Multi-layer IoT Resource Management

Kyriaki Panagidi*

Department of Informatics and Telecommunications
National and Kapodistrian University of Athens, Greece
kakiap@di.uoa.gr

Abstract. Internet of Things is one of the most promising paradigms in the current decade characterized by the use of smart and self-configured objects, like sensors, actuators, wearables etc., that are connected to a network and exchange data by sensing, reacting to events, and interacting with the environment. In this new dynamic landscape, it is necessary to have an adequate architecture that can integrate heterogeneous information streams and provide services with an acceptable quality to the users. The realization of an IoT framework needs to take into account many constraints related to the device (power consumption, network processing, battery lifetime etc.), to the stochastic nature of the underlying network (delay, bandwidth utilization, latency) and to the middleware overlay that is necessary to fuse big volumes of information streams and deliver a service to the user. This thesis proposes the design of a resource management framework which can monitor with no prior knowledge information streams produced by IoT devices, can predict changes with online mechanisms that can disrupt the performance of the IoT framework and can take actions to retain acceptable Quality Of Service while trying to save resources. The online, time optimized and distributed decision making models are based on Optimal Stopping Theory and Change Detection Theory applied on Edge, Communication and Middleware Layers. The findings of such decision making models are promising and solidly supportive to a vast spectrum of real-time and latency-sensitive applications with QoS requirements in IoT environments.

Keywords: Real-time Decision Making · Mobile IoT · Optimal Stopping Theory · Scene Detection · Group-of-Pictures · Change-point Detection · Unmanned Vehicle · Distributed streaming platform · Prioritization.

1 Dissertation Summary

Internet of Things (IoT) is one of the most promising paradigms nowadays characterized by the use of smart and self-configured objects, like sensors, actuators, wearables etc, that are connected to a network and exchange data by sensing, reacting to events, and interacting with the environment. The history of the IoT can be traced in the area of Ubiquitous computing and Wireless Sensor

* Dissertation Advisor: Stathes Hadjiefthymiades, Professor

Networks (WSNs). Mark Weiser proposed the idea of a smart environment: "a physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network". This idea is explored in the area of WSNs, where the goal is to build a system of many cheap computational components, called sensor nodes, wirelessly connected and jointly working towards a common goal. Sensor nodes have specific and, usually, small size, weight and cost. Going a step further and with the technological evolution, new physical devices with enhanced characteristics at both hardware and software parts are introduced daily, e.g. smartphones, wearables, unmanned devices etc, extending WSN paradigm to a "network", i.e. IoT, where every physical device is connected to the Internet ready to transfer data without requiring human-to-human or human-to-computer interaction. Especially robotic devices take part in IoT as long as they carry sensing equipment and on-board computing elements. IoT embodies a vision of merging heterogeneous objects while utilizing the Internet as a backbone of communication to establish interaction among physical and virtual entities. These seamless interactions among heterogeneous objects enable ubiquitous and pervasive applications. Most of these applications pose many challenges due to constrained resources in these miniature and unattended objects.

The technical challenges of the IoT can be identified in several areas:

- **Heterogeneity:** Connecting trillions of devices in the same network is not an easy task. The heterogeneity of the involved devices makes it even more difficult, since many different physical interconnections and system architectures can be expected. These differences can cause problems to certain communications.
- **Constrained resources:** A typical battery-operated IoT device possesses storage, processing, bandwidth, and energy as its resources. Since these resources are limited and the battery replacement is not feasible in many cases, therefore, various energy-efficient lightweight algorithms and protocols shall be being implemented to store, process and transfer the data as per application requirements.
- **Interoperability and integration:** The IoT is built by many distinct vendors, using various technologies. Their seamless integration can only be possible if IoT systems are built on top of open standards. There may be multiple standards for the same areas (e.g. different wireless networking standards), but interoperability between them has to be established.
- **Quality of Service**: With the advancements in embedded devices, the processing power of IoT devices is increasing day by day, but this results in increased energy consumption. To overcome that, IoT devices can rely on more powerful devices or servers for processing of data, but this introduces a delay in data processing and increases network delay and cost.
- **Computational and storage complexity:** The devices that comprise the IoT generate massive amounts of data. These data can be continuous or in bursts, and be in structured or unstructured form. In order to extract the

most from these data, they have to be transported, stored and analyzed. These operations put enormous pressure on networking, storage and computational infrastructure.
- **Security, Trust and Privacy:** The penetration of the IoT in daily lives emphasizes the need of proper secure solutions. The large number of devices involved makes the design of a completely secure system difficult, as there are many points of potential attack. Then, any solutions have to be portable to a wide set of devices, despite their intrinsic differences.

This thesis proposes the design of a resource management architecture that addresses the challenges of Heterogeneity, Quality of service, Resource constraints and Computational complexity applied on the Edge, Communication and Middleware layer. The Edge Layer is referred to the device and the computational complexity of different tasks. One daily energy demanding task in IoT is multimedia streaming, which causes the energy drainage to network resources and lifetime. Therefore efficient compressing methods are needed in order to minimize the consuming power but without harming the content of the distributed data. The Communication layer is based on wireless network technologies in order to enable interactions between various heterogeneous devices and information streams. At this layer information streams produced by heterogeneous sources are gathered in real time while taking into account the rational use of IoT devices. This mere data need to be combined in order to extract knowledge. At Middleware layer distributed data streaming solutions are targeted because they are extensively used to manage the big data flows of generated information streams by IoT devices. It is necessary these platforms to support reliable and timely communication despite poor performance of underlying units like lossy channels and failed components. At this these we design and implement online decision making models based on Optimal Stopping Theory in order to monitor the performance of units in different layers and predict disruptive changes. For example in edge layer during the multimedia compressing task a change can be defined as a scene change during the transmission of a multimedia sequence or an unknown object shown suddenly in the frame; a change in communication layer can be defined as a network blind spot of communication link during a flight of a drone in an unknown area. Changes trigger actions like the reconfiguration of the input system at Resource management layer. The main challenge is with no prior knowledge to monitor, predict changes and proactively act to sustain the continuous performance of a task efficiently without the energy drain of IoT devices.

At the first part we include our study of a content driven model applied to infrastructures with restricted resources like Wireless Sensor Multimedia Networks (WSMNs) in order to support multimedia application in such infrastructures [3]. Currently WSMNs are attracting significant attention due to the variety of applications in which can be applied such as traffic congestion, environmental habitat patient monitoring, etc. Although providing better quality for images and videos is necessary, it shortens the network lifetime as the energy battery operated sources are rapidly drained. Going inside the device, we propose a dynamic video encoding model that detects scene changes and tunes the synthesis of MPEG Group of Pictures (GOP) to meet Quality of Service objectives, i.e. transmit video sequences in acceptable quality, with a rational use of the IoT resources. The decision making process is based on Optimal Stopping Theory decision rule for the conclusion of the GOP in the encoder and the transmission of intracoded frames. We have used an MPEG-2 simulator. MPEG-2 simulators is enhanced with additional functions in order to support the creation of GOPs with dynamic size based on an OST rule.

The second part of the thesis presents a study of performance changes in communication links between IoT devices. The idea behind this research is the efficient monitor and control of unmanned devices operating in critical missions like natural disasters. We propose a real-time control mechanism to *adapt* to changes in network quality by dynamically *pausing* control telemetry and control messages based on *optimal sequential decision making rules* named as TOPCP-DRP [4] [2]. This is expected to ensure the trouble-free delivery of critical information subject to the dynamic network status that unmanned devices encounter while dispatching a certain mission. Our rationale is that should the network be performing properly, then the transmission control can be 'relaxed' to exploit the available resources in the resource-constrained IoT device. Our model introduces two sequential optimal stopping time decision making mechanisms based on the Change Detection theory (TOCP) and an application-specific discounted reward process (DRP).

At the final part we study the performance of a distributed message platform implemented as a midleware in an IoT system. The huge amount of data generated by sensor-instrumented objects of the real world in an IoT environment impose a great demand on processing and storage to transform the data into useful information or services. Some applications can be latency sensitive, while other applications can require complex processing including historical data and time series analysis. Therefore, considering the typical resource constraints of IoT devices, it is difficult to envision a real world IoT ecosystem without including a cloud platform or at least a distributed data streaming platforms. Distributed data streaming solutions manage big data flows of relevant data to/from devices, services and micro-services and are critical centerpiece of IoT deployments. These platforms are necessary in IoT infrastructures to process such enormous volumes of data against resource constrained IoT devices. The key challenges arise when supporting reliable and timely communication over constrained networks (e.g. due to lossy channels and failed components). To overcome these challenges, we propose a stochastic optimization framework of on-line control unit applied in the Publish/Subscribe of middleware data exchange platform, in our case Apache Kafka, adaptive to changes in performance of the studied platform [5]. We enhance our messaging distribution platform by applying prioritization policy of different types of messages when key performance indicators change. The optimality of the proposed mechanism is achieved by applying optimal delivery decision making policy in different priority queues. Optimal delivery decisions involves whether a consumer/producer in the device edge shall pause the pull/push requests in order not to overload a saturated message bus, to cause synchronization issues or to risk to completely lose the messages.

2 Results and Discussion

2.1 Optimal Stopping Theory applied on Time-optimized Grouping-of-Pictures

The performance metrics of the proposed encoder [3] with dynamic grouping of pictures of GOPs adapted to stream behavior are i) the produced error of the encoding process - SATD and ii) the size of generated video stream in bits. The dynamic grouping of pictures method is compared with a classic fixed-length version of an MPEG-2 encoder which creates a GOP with one I frame and then adds a constant number of P frames e.g. IPPPPPPPPPP. In our case the length of P frames is equal to 10. The pool of videos is downloaded from [1]. Every video was examined in a sequential stream of 100 frames and can be characterized as slow, medum and fast motion video. We use the abbreviation DGPE describing the dynamic grouping of pictures encoder for the gamma distribution and NDGPE describing the normal distribution.

Video	$DGPE_{GoPs}$	$NDGPE_{GoPs}$	$DGPE_{Size}$	CE_{Size}	$NDGPE_{Size}$
bridge	8	7	832477	876182	841583
waterfall	2	3	1499705	1642998	1500144
hall	9	15	1019986	1032109	1098023
container	15	11	2158422	2017824	2084643
foreman	6	9	2705621	2819314	2847462
football	27	13	6608428	6510336	6288473

Table 1: Size of bitstreams transmitted in network

The classic encoder (CE) created 10 fixed length GOPs. The number of GOPS created by DGPE and NDGPE are depicted in table 1. In the same table we compare the total size transmitted for each video (*inbits*) from the CE and DGPE encoders. We can notice that in slow motion videos the GOP size is extended in order to avoid unnecessary transmissions of I frames. For example in the waterfall video the number of GOPs is reduced to 2 and 3 per 60 frames in DGPE and DGPE respectively. In contrast in fast motion video the GOPs created are increased while the size of the generated bitstream stays below the generated bitstream of CE in average. It can be noticed that the volume transmitted in most of the cases from dynamic encoder is smaller than classic encoder. This is expected as fixed encoders are not content-driven and lead to waste of bits and resources. By comparing the dynamic encoders, we may notice that DGPE is more "sensitive" in fast-motion videos by capturing more scene changes than DGPE while DGPE shows tolerance to the medium motion videos. In addition, through Figures 1, 2, 3 and 4 we provide a comparison overview of

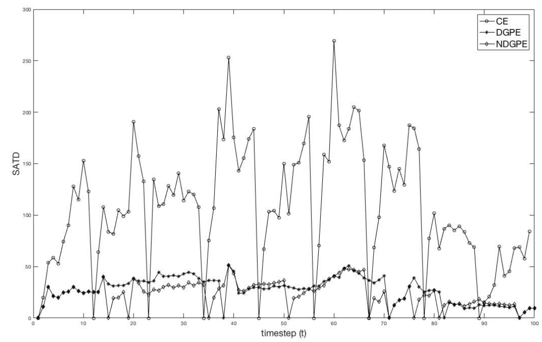
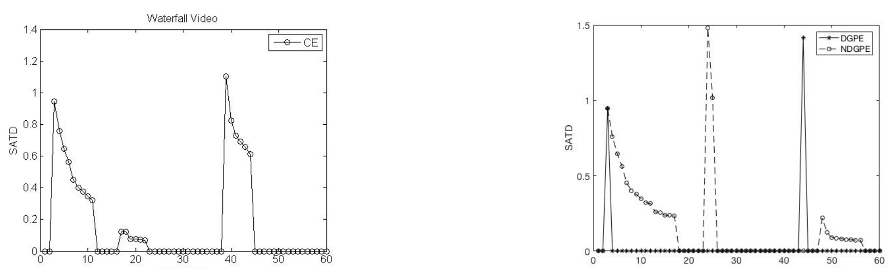


Fig. 1: SATD between classic approach and OST- football video

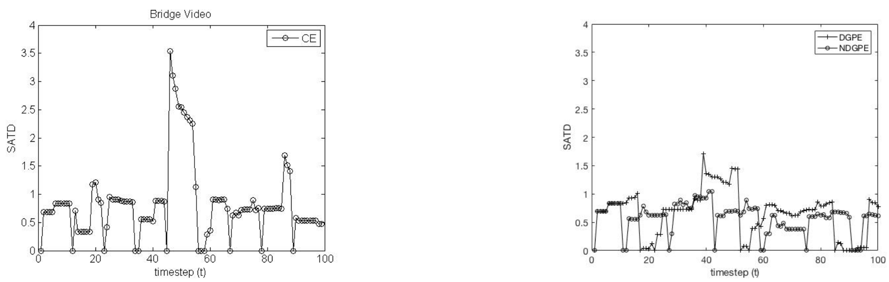
SATD measured between CE and DGPE. In Figure 1 it is observed that the error values coming from CE are higher than the dynamic encoders DGPE and NDGPE. The median value of SATD corresponds to 107.4 for CE. The median value of DGPE is 27.56 and 23.52 of NDGPE. The fewer GOPs created by truncated normal encoder also corresponds to a reduction of 4% in the total transmitted volume of bits as shown in table 1. In Figure 2, DGPE has the best



(a) SATD of CE (b) SATD of DGPE and NDGPE

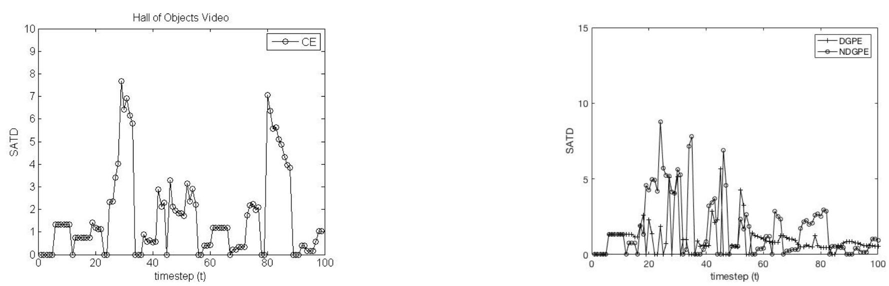
Fig. 2: SATD between classic approach and OST - waterfall video

video stream performance. The error remains close to the zero values. The first GOP is based on initial mean values of α and β and the next GOPs are based on the refitting of the design values to the incoming data distribution. NGOE needs time to fit μ and σ values to the slow motion video distribution. The mean and std values of the output error are the following: DGPE$\{0.0394, 0.2177\}$ and NDGPE$\{0.1625, 0.2934\}$.



(a) SATD of CE (b) SATD of DGPE and NDGPE

Fig. 3: SATD between classic approach and OST - bridge video



(a) SATD of CE (b) SATD of DGPE and NDGPE

Fig. 4: SATD between classic approach and OST - hall video

From the results in Figures 3 and 4 the encoder which uses normal distribution to compute t^* performs better than the other assessed encoders. We can notice that NDGPE needs more time to be adaptive to the changes of the incoming distribution but then SATD error values generated between the frames in the GOP created correspond to small values. For example at hall video the error values after the first 30 frames are quite low when compared with DGPE and CE methods.

From the description above, it shown that the dynamic encoders perform better than the fixed length encoder. The notion of adoption to video content is important as I frames are depended on scene changes and thus the encoding efficiency suffers from the error drifting on video transmission. The NDGPE shows better performance but a number of training incoming frames are required to fit the data distribution. If video streaming is quite short then the gamma-based encoder DGPE is the better candidate.

2.2 Real-time stochastic control mechanism adaptive to changes in network quality based on OST making rules.

We report the experimental evaluation of our framework and mechanisms to examine their performance. The unmanned ground vehicles (UGV) and the ground control station(GCS) are part of the Road-, Air- and Water based Future Internet Experimentation (RAWFIE) platform, which offers an experimentation framework for interconnecting numerous testbeds over which remote experimentation can be realized. Our TOCP-DRP [2] optimal mechanisms extended the functionalities of RAWFIE and can be applied to any mobile IoT device, i.e. aerial, ground and sea surface vehicles. The used UGV in our experiments offers the convenience to make multiple repetitions of the same experiment in the campus of the University of Athens, Greece, unaffected from weather conditions and with real users.

The UGV was used in two real case applications: (1) scanning search for a specific sensor value or a detection of an event designed by a user (mission 1-M1) and (2) exhaustive scan of a room (mission 2-M2). In both missions, the user creates a path and the UGV should follow the way-points in order to reach the final destination. The tested area was an amphitheater of the Department of Informatics & Telecommunications of the University of Athens and a corridor outside. During the execution of the experiments, the area was used from students and staff members that are moving around and their mobile devices are connected to the same WiFi network.

We performed 100 runs of 10 mins duration each, where each run involves sampling for more than $N = 100$ observations for every sensor integrated on UGV. The comparative assessment is based on four different policies of decision making: (i) the no-policy model, (ii) the heuristic threshold-based model, in which the transmission of messages is paused when Quality Network Indicator (QNI) falls below a threshold, (iii) TOCP model based on [4], which applies a change detection policy triggering the 'pause' mode operation (the passive mode lasts for Th and then it is activated again) and (iv) the hybrid TOCP-DRP model applied on both UGV and GCS. The performance metrics are QNI measured, Packet Error Rate (PER), based on packets sent and packets lost, and the end-to-end message latency. Figure 5 plots the QNI performance of the four policies. We can observe that in mission M1, two areas of poor connectivity exist in time-steps [35-45] and [75-90]. The no-policy, the threshold-based policy and TOCP policy reach QNI values less than 50%, while our TOCP-DRP policy has a mean value close to 68%. In addition, for $N > 60$ the TOCP-DRP is more intolerant to network changes with mean values around [70-85].

The PER maximum values are for all the policies: $\{no-policy, threshold-$ *based* policy, $TOCP$, and $TOCP-DRP\}$ are $\{25, 45, 15, 10\}$, respectively, with

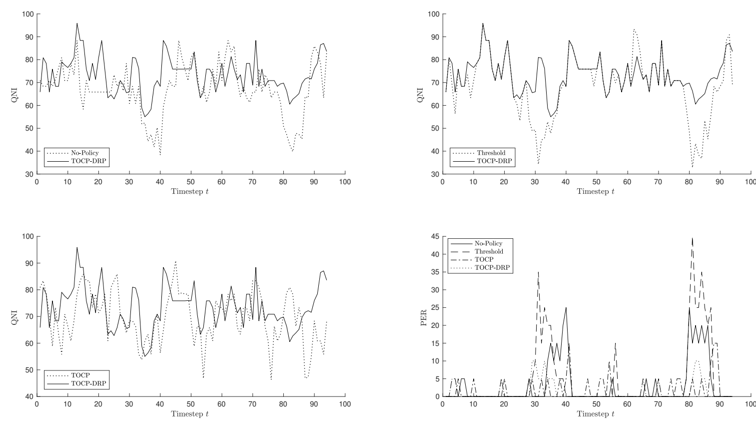


Fig. 5: The QNI for all the compared policies regarding the mission M1: Exploration of a Path.

TOCP-DRP achieving the minimum PER, i.e., we obtain up to 20% less PER compared with the other policies. The TOCP-DRP has better performance than the TOCP policy because TOCP overviews network data only in active mode and TOCP-DRP monitors QNI in both active and passive mode. The deactivation of passive mode in TOCP happens when the threshold is reached and this means that the algorithm is triggered in random time-steps independently of the network status. This is the reason for observing relatively small PER values every 50 steps when the algorithm recognizes a change detection.

Figure 6 shows the QNI performance of the four comparison policies for scanning missions. The M2 mission is performed indoors where areas of low connectivity and objects exist as obstacles to the UGV. The QNI has greater fluctuation in this mission relative to the M1 mission. Our TOCP-DRP mechanisms from the early beginning of mission M2, where UGV is positioned in one random corner of an amphitheater, outperforms the other policies. The average values of QNI for all policies: $\{no-policy, threshold-based$ policy, $TOCP$, and $TOCP-DRP\}$ are $\{68.4446, 70.8197, 65.8525, 76.3498\}$, respectively. The performance of the PER is similar to the M1 mission. The PER is minimized in our TOCP-DRP policy, where the maximum value is 10% in observations. In the remaining policies, the PER achieve values between 20% and 30% .

We plot the latency of the no-policy and our TOCP-DRP policy in Figure 7a(a) and Figure 7a(b) for the missions M1 and M2, respectively. The TOCP-DRP policy is considered more efficient than the no-policy for all the observations in both missions. In particular, in M1 we can measure 24% less end-to-end message latency compared to the original no-policy decision making of UGV. Moreover, the TOCP-DRP policy achieves systematically a message latency value which is close to 9% less of the original message latency. We can conclude that

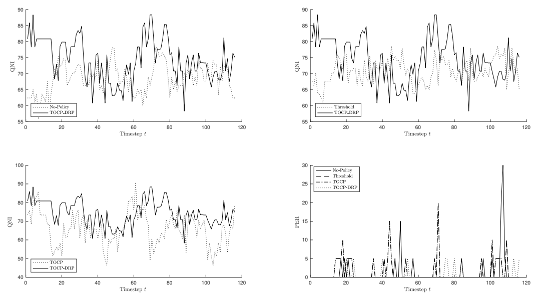


Fig. 6: The QNI for all the compared policies regarding the mission M2: Scanning of an unknown Area.

the double hybrid optimal stopping model in the two phases of the network, i.e., active and passive, based on the network assessment monitoring results to missions with low end-to-end latency and low expected PER.

(a) (b)

Fig. 7: The latency (ms) measured during the no-policy and the TOCP-DRP policy in mission M1 (a) and mission M2 (b).

The experimental performance evaluation and comparison assessment showed the successful delivery of messages in poor network conditions and the moderate production of messages so as not to burden an already saturated network.

2.3 Time-optimized prioritization of Kafka Message Scheduling for Unmanned Vehicles in IoT networks

To perform experimental evaluation in [5] two kinds of Apache Kafka clients were tested in both good and saturated network performance conditions. The first set (no-policy) of producers and consumers were simple Java clients producing and consuming constantly messages unaware of the network conditions to measure the performance of a non-priority Apache Kafka installation. A pair of producers and consumers were deployed in the cloud emulating a service that can be in the same datacenter with the Apache Kafka thus having a small delay by requesting or transmitting messages to the broker. Another pair of producer/consumer was placed on an edge device that was vulnerable to network conditions. Both producers and consumers were sending and receiving messages equally from all the topics without distinguishing the ones with high or low priority. In the opposite scenario (DMP), a pair of enhanced clients aware of the network conditions were installed on the edge device to measure the performance of priority queues. The Key Performance indicators (KPIs) performance of the distributed message bus was measured as the affine combination of the round trip time (RTT) and the Packet loss.

We have used a unmanned aerial vehicle (UAV) simulator and the role of control station was handled by a fixed server. UAV simulator and GCS are part of the *Road-, Air- and Water-based Future Internet Experimentation* (RAWFIE) [1] platform which offers a framework for interconnecting numerous testbeds over which remote experimentation can be realized. We performed 100 simulation runs with duration 10 minutes. In each second 2000 messages were produced and distributed in the message platform for each priority queue.

(a) (b)

Fig. 8: Decision making between states in S_n to S_s and then to S_n.

Figures 8a and 8b plot the packet loss measured in all the priority queues from Normal Performance State (S_n) to saturated state (S_s) and then back to (S_n). In both Low and High priority queues the DMP policy outperforms against no-policy. Especially in the High priority queue, in which essential information is exchanged for the user, the packet loss is less than 10% while in the no-policy the mean value of the packet loss is in range $[25\%, 30\%]$. The small improvement for DMP policy in the Low priority queue can be explained from the brokers' side. Brokers in DMP handle less "bursts" of messages in total that cannot handle due to bad network performance. The latency issues is more evident in Figures 9a and 9b. Applying the DMP decision making model the average value of delay in the messages successfully transmitted in Kafka message bus in saturated state S_s is between for High Priority Queue $[25-45]$ ms and for Low Priority Queue

[1] www.rawfie.eu

(a) (b)

Fig. 9: Delay in two priority queues between states in S_n to S_s and then to S_n.

between $[45, 60]$ ms. This is not the use case for the non policy model where the delay metrics is twice the delay of DMP. The performance evaluation showed the successful delivery of messages in poor performance conditions and the moderate production of messages of High Priority messages so as not to burden an already saturated queue which leads to loose completely the messages.

3 Conclusions

In this thesis decision making models were investigated which can monitor with no prior knowledge information streams produced by IoT devices, can predict changes with online mechanisms that can disrupt the performance of the IoT framework and can take actions to retain acceptable Quality Of Service while trying to save resources. The online, time optimized and distributed decision making models are based on Optimal Stopping Theory and Change Detection Theory appied applied on the Edge, Communication and Middleware layer of a multi-layer resource management architecture.

References

1. Hoelzer, S.: Xiph org. https://media.xiph.org/video/derf/ (2010 (Accessed: 2017-02-24))
2. Panagidi, K., Anagnostopoulos, C., Chalvatzaras, T., Hadjiefthymiades, S.: To transmit or not to transmit: Controlling communications in the mobile iot domain. ACM Transactions on Internet Technology (TOIT) - In press (2019)
3. Panagidi, K., Anagnostopoulos, C., Hadjiefthymiades, S.: Optimal grouping-of-pictures in iot video streams. Computer Communications **118**, 185–194 (March 2018). https://doi.org/10.1016/j.comcom.2017.11.012
4. Panagidi, K., Galanis, I., Anagnostopoulos, C., Hadjiefthymiades, S.: Time-optimized contextual information flow on unmanned vehicles. In: 2018 14th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). pp. 185–191 (Oct 2018)
5. Panagidi, K., Gerakos, K., Hadjiefthymiades, S.: Time-optimized prioritization of kafka message scheduling for unmanned vehicles. In: 21st IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM 2020). vol. under submission (June 2020)