are now legacy services.

D.1. Caveats relating to bandwidth measures

When interpreting the statements in these SINS (as in the vast majority of technical documentation in this area), it is important to note precisely where traffic management is being used, or where costs are being calculated/accrued on data flows. Measurement/management is based on *the size of the protocol data unit*⁵ at a specific point in the end-to end-path, including any protocol overheads. Thus there is no uniform way to compare “bandwidth” at one location (say the ISP ↔ Wholesaler boundary) with that at another (say the BRAS ↔ DSLAM boundary). The fungibility of bandwidth in a circuit-switched environment such as TDM is not present in broadband.

As the size of any PDU is likely to change many times along the end-to-end path⁶, the actual user data rate delivered by a reported “bandwidth” can vary substantially⁷. One way BT Wholesale addresses this fact is by distinguishing between “SYNC” rate (the rate at which signalling is occurring over the final connection to the premise) and “BRAS” rate (the rate at which traffic is shaped in order to avoid queuing in the DSLAM).

⁵A protocol data unit (PDU) is the composite of the protocol headers and the service data unit (SDU). The data relating to an application is within the SDU, but there may be several additional intermediate protocol layers.

⁶This size change is because a user IP packet is encapsulated/de-encapsulated by various transport technologies as it traverses its path.

⁷The most extreme case the authors have encountered was in a ADSL-based VoIP system where (through a configuration choice) voice packets, by only one octet, occupied two ATM cells rather than one. This meant a 96% overhead, and thus halved the effective capacity of the system.

E. Additional Literature

There is a further body of relevant literature, as represented by the citation graph in figure E.1. This was discovered by: making an initial survey; referencing all papers cited by the initial cohort; and then searching for further papers citing those.

Interesting material includes [33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 32, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 21, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110].

The techniques studied in detail in §2 are marked with square nodes.

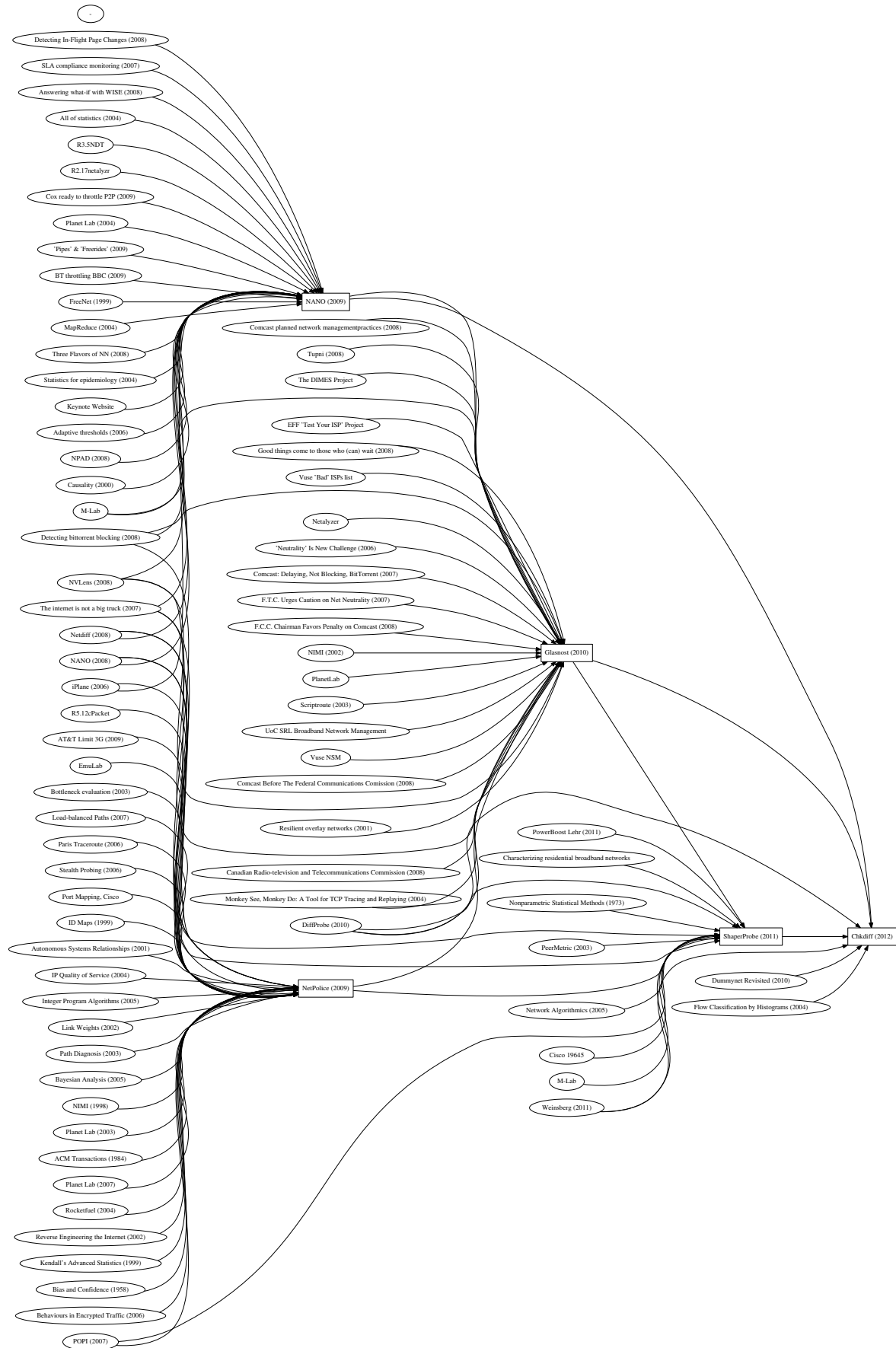


Figure E.1.: Citation relationship between relevant papers