Deep Science Publishing
https://doi.org/10.70593/978-93-49910-26-3_9

DeepScience
Open Access Books

Chapter 9: Applying predictive algorithms to monitor, maintain, and evolve OSS network performance efficiently

9.1. Introduction

Increasingly pervasive Access Networks have become known as Optical Subscriber Services (OSS). Distributing high-bandwidth Internet Protocol-enabled services across all market segments is seen as essential in making the necessary returns on investment over large service areas. Delivering OSS infrastructure and services must occur within a competitive economic environment with levels of sensitivity unheard of in traditional telecommunications practice. Telecom and cable corporations are investing heavily to migrate to an OSS infrastructure for broadband data services, and the subscriber service continues to steadily evolve along with the challenges for both service providers and equipment vendors as new layers of performance and services are built up above basic functions of capacity, scheduling, and supporting interfaces with the user. OSS will support new classes of performance-sensitive multicasting services, in particular, video-on-demand over Multicast Point-to-Point Protocol. Bandwidth fluctuations seen in real-time traffic from media sources must be accommodated by the underlying infrastructure that OSS provides. New enabler architectures must predictively manage OSS bandwidth performance or face regulatory scrutiny relative to new local competition guidelines. Existing technologies that have been used to manage First Generation Broadband-in-Broadcast networks do not scale appropriately to a large number of individual subscribers (Akella et al., 2003; Guo et al., 2019; Miao et al., 2021).

The objectives of a new class of predictive algorithm, such as the bandwidth performance management capabilities of Link Utilization Allocation, must meet or exceed run-time performance, sensitivity, and scalability guidelines that have been developed. For an ever-increasing number of individual bandwidths in Distribution Hierarchies, the difficulties will be demand-sensitive, nonlinear network utilization. The proposed new scalable enabler architectures for telecommunications OSS networks examine the relationship between demand sensitivity and performance, and the

simulation techniques are developed based on queuing networks, which validate capability (Zhang et al., 2020; Singh et al., 2022).

9.1.1. Overview of the Document's Focus and Objectives

Dramatic growth in the communications industry has given rise to an ever-increasing demand for higher bandwidth transmission capabilities from the end-users, together with an equally continued requirement for improved reliability and reduced costs throughout the whole system. To fulfil these escalated user requirements, the Optical Super Highway concept has been proposed to implement Long-Haul Optoelectronic Interconnects having Distances in the Meters-to-1000s Range and Data-Rate Capacities in the Terabits-Per-Second Range to be utilized to reconstruct a vast majority of the interconnections between circuits on Circuit Boards. This optical solution employs parallel single-mode fiber optic technology and massive Signal Wavelength Division Multiplexing Transmission Technology together with innovative optoelectronic devices having Vertical-Cavity Surface Emitting Lasers and Ultra-Thin Semiconductor Optical Amplifiers. Designed to be included inside numbers of the local interconnections between consumer Products and Remote-Purchasers' Transport Centers as part of the OSS, these devices are based on innovative fiber-grating chip coolers arranged in special arrangements to allow the original massive fan-in and fan-out connections together with specially designed threaded-lens Fiber Optic Couplers to achieve madly ultrafast signal rate optical interconnections. Predictable failures and performance degradation are common for both the core optical transport networks and the access networks. Tools supporting fault localization, diagnosis, and prediction are needed to keep the carrier networks running smoothly, maintain minimal downtime, and guarantee the performance and quality of service requirements. Solutions targeting prediction would be able to accomplish maintenance proactively versus the reactive, and thus much longer, duration of detection and repair resolutions for real faults that have been already introduced into the networks. Compared to the existing solutions for either network traffic or fault monitoring, the granularity of our measurements, which come from carrier core routers at the gigabit level and that of a subset of the active components, has also significant advantages. It would allow us to collect deep network state data to support accurate machine learning-based prediction algorithms.

9.2. Understanding OSS Networks

OSS Networks are systems of equipment that work together to deliver telecommunications services or products, such as the capability to send and receive telephone calls, utilize a long distance or local toll service, or exchange voice, fax or data

for, through, to, or from a location in the telecommunications network. The pieces of equipment can be made from a variety of manufacturers, with different pieces of equipment comprising their own subsystem, but together they interconnect and cooperate to provide the telecommunications service. Other systems can be used to describe these and/or larger groups of equipment, such as transport systems and service systems. Service systems describe the influence of such things as converters or protocol translations. Transport systems describe the influence of wide bandwidth electronic or optical devices, such as multiplexers, either time or wave division, or cross connect switches.

Fig 9 . 1 : OSS Networks

The actual services provided can be TDM switched services and ATM or IP routed services, and/or a support of such things as voice, fax, data or video calling for the telecommunications user. The equipment combinations can encompass a variety of media, example twisted pair, coaxial cables, copper and/or optical fiber conduits, microwave towers, loanings, etc., and a distribution of density, locally having short distances to terminals and long distances to repeatered central offices, to be interconnected. The differences in scope and scale not only qualify the utility of terminology, but require industry monitoring of system performance for proper basic research.

9.2.1. Definition and Scope

The great advances in Information and Communication Technologies, such as the growth of both networks and services, their diversity, distribution, and increasing interdependence, have made users more dependent on the normal working of the services provided by these Networks and other subtle and hidden mechanisms. Providing high-quality and high-performance services is a deeply rooted concern among the communications community. However, the actual evolution and near future of these Networks is more and more biased towards self-configuration, self-optimization, and self-healing mechanisms, which are responsible for detecting and performing the necessary compensate actions, such as rerouting the traffic towards alternate paths to avoid congestion. The work in this chapter is aimed at these specific performance enhancements in Optical Switched Networks.

These Networks are at a mature and decisive stage of their actual deployment, and many real concerns still remain open. Traffic profiles have shown, over the last years, a variable behavior that shows also a peak in its use, which has caused the appearance of congestion problems, such as packet drops and reported delays. Optical Switching Networks are one of the possible solutions proposed to improve the behavior of the traditional electronic packet-switched networks. These specific networks are based on the use of optical-label-switching techniques, which avoid the transformation of the packet signal from the optical domain into the electrical one, and are able to switch packets at the optical level. The importance of Packet Optical Label Switching technology is now being re-evaluated, in connection with the development of Optical Burst Switching Networks and the configuration of packet flows in Multi-Protocol Label Switching. The use of Optical Buffers Network technology can also be used without any packet transformation between the two domains, thus supporting packets in one domain and using their inner structure to route them in the other domain. The presentation of a new topological structure for the implementation of a POLLS Network is also in the background of this chapter.

9.2.2. Importance of Performance Monitoring

A significant amount of empirical networking studies has signaled how control loops for self-configuration and self-healing within the OSS system are merely automation mechanisms to configure as run current operations. They do not offer an efficient tool to optimize the way Internet service providers (ISPs) provide Internet services are available all around the globe at affordable prices. These studies, in addition, focused on diagnosing short-term performance trend changes, and only recently long-term trends have been investigated. Trend prediction, proposed two decades ago, needs to be done: (i) with periodic sampling of affected technology areas; (ii) by means of statistical

graphics such as Scatter Diagrams; and (iii) with automatic calculation of simple statistics which today's huge computational powers allow.

Conversely, the continuous instruments, although recently extended to document technical debt information, are based on operating system (OS)-based file size sampling of affected technology areas and need to be complemented with hot-spotting, root-cause analysis, profiling, stack analysis, and canary-based performance comparison. These two classes of instruments are collected by separate groups of engineers, within disparate life-cycles. Operating regarding OSS network technology area monitoring, this means that communication carrying meta-pertinent information about queried parameter value is carried out in one direction only. Being able to propose an automated statistically sound algorithmic pattern able to discover performance anomalies allows optimization of alert generation: Configuration-based alerts are identified by the distributed intelligence query format and are sent to the appropriate ops team.

9.3. Predictive Algorithms Overview

Predictive algorithms work with a set of operational data to identify relationships and trends. They then evaluate these models against historical use of the technology. In the context of OSS data analysis, predictive algorithms are used to detect a to-beyond threshold based on some measure such as bandwidth, CPU, loss, or uptime. These tools create the predictive outcome event from the historical use of the technology, usually for several months to a year, and use them to identify a probability of a to-beyond event occurring. This predictive outcome is then further processed by built-in event correlation engines to not only reduce the number of events but to create the context of those events. This context allows the staff to focus on the higher priority events while further exploring situations for lesser priority technologies. The predictive to-beyond events help the operations centers anticipate these troublesome situations and resolve them before they become serious outages, as well as reduce the number of recurring outages.

There are several types of predictive algorithms. Statistical algorithms estimate the probability of future activity based on what has happened to previously available data and require historical data to predict the expected future activity. These algorithms include regression analysis, autoregressive integrated moving average models, and smoothing techniques. While widely used and easy to create and interpret, statistical algorithms can only predict short time frames and do not work well with multiple inputs. Methods based on classification trees and neural networks are also available. These methods are more complex to create and require much more processing and cross-validation but support the prediction of larger time frames and are more flexible regarding input. Consumers who have been the most successful in using predictive

algorithms are those whose business is heavily reliant on standardized data and transactions.

9.3.1. Types of Predictive Algorithms

Algorithm is traditionally defined as a finite, clearly specified sequence of operations to solve a problem or achieve a certain goal. Algorithms may be classified in many ways, and the classification may vary according to different criteria. Algorithms may be classified by their design or computational approach, or by solving different types of problems. Based on the design approach, an algorithm may be classified as a sequential, branch and bound, divide and conquer, dynamic programming, greedy, or randomized algorithm. If we classify algorithms based on the types of problems that they solve, the categories may be control algorithms, combinatorial algorithms, data compression algorithms, encryption algorithms, etc.

There are different characteristics and criteria to classify predictive algorithms. In essence, predictive algorithms are divided into two main categories: parametric and non-parametric predictive algorithms. A predictive algorithm is a solution of estimating an unseen value of a random variable given observations. In principle, this estimation can be done using statistics and statistics based algorithms. These algorithms are the traditional, most known predictive algorithms and they basically rely on parameters. These algorithms postulate a statistical model for the random variable to be predicted, which contains a small number of parameters. These predictions are simple functions of previously known examples plus some statistical inference regarding the parameters of the model. Examples of such algorithms include linear regression, parametric density estimation, Kalmar filtering, hidden Markov models, etc.

9.3.2. Applications in Network Performance

The combination of predictive algorithms to optimization problems in network performance have numerous applications. Networks must cope with changing traffic patterns, but at increasing levels of performance, reliability and security. Networks change over time, both in terms of topology and in terms of the demands placed upon network resources. The changes in demand can be both predictable, in terms of temporal patterns that could be forecasted, and random, in terms of short bursts of irregular behavior that could be modeled with predictive probabilities. A primary method for responding to changes in network demands is to route traffic to the expected least-congested paths, and routing is generally performed in a static mode. Thus routing tables, which map each destination address to the corresponding next hop node to which packets must be sent, are set and remain fixed over some length of time. However, demanding

applications and unpredictable user behavior have made it difficult to specify these time periods, leading to a growing interest in dynamic routing. And although it is true that dynamic routing incurs significant network overhead that leads to new problems, it may be best not to route often, but rather to let routes remain fixed over a majority of traffic, with infrequent changes to combat increased user activity.

Routing is, of course, not the only method for accommodating temporal changes in the utilization of network resources. Many integrated services provide guarantees of performance that include bounds on packet delay, loss, and delay jitter. These bounds may be defined in terms of the likelihood of experiencing more than the allowed amount of delay. When a given flow approaches its worst-case delay, the arrival of additional packets from that flow must be curtailed. Such constraint-based approaches to traffic management avoid the complexities of global routing decisions by dealing with the behavior of all traffic on a component basis. For virtually all applications, it is better to have delays remain small, so that user dissatisfaction is minimized; furthermore, by treating maximizing network utilization as the objective while delaying probabilities remain small, the need for rate limiting on individual flows can easily be relaxed.

9.4. Data Collection and Analysis

This chapter addresses the data collection and analysis techniques used in the models presented, more specifically, in the CTR-based model called CTP-OSS and in the TCF-based model called SPF-OSS. We need to track two types of information to model the network performance: the queries the user sends to services and the time response provided by services to process user queries. To build our model, we needed data for various user-spider time interaction patterns under different network and traffic conditions in the recipe listed below. The interaction patterns were built over time using an API. The network conditions were modified by periodically varying the nozzle bandwidth, and the traffic conditions were modified by transiting the architecture features from optimized to overlaid DNS and network performance absence. We collected data for three application types: search and generic adoption type, Java Applet based commercial application of less than 50Kbytes that uses a commercial service and requires about ten seconds on average to respond and would be generally object of at least one request a day during the whole expected winter solstice episode, and image request with a recovery time of about five minutes during the episode.

Unhappily, the data collection process is not as simple as described in the recipe. The most important point concerns the expected service recovery time for standard applications, including the required bandwidth patterns for images, documents or service priorities. These limits condition the probability of at least making one request to it per request-pairing period. In our case, we also had to take care about the constraints of the

query verbosity. In practical terms, the configuration described is a complementary adjunct for the Web. In other words, whenever predictors suggest at least a fluctuating service demand in the usual queries, they trigger actions to change the bandwidth knobs of the previously configured bandwidth conditioner-tracer. The novelty in the process, which also applies for all database traffic, is the pairing the response and pairing queries prior processing times.

9.4.1. Data Sources for OSS Networks

Data sources in OSS networks application domains are typically classified as administrative and operational data. Administrative data in OSS networks correspond to the management of various resources. Data sets supporting engineering, maintenance, traffic management, service quality, billing, customer, network security, interconnect, and mobile virtual network operator repouring function have been defined. Administrative data are populated by different sources, having different characteristics, and requiring different collection techniques and periodicities. Adverse effects of incorrect hardware and software configurations on the performance of networking elements are well-known. Accounting, configuration, and audit management OSS operations collect and store information on supported services, network elements, customers, installed software, versions of updates, and availability of hardware resources. NMS and PM processes also collect data on the state of network elements and services.

Information useful for the development of incident and problem management tools is contained in trouble ticket data. Trouble ticket systems track and help manage customer complaints problems until they are resolved. Data required for billing processes come from different sources: either from within the telecommunications network and reported by network monitoring systems, or external, received by processing the customer hard copy and electronic media and care and customer feedback processes. Billing data processing allows the identification of failure patterns and problem areas that, introduced to NMS and PM solutions, lead to tool improvement. Quality control data are generated by the operation of different systems. Data regarding service performance come from customer care representatives, sales personnel, workflows, focus groups, customer view phones, and logging of unsatisfied user experience. Data logged also contain subjective measurements of voice quality and data throughput, and objective measurements of call completion ratio and packet error rate.

9.4.2. Techniques for Data Analysis

In this section, we will discuss how to identify which parts of an OSS headset's network affect performance, in addition to which data parameters are affected by performance degradation, using a number of data analysis techniques, which can be broadly classified into two types: correlation and regression. The latter is usually a more powerful method, especially in the case of nonlinear dynamic systems. In this case, we have more variables of interest than are present in the data. So, this technique is suited for systems that do not go through long nonoperational periods, in order to learn the underlying joint probability density model. Modeling programs can be used for the regression task, where the architecture is selected based on the amount of training data and error rate on the test data, as well as the speed of convergence.

Most OSS performance analysis needs to conduct these minimum analyses before making further conclusions regarding root-cause or effect. This section describes only a few such algorithms that can be implemented on the data presented in the previous chapter. Data from the following fields can sometimes be correlated to system performance: Post-connection Setup Delay, Pre-RTC Connection Making Delay, RTC Completion Time, Post RTC-connection Changing Delay, Core-Network Connection Making Delay, Connection Releasing Delay, RTC Modem Crashed, Traffic Uplink Delay, Traffic Downlink Delay, Traffic Dual Link Delay, Traffic Data Rate and Core-Network Connection Making Status. Note that in the correlation example, we used the shortest Uplink Delay over the 6 hr POST downtime from rnch and correlated it to Long Term Diff Serv Control Uplink and Downlink Accumulated Deferred Send from perf.

9.5. Implementation of Predictive Algorithms

In developing a predictive algorithm, care must be taken to follow the path already examined in the schema of algorithm selection. The success of any predictive algorithm in addressing the three problems in the area of performance enhancement through predictive capability is directly related to its knowledge acquisition component. There have been advantages and disadvantages offered for different forms of intelligent control in the specific areas of learning, representation, and reasoning. These serve as appropriate criteria for understanding the proposed predictive algorithms for monitoring and control and to assist in making a selection of an algorithm for a specific implementation of a predictive monitoring and control process. Performance suggestions without ensuring accuracy in intelligent decision-making wield very little strength in optimizing decision-making capability. In this section, we address the parameters against which decision-making algorithms can be addressed.

In selecting an algorithm, it is important to recognize what trading of capabilities in each of these components are receded by the choice of a particular algorithm. Also relevant, is how previously acquired knowledge is utilized through an intelligent selection of parameters unique to that domain or physical region in a chosen algorithm. Learning is a key part of intelligence and an algorithm may not learn in an appropriate way, or it may fail to learn at all. Nevertheless, this is the stage at which we set up the predictive algorithm system. The predictive algorithm is designed to be implemented on top of existing predictive algorithms. In this research, we are concerned with developing algorithms that are general to many sites and systems. Thus, the algorithms may need supervision when first applied to a new site. We propose algorithm wrappers that can be inserted in front of existing foresight and optimization algorithms.

9.5.1. Algorithm Selection Criteria

There are three sets of criteria with which we must assess potentially useful learning algorithms: efficiency, domain competence, and algorithm balance. When working with a domain-dependent search control knowledge system, one of the goals is to avoid algorithm inefficiency during search space match. The fact that the system builds problem-solving, search-control and competence knowledge during each solving episode allows it to retain efficiency for the problems encountered frequently enough. However model reuse implies selecting among available algorithms during each selection occasion. An inevitable consequence of multi-domain algorithm selection is a certain comparative level of inefficiency as all domains are not modeled in the same way. It is thus necessary, and possible, to reduce this selection overhead by retaining only algorithms that are efficient for a given domain within some specified range.

Algorithm domain competence refers to the capability of an algorithm to deliver efficient solutions for specific types of problems within a given domain. Various issues, related to the types of solutions expected, the nature of the problems to be solved, the characteristics of the contexts in which problems arise, and the frequency of such problems within a given domain adversely affect search-control algorithm competence for that domain. Algorithm competence mining attempts to relieve the necessity for an algorithm to be globally competent at all times. Competence mining allows control knowledge to be attached to domains, through expert opinions and experience-based on similar problems, and to specific thinking episodes with details on problem characteristics. It thus allows algorithm competence particularly when there is a large pool of candidate algorithms to be turned into limited per-domain experts. Without such competence mining, an indisputable requirement for algorithms used in a multi-domain selection architecture to be globally efficient is liable to make the process inefficient.

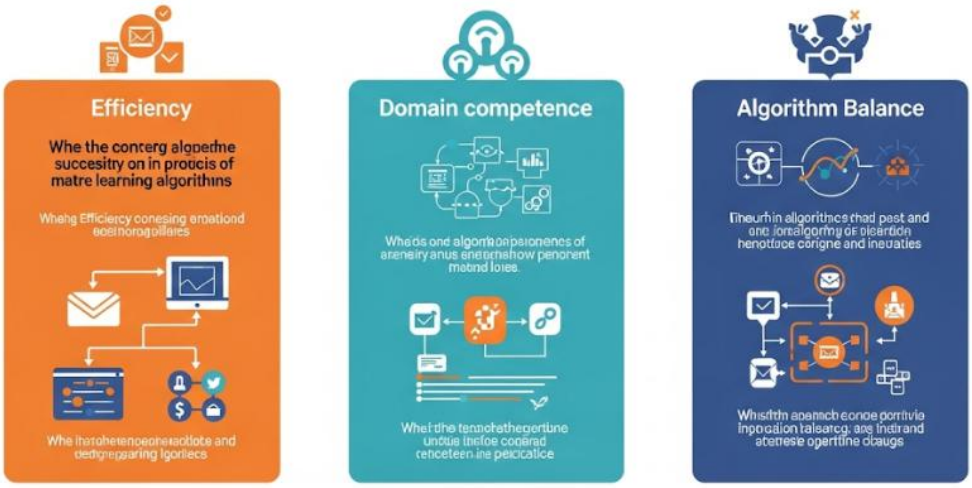

Fig 9 . 2 : Learning Algorithms

9.5.2. Integration with Existing Systems

Optimization of telecommunications network performance through predictive algorithms has mainly two limitations, data requirements for training models and execution time that can exceed real time. Even with these limitations, OSS systems are designed to exploit models that are trained and deployed in prediction servers. Models can be trained using historical data stored by the operations support systems. In many cases, active alarms and service disruptions can be easily mapped to faults analyzed by engineering teams using various methods. Typically, these methods include using experience-based rules, using simulation software, or using sorting tools. Predictions are essential inputs into decision making and resource management, especially when decisions have to be made on a proactive basis in order to anticipate faults before they affect the network. As a rule of thumb, more sophisticated and expensive models are generally used on a smaller number of complex and critical network elements.

The reaction to a prediction made by an algorithm will be driven by rules and procedures set up by engineering teams for each network element type using predictions from the algorithm. Prioritizing of predictions will also be done by heuristics. Telecommunication networks can be predicted to change state. The time duration, the fault condition, credit loss impact, and the importance of the risk in overall network management will determine if the prediction will be addressed and when. This is particularly important when the cost of remediation is high or loss of revenue is probable in the short term.

9.6. Monitoring Network Performance

In the dynamic landscape of telecommunications networks, the effective operation and swift resolution of issues related to service quality and traffic handling is made possible only through constant performance monitoring. Various parameter measurements are continuously relayed to Network Operations Centers, where they are evaluated against established alarms, thresholds, or compared with predicted behaviors. When necessary, NOCs trigger field events to fix any issues. Network management can predict traffic intensity on links by employing simple algorithms, such as smoothing, average deviation from average, or more complex algorithms, such as short and long term smoothing or statistical validation.

Key Performance Indicators have been defined by the industry and need continuous monitoring. For these KPIs, composite alarms based on sensitivity, actionable alarm time frames, and risk concentration are considered critical for concentrating network operations resources on crucial network areas. Novel composite alarms based on sensitivity can maximize network operator return on investment, but require commercial agreement with network vendors regarding exposure of proprietary algorithm details for composite alarm definition – or alternatively, fully cooperative vendor-neutral network performance monitoring companies to define a proprietary KPIs – Composite Alarm Matrix map that can enable joint multi-vendor composite alarms with few cross-domain false alarms.

Traffic and service performance are impacted by network events such as link outages and change service level impact. There has been ongoing research in real time monitoring of traffic and service performance using network component parameters. Traffic, packet loss, and packet delay are monitored using a system of snoopers embedded in routers offering constant feedback to network operations regarding network performance status. In a deployment, dedicated snooper services become inexpensive because snooper operations are free of charge and widely distributed.

9.6.1. Key Performance Indicators (KPIs)

In simple terms, KPIs tell us how a business is performing. They are measurable values that demonstrate how effectively a company is achieving its key business objectives. Organizations use KPI at multiple levels to evaluate their success at reaching targets and achieving deliverables. High-level KPI may focus on the overall performance of the organization, while low-level KPI may focus on processes in departments such as sales, marketing, HR, support, and others.

KPIs are essential in network performance management and measurement. They are critical in ensuring that the objectives of organizations are met and that any shortcomings

can be corrected to achieve the required results. KPIs are the physical metrics of performance measurement, monitoring, and evaluation. By their very nature, they can be checked repeatedly over time to identify deviations and trends. Further, they help a network manager to confirm an assumption about a performance issue, facilitate diagnosis, and validate or discard an alternative hypothesis. But not all metrics are necessarily good indicators of performance — a good KPI must be carefully selected as it determines success or failure. A performance indicator provides a good indication of success if it shows a direct and reliable link between performance and achievement of the desired goal or standard. Using the wrong metric can lead to wasted resources, incorrect conclusions, and even dangerous decision-making.

Many have formalized lists of essential network KPIs. These KPIs span operations and management activities including capacity management; configuration management; fault management; service level management; and security management. Of course, since networks are much more complex, a complete list of KPIs, in detail, is still a work in progress. Nonetheless, a limited number of KPIs that cover the most important areas of the network and those areas that impact the bottom line are necessary. Most reports of KPI use only briefly mention the specific types of KPIs that may be valuable in the area of interest.

9.6.2. Real-time Monitoring Techniques

Typical passive performance monitoring approaches rely on the continuous snooping on and inspection of certain packets to make inference about the overall or individual user performance. This procedure-based approach thus requires only the packet header information, which incurs very little per-packet overhead. An active performance monitoring system sends fixed bit rate test packets, and measures the time-to-forward the packets in each direction, as well as the time it took for the destination to receive the packet, packet loss and jitter. These measured performance parameters can then be compared to the required Quality of Service specification. However, this procedure incurs non-trivial packet overhead, which may change the traffic characteristics and thus may also distort the results.

The second approach, Packet Pair, which entails measuring the characteristics of packet arrival times. If the arriving packets are part of the same flow, the spacing between the arrival times is the packet spacing. If the adjacent packets belong to different flows, the spacing is equal to the minimum of two spacing values for both flows. The Packet Pair correlation technique takes advantage of the fact that the timing characteristics vary considerably for the packets that belong to the same flow as opposed to adjacent packets that belong to different flows. The technique provides a reliable detection rate for large packet sizes and longer flows. In addition, the Packet Pair correlation technique also

successfully detects ON/OFF traffic even in the presence of bias. As such, the Packet Pairing technique provides a powerful approach to distinguishing packet characteristics. In the absence of a suitable procedure for measuring the large traffic parameters continuously, available bandwidth measurement at selected intervals may be required.

9.7. Maintenance Strategies

In this section, we discuss maintenance strategies, beginning with an overview of how proactive and reactive maintenance policies differ. Then we discuss automated solutions, exploiting self-healing and data-driven models to perform maintenance autonomously.

1. Proactive vs Reactive Maintenance

Maintenance functions are typically performed according to two strategies: reactive maintenance returns the system to an operational state only when a performance limitation has been exceeded. In this case, it is assumed that a malfunctioning condition will be detected and notified by monitoring agents and tools, or inferred by humans through careful examination of management data. In contrast, proactive maintenance takes certain actions regularly, or based on models of normal operation, or predictions of resource availability, to avert future problems. Proactive maintenance policies are of two types: those which impose an additional short suspension in the regular workflow of the system and thus are called intrusive, and those that don't and thus are called non-intrusive.

Non-intrusive proactive functions thus have great advantages if they can be implemented properly. To help this implementation, predictive models need be as reliable as possible. By far the most common way to achieve a better prediction performance, this is to rely on more data. However, in many cases, data colonization becomes a problem. Data colonization is often more severe for proactive models, since for these models a larger completeness when hired is more desirable (more events at places where predictions are required). In general, both predictive algorithms and monitoring options are an active area of research. The knowledge procedures built around performance predictive models can be potentially much more efficient than conventional methods based exclusively on business-rule specifications. However, for this efficiency to be achieved, the predictive power of the model needs to be very high. This makes reliability of the predictive models even more crucial in proactive applications than in reactive cases, where false positives always generate some costs, but often much less than a sharp error in estimating bad performance.

9.7.1. Proactive vs Reactive Maintenance

Traditionally, telecommunications networks have been designed to remember faults and to correct them when they occur. This is the reactive maintenance model. The argument is that end-users should expect a reliable service and that all the operator has to do is to fix problems quickly. In principle this is indeed how telecoms networks were designed - to be resilient and to recover quickly. This model is however expensive. First, the repair work is expensive. Two, the fact that the networks need repairs means down-time. Service is interrupted while repairs are made. Although an operator may claim to recover low mean-times-to-repair, they do not tell customers how frequent such repairs are and how long services are lost whilst operators work away to fix problems. In short, the customers may have to endure long losses of service waiting for problem to be fixed. Frustrating. Given the fierce competition among operators, they must provide a suitable level of service and frighten consumer complaints. Particularly as many services are now mission critical.

A better model can help all sides, customers and operators alike. Not only is the fact that something has gone wrong predicted, but a solution is applied before the customer actually becomes aware of a problem. Service outages are predicted ahead of time to allow suitable solutions to be found. This is the proactive maintenance model. Operators have been using some limited forms of proactive maintenance for some time but only in a limited way. Why not try to make best possible use of the multitudes of fault data that is generated? Quality of Service has tended to be measured on the basis of repair times and response to complaints while actual outages are occurring. Although a customer-friendly approach to service is admirable, efforts should also be going into ensuring that outages and faults are kept to the lowest number possible through suitable foresight and maintenance.

9.7.2. Automated Maintenance Solutions

Software maintenance is an integrated part of software engineering and one of the major items on the design quality managers' agenda, and both user and public authorities count on it to uphold reliable, safe, and secure services, especially in critical infrastructures like utilities, banks, and aeronautics. Traditionally, network operators are assisted in maintaining, repairing, and modifying their systems by software tools. Still, there is a growing demand, also from users, for more automated solutions to detect and remove faults, security threats, and vulnerabilities, to identify system aging and to trigger software system refurbishment, repair, and even replacement. The road towards this additional degree of automation is slowly paved by the growth of artificial intelligence systems and the overall appearance of a more commonplace state of art of AI applications, like expert systems able to provide corporate consultancy.

New techniques for software system safety assessment and for the assessment of security through tampering are needed. The detection of critical events at system and service level has become more critical than in the past, together with the degradation assessment of the response of services. Many solutions are currently in preparation using statistical information distance to decide, for instance, upon fault or attack recognition, or cycle-based models for system condition estimators, like Service Level Indicators that correlate system functions and infrastructure dimensions. These techniques ensure a quicker recognition of system problems than classical solutions based on mean inverse minutes between failure, which are either highly efficient when trouble occurs, but incomplete, or highly predictive as long as system or service state is stationary, but subject to frequent false alarms.

9.8. Evolving Network Performance

1. Adapting to Changing Conditions

For a long time, the dominant paradigm for OSS networks has been one of static configuration and performance. Networks are designed, built and then are assumed to operate as they originally planned. This arises primarily because there is a long delay between when a problem is detected and when it is acted upon, making it hard to dynamically adapt to changing conditions. This plan – build – forget paradigm will not be adequate as OSS networks change and evolve to become dynamic themselves. At the very least, operators will want to make enhancements to overcome problems they know they have. To support this procedure, we stress that predictive performance algorithms and general system monitoring is vital. Without this crucial additional information, it is difficult for operators to know what enhancements they need to make.

In fact, we think that proactive – "what enhancements do we need to make" – methods will soon move into a gradual interactive style. Operators don't want to be swamped with details each time they make an adaptation, so it would be best if operators were only presented with detail on a small number of options from which they could select. An adaptation controller or exploration strategy should handle the larger part of the adaptation process; note that this could include consideration of using different enhancements in different control contexts. Here we envisage a two-level architecture with a longer time-scale higher-level controller and a fast reactive lower level where the lower level is capable of fast adaptation to changing conditions.

2. Future Trends in OSS Networks

Currently, OSS networks are designed for a very particular set of common services. More variations in service than are currently designed into networks are existing in practice. We see this trend expanding. It is a natural consequence of the trends which we

explore in the other papers throughout this book. The move to an integrated services environment with the development of new varied types of multimedia services provides a basis for this expansion. In addition, greater interaction with human users by use of intelligent agents and with end-user adaptability of both service preferences and content as provided in some new Internet pages would not be possible without the capability of supporting many more varying services than we have had in the past.

9.8.1. Adapting to Changing Conditions

An important characteristic of many optical networks is their ability to adapt relatively easily to changes in network conditions. Such changes may arise from normal diurnal variation in traffic demands, the appearance of new flows, loss or addition of client services, or circuit failures. Traffic changes within the OSS portion of an optical network, caused by failover events followed by restoration, can last for some time, and it is generally possible for the optical layer resources to be set up to rout the traffic during such transient processes with a parametric COST function on some partitioned time windows. Although the dynamic control of the OSS path selection is hardly applicable with respect to circuits, more general degree and link loadings of the OSS ring must nevertheless rely on the same or a very similar distributed dynamic optimization approach.

If events causing changes in network conditions last only for relatively short time intervals of a few seconds, OSUs are likely to do little more than monitor fluctuations in condition and apply them to local routing changes on the fastest time scale in an event-driven way. For longer and more predictable time intervals of the order of a few minutes, simple QoS models can be expected. We therefore obtain a more precise routing of selected OSS flows, better resource utilization for the remaining portion of the OSS, and better OSS QoS in terms of lower packet loss ratios and shorter maximum queue lengths. In this case, if queue length sampling is done at fine intervals, we can rely on an instantaneous view of the OSS's general health state at the times of implementing the route changes; however, it is likely that drastic variations in traffic may still jeopardize the provisioning of a given delay level.

9.8.2. Future Trends in OSS Networks

The evolution of algorithm complexity is driven by technological advances and economic efficiency of new solutions. As computers become more powerful, predictive algorithms can employ advanced machine learning techniques such as deep learning. These algorithms can extract hidden patterns from string data. To support such pattern

discovery, real-time servicing of massive amounts of log and statistics data is required. Such large databases are present in the communications industry.

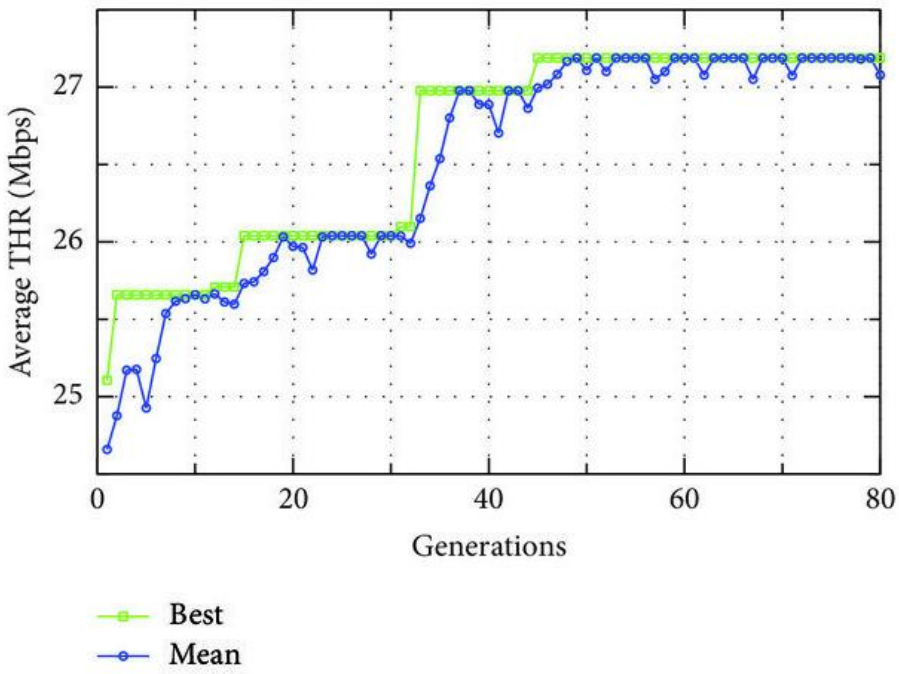

Fig 9 . 3 : Evolution of the average throughput in the whole network

As prices of computer servers continue to fall, the cost of processing predictive algorithms will also decrease. Active predictive analytics is the next evolutionary step beyond traditional analytics. This new research field combines mathematical prediction models, knowledgeable computational agents, and learning approaches such as heuristics and neural networks to automate the cycle of observing, learning, predicting, acting, and learning in closed loop. It needs new methods at the crossroad of several fields. Advances in computer power and miniaturization have revolutionized tiny sensor design. Embedded sensors are increasingly sampled on a real time basis. The ability to make better predictions will rely on increasingly complex, multilayer models operating on information from new sensors such as smartphones, especially with increased penetration of high bandwidth smartphones and other devices into more consumer markets. Formulae of current device and application behaviours will allow calculations of probabilities that different users will use different devices during future time intervals.

New software developed will allow engineers to decide important aspects of the prediction cycle. These include which prediction parameters can be ignored, the degree of mathematical model complexity needed to optimally balance accuracy and cool down time, the required frequency of model recalibration requests, etc. The predictive

backtracking will allow users to backtrack into the past when present unusual situations and behaviours were last seen.

9.9. Conclusion

In this paper, we provided insights into how predictive algorithms can be used to optimize network slicing gateways operating Multi-access Edge Computing infrastructures with copper fiber hybrid last miles, in order to provide high quality of service for a low operational cost. The predictive algorithms allow the MEC gateways the capability of estimating network traffic congestion during dynamic events, such as massive outdoor sports events, and thus automatically configuring and adapting the system parameters to optimize configuration and adaptation times, and minimize costs and latency with respect to the anticipated event. Moreover, these algorithms can estimate the point to point transmission delay associated to a given quality of service, for a given traffic intensity and system configuration, and for any PON technology. As such, these algorithms can optimize the configuration and adaptation of the MEC gateway system during dynamic vectored VDSL2 PON.

The results presented may be partly applicable to other wireline systems, such as flexible wavelength assignments Optical Circuit Switching or Ethernet Passive Optical Networks. In this case, because of their tee ball tree structure, we would need to change the algorithm for estimating the point to point transmission delays associated to a given quality of service. However, we believe that most of the results are transferable, because dynamic events are becoming more frequent and critical for fixed network communications, and wireline optimization algorithms are still in their infancy, while due to their lower complexity, computational time, and flexibility with respect to traffic prediction, machine learning algorithms offer a great opportunity for optimizing wireline resource management during dynamic events.

9.9.1. Summary of Key Insights and Recommendations

Since the 1990s, operators have devoted engineers and operational resources to enable and guarantee the real-time performance of Elastic and, more recently, Cloud-based applications. But their demand had been an order of magnitude greater than the available investment in manpower and algorithmic capabilities available to the operators. In this essay we described the methods used in the optical sector of the telecommunications market to lessen this labor and resource burden, and thus keep up with demand. The commercial importance of making sure these applications work correctly cannot be overstated, and the economic importance of investments in algorithm and predictive modeling capabilities for operations is huge. We expect these expenditures to grow as

revenues fall in the Box-Dominated marketplace, and predict that Predictive modeling and other information-theoretic research will enhance revenue growth as speed and delay sensitive applications proliferate in the Deep Content Internet favored by consumers, small, and large enterprises.

Within this more measurable domain, we presented vertical market after vertical market insights about where prediction and forecasting capabilities were underutilized, and highlighted where potential investments in algorithm resources could improve the SSA and incident optimization stages by diminishing the time and manpower of the advanced services sector rebuilding connective paths. These investments in algorithmic and predictive modelling sector would then enable the shift of focus for these small but revenue important services away from obnoxious seconds after detection repair, to more customer and revenue friendly minutes and hours long predictions. Such timelines describe the shifted paradigm for the Deep Internet, a paradigm where optimization is a regular and absorbing event for a protected and monitored network, not an emergency lightning fix for an important provisioning fiber.

References

Zhang Y., Wang S., Wang X. (2020). A Survey on Predictive Maintenance: Systems, Purposes and Approaches. IEEE Communications Surveys & Tutorials, 22(4), 2462–2495. https://doi.org/10.1109/COMST.2020.2988377

Guo Y., Lin W., Li Z., Shen S. (2019). Network Performance Prediction Based on Machine Learning: A Survey. IEEE Access, 7, 130632–130646. https://doi.org/10.1109/ACCESS.2019.2938683

Miao J., Ding Y., Guo Y., Li J., Wu J. (2021). Deep Learning Based Fault Prediction in OSS Networks. IEEE Transactions on Network and Service Management, 18(3), 3011–3024. https://doi.org/10.1109/TNSM.2021.3071735

Akella S., Feamster N., Snoeren A.C. (2003). Self-Configuring Networks: Principles and Practice. Proceedings of the ACM SIGCOMM, 89–100. https://doi.org/10.1145/863955.863963

Singh S., Jain A., Aggarwal S. (2022). Real-Time Monitoring and Fault Diagnosis in OSS Using Machine Learning. IEEE Systems Journal, 16(1), 1131–1142. https://doi.org/10.1109/JSYST.2021.3116847