Using Big Packet Protocol Framework to Support Low Latency based Large Scale Networks

Kiran Makhijani, Renwei Li, Hesham El Boukary

Future Networks, America Research Center
Huawei Technologies Inc., Santa Clara, CA, USA
email: {kiran.makhijani, renwei.li, hesham.elboukary}@huawei.com

***Abstract*—The performance of many automation-based services over networks continue to demand lower latency and higher reliability. With Industry 4.0 initiative, the scale of such applications will grow over time requiring large-scale *High Precision Communications* as its foundation. IP-based networks with existing service delivery models do not support time related guarantees. In contrast, Time Sensitive Networking (TSN), which is a data link layer technology, supports many paradigms of *High Precision Communications* but capabilities are limited to subnets with only a few number of connected devices. A network layer (IP-based) solution is needed to overcome limitations of flooding and control overheads in layer 2, in order to expand the use of TSN services over broader domains. In particular, an extensible IP-based data plane approach exploiting already available hardware capabilities of TSN solution can be envisioned. This paper discusses one such approach using Big Packet Protocol (BPP) and develops a cross-layer forwarder method to combine benefits of BPP high-precision directives in network-layer with time-sensitive capabilities of TSN.**

***Index Terms*—Big Packet Protocol; BPP; High-Precision Networking; Programmable Networks; SLA.**

I. INTRODUCTION

Low latency applications are fast gaining mainstream momentum at large scale. That is, they are not limited to a single factory floor or studio networks, purpose-built for use with proprietary services. The number of such applications is continuously growing not only in private network domains such as factories, but are expanding to mass-consumption in service providers' networks as well. Interfaces such as human-to-machine and machine-to-machine are the basis of next generation connected services that aim to deliver digital world interactions with a very real user experience. Several such applications are sensitive to the precise time of delivery of information.

In networks, a layer-2 mechanism is offered by TSN protocol suite in bridged networks and has seen wide adoption in industry control, automotive networks and Audio Video Bridging (AVB). However, this can easily become difficult to scale as the density of end-stations, and bridges go beyond a certain number. A further proliferation of several latency-bound applications in service provider infrastructures is expected with the onset of 5G network slices for verticals such as Vehicle to Everything (V2X), critical infrastructure, Internet of Things (IoT) and so on. Many such service verticals need to stretch beyond layer-2 boundaries in order to scale better. Therefore, a TSN-like support in IP-based networks will be necessary.

High precision communications are the ones that guarantee packets of a flow associated with a service are delivered accurately on or before the prescribed time. Networks that are capable of providing high-precision communications are expected to have necessary network resources on each node in terms of buffers and deploy deterministic scheduling algorithms to achieve time-guarantees of such services. However, for the most part, packet forwarding technologies in large-scale IP-based and/or traffic-engineered networks continue to serve only statistical resource requirements, i.e., allocating from shared resources. This often comes with high cost of provisioning of network elements and their resources. To operate at large scale for diverse set of applications over larger area, minimizing configurations while providing higher level of customization and finer granularity of resource specification is necessary. It requires additional capabilities to be defined for IP, which is done via BPP.

In this paper, we propose a generic routed network solution based on BPP and extend the use of existing low latency TSN bridged networks. This paper describes the use of BPP framework [1] (a.k.a. New IP) to provide high-precision communications specifically for bounded-latency applications as covered under TSN. The New Internet Protocol (New IP) or BPP delivers high precision services over IP networks. We explore BPP as a solution to deliver time-sensitive services in layer-3 domains.

The New IP (will be referred to as BPP in remainder of the paper) defines high-precision communications suite which comprises of (a) in-time, (b) on-time, and (c) coordinated delivery of services - all of which are factor of time. It is a network layer solution that may easily be deployed at scale by any application. BPP is a new technology that provides building blocks both for customizing data plane forwarding from a user's perspective as well as in-node mechanisms to process many network parameters to manage packet latency and scheduling. At the same time, TSN is a well-established ethernet-based protocol suite built on the foundations of real-time Ethernet, e.g. Profinet, EtherCAT, etc. TSN is a part of IEEE802.1 standard and is widely used in AVB studio and factory floors networks. It consists of well-designed resource reservations and scheduling algorithms to support end-to-end bounded latencies. We demonstrate how TSN can be expanded to provide Ethernet services at a higher layer, while simplifying operation, control and monitoring.

This paper makes the following contributions: (1) introduces fundamental requirements for high-precision services with respect to the growing demand for latency-sensitive applications over bigger regions, (2) provides an overview of capabilities of TSN protocols along with their limitations, (3) provides a vision of BPP router node to support high-precision forwarding paradigm, and (4) finally elaborates a cross layer forwarder to combine capabilities of TSN with BPP to extend them to wide area applications.

The paper is organized as follows: Section II describes the motivation behind our work and provides a background to time-sensitive networking. Section III is an in-depth discussion of high-precision services and discusses BPP technology as means to achieve such services. Section IV very briefly discusses related work. Sections V and VI discuss in detail the contributions of this paper, finally covering future work in this area under Section VII.

II. BACKGROUND AND MOTIVATION

In industry operations, typical requirements for automated control of production floor requires bandwidth ranges of the order 100M-1G with latency 1ms to 200ms (it may even be lower for isochronous control such as PLC and embedded control) requiring interactions with Cyber Physical Systems (CPS) involving Machine to Machine Communications (MMC) [2]. Since these tight latency requirements originated from on-premise networks built on bus-like or LAN communications, the TSN solution was developed as a part of Ethernet protocol suite. Formally, IEEE 802.1 group defines TSN applications as those responding to external stimuli within a fixed, and often small, period of time.

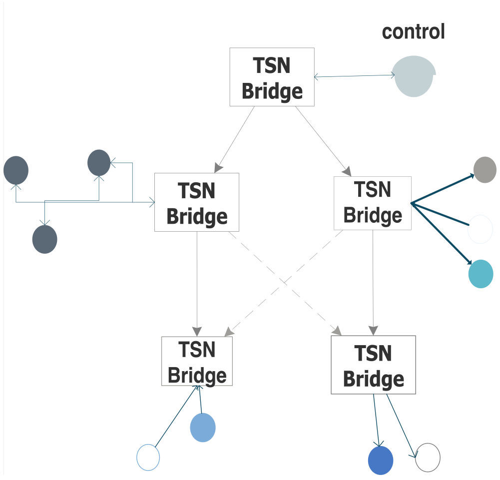

Fig. 1. Industry control network reference model

A factory floor will typically use a bridged topology as shown in Figure 1. The TSN-bridges connect with end stations and support usual layer-2 protocols such as spanning tree, VLAN etc. while providing latency guarantees in bridged network. TSN bridges provide bounded latencies, completely automated and reliable connections between the end stations such as machine equipment, sensors, PLCs etc. (shown as small circles directly connected with different TSN bridges Fig. 1 above) and a command controller shown as just another device in the network at the root bridge for simplicity but may be connected anyhere in this network. Such models have been widely deployed not only in industrial automation but also in Car Area Networks (CAN) and AVB production studios.

Contrary to this, Industrial 'internet' by very definition means interconnecting different end stations and industrial applications across multiple network domains, not just limited to local area networks. The principal goal of network slicing is to use common and public communication infrastructure for different types of services. An ultra-reliable low latency slice being just one 'type' of network-service and may have several instances based on market verticals. Obviously TSN by itself is not suitable for building applications at this scale. Either TSN should be extended beyond LANs or we need similar technology in layer-3 networks. Arguably, IP networks are best-effort and such bounded latency services were not inherently supported layer-3 until BPP [1]. BPP formally defines high-precision communication services as a group of technology and capabilities in the scope of layer-3 to serve similar but broader purpose than TSN. TSN is a well-known standard and provides several hardware capabilities for serving real-time applications but is not capable of scaling beyond bridged networks. Our contribution is to address this gap by introducing a cross-layer forwarder, which leverages information from both BPP (the network layer) and TSN (data link layer) to provide high-precision services in a generic manner. Our objective is to maximize re-use of existing TSN technology and build large scale high-precision networks with minimal disruption. We show that using cross-layer forwarder with minimal changes we can make use of many hardware components without much overhead. We then demonstrate the validate our approach by studying the forwarding path with BPP cross-layer forwarder.

III. HIGH-PRECISION COMMUNICATION SERVICES

Applications in automation, such as machine to machine interactions, vehicle to infrastructure, smart cities, remote surgical procedures, etc. have diverse and variable requirements from networks. While TSN primarily supports technology to guarantee worst-case end-to-end latency, there are in fact, more critical factors to time-sensitivity which are collectively described as High-Precision communication (HPC) services [3]. The HPC services can be broken down into a) in-time packet delivery - like TSN applications, b) on time service - with an extremely low delay variation between actual and planned packet arrival time and c) coordinated service - having more than one data stream arriving in specified bounds of time. This classification gives a better sense of criticality and time sensitivity in terms of how networks elements should treat such services. Regardless of the classification, there are certain common requirements to be met in delivering high-precision services and are described next.

A. Requirements For High-precision Services

Firstly, a knowledge of different HPC profiles is needed based on which appropriate bandwidth and path computations can be done. There may be different service-operation profiles based on the resources required for each end device (bandwidth, latency, etc.). The same network is also integrated for Information Technology (IT) services, such as management, monitoring, telemetry collection, etc. which do not have stringent service delivery requirements and no special traffic treatment is necessary. However, they may contend for same resources in the network at random period of times. Therefore, a discrimination of HPC services from normal traffic is necessary.

The second requirement is that in industry automation all network nodes should have an ability to compute transmission scheduling criteria and position in queue with high accuracy on network nodes. It requires knowledge of both complete topology and the path associated with the streams, all of which need high-precision time synchronization in the network.

Third, flow classification is required for a network node to identify what kind of service profile the flow belongs to in order to process it as per the constraints of that profile.

Then, determination of timing behavior is necessary. Special time-aware flow processing functions are required to ensure all packets are forwarded with in the desired timed-accuracy. These functions reside in the network element and support different varieties of scheduling and shaping mechanisms suitable for both time-sensitive and normal service profiles. These functions operate with the knowledge to resource specification of a service profile and will include parameters, such as bandwidth, jitter, and latency.

Finally, reliability is of prime importance in industry control. If a packet is dropped in transit, entire synchronized and automated factory pipeline may come to a halt or even worse may lead to commands being processed out of order causing several anomalies in production. Therefore, at the network level, path redundancy and extreme reliability functions are very important.

B. Challenges Beyond Switched Networks

TSN satisfies above requirements within the scope bridged-network (e.g. a few kilometers) for limited size of fixed topology through a suite of new provisioning and forwarding techniques, but there exist some limitations in delivering time-sensitive services.

Scalability. In traditional manufacturing, automation distances are few kilometers (less than 10) and limited number of devices, however as Industrial internet grows, it will be hard to scale the networks in layer 2 and even more complex to partition on per-service basis. The disadvantages of such large-scale bridged networks are well-known from data center networks (flooding, slow STP convergence, and so on), that have already transitioned to IP based solutions.

Infrastructure sharing. 5G brings automation in all kinds of applications and economy of scale demands that networks for edge services, V2X applications, critical services and industrial networks etc. shall share the infrastructure and hardware. Therefore, isolating applications is far simpler and scales better with IP based networks.

Complexity: A closer examination of TSN will show that it has evolved into a complex set of protocol suite. Several protocols have to be well understood and provisioned on all the bridges. The challenges of provisioning at scale, their response to topology changes, update and withdrawal of stream specific resources is an expensive procedure collectively generating several Bridged Packet Data Unit (BPDU)s and other control PDUs in the bridged network.

In comparison, BPP framework is a simple and customizable data plane for delivering high-precision services. The BPP has two primary artifacts, first that specifies what goes on the wire and signals per packet Service Level Objectives (SLO)s in a ***contract*** (BPP Block in Figure 2) to intermediate network elements, such as routers. Second, a programmable compute, forwarding and schedule engine on those routers to implement components necessary for scheduling and shaping functions. Because of these two factors, BPP is capable of combining on-the-wire contract with any hardware element that supports high-precision functions such as a TSN-bridge. We propose a simple method of cross-layer forwarder between BPP contracts and TSN scheduling and shaping functions by allowing coordination of technologies crossing IP and MAC layers. In the following section, we briefly describe end-to-end control and flow processing and forwarding based on TSN protocol suite.

IV. RELATED WORK

Historically, IP services have been implemented using Diffserv [4], IntServ [5] and RSVP [6]. These techniques can provide bandwidth assurance but latency guarantees are not possible, especially with interfering traffic. The most common queuing discipline used in IP based networks is deficit round robin method. Traffic engineering mechanisms for IP focus mainly on providing paths based on certain resource requirements, mainly bandwidth guarantees but often low-latency paths may also be computed. Recently, Internet Engineering Task Force (IETF) formed a deterministic networking work group (DetNet WG) to address a similar problem of time-sensitive networking. The scope of DetNet [7] does not include pinning time-sensitive low-level capabilities on the nodes but covers only the data plane that carries the deterministic service. This is limiting because the aggregated resource reservations have to be made in advance for the DetNet flows. Finn [8] discusses computation of worst-case end-to-end bounded-latency, but refers to external queuing mechanisms without details on how to integrate them in IP-based networks.

While IP-based networks lack sophisticated queuing and scheduling capabilities, Ethernet based TSN scheduling algorithms are well-established and thoroughly defined. Thus, our approach to achieving high-precision communications, at large-scale on per-flow basis is based on utilizing capabilities of underlying time-sensitive bridges.

V. HIGH-PRECISION NETWORKING WITH BPP

A. BPP Overview

BPP was first described in Big Packet Protocol framework [1]. The basic idea is that of injecting meta-information into packets to provide guidance to intermediate routers about processing those packets. This is done by attaching BPP Blocks (or contracts) with directives that provide guidance for the packet treatment such as what resources must be made available for the packet, as well as the flow that the packet is a part of. Rather than relying on in-built logic provisioned statically through management or control plane of networking devices that may result in best-effort treatment of the packet, a BPP network device will act on those directives and metadata to handle the packet, overriding any regular packet processing logic that is deployed on the device. This is in particularly important when dealing with resource-centric commands, for example, to determine conditions when to drop a packet, which queue to use, when to allocate a resource, or to measure a service level and compare it against its SLO.

This concept allows behavior of packets and flows to be programmed by injecting BPP Blocks (contracts) into packets at the edge of networks. There is no need to program networking devices or network controllers directly. At the same time, the programmed behavior is isolated from other flows and restricted to the packet and its flow. A BPP packet is structured as depicted in Figure 2. It consists of a pseudo-header of its host protocol (pointing to BPP Block as next protocol), as well as one (or more) BPP Blocks.

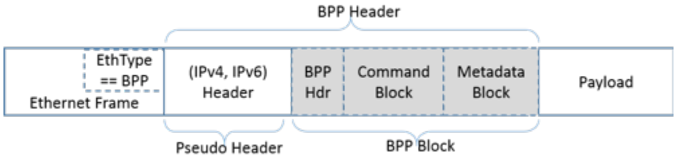

Fig. 2. BPP Packet Structure with Contract as BPP block

B. Sample Description Of High Precision Services

For example, the meta-information that BPP carries can easily be used to describe a high precision service requesting end-to-end latency of 14 ms and peak rate 512 kbps. In BPP contract, it requires two instructions 1) for latency-constraint and b) for peak rate as below. The actual encodings of these instructions are not described since they are implementation specific.

$$LatencyFn(intime|e2eFn(14, ms)|residual(time))$$

$$BandwidthFn(peakrate(512, kbps))$$

C. BPP Forward Processing On The Node

BPP contract processing requires a forwarding component that is capable of parsing and understanding the semantics of the contract carried by BPP. A simplified BPP node is shown below in Figure 3. In this figure, an enhanced BPP forwarding plane is shown with all the usual forwarding plane tables and components such as packet memory, classification tables, forwarding information base (FIB), Access Control List (ACL), and policy tables along with port specific queue manager, scheduler and shapers. In addition, there is a BPP parser that parses the instructions, metadata and state in the BPP Block, generating an output result that can be directly used by queue manager to schedule packets in output queue. Essentially, we show clear separation of three blocks: (a) BPP parser and compute, (b) Traffic manager, (c) lookup and state tables.

As a packet is received on a port, it is classified and checked for any ingress filtering, then BPP contract processing of different instructions happens based on the outcome of parsing logic. The results of instructions are generated and fed into the final scheduling and queuing functions. This forwarding pipeline is quite similar to Ethernet forwarding Section V-E described later, just that the functions are specific to layer 3 headers and in Ethernet, they are layer 2 port specific.

D. Leveraging Time Sensitive Networks Capabilities

A comprehensive detail of IEEE 802.1 TSN Task Group (TSNTG) [9] work is presented in [10] survey and therefore, we do not discuss details of the individual protocols and only cover the broader area in the context of the paper. A summary of TN work is shown in Table I.

Provisioning and reservation: The reservation of resources is mandatory in TSN networks. To this end, the reservation protocols have been enhanced to support centralized (Centralized Network configuration (CNC)), distributed (Stream Reservation Protocol (SRP)), and hybrid modes. SRP utilizes signaling between talkers, stations producing streams and advertise network resource attributes of the stream towards listeners, the devices consuming those same streams and declaring resources available for their reception. Reservations are created when these events are combined in a bridge.

Classification and marking: There is a new SR traffic class associated with two additional queues that enjoy higher precedence than usual priorities or priority code points. THe default settings map priority 2 and 3 to SR class B and A, respectively. In distributed SRP, bridges use credit-based shaper (CBS) data plane (Table I second row). The end stations are required to mark the packets with SR class A or B. Later, stream configuration enhancements were introduced

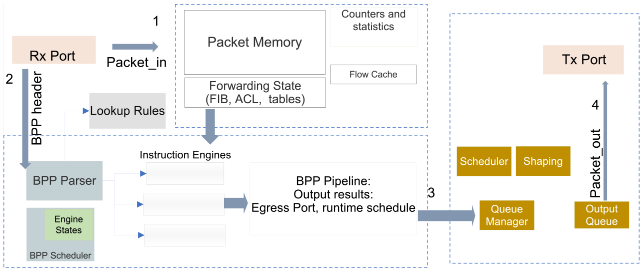

Fig. 3. New IP functions on Network Element

to support all TSN functions such as shaping, preemption, and redundancy through the CNC model. RAP is introduced to do distributed reservation for hard real-time control type applications. LRP provides local port services.

Packet scheduling and shaping: As shown in Table I first row, TSN supports different types of scheduling and shaping algorithms to address different functional requirements for bounded latency with rate constrained traffic, scheduled and best-effort traffic. Both queue and transmission algorithm selections are done based on traffic class. Credit-based shapers handle near real-time traffic which do not satisfy industry control requirements of tighter and lower latency guarantees. These are requirements are met using time-aware shapers with repeating schedules at pre-determined intervals. In-traffic-class interference among time-critical streams has to be further eliminated.

Synchronization and Reliability: Ethernet systems use 802.1AS Timing and Synchronization for Timing-Sensitive Applications, and 802.1CB Frame Replication and Elimination for reliability.

E. TSN Forward Processing On The Node

End to end forwarding of time-sensitive packets is well integrated in 802.1Q bridged networks. The MAC bridges are associated with separate learning, filtering and forwarding functions on receiving ports along with egress filtering, transmission selection and queue management functions taking place on transmit port as seen in Figure 4. The information required for forwarding comes from different dynamic and management configured tables. For example, implementation of the decisions governing where each frame is to be forwarded is determined by the relay function using forwarding rules that are populated in the Filtering database (FDB). The operation of relay function includes verification of active topology, classifying frames to expedite time-critical traffic, and frame format conversion for destination stations.

F. Layer-2 Service to Cross layer forwarder

Our motivation is to reuse queues of TSN ports with forwarding logic of BPP to provide scalability to time sensitive applications by using IP networks. We see benefits with the use of pinned-down hardware capabilities of TSN, and do away with layer-2 protocols, forwarding rules, policies and filter tables in favor of higher resolution of service control in layer-3. Ethernet bridges support relay function in which packets from a receive port are processed and sent to the transmit port. The bridge design provides information to relay function or tables in 3 ways, 1) learnt dynamically through incoming port, 2) management interface, 3) higher-layer MAC Service to higher layer entities using bridge port functions. In all the above scenarios, the user of relay functions essentially provisions various layer-2 tables for relay function with different forwarding, filtering, and policing rules.

TABLE I. AN OVERVIEW OF DIFFERENT TRAFFIC CLASSES IN TSN

Traffic class	**Best Effort**	**Rate-constrained**	**Scheduled**
Data plane techniques	Strict priority algorithm	Traffic shaping with Credit-based Shaper, 802.Qav	Time-aware scheduler, 802.1Qbv-Scheduled traffic, 802.1Qch Cyclic Queuing, 802.1Qcr Async Shaping
Control plane techniques	STP & so on	Dist. stream config bandwidth reservation resource allocation using Stream Reservation Protocol (802.1Qat)	Centralized stream config path, schedule calculation. management with a central controller (802.1Qcc)
Target Latency	Non-deterministic	Bounded max. latency	Guaranteed lowest latency

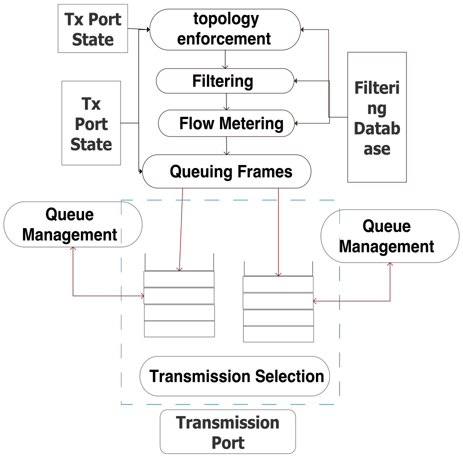

Fig. 4. TSN Forwarding process

None of the above three options are usable with BPP, nor do they provide the value of scale and simplicity we aim to achieve. Therefore, our proposal is to connect bridged ports with BPP forwarding pipeline, as explained in next section.

VI. EXTENDING TSN WITH NEW IP FORWARDER

In addition to looking at TSN bridges, we investigated well-known classical methods of cross-layering; thematically, in these approaches each layer continues to perform its function, while those functions are improved or optimized for the purpose of overall application response time or quality of experience. We found that cross-layering research is more significant either at transport level [11] [12] for better integration with applications, or in wireless networks [13] [14] for better feedback about signal strength and availability between MAC and PHY.

We did not come across a lot of literature on cross-layering between layer-2 and layer-3. There may be several reasons behind this such as ossification, specific segregation of switching and routing domains (i.e. either switched or routing policies, rules and management is used) etc. However, we believe the main reason is that in deployed communication protocol stack, over time layer-2 and layer-3 forwarding and control methods have essentially evolved independently. Yet, interestingly, the foundations of forwarding, policing, and scheduling are often found to be quite similar. For example, commonalities are seen in 'forward to next hop' functions; both layers use destination address based lookup, priorities are marked in packets header, and in particular development of TSN solution, several IP protocols were used as a reference to design similar requirement in TSN protocols such as, RSVP & SRP, and PTP & Time synchronization. Thus, in building high-precision services with BPP, we find the greatest benefit in cross-layering of 'functions' instead of information. Doing so, we combine the good of layer-2 and layer-3 as:

- Forwarding functions of New IP, since they provide high-degree of customization, replace forwarding of TSN
- Scheduling functions of TSN, since they exist already in deployments are reused.

As shown at the top part of Figure 5, functions of BPP parsing and forwarding are linked with the queue management functions of TSN in the same figure. We do not show different tables that will be populated and managed at runtime because BPP is the runtime execution-pipeline; it is not a chain of different tables in the traditional sense of forwarding pipelines.

We use directives from BPP block, such as those defined earlier in section V-B, to process high-precision requirements, run them through the BPP parser and processing engine, derive the result as to which algorithm (among the TSN supported algorithms) guarantees service, determine queue, and its parameters to schedule the packet in TSN queue component. BPP parser and processing engine determines the algorithm, the egress port, and the schedule. The BPP traffic manager entity maps schedule to traffic class and interacts with TSN queue management function.

A. Cross-layer forwarder Initialization

TSN Queue Manager/Schedule configuration. The way queue management (QM) algorithm will work is decided at the management plane. i.e., the operator determines what kind of scheduling behavior is required and accordingly TSN QM can be configured. TSN provides several managed objects and external configuration parameters through which queue, traffic class and algorithm specific information can be programmed in the TSN switch. Through configuration knobs, the exact timing behavior of queues can be provisioned such as bandwidth allocation for a particular traffic class, gate control list, queue max Service Data Unit (SDU) and even the list of algorithms supported. Queue configurations are completely unrelated to forwarding tables such as FDB and learning tables.

BPP Traffic Manager. Initialization for BPP is only needed to determine mappings in traffic manager, the traffic classes are to be mapped to the ones configured by TSN QM; BPP does not require any markings like .1P priority bits or even VLAN tagging. The result of the parser can determine the traffic class as an internal parameter (similar to internal priority value in TSN). ***Resource Reservation***. No TSN or IP reservation algorithms are used for profiles of different type of services. This is because the Tspec equivalent information is carried along with the packet in New IP (in-band). The hint for reservations happens in flow cache of BPP engine with first packet. The BPP schedules to send the packet according to embedded TSpec not according to what is reserved on the node. This is possible because BPP traffic manager uses cumulative port state based on queue-depths, if a particular packet can be sent in time.

B. Runtime Forwarding path

Figure 6 below shows a high-level packet processing and forwarding through BPP block to determine how latency instruction can be mapped to a queue or a traffic class and what algorithm can be used. As a high-precision service packet is received, the following processing is done.

1) In a high-precision network (large- or small-scale), an operator has a knowledge of the type of service. At the head-end node when packet arrives, a BPP block is injected with the high-precision service profile. BPP-Header insertion is a function of forwarding pipeline installed as a police on edge routers (not shown in the 6).
2) For an incoming packet, layer-3 lookup
    - uses FIB to determine the egress port

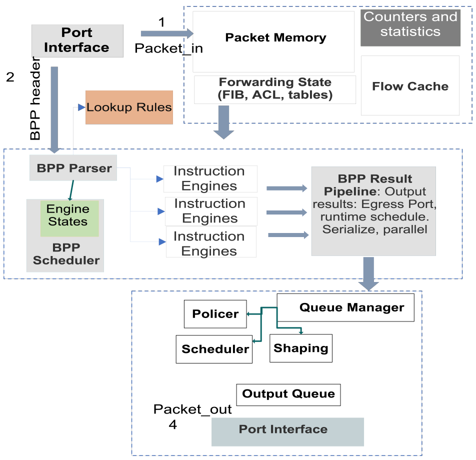

Fig. 5. Cross layer New IP forwarder functions

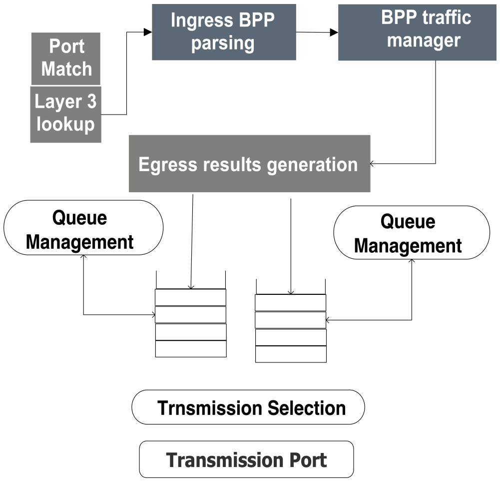

Fig. 6. Cross layer forwarder: packet processing

Copyright (c) IARIA, 2019. ISBN: 978-1-61208-711-5

    - recognizes that it is a BPP packet and dispatches it to BPP parsing engine. BPP parsing engine sees latency directive, determines type of high-precision service, and sends to egress results generation.
    - Egress results generation combines FIB and BPP results to assign appropriate algorithm and determines traffic class.
3) Internal state of BPP engine maintains the knowledge of resource budgets and keeps 'available resource' repository updated.
4) BPP traffic manager function finds the mapping to traffic class, peeks into queue tables to find runtime queue depth for that traffic class and determines if the latency-goal is feasible or not; if yes, it transmits the packet to that queue.
5) On the tail-end before delivering the packet to end-station, the BPP block is removed and plain IP packet is delivered.

C. Solution Analysis

The cross-forwarder solution is examined briefly in the following manner:
Shorter lead times: BPP is a new technology in which many new function components need to be developed. BPP framework requires hardware-support to implement high-precision networking and TSN switches can fulfill this gap. Given the cost and lead time for procuring prototypes, BPP benefits from the reuse of existing well-proven hardware schedulers.
Reuse Benefits: We proposed a theoretical model to tweak existing layer-2 forwarding pipeline in the layer-3 pipeline and make it usable with IP-based networks. The forwarding pipelines are based on match table look ups and assigning proper egress queues. Such tables can be programmed from the control plane with vendor-specified interface or drivers without having the need to change scheduling behavior or spin a new hardware. This was demonstrated via Figure 4 and Figure 6. It allows for new services to describe their service requirements with a fine granularity using BPP and have them treated using time aware schedulers.
Forwarding path validity: We provided a systematic examination of the forwarding path and corresponding BPP components necessary to develop the cross-forwarder. It exploits the fact that queue identifiers in the TSN switches are independent of link layer specific addressing. The feasibility of the cross-forwarder can be proven through empirical analysis which will be our next area of focus.
Performance: The performance comparison between TSN switch and proposed cross-forwarder will depend on the cost of processing BPP directives versus BPDU processing and is a part of our future work.

VII. FUTURE WORK

So far, we have presented the feasibility concept of cross-layer forwarder. Although this discussion is quite thorough, it still needs validation with implementation. There are not great options of opensource TSN algorithms or comprehensive SDK for the existing TSN switches. At the time of writing this document, New IP development work is in progress; once available it can be used to integrate and further evaluate our approach presented in this paper. While it is simple to develop this concept in software, it is still necessary to explore the amount of driver or FPGA changes required to use TSN switches.

VIII. CONCLUSION

As is evident from the previous section that TSN solution requires several protocols leading to overall higher operational complexity. The biggest limitation remains that it is only a layer-2 solution; in order to scale over wide area networks, a network layer approach is desired. In this paper, we propose that new data planes like BPP can tremendously reduce provisioning protocol complexities. We demonstrated emulating a TSN switch as a layer 3 high precision router is feasible which will allow a fast adoption of high-precision services in networks.

ACKNOWLEDGEMENT

The authors would like to thank members of their team who provided valuable information. These are Lin Han, Lijun Dong, Padma Esnault, Yingzhen Qu, Uma Chunduri, Toerless Eckart, and Alexander Clemm.

REFERENCES

[1] R. Li, A. Clemm, U. Chunduri, L. Dong, and K. Makhijani, "A new framework and protocol for future networking applications," *ACM Sigcomm Workshop on Networking for Emerging Applications and Technologies (NEAT 2018)*, pp. 637–648, May 2018.

[2] M. J. Teener, "Keynote: A time-sensitive networking primer: Putting it all together," *International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication, ISPCS*, 2015.

[3] R. Li, "Network 2030: Market Drivers and Prospects." "https://www.itu.int/en/ITU-T/Workshops-and-Seminars/201810/ Documents/Richard_Li_Presentation.pdf", Oct. 2018. First Workshop on Network 2030, New York University, Brooklyn, US.

[4] D. Black, and P. Jones. Ed., "Differentiated Services (Diffserv) and Real-Time Communication," *IETF, RFC 2205*, Nov. 2015.

[5] Y. Bernet et al., "A Framework for Integrated Services Operation over Diffserv Networks," Nov. 2000.

[6] R. Braden et al., "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," Sept. 1997.

[7] N. Finn, P. Thubert, B. Varga, and J. Farkas, "Deterministic networking architecture," *draft-ietf-detnet-architecture-09 (work in progress)*, September 2018.

[8] N. Finn, E. Mohammadpour, J. Le Boudec, J. Zhang, B. Varga, and J. Farkas, "Detnet bounded latency," *draft-finn-detnet-bounded-latency-02 (work in progress)*, October 2018.

[9] "Time-Sensitive Networking Task Group." http://www.ieee802.org/1/ pages/tsn.html. Last accessed 14 July 2018.

[10] A. Nasrallah et al., "Ultra-low latency (ULL) networks: A comprehensive survey covering the IEEE TSN standard and related ULL research," *CoRR*, vol. abs/1803.07673, 2018.

[11] S. Denny et al, "QoE Analysis of DASH Cross-Layer Dependencies by Extensive Network Emulation," *Internet-QoE '16: Proceedings of the 2016 workshop on QoE-based Analysis and Management of Data Communication Networks*, pp. 25–30, Aug. 2016.

[12] X. Corbillon et al, "Cross-layer scheduler for video streaming over MPTCP," *Proceedings of the 7th International Conference on Multimedia Systems (MMSys '16), ACM, New York, NY, USA*, pp. 1–12, Aug. 2016.

[13] S. Ket, V. Shinde, R. Khandare, and R. N. Awale, “Cross layer communication for wireless networks,” *Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3 ’09). ACM, New York, NY, USA*, pp. 629–632, 2009.

[14] F. Foukalas, V. Gazis, and N. Alonistioti, “Cross-layer design proposals for wireless mobile networks: a survey and taxonomy. Commun. Surveys Tuts,” *Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3 ’09). ACM, New York, NY, USA, 629-632*, p. 70, Jan. 2008.