Novel Approaches in Network Fault Management

Ankur Gupta and Purnendu Prabhat
Model Institute of Engineering and Technology - Jammu

As computer networks increase in size and complexity, managing them to ensure 24x7 uptime while meeting increasingly stringent Service Level Agreements (SLAs) and customer expectations, become critical issues. Although Network Management solutions have progressed significantly in recent years, issues such as extreme scale, new network paradigms, protocols and increasingly heterogeneous networks make the task of efficient fault management non-trivial. This research paper identifies and reviews novel strategies, techniques and ideas, either in commercial solutions or in literature to alleviate some of these challenges in the network fault management domain. Some ideas of the future evolution of the domain are also presented.

Keywords: network management, novel approaches to network management

1. INTRODUCTION

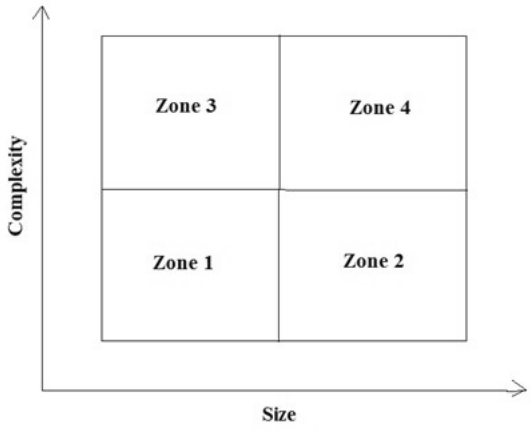
Network Management encompasses five functional areas: Fault, Configuration, Accounting, Performance and Security management as defined by the OSI FCAPS [Raman 1998] model. Large networks with multi-vendor devices, varied protocols, plethora of applications and services deployed over the network introduce complexity in the network management process. These challenges are usually overcome by deploying multi-vendor specialized management solutions focusing on specific aspects of the network to be managed. This requires serious integration efforts to tie-up the different pieces of management software. A truly comprehensive, vendor-agnostic solution is still a few years away. An earlier work on the shortcomings of existing network management solutions, the state-of-the-art and possible future trends is available in [Gupta 2006b].

Fault management is a key functional area in network management. With increasing scale the efficiency and accuracy of fault detection is impacted. Generally, fault detection becomes delayed, root-cause analysis for network faults incurs significant computational overheads and there is an information overload for human network administrators due to an avalanche of alarms and events. This has necessitated building some intelligence into network management software, push for increased automation, apply new techniques to network management and early forays into self-managing autonomic networks or management software. Further, with the advent of new paradigms such as Software-Defined Networking (SDN), programmable networks and cloud computing through large-scale distributed data centers, the complexity involved in managing faults has increased significantly. This paper reviews some interesting, novel approaches and ideas which are currently being used in network management or which have the potential of being deployed to address some of the existing challenges in fault detection and management.

We identify two parameters, size and complexity, based on which we categorize network fault management challenges into four quadrants as shown in Figure 1. Size refers to the number of managed entities (devices, device interfaces, groups of devices, clusters, racks, protocols etc.) in the network whereas complexity is a holistic term used as a measure for the degree of heterogeneity (multi-vendor and protocols) in the network, use of virtualization, SDNs or geographically distributed data-centers.

Zone 1: Low complexity and small to medium size (hundreds or few thousand entities). It is obvious that small networks homogeneous in nature are the easiest to manage and sometimes need a simple solution based on polling. This zone typically covers in-company networks which typically involve implementation and deployment of single vendor devices. Here network management is straightforward.

Zone 2: Low complexity and large size (hundreds of thousands/millions of entities). This zone requires distributed network management to overcome scale and collate data from multiple sub-



. Figure 1:Categorization of network management solutions

networks to build an overall view of the network. Typical issues in such networks are the speed of fault-detection, reducing event overload at central management station, event correlation for root-cause analysis etc.

Zone 3: High complexity and small to medium size. This zone necessitates specialized network management software to be employed to address complexity. One approach is the use of ontologies that contain detailed information about the underlying technologies used and different proprietary protocols to construct a vendor agnostic view of the network. The manageable size of the managed network however reduces issues of speed of detection, event correlation complexity etc.

Zone 4: High complexity and large size. This zone is obviously the most challenging in terms of correct fault diagnosis through event correlation across different network constructs and ensuring uptime by reducing mean-time- to-detect for network faults. Such networks exist at large service providers and encompass legacy networks. Other examples are very large-scale data centers operated by the Googles, Amazons and Facebooks of the world. This zone is characterized by multi-vendor solutions and large IT teams, necessitating the need to employ novel solutions.

This paper primarily focuses on solutions designed to operate in Zone 4. While review papers focused on broad aspects of network management exist, this research paper specifically focuses on fault management which is a key functional area in network management. Many of the techniques employed in network fault detection are relevant to the broad domain of anomaly detection across diverse application domains, making their study important. This paper consolidates knowledge in the domain, identifies novel solutions, classifies them and provides insights into possible future research directions in the field.

The rest of the paper is organized as follows: section 2 details the major challenges in network fault management, with each subsection focusing on a particular challenge and reviewing the related novel approaches to address that challenge. Section 3 focusses on fault management in Data-centers and SDNs, section 4 discusses the advances in autonomic network fault management, while section 5 concludes the paper.

2. NOVEL APPROACHES TO NETWORK FAULT MANAGEMENT

This section discusses novel solutions proposed in literature and/or adopted by the industry to address some of the major challenges in network fault management.

2.1 Overcoming Scale

The distributed computing paradigm can be safely applied to network management, since the network in essence is a distributed entity and a centralized management solution seems counter-intuitive.

2.1.1 *Supporting Distributed Management Architectures.* NMSs such as HP OpenView Network Node Manager (NNM) [hp.com] provide in-built support for distributed management of large topologies. This requires manual configuration and deployment of multiple NMSs in a master-slave configuration, with each slave NMS being responsible for a sub-set of the topology. The challenge with this approach is that the management information needs to be synchronized across multiple NMSs and the master NMS does maintain redundant information to provide a central view of the network. Moreover, the data from each of these stations will still have to be correlated entailing significant overheads. What is needed is to build sufficient intelligence into the distributed components and grant them sufficient autonomy to enable real distributed network management. A novel approach integrating distributed network monitoring at the protocol level is described in [Gupta 2001]. This mechanism makes use of CORBA (Common Object Request Broker Architecture) [corba.org] to make distributed SNMP calls for monitoring distributed network entities. This results in the network management station making simple SNMP calls to a master agent which then breaks up the simple SNMP calls into multiple distributed SNMP calls to multiple sub-agents. The master agent then collates management information and returns consolidated status information to the NMS. Thus, monitoring distributed entities is made simple and the NMS does not need to separately track distributed entities or applications.

2.1.2 *Mobile Agents and Swarm Intelligence.* Mobile Agents (MA) are intelligent software agents that are free to roam the network, clone themselves and perform management functions. They are designed to move management functions to the vicinity of the devices rather than centrally collating management data from all devices to the NMS. They are capable of self-propagation and self-deployment as per a fixed mandate in order to maintain local optimality. This decreases network-wide traffic and the response time while attending to the dynamics of very-large-scale networks [Liotta et al. 2002]. With decreased response time and traffic overhead, they make network fault management for large topologies feasible. A novel approach of using mobile agents for network fault management is discussed in [Yang and Chang 2011] where multi-agent techniques are used to create a centralized caching system based on network flow and uses ontology to analyze network-wide events to monitor and predict erroneous events. This has been experimented to reduce the fault recovery time by 61%.

Swarm Intelligence (SI) is the collective behavior of decentralized and self-organized systems and can be used for distributed and fully decentralized network management. Hence, SI and MA are largely complementary concepts. MAs with SI properties can be envisaged to quickly detect, localize, isolate and maybe even correct them autonomously. Researchers have proposed the application of SI concepts employing mobile agents to network management [Gupta and Koul 2007]. It seems feasible to have a community of specialized agents managing the network as a collection of small cells with localized fault-detection and correlation logic. However, swarm intelligence is still to find application in commercial network management solutions, due to their largely centralized architectures.

2.2 Intelligent Fault Detection

2.2.1 *Smart Polling.* Scalability issues result in delayed fault reporting, since the entire topology needs to be polled. One solution to overcoming scale is to build intelligence into the monitoring engine of the NMS so that it can quickly determine network outages and malfunctions in case of large internet scale topologies. Research from Wendell and Goltermann [Wandel and Inc.] shows that network faults also follow the 80:20 rule, which implies that 20% of network devices account for 80% of the network faults. By using such heuristics with solid measures about past

network behavior through network baselining, the monitoring engine can focus more on the 20% of error-prone devices in a network topology, resulting in lower Mean-time-to-Detect for network faults. The hot-spots of network faults can be identified through network baselining statistics and can be polled more frequently compared to the more stable areas of the network. The frequency of polling can be fine-tuned to the network and dynamically adjusted with the use of machine learning. This scheme is discussed in [Gupta 2010].

2.2.2 *Participative Fault Detection.* Router faults result in a cascade of network faults mostly caused by unreachability of network devices which are downstream of a failed router interface or the router itself. Thus, detecting router/routing faults quickly is essential to ensure network connectivity and correlation of cascading network faults. In Packet Designs Route Explorer [packetdesign.com], a dummy router device is connected to the routing backbone (a network of routers). This dummy router does not participate in the active routing process, but receives all protocol updates including change of routing tables, unavailability of routers, router configuration changes etc. Thus, it is the first to know of any faults at the router level and notifies the NMS immediately. Thus, fault detection is off-loaded from the NMS to the dummy router which is connected to the active routers. Such approaches are applicable to specialized protocols and other special purpose devices helping reduce the mean-time-to-detect network faults significantly.

2.2.3 *Peer-to-Peer Network Management.* The application of P2P networks to diverse domains including network management is discussed in [Gupta and Awasthi 2012]. P2P networks are distributed, highly scalable, resilient to faults and self-organizing making them suitable to address the challenges in network fault management. Security does present a challenge in P2P networks, although solutions to protect individual peers from other malicious peers have been proposed in literature [Gupta and Awasthi 2008]. Some of the advantages which P2P systems can expect to bring to network management include immediacy of fault detection (since each peer is monitoring immediate neighbors) irrespective of topology size, localized network monitoring traffic and non-contribution to network-wide traffic overheads. Challenges with P2P networks would include collating data and creating centralized views of the network as provided by existing NMSs. A hybrid approach might be practically feasible in which light-weight localized monitoring may be performed via P2P, while centralized polling for performing root-cause-analysis and correlating cascading or interrelated faults may be carried out. Researchers have examined the feasibility of JXTA-SNMP interoperability allowing tight integration of network monitoring in a P2P manner. The architecture of a collaborative monitoring system based on collaborative peers is described in [Zhou et al. 2014].

Some work exists in this area. Madeira [Plus] aims at providing system technologies for very large and highly diverse network element technology. It approaches decentralized and iterative alarm processing and correlation by peer-to-peer communication facilities and a logical overlay network. SELFMAN project [Roy] intends to develop self-managing applications based on peer-to-peer networks that do not break with network faults. Their approach keeps the communication alive by using a structured overlay network that can survive network partitions. PeerMon as described in [Newhall et al. 2010] provides a system wide data collection framework which can be modified to be used for network management.

2.3 Reducing Information Overload on Network Administrators

Network administrators deal with large volumes of events and traps signifying abnormal conditions, state changes on network maps initiating corrective actions to restore the networks operational state. Hence, there is a real need to reduce the information overload on network administrators. Comprehensive graphical visualization [Zhang et al. 2014] of the whole network can, to some extent, reduce the burden on human administrators but the pace of evolution of the networks is expected to render such solutions infeasible in the near future.

2.3.1 *Advanced Root-Cause Analysis.* Event correlation is a standard technique employed by commercial NMS solutions to discover event relations and suppress events which may not be important from the perspective of the network administrator. Correlation logic based on Rule-based expert systems [Cronk et al. 1988] and machine learning [Kumar and Venkataram 1997] exists. SMARTS code book [emc.com] technology from EMC is one such application of an expert/decision-support system which is widely prevalent in industry. However, the success of the rule-based systems is based on close working relationships with device vendors; hence the solutions do not work well across all classes of devices and vendors. With increased complexity, event correlation logic is becoming computationally intensive due to large number of potential correlations between diverse network events. The trend in this sub-domain is the move towards automatic discovery of event patterns to work in diverse and heterogeneous environments. For instance, the fault localization architecture proposed in [Natu and Sethi 2008] captures network dependency changes with respect to time and uses temporal correlation algorithm to perform fault diagnosis.

2.3.2 *Early-Warning Systems.* An NMS that could predict network outages, device malfunctions or even provide an early warning for impending network faults would allow network administrators to take precautionary measures in reducing the impact of the network faults or even preventing the fault from occurring. Such a system could be built by analyzing past network behavior through events and then figuring out spatial and temporal relationships between network faults and the elements involved. Thus, patterns of occurrence of past events that lead to the final error can be identified. As real-time events come in they are matched with the existing patterns. Based on the success of pattern matching, early warning events are issued, enabling preemptive action. Past analysis can therefore form the basis for future predictions.

An early direction in this area is provided in [Gupta 2006a] and is discussed extensively in [Gupta and Prabhat 2016], identifying potential adverse network conditions through automated analysis of past network behavior and developing baselines for normal network behavior. The method described discovers and maintains network event patterns (temporal and spatial) that led to final network error in a pattern repository and matches with it occurring event sequence in real-time to generate early warnings with different severity levels. A similar approach is discussed in [Wang et al. 2010] which discover a network signature and use it for Topologically Aware Reasoning (TAR). Few commercial solutions like IBM Netcool [ibm.com] also provide proactive fault management through advanced correlation. Authors in [Caravela et al. 2016] investigate the usage of data mining methods on past data to generate knowledge which in turn trains a machine learning system to predict alarms and allow for preventive network maintenance. However, a truly predictive solution which is also adaptive proves elusive as existing systems work well when network topology and dependencies remains static for long periods of time.

An overview of the major network fault management issues and broad approaches to address those issues is presented in Table I.

3. MANAGING FAULTS IN DATA-CENTERS

According to Cisco [cisco.com], the traffic flowing through data centers was 3.4 zettabytes in 2016 and it is forecasted to be 10.4 zettabytes in 2019. In cloud computing around 75% of the traffic flows within a data center, 7% flows between data centers and the remaining flows between user and data center. The new-age data-centers comprise millions of physical servers, potentially billions of virtual machines, hundreds of thousands of network entities which need to be operating at optimal performance to deliver overall performance goals. Further, with ever-increasing end-user expectations on QoS parameters and SLAs, fault management and alternate resource provisioning mechanisms need to operate at unprecedented levels of efficiency and effectiveness. Data centers are extremely dense and large networks generating hundreds of thousands of syslog and other monitoring events per hour. Add to that proprietary hardware, software and networking employed by the big players like Amazon, Google and Yahoo! requires that customized fault

Issue	*Approach*	*Implementations/Projects/Frameworks/Ideas*
Scalability	Distributed NMS Architectures	HP OpenView NNM [hp.com] Master-Slave deployment architecture.
	Protocol level support	CORBA-SNMP gateway to manage distributed applications/protocols [corba.org]
	Mobile Agents/ Swarm Intelligence	Ontology-based multi-agent techniques [Yang and Chang 2011], SWAN [Gupta and Koul 2007]
Fault Detection	Smart Polling Strategies	IntelliMon [Gupta 2010]
	Participative Fault Detection	HP OpenView Route Explorer by Packet Design [packetdesign.com]
	Peer-to-Peer Network Management	collaborative monitoring system based on JXTA[Zhou et al. 2014], Madeira [Plus] , SELFMAN [Plus], PeerMon [Newhall et al. 2010]
Information Overload	Expert systems to codify network management knowledge	SMARTS [emc.com], Temporal Correlation Algorithm [Natu and Sethi 2008]
	Early-Warning Systems, Predictive network management	Early Warning System [Gupta 2006a], TAR [Wang et al. 2010] , IBM Netcool [ibm.com]

Table I. Summary of novel approaches to address issues in network fault management

detection and management techniques are employed and traditional network management solutions are no longer relevant. Thus, data-centers require novel fault detection and management techniques to handle the inherent complexity and scale.

3.1 Real-time fault management

A data center is composed of heterogeneous devices and multiple links connecting them; both fail and cause outages in the network which needs to be dealt with immediately. According to a study by Gill et al. [Gill et al. 2011], links fail mainly because of hardware and connection faults whereas the devices sometimes go down due to hardware and software faults. Among the devices load balancers have highest probability of failure followed by aggregation servers and top of rack switches. Links connecting load balancers to aggregation servers fails most followed by those connecting routers and those connecting primary and backup switches. Vincent et al. developed F10 [Liu et al. 2013], a fault tolerant network topology with custom made protocols that claims almost instantaneous reestablishment of connectivity and load balancing even when faced with multiple failures.

Recovery from a fault and its resolution, either automated or human-supervised, takes time. In some scenarios it is efficient to mitigate the fault rather than resolving it. Restarting devices is a heuristic way of solving a network problem but doing that should not hamper the system functionality. NetPilot [Wu et al. 2012] does impact estimation and then defines a candidate set of afflicted components on which it takes mitigation action iteratively till the fault is mitigated. This method is viable as backups are a core attribute of a data center network.

Facebook uses NetNORAD [Aijay Adams] which treats the data center network as a black box and does fault mitigation independent of device polling. It uses end-to-end UDP probes from pingers (chosen servers) to responders (all machines). The data from these probes is analyzed in real-time and faults are localized and isolated. Alarms are raised if human intervention is needed. A packet-level telemetry approach called Everflow [Zhu et al. 2015] has been applied by Yebo et al. at Microsofts data center network that filter packets and sends probes to test or confirm potential faults.

3.2 Proactivity through Prediction

Given the event (syslog, traps, faults, monitoring and control messages etc.) diversity and rate in data centers, predicting faults through pre-learned rules is not as effective as in traditional network management [Watanabe et al. 2012; **?**]. A large portion of knowledge gained from these events is short-lived as they become obsolete after any upgrade to the devices or software running on them. Therefore, mechanisms which learn and predict in real-time will need to be evolved.

Issue	*Approach*	*Implementations/Projects/Frameworks/Ideas*
Real-time fault management	Fault tolerant topology and custom made protocols	F10 [Liu et al. 2013]
	Iterative rebooting of inflicted devices to mitigate fault.	NetPilot [Wu et al. 2012]
	End-to-end probing	NetNORAD [Aijay Adams] by Facebook
	Packet-level telemetry	Everflow [Zhu et al. 2015]
Proactivity through Prediction	Message-type pattern matching	Online Fault Prediction by Fujitsu Labs [Watanabe et al. 2012]

Table II. Overview of novel approaches to address issues in network fault management in data centers

Researchers at Fujitsu Laboratories have come up with a unique solution that addresses this issue by classifying messages into types, so they learn these types rather than learning specific messages, and create patterns by handling sets of message types [Watanabe et al. 2012]. These patterns are saved in a dictionary attached with the learned probability of failure. Thus, the learning and prediction are both done in real-time and the operator is alarmed of any predicted failure. Further, advanced correlation rules which connect observed problems to virtual faults to physical faults shall also need to be devised to further improve the effectiveness of the proposed scheme.

Table II gives an overview of the novel approaches in managing data center network faults.

Thus, the major trend in fault management in large data-centers is that rather than focusing on improving the efficiency of detecting faults, as in traditional network management, it becomes imperative to bypass or avoid the faults through in-built redundancy and real-time load-balancing strategies. Moreover, reporting of faults to the human network administrator becomes less relevant than initiating automated actions to mitigate the fault itself. Thus, autonomic approaches to fault management are appropriate in large data centers.

3.3 Fault management in SDNs

With the advent of Software Defined Networks (SDN) and programmable networks there is an additional complexity introduced in detecting, localizing and remediation of faults. The task of correlating faults across multiple planes physical and virtual is non-trivial and computationally intensive. Moreover, the SDN-compliant network entities themselves need to be managed to ensure that the network continues to deliver performance as planned. SDN failures are broadly categorized as:

(1) controller (e.g. OFC) server failure
(2) controller crashes
(3) network device failures and
(4) SDN-App failure

Innovations that address these failure categories are elucidated in Table III.

4. TOWARDS AUTONOMIC FAULT MANAGEMENT

IBM Research introduced the concept of autonomic computing in 2001. Autonomic computing as outlined by IBM has the following major features:

(1) Self-configuration
(2) Self-healing
(3) Self-optimization
(4) Self-protection

These four properties have been extended by other researchers to include awareness, learning, organization, creation, regulation and management.

Issue	*Inovations*
Controller server failure	AFRO [Kuźniar et al. 2013] spawns a new controller and stabilizes it by adding to the rule base for seamless handover.
Controller crashes	Ravana [Katta et al. 2015] is a logically centralized fault-tolerant SDN controller platform.
Network device failures	NetSight [Handigol et al. 2014] Packet History Filter API traces packet history to troubleshoot failures.
SDN-App failure	LegoSDN [Chandrasekaran and Benson 2014] detects and modifies network events to eliminate the crash.

Table III. Summary of SDN Fault Management Approaches

Autonomic solutions for network fault management hold great promise to master scale and complexity. Some early work in this area has started to appear. FOCALE [Jennings et al. 2007] being developed at the Motorola Labs is an autonomic approach to network management. It uses an ontology-based approach to abstract away vendor specific attributes. It uses the concept of closed control loops that compare the state of a device to its desired state and takes corrective actions (if required) governed by a set of context-aware policies which are updated by the human administrator. FOCALE learns these policies and contexts and manages the conflicts in between. With time FOCALE minimizes human intervention.

The relevance of autonomic decisions is impacted by the delay in the receiving of environmental observations from the networked devices and deployment of management actions to them. MMS [Gogineni et al. 2010] is a network-layer module that runs in the management plane of networked devices and the management stations. It automatically creates management channels between the networked devices and management stations to bifurcate management and data communications allowing a seamless flow of environmental observations enabling better autonomic control over the network.

Microsofts in house automatic data center management infrastructure Autopilot [Isard 2007] is principled on fault tolerance and focussed on reducing human intervention. It provides mechanisms to automate monitoring, provisioning, deployment and maintaining hardware . A distributed device manager holds the ground state of the networked devices and the ground truth (the state the system is intended to be in). The satellite services communicating to the device manager via pull and heartbeat messages, sends the state of the managed devices to the device manager and receives in turn the ground state for that device and the ground truth. The satellite services then perform actions to bring themselves and their managed devices up to date. This way Autopilot integrates cluster wide information and becomes robust even to transient failures delivering an auto-healing solution to manage global data centers [Clark].

Autonomic networks which move network management to the network itself are expected to be the future of network management. However, realization of truly self-sustained autonomic networks is some distance away. Table IV summarizes the different autonomic solutions for network fault management and the approaches they follow.

5. CONCLUSION AND FUTURE DIRECTIONS

Network management stations have come a long way from building a network view and then continuous monitoring of the network topology to determine faults. With increased scale, heterogeneity, virtualization and complexity, novel approaches have been designed and will need to continuously evolve to meet the challenges in fault management for very large-scale networks. The advent of Cloud Computing and Software Defined Networks (SDNs) also necessitates that

Autonomic solutions for network management	*Approach*
FOCALE by Motorola Labs	Use of ontology and control loops to deliver vendor-agnostic and corrective solutions that improve with time.
MMS by Carnegie Mellon University	Management plane robustness through an autonomic network layer module.
Autopilot by Microsoft	Robustness to transient failures through auto-healing via device and system state updating.
SDN-App failure	LegoSDN [Chandrasekaran and Benson 2014] detects and modifies network events to eliminate the crash.

Table IV. Some Autonomic solutions for Network Management

Traditional	*Modern*
Fault Detection	Fault Mitigation
Centralized Polling	Decentralized Probing
Reactive	Proactive
Human Involvement	Autonomic
Standards-based	Proprietary

Table V. The shift in network management strategies

management software is extended appropriately to first understand the new abstractions and then monitor them to effectively manage and correlate faults.

Thus, there is a clear paradigm shift in network fault management strategies. Fault management is shifting from traditional ways of reactive, standardized and human controlled management towards increased automation leading to fault mitigation or avoidance. Decentralization of management functions to the network itself is a major trend. Proactivity and autonomic solutions will be the inherent functionalities of next generation network fault management systems. The other major observation is proliferation of proprietary software in fault management which is a result of primarily proprietary hardware and software in the data-centers of large players such as Facebook and Google. Table 5 summarizes this paradigm shift.

We expect the future work in this domain to largely focus on the following areas: Predictive Improving the quality of prediction shall be a major challenge for researchers requiring the application of AI, deep learning and cognitive computing concepts and techniques going forward.

—Autonomic
Devices will tend to become more autonomic requiring very little central monitoring. The network fault detection and management can be expected to largely become a function of the network itself [Kuklinski and Chemouil 2014; **?**].

—Context-Sensitive
The network infrastructure will begin to understand the service/business context of the applications, developing ability to perform differentiated operations and support [Sethi et al. 2013]. Further, we believe that fault management and remediation shall become personalized to the end-user and her requirements.

—Collaborative
Application of P2P concept to the physical network can be expected to boost cooperation and collaboration among connected devices for quick fault detection and recovery [Atwal et al. 2016]. Further, new mechanisms such as blockchain technology hold promise in fault detection at the SDN-level and even ensuring correctness of applications and services running over the network.

—Internet-of-Things
We believe that fault management for IoT will require new and innovative strategies to be evolved. For instance, real-time stream processing leading to Big Data analytics frameworks to uncover fault patterns in internet-scale topologies are entirely feasible. A non-invasive, extremely large-scale edge computing approach will need to be evolved which shall present interesting challenges to the research community.

Thus, network fault management as a domain promises exciting new advances as the network itself continues to evolve, gain complexity and attain higher levels of operational efficiencies to deliver the next-generation of services and experience to the end-users.

REFERENCES

Aijay Adams, Petr Lapukhov, J. H. Z. Netnorad. https://code.facebook.com/posts/1534350660228025/netnorad-troubleshooting-networks-via-end-to-end-probing/.

Atwal, K. S., Guleria, A., and Bassiouni, M. 2016. A scalable peer-to-peer control plane architecture for software defined networks. In *Network Computing and Applications (NCA), 2016 IEEE 15th International Symposium on*. IEEE, 148–152.

Caravela, I., Arsenio, A., and Borges, N. 2016. A closed-loop automatic data-mining approach for preventive network monitoring. *Journal of Network and Systems Management 24,* 4, 974–1003.

Chandrasekaran, B. and Benson, T. 2014. Tolerating sdn application failures with legosdn. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*. ACM, 22.

cisco.com. Cisco global cloud index: Forecast and methodology, 20142019 white paper. http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html.

Clark, J. Inside microsoft's autopilot: Nadella's secret cloud weapon. http://www.theregister.co.uk/2014/02/07/microsoft_autopilot_feature/.

corba.org. Corba. http://www.corba.org/.

Cronk, R. N., Callahan, P. H., and Bernstein, L. 1988. Rule-based expert systems for network management and operations: an introduction. *IEEE Network 2,* 5, 7–21.

emc.com. Smarts- automated it management for the software-defined data center. http://www.emc.com/it-management/smarts/index.html.

Gill, P., Jain, N., and Nagappan, N. 2011. Understanding network failures in data centers: measurement, analysis, and implications. In *ACM SIGCOMM Computer Communication Review*. Vol. 41. ACM, 350–361.

Gogineni, H., Greenberg, A., Maltz, D. A., Ng, T. E., Yan, H., and Zhang, H. 2010. Mms: An autonomic network-layer foundation for network management. *IEEE Journal on Selected Areas in Communications 28,* 1.

Gupta, A. 2001. Network management system and computer-based methods for network management. US Patent App. 09/845,456.

Gupta, A. 2006a. Method and system for identifying potential adverse network conditions. US Patent App. 11/487,248.

Gupta, A. 2006b. Network management: Current trends and future perspectives. *Journal of Network and Systems Management 14,* 4, 483–491.

Gupta, A. 2010. A system and method for reducing the mean-time-to-detect for network faults. Indian Patent App. 2966/DEL/2010.

Gupta, A. and Awasthi, L. K. 2008. Secure thyself: Securing individual peers in collaborative peer-to-peer environments. In *GCA*. 140–146.

Gupta, A. and Awasthi, L. K. 2012. Peer-to-peer networks and computation: Current trends and future perspectives. *Computing and Informatics 30,* 3, 559–594.

Gupta, A. and Koul, N. 2007. Swan: a swarm intelligence based framework for network management of ip networks. In *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on*. Vol. 1. IEEE, 114–118.

Gupta, A. and Prabhat, P. 2016. News: Towards an early warning system for network faults. *International Journal of Next-Generation Computing 7,* 3.

Handigol, N., Heller, B., Jeyakumar, V., Mazières, D., and McKeown, N. 2014. I know what your packet did last hop: Using packet histories to troubleshoot networks. In *NSDI*. Vol. 14. 71–85.

HP.COM. Hp network node manager. http://www8.hp.com/in/en/software-solutions/network-node-manager-i-network-management-software/.

IBM.COM. Reduce outages, automate, gain visibility and control of your network. http://www-03.ibm.com/software/products/en/netcool-network-management.

ISARD, M. 2007. Autopilot: automatic data center management. *ACM SIGOPS Operating Systems Review 41,* 2, 60–67.

JENNINGS, B., VAN DER MEER, S., BALASUBRAMANIAM, S., BOTVICH, D., FOGHLÚ, M. Ó., DONNELLY, W., AND STRASSNER, J. 2007. Towards autonomic management of communications networks. *IEEE Communications Magazine 45,* 10, 112–121.

KATTA, N., ZHANG, H., FREEDMAN, M., AND REXFORD, J. 2015. Ravana: Controller fault-tolerance in software-defined networking. In *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*. ACM, 4.

KUKLINSKI, S. AND CHEMOUIL, P. 2014. Network management challenges in software-defined networks. *IEICE Transactions on Communications 97,* 1, 2–9.

KUMAR, G. P. AND VENKATARAM, P. 1997. Artificial intelligence approaches to network management: recent advances and a survey. *Computer Communications 20,* 15, 1313–1322.

KUŹNIAR, M., PEREŠÍNI, P., VASIĆ, N., CANINI, M., AND KOSTIĆ, D. 2013. Automatic failure recovery for software-defined networks. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*. ACM, 159–160.

LIOTTA, A., PAVLOU, G., AND KNIGHT, G. 2002. Exploiting agent mobility for large-scale network monitoring. *IEEE network 16,* 3, 7–15.

LIU, V., HALPERIN, D., KRISHNAMURTHY, A., AND ANDERSON, T. E. 2013. F10: A fault-tolerant engineered network. In *NSDI*. 399–412.

NATU, M. AND SETHI, A. S. 2008. Using temporal correlation for fault localization in dynamically changing networks. *International Journal of Network Management 18,* 4, 303–316.

NEWHALL, T., LIBEKS, J., GREENWOOD, R., AND KNERR, J. 2010. Peermon: A peer-to-peer network monitoring system. In *Proceedings of the 24th international conference on Large installation system administration*. USENIX Association, 1–12.

PACKETDESIGN.COM. Packet design route explorer, product overview whitepapers. http://www.packetdesign.com//technology/wp.htm.

PLUS, C. Project madiera- celtic plus. https://www.celticplus.eu/project-madeira/.

RAMAN, L. 1998. Osi systems and network management. *IEEE Communications Magazine 36,* 3, 46–53.

ROY, P. V. A self-managing peer-to-peer network. http://www.ist-selfman.org/wiki/images/6/6c/Selfman_A4s_H_Res.pdf.

SETHI, A. S., RAYNAUD, Y., AND FAURE-VINCENT, F. 2013. *Integrated Network Management IV: Proceedings of the fourth international symposium on integrated network management, 1995*. Springer.

WANDEL AND INC., G. Network baselining part-i: Understanding-the-past-to-predict-the-future. https://www.scribd.com/document/331973525/Network-Baselining-Part-I-Understanding-the-Past-to-Predict-the-Future.

WANG, T., SRIVATSA, M., AGRAWAL, D., AND LIU, L. 2010. Spatio-temporal patterns in network events. In *Proceedings of the 6th International COnference*. ACM, 3.

WATANABE, Y., OTSUKA, H., SONODA, M., KIKUCHI, S., AND MATSUMOTO, Y. 2012. Online failure prediction in cloud datacenters by real-time message pattern learning. In *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*. IEEE, 504–511.

WU, X., TURNER, D., CHEN, C.-C., MALTZ, D. A., YANG, X., YUAN, L., AND ZHANG, M. 2012. Netpilot: automating datacenter network failure mitigation. *ACM SIGCOMM Computer Communication Review 42,* 4, 419–430.

YANG, S.-Y. AND CHANG, Y.-Y. 2011. An active and intelligent network management system with ontology-based and multi-agent techniques. *Expert Systems with Applications 38,* 8, 10320–10342.

ZHANG, T., LIAO, Q., AND SHI, L. 2014. Bridging the gap of network management and anomaly detection through interactive visualization. In *Visualization Symposium (PacificVis), 2014 IEEE Pacific*. IEEE, 253–257.

ZHOU, K., TIAN, F., AND KONG, T. 2014. The design and implementation of a collaborative monitoring system based on jxta. In *Computer Science & Education (ICCSE), 2014 9th International Conference on*. IEEE, 989–992.

ZHU, Y., KANG, N., CAO, J., GREENBERG, A., LU, G., MAHAJAN, R., MALTZ, D., YUAN, L., ZHANG, M., ZHAO, B. Y., ET AL. 2015. Packet-level telemetry in large datacenter networks. In *ACM SIGCOMM Computer Communication Review*. Vol. 45. ACM, 479–491.

Prof. Ankur Gupta is the Director at the Model Institute of Engineering and Technology, Jammu, India, besides being a Professor in the Department of Computer Science and Engineering. Prior to joining academia, he worked as a Technical Team Lead at Hewlett Packard, developing software in the network management and e-Commerce domains. He obtained B.E (Hons) Computer Science and MS Software Systems degrees from BITS, Pilani and his PhD from the National Institute of Technology in India. His main areas of interest include peer-to-peer networks, network management, software engineering and cloud computing. He has published over 40 peer-reviewed papers in reputed international journals and conferences and is a recipient of the AICTE's (All India Council for Technical Education) Career Award. He has filed 14 patents in diverse technical domains and is the founding managing editor of the International Journal of Next-Generation Computing (IJNGC). He is a senior member of both the IEEE and ACM and a life member of the Computer Society of India.
Email-ID: ankurgupta@mietjammu.in.

Purnendu Prabhat is an Assistant Professor at Model Institute of Engineering and Technology, Jammu in the Department of Computer Science & Engineering. He received his Bachelors degree in Computer Science & Engineering from Kalasalingam University, Tamil Nadu in 2012 and Masters degree in Computer Science from Central University of South Bihar in 2014. His research interests include Network Management and Software Defined Networking. He has filed 2 patents and published 3 research papers and is also a mentor in Center for Research Innovation and Entrepreneurship (CRIE), MIET Jammu. He is a member of IEEE and Python Software Foundation. Besides research he enjoys programming and teaching.
Email-ID: purnendu.cse@mietjammu.in.