

# **SORT:**

## **A System for Adaptive Transmission of Video Over Delay Tolerant Networks**

Abhishek Thakur, IDRBT, Hyderabad, India

### **ABSTRACT**

In challenging environments, opportunistic networks can provide limited communication features in a delay tolerant manner. It is extremely difficult to transmit large data like videos in such environments, as delay may be hours and part of the information may be lost. This article proposes a novel system that uses partial information from prior communication to estimate the network congestion and delay. The video is compressed and packetized using scalable video coding (SVC). Extensions to Spray-and-Wait routing protocols are analyzed to ensure better delivery video quality and lower wastage. Through simulation, including real-world traces, performance of the proposed solutions under multiple scenarios is evaluated. Experimental results show that adaptive control reduces overall delay and minimizes wastage while improving the quality of video at the receiver. Adaptive SVC transmissions demonstrate almost three times increase in decoded content, as compared to non-SVC transmission.

### **KEYWORDS**

Application Layer Adaptation, Data Forwarding, Delay Tolerant Networks, Media Aware Network Elements, Scalable Video Streaming

### **1. INTRODUCTION**

Wireless ad-hoc networks and mobile ad-hoc networks (MANET) can have network partitions, because of node mobility. Delay and disruption tolerant networks (DTN) are a sub-field of MANET, where the end-to-end connectivity does not exist on a regular basis (Fall, 2003). While mobility broke MANET, the contacts between mobile nodes completes the network for DTN over time (which can be in the order of hours or even days in some cases). Instead of store-and-forward techniques, these networks rely on store-carry-forward. Such a communication approach may also use multiple replicas of the content to ensure better delivery rates and lower delays. This replication can lead to resource exhaustion (e.g., bandwidth, storage or power in case of portable devices). DTN based routing has been widely researched (Cao & Sun, 2012), with applications in defense, disaster management scenario, inter-planetary networks, etc.

Similar to the evolution of networks, video communication technology has also progressed. Early digital video networks would typically broadcast the video, for prior published content and one-way broadcast of live events. Digital TV offerings (including digital cable TV, Satellite-based TV services) are early examples of this. Subsequently, as internet penetration increased, other platforms

DOI: 10.4018/IJWNB.T.20200701.0a2

like Netflix, YouTube, etc., started providing content individually to each user (i.e., unicast). Parallel to this, real-time video communication also evolved, initially in one-to-one mode and later in conference mode supporting multiple sources. Examples of such communication platforms are WebEx, Skype, FaceTime, WhatsApp, Duo, etc. In all these scenarios, end users view the video on heterogeneous platforms connected over a variety of networks. Since the display resolution, buffering and decoding capabilities may vary, the original video may need to be adapted for different receivers. SVC (Unanue et al., 2011) is one of the approaches, which uses the multi-layer encoding of content, to satisfy the needs of the different network and device capabilities. The base layer (BL) has the lowest demands on resources while providing a minimal quality. Addition enhancement layers (EL) help to improve the quality of the decoded video, on devices that have better resources like network bandwidth, screen resolution, processing power, etc.

There is a significant increase in opportunistic capture of content in multiple domains like law enforcement, disaster response, transport, defense, wildlife, agriculture, etc. Here the communication delay may be of the order of minutes to hours. Affordable smartphones and other portable devices with integrated camera have helped increase this trend. Trono et al. (2015) and Shibata and Uchida (2017) have explored multimedia applications, which are both delay and loss tolerant for disaster management scenarios.

In opportunistic networks, the destination may only receive parts of multimedia content. Further, the acknowledgments convey information only about that part of the network, over which the successful delivery of content and acknowledgment has happened. Such path limitations mean that both the source and the destination can only get a partial view of the network. This partial view of the network is used to adapt subsequent transmission from the sender, to maintain video quality without overloading the network. The algorithm adapts three aspects for subsequent transmission – SVC operating points (i.e., number of SVC layers transmitted by source); replication count for different layers; and time-to-live (TTL) for different layers. Savings from the adaptation can be redistributed, by increasing the copy count of the transmitted layers. The adaptation is purely on the end host and does not rely on a modification of the routing protocols. This paper uses Spray-and-Wait (SNW) as the routing protocol (Spyropoulos, Psounis, & Raghavendra, 2005).

The proposed system is named SORT (SNW based adaptive video transmission using operating point, replication count and time-to-live). For scenarios where it is feasible to have media-aware network elements (MANE) (Schierl et al., 2007), authors implemented layer awareness to SNW routing for all nodes and analyzed its performance.

This work extends authors prior work (Thakur, 2020). The additional contributions of this paper are:

1. Experiments and analysis of overheads and buffer occupancy;
2. Explore the impact of extensions to SNW routing protocols on SVC media flow;
3. Analyse the impact of different scenarios (including mobility pattern, load and node density) on SVC media flows.

This work can be applied to multiple domains that may need opportunistic video capture. For large transitional gatherings, the opportunistically collected video provides excellent input for trend analysis, including posterior monitoring and investigations (Trono, 2015). Monitoring of events in remote locations (e.g., elections or exams in sparsely populated areas) is another application where such an approach, backed with tamper-resistant local storage for few hours can provide an efficient enhancement for ensuring that no malpractices take place. This work can also be coupled with energy saving approaches like those proposed by Celebi et al. (2019), for infrastructure based dense 5G solutions.

The structure of rest of the document is as follows: Section 2 is an overview of scalable video compressions and prior work done regarding the transmission of video over delay tolerant networks. Section 3 presents the details of the proposed systems. Section 4 covers the experimental

setup including the aspects of different videos and simulation scenarios. Section 5 presents the experimental results and analysis of the same. Section 6 provides the conclusion and discusses the scope for further work.

## 2. PRIOR WORK AND MOTIVATION

Video communication over opportunistic delay tolerant networks is an active area of research. Additional motivation for this work comes from Ad-Hoc networks (Lindeberg et al., 2011) and peer-to-peer networks (Abboud et al., 2011). Following sections cover a brief overview of a) video compression including scalable video and related technologies; b) video packaging and communication; c) video transmission over DTN and finally d) motivation for SVC on DTN.

### 2.1. Video Compression and Encoding

The raw video takes tremendous bandwidth and hence needs to be compressed or encoded. The choice of encoder provides a trade-off between three things: the size of compressed content, computation and storage complexity during encoding/decoding and the distortion in the video post-decoding. As computation and storage capabilities have increased, the newer encoding schemes use algorithms, which have a higher demand for computation or storage. Video/visual coding experts' group of ITU-T developed the early digital video compression standards and published them as H-26X series with the primary aim of real-time video communication. The moving picture experts group developed compression schemes for pre-recorded content. Subsequently, the two teams have combined and developed some of the encoding standards together. Presently high-efficiency video coding (HEVC or H-265) is among the state-of-the-art video compression standards published by this combined team. The earlier encoding standard is called advanced video coding (AVC/H264).

For heterogeneous devices and networks, video compressed at one resolution or bitrate may not give the optimal quality for a variety of receivers. To solve this problem, three broad approaches exist – 1) transcoding on the network, 2) simulcast and 3) SVC and multiple-description-coding (MDC). For transcoding, the network needs significant computing resources to adapt the content to the receivers needs. Simulcast involves simultaneous transmission from the source on to multiple channels (e.g., using IP multicast, link layer or application layer multicast, etc.). The simulcast receiver selects the optimal channel based on its capabilities and the current network performance. For pre-recorded content, there is an alternative to simulcast where the receiver pulls the content. The source publishes content in chunks, at different bit rates. The receiver first downloads the details about the chunks. It then downloads the first chunk and based on the time taken to download the prior chunk, adapts the bitrates for subsequent chunks (Stockhammer, 2011). SVC generates multiple layers of video. The base layer carries the data for lowest resolution (spatial), lower framerate (temporal) and highest quantization scale (lower quality). Subsequent enhancement layers help improve the spatial, temporal or quality aspects. Quality improvement is also called an improvement in signal to noise ratio (SNR). An alternative to SVC is MDC, which has been analyzed in detail by Kazemi, Shirmohammadi, and Sadeghi (2014). MDC includes redundant information in each stream such that receiver can decode any of the streams independently. The destination node can further improve the decoded video quality, as and when it receives more MDC streams.

For opportunistic networks, one cannot rely on network infrastructure for transcoding. Unless distributed, the transcoder may become a single point of failure. This scheme expects relay of content via the transcoding node(s), thus increasing the delivery path length. The delivery quality (delay and loss) will degrade because of longer path length. Given the fact that many of these network elements may be resource constrained, computation complexity of transcoding will reduce the active lifetime of such nodes. Moreover, the bandwidth from the source to the transcoder will be very high. Because of such issues, one cannot use on-network transcoding approaches for opportunistic networks. Simulcast creates additional packets for the network and may make sense in information-centric deployment for

data dissemination in DTNs (Sobin et al., 2016). Even in such scenarios, the redundant information between the Simulcast sessions will be a huge overhead for the network.

As mentioned before, SVC cannot decode successfully received higher layers, when the base layer or lower layers in SVC are not available. MDC does not suffer from such issues. Irrespective of the stream received, MDC can decode the video, because of redundant information within each stream. When comparing SVC and MDC for peer-to-peer streaming, Abboud et al. (2011) in context of P2P infer that SVC is better in scenarios where the network can prioritize the lower layers. Usage of MDC (with redundant information across the stream) will cause additional overhead during replication-based routing. By leveraging higher replication to protect the lower layers (BL may have double the copy counts as compared to first enhancement layer), SVC is more adaptable than MDC.

Scalable HEVC or SHVC (Boyce et al., 2016) as compared to the SVC (Unanue et al., 2011) adds enhancements for bit-depth, color gamut, and hybrid codec. Moreover, HEVC has inbuilt support for SHVC layers using the high-level syntax (HLS). HLS allows for lower overheads in packaging the layers in SHVC when compared with SVC. This work utilizes SVC and H264 to align with some of the reference work. Moreover, because of the lower overhead of SHVC, the results for SHVC should be an improvement over SVC.

In SVC, video scaling can improve frame-rate (temporal), resolution (spatial) or quality (SNR). For temporal scaling, enhancement layers increase the frame rate. Enhancement layers for spatial scaling increase the resolution for the frames. SNR scaling reduces the quantization scale for enhancement layers, thus reducing the residual error for the decoder. One or more of these scaling approaches can be combined to get multilevel scaling.

Based on the type of video, and choice of scalability order, multilevel scaling can provide different residual errors (Li et al., 2010). E.g., when a surveillance video is captured by the static indoor camera (say in a shopping mall or banks), it may be meaningful to move first on spatial scalability, followed by temporal scalability. This is because most of the information across the scenes stays static, hence missing the temporal aspect, does not affect the SNR values. For such videos, it may be more important to get detailed features of objects, rather than a smooth motion for the mobile objects. On the other hand, for entertainment-focused video recording from a moving vehicle or for a sports event, it is more advisable to have temporal enhancements at lower layers, while higher layers may bring in spatial enhancements. The video to be too jittery to watch if temporal enhancement layer is delayed for such content.

## 2.2. Scalable Video Packaging and Communication

This work explores video communication for opportunistic DTN; hence, topics for regular Internet-based applications or ad-hoc networks are not discussed in detail. Singhal et al. (2014) discuss physical layer and encoding optimizations in multicast scenario for base station or access point like deployments for heterogeneous user groups. Lindeberg et al. (2011) and Trestian, Comsa, and Tuysuz (2018) provide a good survey on these aspects. A brief context for video transmission and the alternatives for opportunistic networks is covered below.

One can broadly classify video communication in three categories, based on the demand of end-to-end interaction and delay– 1) live-interactive transfer with round-trip delays below second; 2) live or pseudo-live communication based on application needs; and 3) transmission of pre-recorded content. For the first two categories, on regular Internet, mostly RTP/UDP based approach is used (Handley, et al., 1997). For both AVC and HEVC (and by that virtue SVC and SHVC), encoded content is packaged with network adaptation layer (NAL) unit. Encoder embeds the parameter used for encoding as well as encoded pictures within the NAL units (NALU). The NAL units on RTP/UDP packets for communication. Transmitters try to ensure that the UDP packet size is small enough to avoid fragmentation when the transmission is over the internet. Each UDP packet contains one or more NAL units of an SVC layer. Actual UDP communication may be multicast (one-to-many) or unicast (one-to-one). Transmission of the RTP/UDP content can use different communication

channels (ports) for NAL units of each SVC layer. The receivers can use approaches like RSVP (Handley et al., 1997). to detect the available bandwidth and get access to as many SVC layers as possible, starting with a channel for a base layer. In the absence of RSVP, lower layer packets may be assigned higher priority so that these packets encounter a minimal drop and have less delay in the network. To achieve such behavior from the nodes on the network, special values in IP header field may be used, or edge routers may have more involved approach using MPLS, etc. The source, using feedback received from clients, adapts the encoder. The receiver implements the network behavior detection while the source does the control of video encoding.

When transmission can be delayed by a few seconds, video can be broken into chunks and stored. These chunks are subsequently transmitted to regular Internet applications. E.g., DASH (Stockhammer, 2011) stores video in chunks of few seconds at different bitrates and the transfer happen from the server to client using TCP. Client downloads and buffers newer chunks, while it continues to decode and playout prior chunks. Since the server and clients have round-trip delays below a second, the client can choose lower or higher quality for subsequent chunks, depending on the time taken to download prior chunks. In this case, the client does the congestion detection and adaptation. Note that the default DASH approach is different from SVC since the source creates different chunks for different bitrates and the client chooses to download only the chunk that it deems fit, as per its adaptation state. Grafl et al. (2013) explore an extension of DASH involving SVC.

As compared to legacy wired networks with static nodes, there is a higher impact to media flow for wireless networks, because of interference, lower bandwidth, and node mobility. Abdulkadir et al. (2019) explore heterogeneous wireless networks but utilize a central SDN controller to decide if a stream of media should be adaptively rerouted to an alternate path. Wireless networks, especially Ad-hoc networks (including MANET) may use link layer optimizations and other network coding approaches (Lindeberg, 2011) to ensure a higher quality of base layers. On the other extreme of Ad-hoc networks are Vanets where the nodes have frequent connections and disconnection. Wu et al. (2014) have explored VANET for routing of the video stream in a dense scenario (more than ten hops and using ten thousand taxi nodes and a radio range of 250 meters). Quinlan, Zahran, and Sreenan (2015) have explored an adaptive layer distribution where the lower few layers of SVC are packaged and transmitted using MDC and measured the impact of loss for such communication across a single hop network.

As compared to few seconds delay for Ad-hoc networks, delay-tolerant opportunistic networks may have delays of hours or more. Hence, the adaptation control from the receiver may be too late, and source may end up underutilizing or congesting the network. Lack of feedback (e.g., acknowledgment getting lost) can further amplify such issues. Instead, it makes sense for the source to estimate the network behavior using acknowledgments that it receives and adapt subsequent transmission. The adaptation needs to ensure that it is neither too fast while reducing the SVC layers, nor too slow in adapting to increased load on the network behavior.

### 2.3. Routing Over DTN

Cao and Sun (2012) provide an excellent overview of numerous DTN routing protocols. One of the classifications involves single-copy vs. multiple-copy routing. Single copy routing frequently involves custody transfer of bundle and works for a controlled environment with oracle based (or strongly predictable) connectivity – e.g., deep space networks or public transport systems that follow a strict schedule. Such DTN deployments can use single copy-based routing, as one can reasonably predict the future location, of the satellites or vehicles with high confidence. At the other extreme of single-copy, routing is epidemic routing, wherein the replica of original bundles are shared (relayed), at each contact with the assumption that one of the nodes will eventually get in contact of the destination and deliver the data. This approach works well for opportunistic networks where little correlation exists between node contacts (Nayyar et al., 2018).

One should note that epidemic routing works well when payloads are small, or the count of nodes is less. As the payload on networks increase or node density increases, epidemic routing creates significant overhead on relaying bundles (higher node count) and usage of buffers to store the bundles (higher payload). In case some correlation exists in the contact patterns of nodes, many proposals exist to minimize the number of the replica. E.g., multiple social contacts-based routing approaches attempt to exploit features like community, between-ness, centrality, etc. when deciding whether to relay a bundle or not. Some of the other approaches rely on the history of encounter times and transitivity of the same between nodes (e.g., Prophet). Another group of algorithms attempts to solve the routing challenge as a resource optimization problem across nodes in a distributed manner (e.g., RAPID, MaxProp, etc.). These approaches try to find the utility of the other node in successfully delivery of the content and give good results when such a correlation exists, the computing and sharing of information on contact add to the complexity of these protocols.

One of the most widely used approaches to limit the overheads of Epidemic routing is using spray-and-wait (SNW) proposed by Spyropoulos et al. (2005). Here the source application decides the maximum number of copies that can exist on the network (denoted as  $L$ ). SNW allows any node to relay content only if it's copy count is more than one. When copy count is one, a node can only deliver the content to the destination. When compared to epidemic routing, there is only a small overhead of tracking the notional value of  $L$  for each bundle.

## 2.4. Video Over DTN

Prior work of video streaming over DTN has been mainly in the context of distributing content or streaming video over oracle-based delay tolerant networks. In some cases, video streaming is done using DTN enabled VANETs. Lenas et al. (2015) have proposed a bundle streaming service using single copy forwarding. They also verified the proposal for the single-copy routing on the real-world experimental test bed. Morgenroth et al. (2011), Blanchet (2012) and Cabrero et al. (2012) have demonstrated media streaming capabilities in a controlled environment with a limited number of nodes. Raffelsberger and Hellwanger (2015) adapted DTN to HTTP for video streaming in a simulated scenario of an explosion in a chemical facility. Such approaches do not fit well for opportunistic networks where the contacts are intermittent, and count of nodes may vary. Pan et al. (2016) have explored routing hierarchical video in opportunistic networks. They have optimized video streaming, by aligning buffer management of intermediate nodes. However, making the DTN forwarding strategy optimized for hierarchical media flow puts other applications at a disadvantage.

Sandulescu et al. (2015) in their work have compared the performance for a video like flow between the stationary source and destination (e.g., public library and airport in one of the use cases) for bandwidth estimation to periodically send video bundles based on the estimates of available capacity. They have not included continuous media flow and video quality measurements. The experiments relied on data from prior round of transmission, with a large time gap between the transmission rounds (e.g., transmission at 06:00, 14:00 and 00:00 hours). Moreover, their work focussed on demonstrating the accuracy of bandwidth and delay estimation, rather than proposing a complete system for streaming video chunks.

Klaghstan et al. (2013, 2014, 2016) have analyzed the performance of SVC using multiple layers. While (Klaghstan, 2013) maps SVC layers to separate bundles, (Klaghstan, 2014) and (Klaghstan, 2016) attempt media flow using NAL units. In (Klaghstan, 2013), they have analyzed the delay and delivery ratio based on single transmission over different network scenarios and observed that the base layer should have higher copy counts, and the higher enhancement layers would have lower copy count. Their subsequent work (Klaghstan, 2014) using NALU implemented media-aware network elements (MANE) that combined multiple NALU into a DTN bundle and exchanged them during node encounters. The size of the bundle was estimated based on the contact duration spread observed by a node. All nodes transmit base layer NALU in a single bundle without forcing it for fitting it into optimum bundle size. (Klaghstan, 2016) explored improving the delivered quality further by

implementing pull requests from the receiver to fill intermediate SVC layers. If the receiver has got a significant number of NALU for higher SVC layer but missed some NALU for intermediate layer(s), it solicits the retransmission from nodes in the network. The nodes in vicinity created additional copies of NALU based on the solicitation requests. To avoid overload on the network, the solicitation is triggered only after a threshold. The threshold is determined based on expected play out time and percentage of NALU received for intermediate layers. In all their experiments. (Sandulescu et al., 2013, 2014, and 2016), they ran the simulation multiple times, but each run involved only one burst of video transmission.

When transmitting a series of video bundles, some of the DTN bundles from the source may not reach the destination because of multiple reasons. One of the most commonly studied reasons for the delivery delay or failure is related to congestion or excessive loads on some nodes (Cao & Sun, 2012). If some nodes run out of space when receiving a bundle to be forwarded, they may drop older bundles. Even if large buffers are available, the contact duration between nodes may constrain and limit the exchange between nodes. In some cases, delay and drops may occur because of the deployment scenario, e.g., location and mobility patterns of the source and destination nodes are such that they have very few contacts with other nodes.

### 3. PROPOSED SYSTEM

Based on analysis of media flow (both scalable and non-scalable) across different scenarios authors identified the following goals as ideal for media transmission on opportunistic networks:

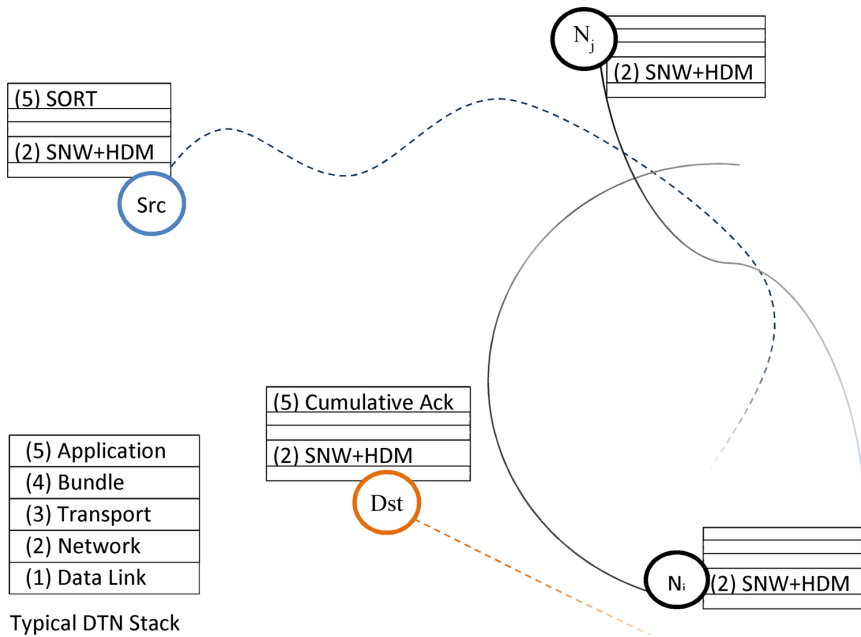
1. **Align with end-to-end principles (Blumenthal & Clark, 2001):** This implies that the optimization of media flow should not require explicit support from DTN routing layer;
2. **It should be possible to use optimizations from the network (if available):** To improve media flow performance, as long as it does not impact other (non-media flow) communication.

As mentioned in the prior section, multiple DTN routing protocols exist. Some of them are quite complicated and may cause unpredictable feedback with end-to-end adaptation. Hence binary SNW is chosen for adaptation at source node, for its simplicity. The only extension to SNW for intermediate node was to accommodate cumulative acknowledgment, wherein on receipt of the newer acknowledgment, the node discards the older acknowledgment.

Another key decision point is the mapping of communication units to SVC layers or NAL units. Some of the prior work, e.g. (Klaghstan, 2014) and (Kloghstan, 2016) have directly mapped NAL units to payload for transmission over DTN. This required changes to all DTN nodes so that they pack/unpack the NAL units into bundles as needed. This packaging comes with significant processing costs. Since each video frame will generate at least one NAL unit (besides multiple SVC layers and compression parameters will add more payloads), such mapping will create a significant overhead for continuous media streaming. Typically, DTN is not-live (delay of the order of hours). Hence authors chose to periodically chunk the video and create a burst of DTN bundles for every video chunk. The source will transmit each SVC layer for the chunk as a separate bundle. This mapping of video chunks to bundles is similar to the communication approach suggested in Klaghstan (2013) and Sandulescu (2015) and conformant with end-to-end principles. Except for source and destination, other nodes need not be aware of SVC mappings and chunk details.

Figure 1 provides an overview of the changes proposed in this work. Note that across different scenarios, the source (S), destination (D) as well as multiple intermediate nodes (Ni, Nj, etc.) may be mobile. End-to-end adaptation by SORT only involves application layer at the source and destination nodes. The destination generates the cumulative acknowledgment. All other nodes can have regular SNW implementation. Later part of this section discusses extensions of SNW, where nodes give

Figure 1. The DTN stack and its modifications for different nodes with the indication of their path



preference to higher copy count (H), delete delivered messages (D) and MANE (M). Figure 1 captures these optional extensions as HDM.

The source (Src) generates a burst of DTN bundles carrying different SVC layers. Intermediate nodes (N), relay / carry these bundles to the destination (Dst). On receipt of the bundle at the destination, a cumulative acknowledgment is generated capturing the details covered in Table 1. Figure 2 depicts such a media flow over time.

Note that Figure 2 does not capture the mobility of nodes. The source node adapts the next burst based on acknowledgments that it has received. Intermediate nodes relay the bundle. If the SNW extensions for higher copy count (H) is enabled, the choice of the relayed bundle gets affected by the value of copy-count(L). As and when destination node gets the SVC bundle; it generates a new cumulative ack. The nodes share the cumulative acks. On receiving a new ack, nodes may trigger the SNW extensions for MANE(M) and delete(D), if they are enabled.

This section first discusses the video packaging and transmission bursts of bundles with different SVC layers from the source. Subsequently, the logic of cumulative acknowledgment is covered. The algorithm for the estimation of state (congestion, delay) and subsequent adaptation from source is covered after that. Finally, the optional extensions to SNW are covered.

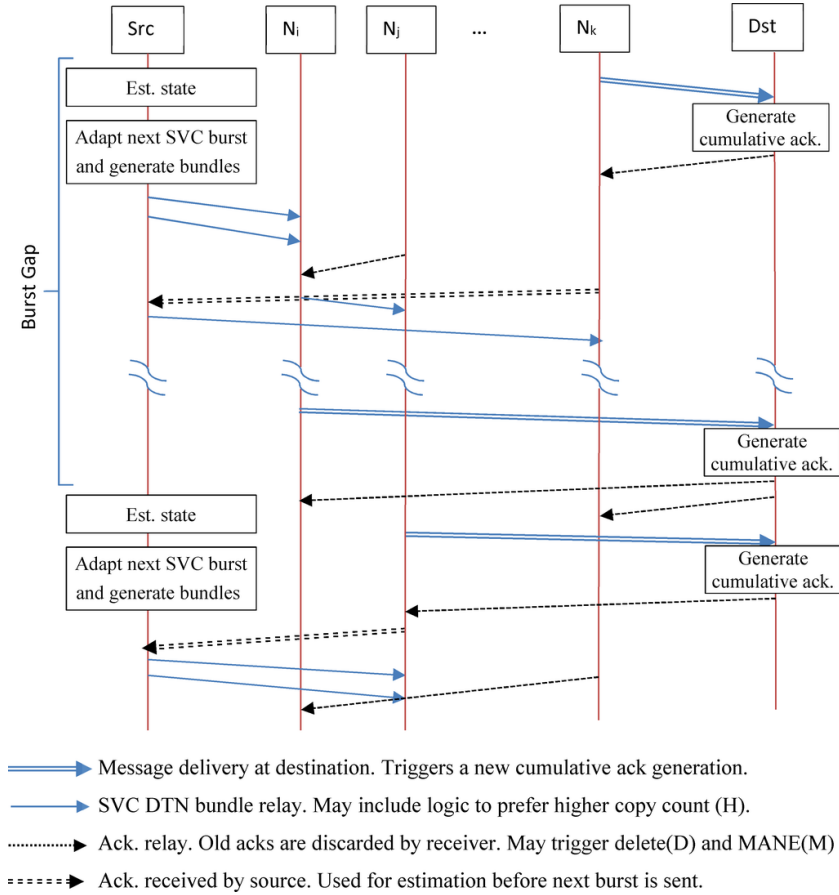
### 3.1. Video Packaging and Transmission

For communicating video over delay tolerant opportunistic networks, the video is captured in chunks of few seconds to few minutes and subsequently compressed using SVC. Each of the SVC layers is transmitted using different max-copy-count threshold (L). Similar to Klaghstan (2013), the base layer gets highest counts, and copy counts are gradually reduced for higher enhancement layers. The source generates a burst of DTN bundles every few minutes. The inter-burst-gap will be same as the duration of the video chunk.

The receiver (destination) will have application and deployment specific delay targets, for playing back the received video, from the time the source sends it. The source will set the time-to-live (TTL) at a value much higher than the delay target (e.g., twice that of delay target). TTL values in DTN



Figure 2. A temporal illustration of various components in the system



ensure that in the absence of acknowledgments, the intermediate nodes do not continue to cache the data forever. They delete the bundles whose TTL has expired.

### 3.2. Cumulative Acknowledgment

DTN routing has an optional feature to send an acknowledgment. The acknowledgments are small bundles that may be flooded on the network or communicated using the same routing protocol as the one carrying the application payload. These acknowledgments can be used to delete delivered messages if the feature is enabled similar to Cao and Sun (2012) and Ding et al. (2018). While earlier work used separate bundles for acknowledgment of each delivery, the authors have implemented cumulative acknowledgment to increase the chance of delivery for the acknowledgment to the source of media flow.

The cumulative acknowledgment includes the delivery delay for last few bundles in the media flow between the source and destination pair. If the source and destination nodes have clock skew below few seconds, the data shared in cumulative acknowledgement will not have major impact. Since the acknowledgments are cumulative, missing some of the intermediate acknowledgments will not significantly impact the estimation at the source.

Table 1 provides a simplified representation of the metadata within the cumulative ack. Note that the cumulative acknowledgment differs significantly from TCP cumulative acknowledgments. It allows holes to exist in the sequence, there is no expectation of retransmission of missing bundles,

**Table 1. Metadata in cumulative acknowledgment from the destination**

Source of ack: 17 {identity of Dst}						
Destination of ack: 44 {identity of Src}						
Cumulative ack Sequence number: 109						
Oldest burst id: 41						
Entries for delay in minutes	Burst Id	BL	EL1	EL2	...	ELN
	41	74	73	-1	...	-1
	42	94	-1	33	...	-1
	..	...	...	...	...	...
	90	76	-1	77	...	...

and it includes the delivery delays for each received bundle. For example, in Table 1, for burst 41, EL1 had a delay of 73 minutes, while EL2 was not received when the ack was generated. Using simple bitmaps and rounding off for the receipt delay to closest minutes, one can communicate cumulative acknowledgment for last four hundred bursts in less than one kilobyte. Application of other lossless compression schemes can reduce this data further. The design of optimal compression scheme for these acknowledgments is outside the scope of this paper.

On receipt of a new acknowledgment bundle (with higher cumulative ack Sequence number, for the same source-destination pair), the intermediate nodes discard the earlier acknowledgment. The SNW routing first attempts to deliver the bundles to the destination then exchanges the acknowledgments before forwarding other bundles.

### 3.3. Streaming State - Congestion and Delay Estimation

The following algorithm is used to estimate the likely congestion and delay between source and destination in the DTN network. Note that the computation of streaming state uses only the data about SVC layers and bursts of transmission between the source and destination pairs. It is not an estimator of congestion in the overall DTN network.

Table 2 covers the details of the constants used in the algorithms, which are discussed below. Based on the scenario and requirements for the deployment, application running at the source can set it. Note that the values used for these constants are based on limited experiments across different scenarios. Authors have not attempted to identify optimal value for them as they found that the optimum values are scenario dependent.

**Table 2. Algorithmic constant used in SORT and HDM**

Identifier	Description / Value
DT	Delay target -ideally playback at the destination is for a burst sent at (Tnow-DT)
BG	Burst gap – gap between each SVC burst
RMDP	Ratio of missed bundles penalty to delayed bundles penalty (value = 9)
EstChoice	The choice between average, minimum, maximum of the two estimates
MxLayers	Maximum number of SVC layers.
MxTtlDrop	Maximum factor by which TTL for the higher layer is reduced (default = 0.2)
ActualTtl	The default value of TTL

The estimate of congestion using NumSample involves counting the number of delayed or missed base layer and enhancement layers. Since base layer is an absolute must for decoding at the destination, it is given half the weight in the estimate (line 5 of Algorithm 1). All enhancement layers contribute to the other half. Moreover, missed bundles are much worse than delayed bundles. Hence a higher weight is assigned to missed bundles using RMDP. Note that division by “ $2 * (1 + RMDP)$ ”, ensures that the estimate is range bound between 0 to 100.

**Algorithm 1:** getStatisticalEstimate( numSamples)

1. Tnow = getPresentTime()
2. mxBurst = (Tnow-DT) / BG
3. mnBurst = mxBurst - numSamples
4. Using prior received acknowledgments for [mnBurst to mxBurst ]
  - cBLmsd = count of base layer missed (no-ack);,
  - cBLdlyd = count of base layer delayed (delay > DT)
  - cELmsd = count of enhancement layer missed (no-ack);,
  - cELdlyd = count of enhancement layer delayed (delay > DT)
  - tELtx = total count of enhancement layers transmitted
5. 
$$\text{newEst} = 100 * \left\{ \frac{RMDP * cBLmsd + cBLdlyd}{\text{numSamples}} + \frac{RMDP * cELmsd + cELdlyd}{tELtx} \right\} / \{2 * (1 + RMDP)\}$$
6. Return newEst

**Algorithm 2:** estimateCurrentState ()

```

sSamples = DT/BG
shortTerm = getStatisticalEstimate(sSamples)
lSamples = (2 * DT) / BG
longTerm = getStatisticalEstimate(lSamples)
avg = (shortTerm + longTerm)/2
min = Min(shortTerm, longTerm)
    
```

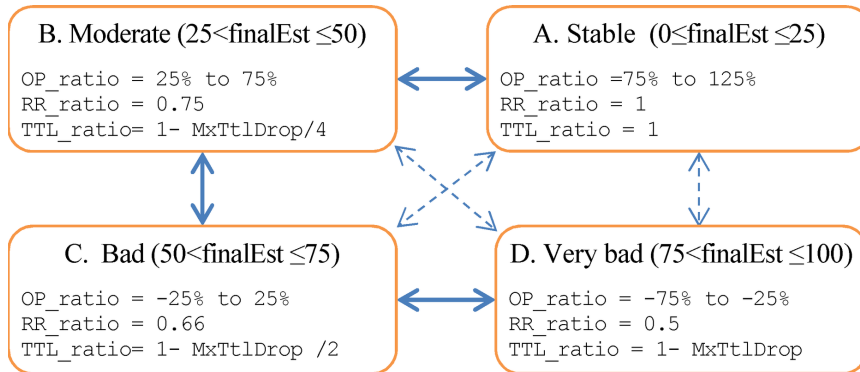
Since even the cumulative acknowledgments may be delayed or lost, sometimes the state may be wrongly represented at the source. Moreover, network behavior can vary over time (because of node mobility). Hence, two intervals (DT and 2\*DT) are chosen to estimate the congestion. The choice of average/min/max (EstChoice) of the two estimate is driven based on whether the SVC media flow can be aggressive on resource usage or not. Algorithm 2 combines the two estimates based on the value of EstChoice at the application level.

### 3.4. Source Adaptation Based on State Estimation

Based on the congestion estimate, the source may infer one of the following states; No Loss or Delay, Occasional delivery beyond delay target; Sustained loss at delay target; Sustained loss beyond delay target (at TTL). Figure 3 captures these states and adaptation values associated with each of them. Based on experiments, authors identified these adaptation values as a reasonable trade-off across different scenarios. The choice ensures that end-to-end adaptation reduces the load on the network for simple SNW deployments while giving significantly better results for decoded video quality. Algorithm 3 utilizes these adaptation values (OP\_ratio, RR\_Ratio, and TTL\_ratio).

In general, the transitions should happen along the solid edges of Figure 3, but in the presence of sudden changes in the networks conditions (say bursts between other nodes or extreme mobility of

Figure 3. The estimated state for adaptive video transmission



source or destination) alternate state transitions along the dashed lines cannot be ruled out. The finalEst from Algorithm 2 is mapped to State A [0,25]; State B (25,50]; State C (50,75] and State D (75,100].

One cannot apply conventional TCP/IP like flow and congestion control for opportunistic Video transmission, on three accounts. First of all, like the scenario of regular internet multimedia communication, Video transfer can be loss tolerant but would expect reasonable bounds on delay. Secondly, unlike TCP transfer, some parts of the video can be discarded by the source when it observes severe congestion. Finally, based on the congestion, it's possible to control replication overheads in DTN. The first few SVC bursts are transmitted without any adaptation. Subsequently, based on congestion estimate, the source adapts future bursts. Usage of SNW over DTN, provides the following three levers to adapt:

1. **O - SVC Operating point:** In the presence of severe congestion, the source does not transmit one or more of the higher layers. Operating point corresponds to maximum enhancement layer transmitted. Less number of bundles will imply smaller load on the network. OP\_ratio from Figure 3 is used to control this value in Algorithm 3;
2. **R - Replication factors of the bundles:** The source reduces the replication count for higher layers to ensure lower load on the network. RR\_ratio controls this value;
3. **T - TTL values:** The time to live for higher layers can be reduced to ensure that they spend less time in the buffers. When buffers are full, and some messages need to be deleted to create space, smaller TTL values for higher layers will ensure that nodes which are running out of storage, delete bundles for higher layers before deleting lower layer bundles. This adaptation uses TTL\_ratio from the corresponding state in Figure 3.

**Algorithm 3: adaptNextBurst()**

Initial values: eWMA\_PathLen = MaxLayers

est = estimateCurrentState()

eWMA\_PathLen =  $(1-\alpha) * \text{eWMA\_PathLen} + \alpha * \text{MxLayers} * \text{OP\_ratio}$

ensure eWMA\_PathLen is below MaxLayers and above 1 (base layer).

for (layer = 0; layer < MxLayers; layer++)

    if(eWMA\_PathLen < layer) break

    ttl[layer] = ActualTtl \* (TTL\_ratio) ^ layer

    if( layer > 0 and layer >= eWMA\_PathLen - 2)

Furthermore, when the source adapts O/R/T, it can measure the amount of traffic load being taken off the network and redistribute (when ReDist=true in Algorithm 3) the same for lower layers. Typically, the lower layers of SVC are smaller in size. Hence ReDist can increase the copy counts for the base layer and lower enhancement layers, thus improving the delivery rate and reducing the delivery delay for them.

To ensure that changes in path length are gradual, an exponential weighted moving average is done for operating point using  $\alpha$  as 0.2 as captured in line 2 of Algorithm 3.

Assume that five SVC layer is used in transmission (MaxLayers=5) and prior eWMA\_PathLen is at 4. Let us consider the impact of Algorithm 3 when the estimate is at 98 (state A, OP\_ratio = 121%) and 35 (state C, OP\_ratio = -5%). For state A, the new value for eWMA\_PathLen will increase to 4.41 ( $0.8 * 4 + 0.2 * 5 * 1.21$ ) and hence logic on line 5 will transmit four layers. There is no impact to TTL, and copy count in State A since both TTL\_ratio on line 6, and RR\_ratio on line 8 is at 1. On the other hand, for estimates leading to state C, the eWMA\_PathLen will reduce to 3.15 ( $0.8 * 4 - 0.2 * 0.5 * 0.05$ ). Hence, it sends only three layers (BL + EL1, EL2). Moreover, for enhancement layers, copyCount and ttl will reduce. Value for ttl[1] will drop to 90% while it will be 81% for ttl[2]. Similarly, copyCount will reduce to 66% for enhancement layer 1 and 2.

### 3.5. SNW Extensions

While SORT adapts SVC media flow without changes on intermediate nodes, authors explored alternatives to improve delivery of content, by making the following three changes to SNW routing. The first two extensions are quite generic and would not give undue resource allocation to SVC media flow:

1. **H - prefer higher copy count:** When two nodes are in contact and need to relay bundles, vanilla SNW proposes a random order of relaying or relaying based on earlier packet first. For SORT+SVC, lower layers will have higher copy counts. Hence authors extend SNW to order the relay of bundles, based on the count of replica (L) that is held by a node;
2. **D - delete on acknowledgment:** On receiving a cumulative acknowledgment, intermediate nodes delete the existing bundles for the delivered SVC layers;
3. **M – media-aware network elements:** Source based adaptation is a push-based model and does not allow the destination to make pull requests. In case some of the higher layer bundles are received at the destination, with missing intermediate layers, it makes sense to increase the counts for missing intermediate layers to ensure that they also are delivered. Only after delivery of missing intermediate layers, prior received higher can be used for decoding. When MANE extension is enabled, the nodes periodically check the acknowledgments to identify the gaps. MANE uses a timer-based threshold and doubles the replication count (L) for bundles matching the missing intermediate layers.

The MANE implementation creates additional copies of some bundles, on intermediate network nodes, when it detects gaps in delivery to destination. This deviates from the network-neutrality principles and breaks the end-to-end paradigm. Such hard coupling of application on all the network nodes negatively impacts other applications that may be deployed on same network.

## 4. EXPERIMENTAL SETUP

The experiments, simulate the flow of SVC bursts in different network topologies using open source simulator ONE (Keränen, Ott & Kärkkäinen, 2009). Based on the bundles received at the destination, it measures the peak-signal-to-noise-ratio (PSNR) for various delay targets. Thereafter, similar to (Klaue, Rathke & Wolisz, 2003), the PSNR values are mapped to mean-opinion-score (MOS). This mapping is primarily to simplify the plots (since single curve represents the value interpreted from the nine SVC curves).

#### 4.1. Input Video Characteristics

The experiments utilize publicly available test video data Highway with 2000 frames at 30 frames per second ([media.xiph.org/video/derf/y4m/highway\\_cif.y4m](http://media.xiph.org/video/derf/y4m/highway_cif.y4m)) and CIF resolution (352x288) as video load. JSVM [9.19.15] is used to create nine-layer SVC content with layer-specific details as captured in Table 3. Further x264 (0.148.2643) is used to create a non-scalable video with a target bitrate of 100 kbps. The resultant file is 836 KB in size with luminance PSNR of 28.7159.

Author also experimented with seven-layer SVC, where temporal scaling at QCIF (BL + 4 temporal EL) was followed by spatial and quality scaling. Across simulation runs, the trends for seven-layer plots were similar to those for nine layers, and hence they are not included in this work.

The chrominance PSNR values (U, V) were always above 38 dB across the SVC layers for this video set., Hence the results have only used the luminance(Y) PSNR as generated by the PSNRStatic tool of JSVM.

#### 4.2. Simulation Settings

Table 4 captures the three different simulation scenarios used in the experiments. Besides, variants of RWP with different node densities, delay targets and burst gap are used in experiments.

The network interface (range and speed), delay target, default TTL and buffer size values are chosen for RWP to align with the settings used by Klaghstan (2013). For SFT and WDM, they were updated to ensure that reasonable communication could happen even without the optimizations proposed. The intent was that for non-SVC video without using the adaptation or SNW extensions, approximately half the transmissions reach the destination within the delay target.

For real-world traces (SFT) the cars move quite frequently, but their paths are different, and they cover a significantly larger geographical area. Hence the choice of Wi-Fi and similar delay target as RWP suits this model. In case of WDM, the nodes travel from home to office and sometimes to evening activities after work. Since there are large periods where the nodes stay static (office in the day, and home in nights) the delay target was taken as 12 hours. Since the map area is large and nodes do not move for a significant period, Wi-Fi like interface is used in WDM scenario to create more contacts.

Each simulation is executed fifty times with different random number seeds. For analysis, bursts in the beginning and end of simulation runs are skipped, since the network is not saturated in these cases. Skipping of initial flows also provides adequate time for adaptation to settle itself. For RWP and SFT, the data analyzed is from bursts 200-299; while for WDM, the analysis is for bursts 432-

Table 3. Nine-layer SVC operation points for highway test video sequence

Layer	Size (KB)	Luma PSNR	Type	L at Source
BL	92	27.0608	176x144 at 1.875 fps	32
EL1	212	27.1757	352x288 at 1.875 fps	16
EL2	76	29.1398	352x288 at 3.75 fps	12
EL3	92	31.3958	352x288 at 7.5 fps	10
EL4	108	33.1269	352x288 at 15 fps	8
EL5	132	35.7351	352x288 at 30 fps	7
EL6	556	36.7759	Quality enhancement 1 (36-33)	6
EL7	1056	37.954	Quality enhancement 1 (33-30)	5
EL8	1232	38.7324	Quality enhancement 3 (30-28)	4

**Table 4. Simulation parameters**

	<b>Random Way Point (RWP)</b>	<b>San Francisco Taxi Traces (SFT)</b>	<b>Working Day Model (WDM)</b>
Mobility model for nodes	Shortest Path Map Based (Keränen, 2009)	Real world taxi traces (Piorkowski et al., 2009)	Working day Model (Ekman et al., 2008)
Node Details	50 nodes at pedestrian speeds	50 taxi nodes	2 Buses, 48 nodes
Map details	Helsinki with points of interest	San Francisco and its vicinity	Part of Helsinki - "Area A" (Ekman et al., 2008); 10 offices and 10 meeting spots
Mobility Speed and patterns	0.5 – 1.5 meters per sec.	As per trace data	50% car ownership
Network Interface (range, speed)	10 meters, 2 Mbps	50 meters, 24 Mbps	50 meters, 24 Mbps
Default Delay Target	2 Hours	2 Hours	12 Hours
Default TTL	4 Hours	4 Hours	24 Hours
Default Burst Gap	5 minutes	5 minutes	5 minutes
Number of Bursts	600	600	800
Buffer Size	100 MB for the source, 10 MB for all other nodes	100 MB for the source, 10 MB for all other nodes	600 MB for the source, 60 MB for all other nodes

719. Usage of 288 samples in WDM, ensures that execution cover a complete day (with the default delay of 5 minutes between each burst). SFT and RWP did not exhibit such daily pattern; hence, 100 samples suffice for them.

### 4.3. Video Decoding and Quality Measurements

The receiver may not get all the packets sent by the source. Some of the packets may be lost or delayed and would not arrive in time for decoding and playout at the receiver. To compensate for missed higher layers, the decoder may use multiple approaches like spatial scaling, repeating frames for temporal scaling, etc. (Unanue et al., 2011). If none of the layers is received, the last decoded frame can be repeated to compensate for missed parts of the video. Such compensation approaches improve the user experience, but the played-out video can be significantly different from the original content.

One can measure the quality of decoded video using multiple approaches (Klaue et al., 2003). Average value of PSNR across the video frames is one of the most common approaches. PSNR measurements can be automated, when both the original content and received data is available. Mean-opinion-score (MOS) is another metric used in multiple research work. Since MOS involves manual feedback collection from multiple users who watch the video, it is very costly to implement and can be prone to bias if not properly implemented. For automatically determining the received video quality, the PSNR values have been converted to mean opinion score (MOS) using heuristic mapping as proposed in (Klaue et al., 2003).

While DTN may deliver multiple SVC layers, PSNR measurement considers only the successfully received base layer and consecutive enhancement layer for each burst. If enhancement layers are missing in the spatial domain, lower resolution decoded frames are scaled to a higher resolution before computing PSNR. Similarly, if temporal frames are missing, FFmpeg (2.8.11) is used to generate the intermediate frames. For video bursts where the base layer itself is not received, MOS of 1 (bad) is used (Table 5).

Table 5. Heuristic mapping for PSNR to MOS

PSNR (dB)	MOS
>37	5 (Excellent)
31-37	4 (Good)
25-31	3 (Fair)
20-25	2 (Poor)
<20	1 (Bad)

#### 4.4. Cost for DTN Communication

To identify the savings on cost, one of the common metrics used in DTN applications is the overhead-ratio. This is the ratio of all bundles relayed and created across nodes, to the number of delivered bundles (see Equation 1). For SVC, given that the size of bundles varies significantly, this metric does not completely capture the associated cost. Hence, cumulative buffer occupancy (CBO) over time in MB-hours is used for analyzing the communication cost. Equation 2 computes CBO, by multiplying the time for which the bundle is in buffers of a node, with the size of the bundle, across all the nodes. Note that for SVC bundles created on source node  $ReceiveTime_{M,N}$  in Equation 2 will be the time of creation of the message:

$$overhead = \frac{total\ bundles\ relayed\ and\ created\ across\ all\ nodes}{total\ bundles\ delivered\ across\ all\ nodes} - 1 \quad (1)$$

$$CBO = \sum_{N=1}^{Num-Nodes} \sum_{M=1}^{Num-Messages} (DeleteTime_{M,N} - ReceiveTime_{M,N}) * Size_M \quad (2)$$

The total count of delivered bundles normalizes overhead computation. For lightly loaded networks if TTL is large, almost all the bundles are delivered. In the absence of MANE overhead for SNW will be upper bound to the values of L used by the source application. Similarly, CBO will be upper bounded to values used by source (at  $L \times TTL \times Size_M$ ) for the bundles created.

As network load increases, overheads and CBO can increase in two different patterns. If the bundles are deleted because the source (or nodes in the vicinity of source) run out of space, the “total bundles relayed” as well as “receive to delete time” will be small. Since bundles delivered to destination will be low in count, overheads will be high, but CBO will not increase significantly. On the other hand, if the nodes have large buffer capacity and node contact durations are large, (e.g., in case of WDM) then the bundles will have a higher count for relayed bundles. They will also stay in buffers longer. For such cases, the delivery rate affects the overhead, but the CBO values will stay high, irrespective of delivery rate.

## 5. PERFORMANCE EVALUATION

The results in this section have the following order, a) RWP scenario for consecutive bundles received and delay in receiving the bundles; b) the MOS based results for the RWP as well as WDM and SFT; c) communication costs using overheads and CBO for the three scenarios; d) impact on HDM and SORT for faster/slower bursts with smaller/larger delay targets; e) impact on SVC media flow



when the node density is varied; and f) the individual contribution from each of the components of SORT and HDM.

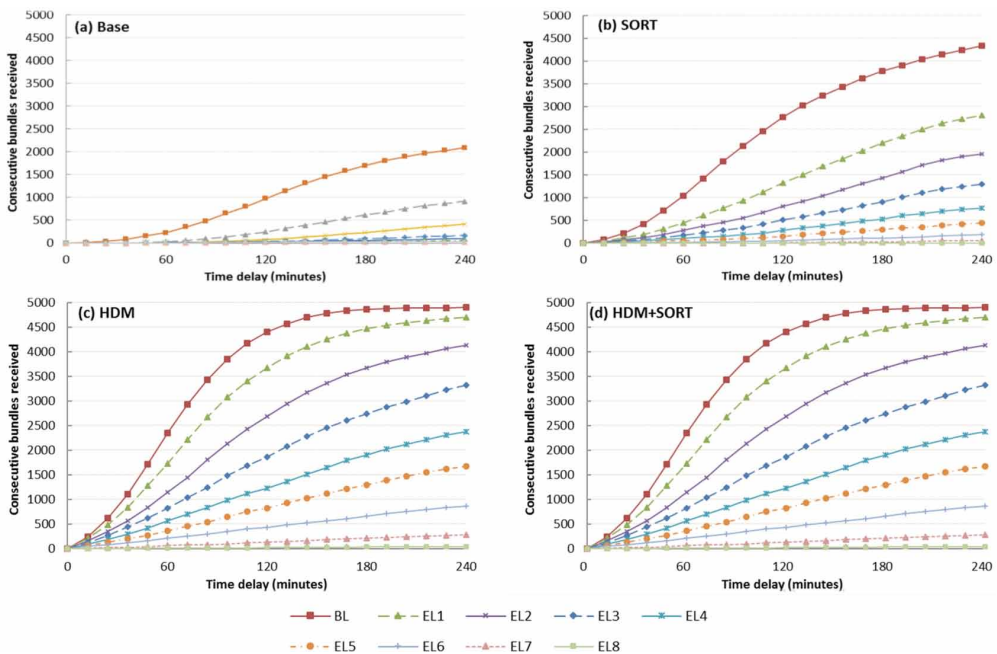
### 5.1. Base RWP Scenario – Consecutively Received Layers

Figure 4 captures the plot for RWP mobility scenario across four different settings. The subplots show the cumulative count for maximum consecutive layer received (for 5000 SVC bursts for Sequence 200 to 299 for fifty simulation runs) on the vertical axis against the delay in delivery on the horizontal axis. First sub-plot, Figure 4(a), shows the performance of nine-layer SVC video transmission with SNW implementation supporting only the cumulative acknowledgment. At two hours (default delay target) less than 20% of the bursts are decodable at the destination since BL bundles reached the destination for less than 1000 bursts. At the end of 4 hours, 42% of the bursts received at the destination are decodable. Similarly, at delay target, 23 flows (0.5%) reach EL5 or higher. Less than 51 bursts (1%) reach EL5 or higher with a delay of 4 hours. Note that the plots are cumulative – for example, the plot for EL5 is the sum of all bursts that are decoded till enhancement layer 5 or higher.

Figure 4(b) shows the performance when the source adapts the transmission. Higher enhancement layers (EL7 and EL8) still have very low counts. Values for all other layers have improved significantly. BL improved from 20% to 55% at delay target and 42% to 87% for 4 hours. The metrics for EL5 corresponding to a two hour delay is 3% and a four hour delay is 9%. At delay target of 2 hours, the destination could decode almost thrice the number of bursts for SORT when compared to the Base scenario.

Figure 4(c) shows the performance when the SVC bursts are transmitted with SNW extended to support higher copy count, delete delivered message and MANE. It does not include SORT adaptation. As compared to Figure 4(a), HDM improves the quality for all the layers. When compared to SORT, SNW-HDM shows improvements for all the layers. At delay target (120 minutes), BL achieves 88%, and EL5 achieves 16%. At the end of four hours, BL achieves 98% while EL5 achieves 33%.

Figure 4. SVC maximum consecutive layer received for RWP base scenario



Combining both HDM and SORT, as captured in Figure 4(d), improves the results at delay target to 89%(BL) and 18%(EL5). The values for a four hour delay is 98% for BL, and 32% for EL5. For this experiment, network-based optimizations (HDM) give better results than end-to-end optimization (SORT).

## 5.2. Quality (MOS) and Transmission and Storage Costs Across Scenarios

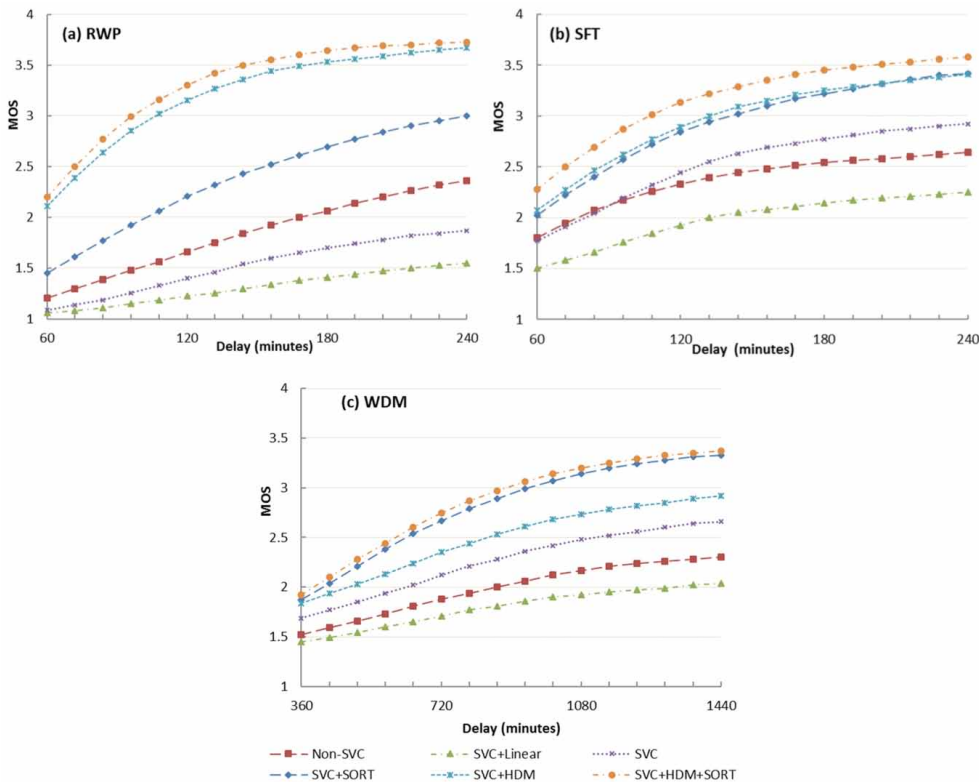
To properly understand the viewing impact of the layers that are received, the Figure 5(a), plots the above four scenarios with MOS values across delay in delivery. The baseline plots also include non-SVC video and SVC video using constant copy count of eight for all layers (labelled as SVC-Linear). For ease of visualization, plots are from  $0.5 * DT$  (60 minutes) to  $2 * DT$  (240 minutes). Fig 5(b) and 5(c) are the corresponding plots for SFT and WDM scenario.

Comparing SVC to SVC+SORT shows that at delay target, across the scenarios, MOS score improved by 0.5 to 0.8. The improvements are much more significant at TTL (twice the delay target).

Combined HDM + SORT perform the best for all scenarios. HDM outperforms SORT in case of RWP, while it is the other way around for WDM. This is because the infrequent contacts between nodes in WDM limits the role of HDM. In case of SFT, both HDM and SORT are very close to each other. Between the non-adaptive versions, SVC-Linear is the worst in all scenarios, as it does not give higher copy count to the base layer.

For SFT and WDM, just moving from Non-SVC (single layer, 836 KB content with  $L=16$ ) to SVC (9 layers with more than 3 MB in total, but with BL of 92 KB and  $L=32$ ) improves the quality even when HDM and SORT are not enabled.

Figure 5. Mean opinion scores for RWP, SFT and WDM scenarios



In Figure 6, CBO is lowest when both HDM and SORT are enabled. Individually, SORT has highest costs for RWP and SFT since it does not benefit from delete optimization of HDM. In case of WDM, the costs on CBO are similar for HDM and SORT.

Since overheads only include relay-counts and do not give weight to the size of the bundle and time occupied by the bundle in the buffers, non-SVC, SVC, and SVC-Linear flavors show a lower value on overheads. But it should be kept in mind that MOS values for these runs are significantly poor than those involving HDM or SORT.

### 5.3. Load and Node-Count Related Performance

The message generation frequency and node density are varied, for RWP with both HDM and SORT enabled. Figure 7(a) plots results for 50 node RWP runs when video bursts are generated every 37 seconds, 75 seconds, 150 seconds, 300 seconds and 600 seconds while keeping all other values same as in Table 4. Figure 7(b) plots the MOS values for simulation runs while varying node density from 25 to 400 for the same map area. Figure 7(c) and (d) plot the costs regarding overhead and CBO.

At a high frequency of SVC burst generation, the decoded quality is low, the overheads are high, and CBO is low. The overheads are high because most of the content created at the source is deleted either on the source or in its vicinity. Source generates new bursts even before the prior burst's content

Figure 6. Communication and storage costs for RWP, SFT and WDM

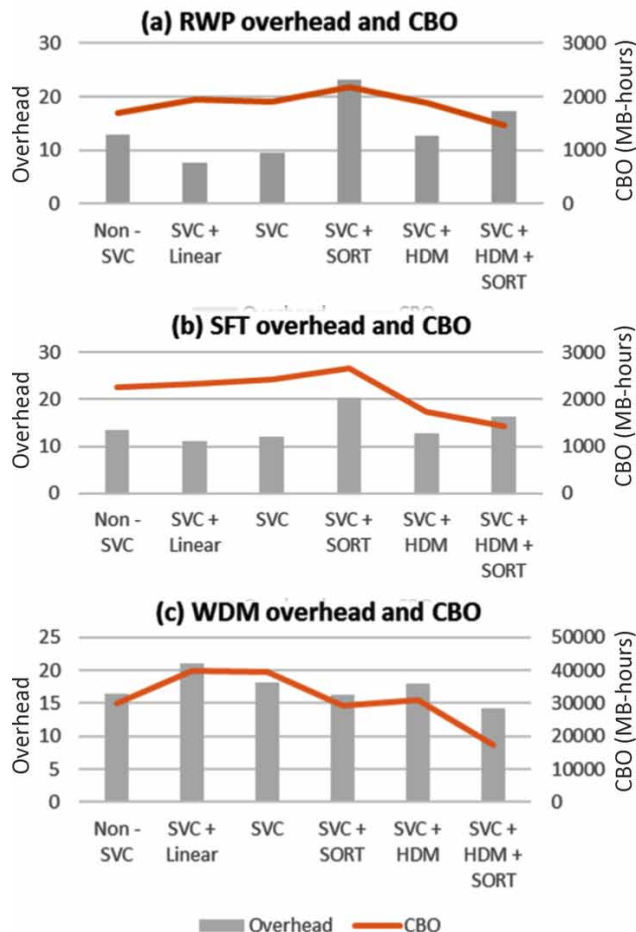
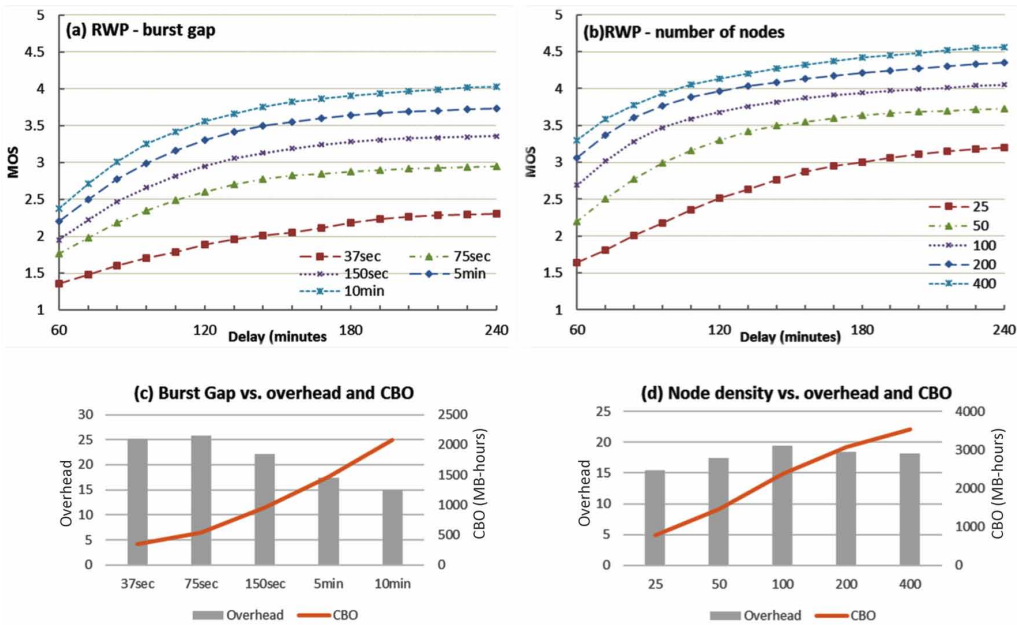


Figure 7. Impact of load and node density in RWP scenario



can be shared, thus causing buffers to become full at the source itself. Note that the source has ten times buffer space than other nodes; still, it suffers from buffer overflow for very high frequencies (37 seconds and 75 seconds) as the gap between contacts with other nodes is higher than the burst gap. For 150 seconds and beyond, overheads drop as more bundles reach the destination (and hence higher MOS scores) while CBO continues to increase since more messages reach the destination and the messages in transit stay in the buffer for longer intervals.

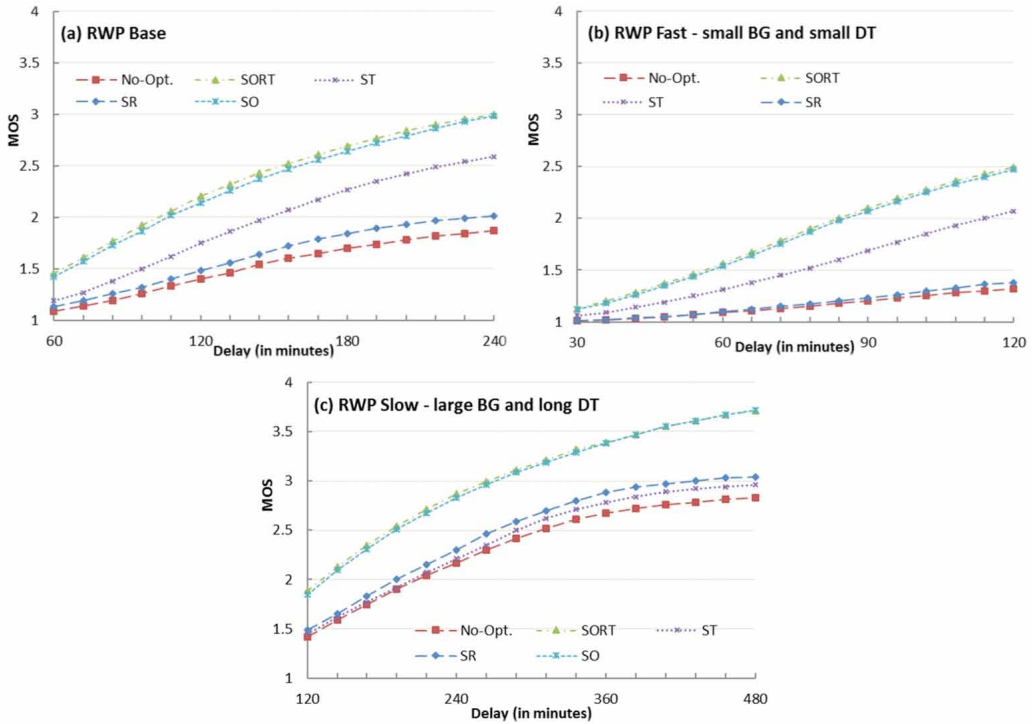
Figure 7(b) and (d) for change in node counts have a similar trend as Figure 7(a) and (c). When few nodes are present, the buffers at source or nodes in the vicinity of source quickly run out of buffer space. Hence they delete older bundles to accept newer bundles, leading to relatively high overheads (order of 15-18 for node counts of 25<sup>1</sup> and 50). For low node density, buffer space and relay opportunities are limited. Hence CBO values are low. Overheads taper off around 100 nodes or beyond since maximum copy count used for base-layer is 32. CBO increases slightly while delivery rates increase significantly for 400 nodes, taking the MOS scores above 4.5.

#### 5.4. Analysis of Components in SORT

This sub-section analyzes the impact of Operating point control, Replica control, TTL control, Re-distribution and choice of estimation. HDM is disabled in these runs (except for ReDist, where the variance is more when HDM is enabled). To analyze the impact of load and delay targets, in addition to the RWP setting captured in Table 4, two additional variants of RWP are used in these runs. *RWP-Fast* generates at twice the base frequency (at 150 seconds interval) and targets delay of 1 hour and TTL of 2 hours. *RWP-Slow* generates messages at half the speed (at 600 seconds) while it also doubles up the TTL and delay-target when compared to the base (to 8 hours and 4 hours, respectively).

As shown in Figure 8, in all the variations of RWP, for quality measured as MOS, operating point adaptation provides the maximum improvement. When all three components (ORT) are enabled, it provides slightly better performance than only the operating point. TTL control provides moderate improvements for Base and Fast scenario, while Replica adaptation provides small improvements for

Figure 8. Contribution to MOS from SORT components across different load and delay in RWP



Base and Slow scenarios. Algorithm 3 applies Replica control only on two highest layers. Hence, its isolated impact is smallest amongst the three.

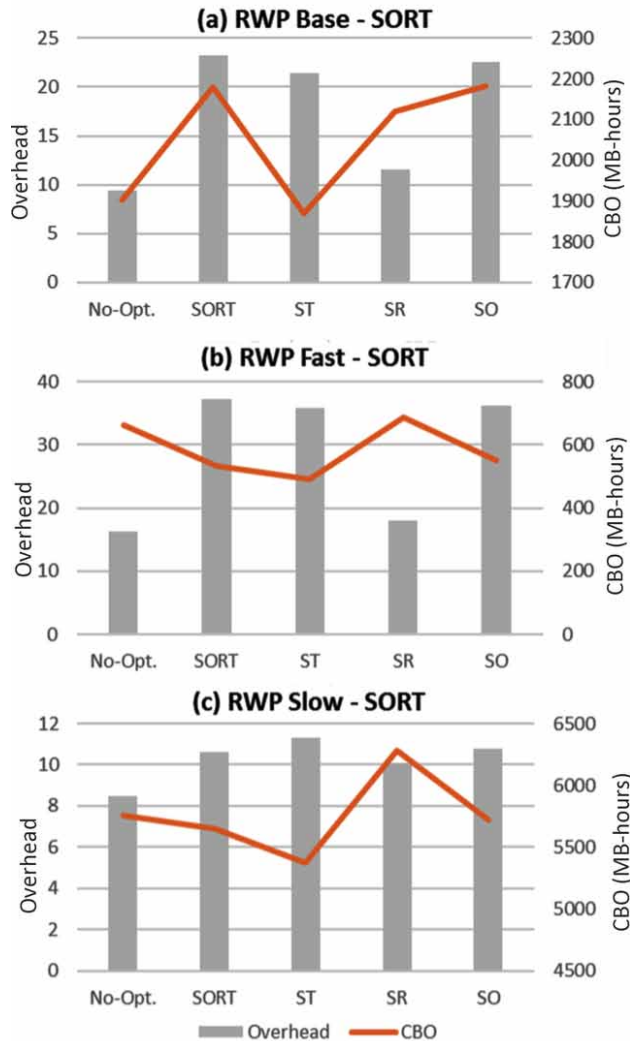
Figure 9 shows that replication control (R) provides maximum savings on overheads for Base and Fast scenarios. But CBO values are relatively high for R since it keeps the higher layers in the buffer (and HDM is disabled on the nodes). ST provides the maximum savings on CBO by reducing the TTL values when it detects congestion/delay.

Similar to MOS figures, when all three (ORT) are enabled, both the overhead and the CBO values are almost same as the values for operating point (O).

Figure 10 analyses the impact of ReDist estimation features of SORT. As expected, redistributing the copy counts to lower layers impacts the results only if delivery rates are low. In the Slow scenario, MOS scores as shown in Figure 10(a) stay the same irrespective of whether the algorithm redistributes or not. On the other hand, for Base and Fast scenario, ReDist increases the MOS while also increasing the CBO marginally. The overheads for ReDist are very high under the Fast scenario. In-depth analysis showed that RWP-Slow operates in State-A most of the time, while RWP-Fast computes the State as B and C most of the time. Algorithm 3 in State-A does not remove the higher layers; hence there is very little scope to redistribute. On the other hand, in State B and C, adaptation removes higher layers and reduces copy count as well as TTL for some of the enhancement layers. This increases the ReDist to lower layers. In RWP-Fast, the bundles are deleted closer to the source, they do not increase the CBO, but the overheads are high because of lower delivery rate.

When experimenting with choice of estimator, when HDM is disabled, the delivered quality was poor, and all the three estimates (min/max/avg) were very close. Fig 10(b) and (d) plots the results for the estimate as min/max with HDM enabled on the network. The plot of avg. was in between these values, hence its removed for ease of visualization. Note that minimum estimator pulls the adaptation towards State-A, thus creating more overheads and increasing the MOS scores.

Figure 9. Communication and storage costs for SORT components



### 5.5. Analysis of Components in HDM

This sub-section identifies the improvements from the three enhancements in SNW - the preference of higher copy count (H), deleting delivered messages (D) and media aware network extension (M). To better identify the contribution from HDM components, SORT is disabled for these experiments.

Figure 11 shows that maximum improvements are because of H (preference of higher copy count). As discussed before, this is because lower layers have higher copy counts and hence enabling H, helps relay the lower layers faster. MANE provides slight improvement for Base and Slow scenarios, while D (delete) does not provide any noticeable gain in MOS score. However, when all three (HDM) are combined, there is a slight improvement in quality across the three simulation runs.

Deleting by itself is not expected to improve the MOS score as it comes into play only after bundles are delivered. Analysis of logs showed that the delivery rate increased when delete feature was enabled, but it delivered bundles for multiple layers, and hence the destination did not see any appreciable increase in consecutive bundles received for different bursts. Note that improvements



Figure 10. Impact from ReDist and estimation for RWP scenarios

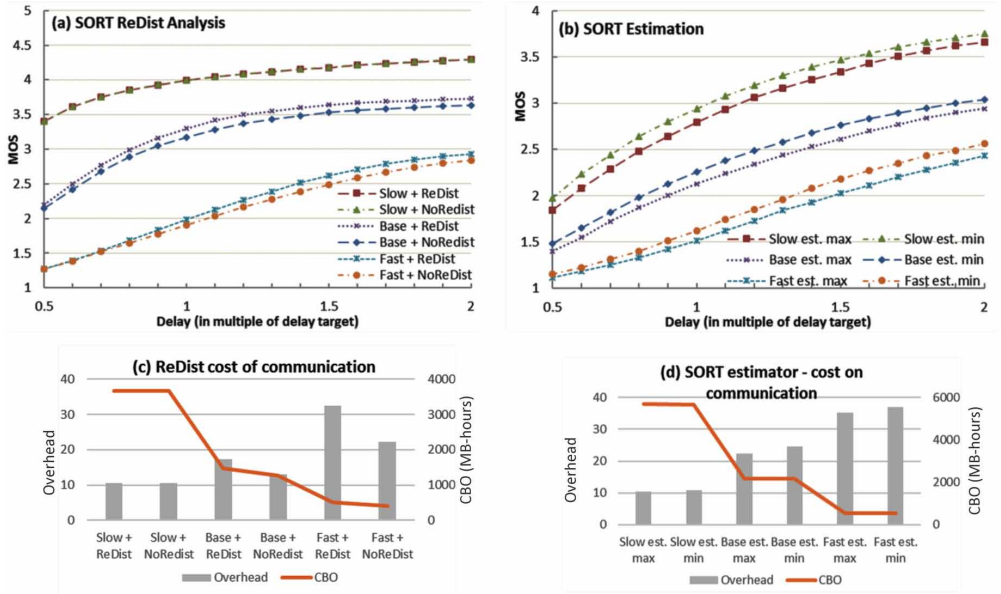
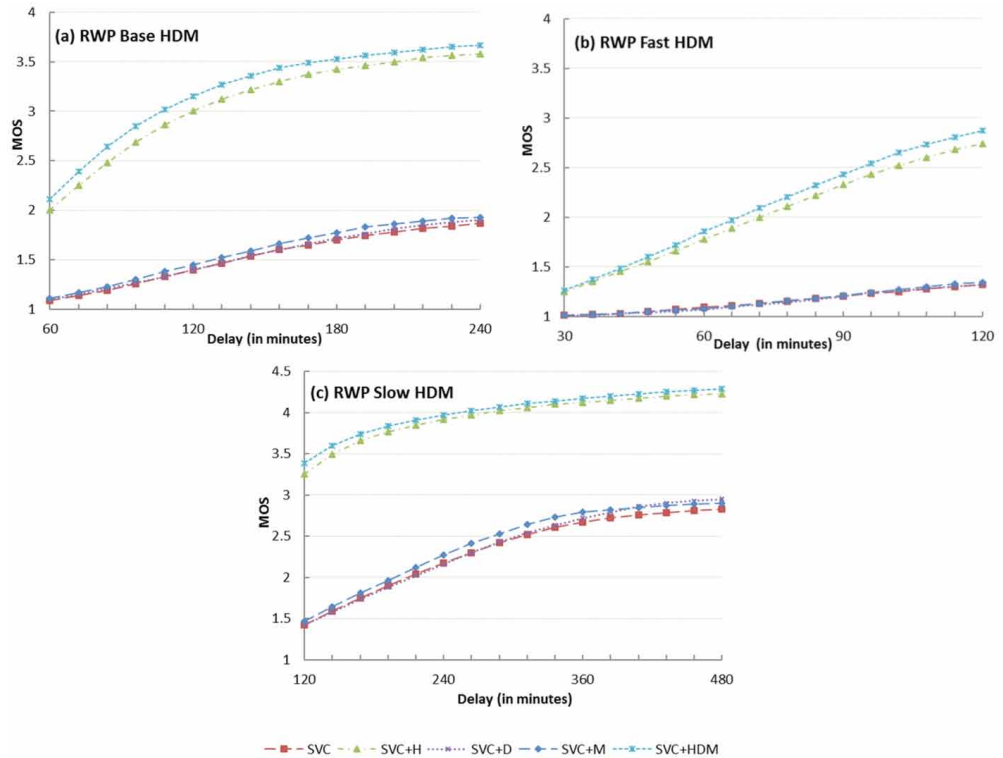


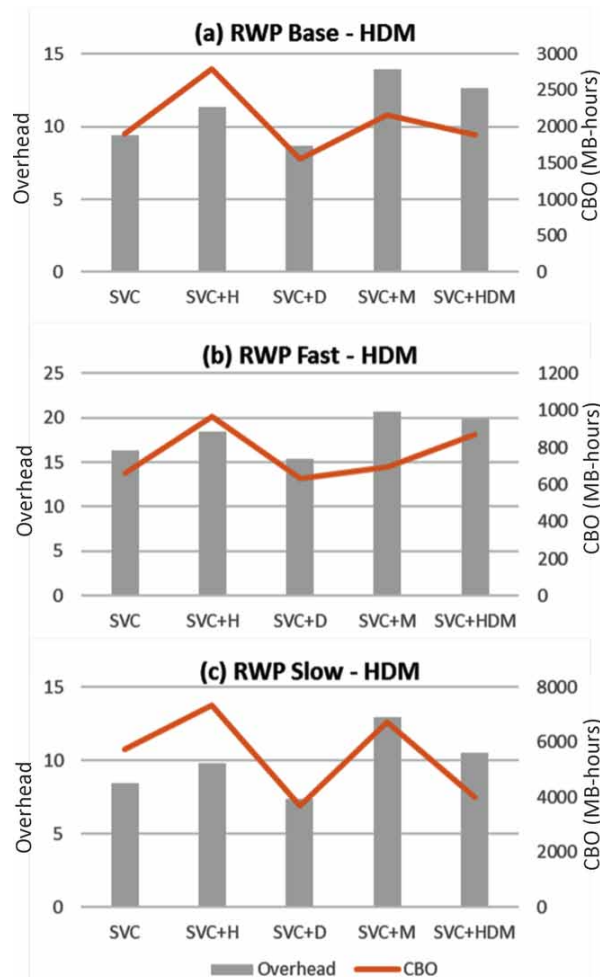
Figure 11. Contribution to MOS from HDM components across different load and delay in RWP



from MANE are lower than that observed by Klaghstan (2016). This is because DTN bundles are have mapped to each SVC layers for the burst. This mapping is much coarser (order of hundreds of KB) than the mapping to NALU, which are hundreds of times smaller. When MANE triggers for such large DTN bundles, the creation of additional copies causes deletion of some of the other SVC bursts, thus offsetting the improvement under heavy load.

Observations on the cost aspects show that “delete” provides a significant saving on both overheads and CBO. Overheads reduce since delivered bundles are not relayed further. CBO improves much more significantly as bundles are deleted as soon as an acknowledgment is received, and additional copies are not created. Figure 12(a,c) show that CBO value also benefits from Delete if all three components are enabled, and network load is moderate. Preference for higher copy count increases the CBO and overhead for all scenarios. MANE also increases the communication costs for Slow and Base scenarios, but for the Fast scenario, MANE does not have a cost impact as most of the bundles are deleted because of heavy load from the source, and MANE is not able to find a local copy for missing layers.

Figure 12. Impact of HDM components to communication and storage





## 5.6. Summary of Analysis

The operating point is the most prominent control for application-level adaptation and preference for higher copy count is the most beneficial extension on the network nodes. Note that different choice for OP\_ratio, RR\_ratio, and MxTtlDrop can make other components of SORT to dominate. On-network optimization for MANE provides benefits under low to moderate load while delete feature is an effective way to control the cost of communication.

Observations across scenarios showed that relying only on routing optimization or only on end host adaptation would not provide the best results. While RWP performs best for on-network optimization, WDM showed better performance, for end hosts adaptation-using SORT. Using both SORT and network optimizations always gave the best results.

## 6. CONCLUSION

This paper has proposed SORT, an end-to-end adaptive system to improve the quality of streamed video over delay tolerant networks, using SVC. The performance of the system has been analyzed using three different mobility scenarios. SORT provides significant performance gains when transmitting scalable video across different scenarios. MOS scores increased by 0.5 to 0.8 across different scenarios. Impact on video quality is analyzed with simple extensions to SNW routing. With SVC layers mapped to DTN bundles, when the source assigns higher copy count to lower layers, simple network optimization to prefer higher copy count during bundle relay, provides excellent improvements. In RWP scenarios, it increased MOS scores by 0.7 to 1.5, with low load scenarios demonstrating the maximum improvements.

An area to explore in the future is the application of SORT and HDM to information-centric networks, especially those involving multiple subscribers (multicast), as well as multiple media flows. The future extension of SORT can use machine learning or similar approaches at the source to estimate the congestion and get the best quality. Other scenario specific optimizations could also be attempted. For example, in WDM it is expected that usage of 24 hours earlier data rather than simple EWMA can provide better results. SORT did not try to optimize the adaptation for specific scenarios. Instead, it used a simple estimator-based approach without changing the algorithm constants for different scenarios.

SORT only considered changes to end-applications for optimization. Feedback from network elements (other than the destination) for adaptation can provide further optimization. MDC based video communications, especially with single-copy DTN routing (e.g., using custody transfer), is another possible area to explore in future. Impact on SORT and HDM by various network coding approaches and heterogeneous connectivity can also be explored.

## REFERENCES

- Abboud, O., Pussep, K., Kovacevic, A., Mohr, K., Kaune, S., & Steinmetz, R. (2011). Enabling resilient P2P video streaming: Survey and analysis. *Multimedia Systems*, 17(3), 177–197. doi:10.1007/s00530-011-0229-x
- Abdulkadir, I., Al-Jawad, A., Shah, P., Vien, Q.-T., Tuysuz, M. F., & Trestian, R. (2019). AROMA: An Adapt-or-Reroute Strategy for Multimedia Applications over SDN-based Wireless Environments. In *Proceedings of the 2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. IEEE Press.
- Blanchet, M. (2012). Postellation: an enhanced delay-tolerant network (DTN) implementation with video streaming and automated network attachment. In *SpaceOps 2012* (p. 1279621). Academic Press.
- Blumenthal, M. S., & Clark, D. D. (2001). Rethinking the design of the Internet: The end-to-end arguments vs. the brave new world. *ACM Transactions on Internet Technology*, 1(1), 70–109. doi:10.1145/383034.383037
- Boyce, J. M., Ye, Y., Chen, J., & Ramasubramonian, A. K. (2015). Overview of SHVC: Scalable extensions of the high efficiency video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(1), 20–34. doi:10.1109/TCSVT.2015.2461951
- Cabrero, S., Pañeda, X. G., García, R., Melendi, D., & Plagemann, T. (2012, October). Dynamic temporal scalability: Video adaptation in sparse mobile ad-hoc networks. In *Proceedings of the 2012 IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)* (pp. 349–356). IEEE.
- Cao, Y., & Sun, Z. (2012). Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges. *IEEE Communications Surveys and Tutorials*, 15(2), 654–677. doi:10.1109/SURV.2012.042512.00053
- Celebi, H., Yapıcı, Y., Güvenç, I., & Schulzrinne, H. (2019). Load-based on/off scheduling for energy-efficient delay-tolerant 5g networks. *IEEE Transactions on Green Communications and Networking*, 3(4), 955–970. doi:10.1109/TGCN.2019.2931700
- Ding, S., He, X., Wang, J., & Liu, J. (2018). Pre-decoding recovery mechanism for network coding opportunistic routing in delay tolerant networks. *IEEE Access: Practical Innovations, Open Solutions*, 6, 14130–14140. doi:10.1109/ACCESS.2018.2813382
- Ekman, F., Keränen, A., Karvo, J., & Ott, J. (2008, May). Working day movement model. In *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models* (pp. 33–40). ACM. doi:10.1145/1374688.1374695
- Fall, K. (2003, August). A delay-tolerant network architecture for challenged internets. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications* (pp. 27–34). Academic Press. doi:10.1145/863955.863960
- Graf, M., Timmerer, C., Hellwagner, H., Cherif, W., & Ksentini, A. (2013, June). Evaluation of hybrid scalable video coding for HTTP-based adaptive media streaming with high-definition content. In *Proceedings of the 2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)* (pp. 1–7). IEEE.
- Handley, M., Crowcroft, J., Bormann, C., & Ott, J. (1997). *The internet multimedia conferencing architecture*. Internet Engineering Task Force.
- Kazemi, M., Shirmohammadi, S., & Sadeghi, K. H. (2014). A review of multiple description coding techniques for error-resilient video delivery. *Multimedia Systems*, 20(3), 283–309. doi:10.1007/s00530-013-0319-z
- Keränen, A., Ott, J., & Kärrkäinen, T. (2009, March). The ONE simulator for DTN protocol evaluation. In *Proceedings of the 2nd international conference on simulation tools and techniques* (pp. 1–10). Academic Press. doi:10.4108/ICST.SIMUTOOLS2009.5674
- Klaghstan, M., Bennani, N., Coquil, D., Kosch, H., & Brunie, L. (2014, August). Contact-based adaptive granularity for scalable video transmission in opportunistic networks. In *Proceedings of the 2014 International Wireless Communications and Mobile Computing Conference (IWCMC)* (pp. 773–778). IEEE. doi:10.1109/IWCMC.2014.6906454
- Klaghstan, M., Coquil, D., Bennani, N., Kosch, H., & Brunie, L. (2013, September). Enhancing video viewing-experience in opportunistic networks based on svc, an experimental study. In *Proceedings of the 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)* (pp. 3563–3567). IEEE. doi:10.1109/PIMRC.2013.6666767

- Klaghstan, M., Coquil, D., Bennani, N., Kosch, H., & Brunie, L. (2016, September). BALCON: BACKward loss concealment mechanism for scalable video dissemination in opportunistic networks. In *Proceedings of the 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)* (pp. 1-7). IEEE.
- Klaue, J., Rathke, B., & Wolisz, A. (2003, September). Evalvid—A framework for video transmission and quality evaluation. In *Proceedings of the International conference on modelling techniques and tools for computer performance evaluation* (pp. 255-272). Springer. doi:10.1007/978-3-540-45232-4\_16
- Lenas, S. A., Burleigh, S. C., & Tsaoussidis, V. (2015). Bundle streaming service: Design, implementation and performance evaluation. *Transactions on Emerging Telecommunications Technologies*, 26(5), 905–917. doi:10.1002/ett.2762
- Li, X., Amon, P., Hutter, A., & Kaup, A. (2010). Performance analysis of inter-layer prediction in scalable video coding extension of H. 264/AVC. *IEEE Transactions on Broadcasting*, 57(1), 66–74. doi:10.1109/TBC.2010.2082370
- Lindeberg, M., Kristiansen, S., Plagemann, T., & Goebel, V. (2011). Challenges and techniques for video streaming over mobile ad hoc networks. *Multimedia Systems*, 17(1), 51–82. doi:10.1007/s00530-010-0187-8
- Morgenroth, J., Pögel, T., & Wolf, L. (2011, September). Live-streaming in delay tolerant networks. In *Proceedings of the 6th ACM workshop on Challenged networks* (pp. 67-68). ACM. doi:10.1145/2030652.2030673
- Nayyar, A., Bath, R. S., Ha, D. B., & Sussendran, G. (2018). Opportunistic Networks: Present Scenario-A Mirror Review. *International Journal of Communication Networks and Information Security*, 10(1), 223–241.
- Pan, D., Zou, X., Liu, X., Zhang, H., & Hao, S. (2016). Buffer management for streaming media transmission in hierarchical data of opportunistic networks. *Neurocomputing*, 193, 42–50. doi:10.1016/j.neucom.2016.01.055
- Piorkowski, M., Sarafijanovic-Djukic, N., & Grossglauser, M. (2009). CRAWDAD data set epfl/mobility (v. 2009-02-24).
- Quinlan, J. J., Zahran, A. H., & Sreenan, C. J. (2015). ALD: Adaptive layer distribution for scalable video. *Multimedia Systems*, 21(5), 465–484. doi:10.1007/s00530-014-0421-x
- Raffelsberger, C., & Hellwagner, H. (2015, March). A multimedia delivery system for delay-/disruption-tolerant networks. In *Proceedings of the 2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)* (pp. 530-536). IEEE.
- Sandulescu, G., Niruntasukrat, A., & Charnsripinyo, C. (2015). Guaranteed capacity bounds in intermittently-connected networks: A resource-aware, holistic evaluation. *Computer Communications*, 59, 12–23. doi:10.1016/j.comcom.2014.12.008
- Schierl, T., Hellge, C., Mirta, S., Gruneberg, K., & Wiegand, T. (2007, May). Using H. 264/AVC-based scalable video coding (SVC) for real time streaming in wireless IP networks. In *Proceedings of the 2007 IEEE International Symposium on Circuits and Systems* (pp. 3455-3458). IEEE. doi:10.1109/ISCAS.2007.378370
- Shibata, Y., & Uchida, N. (2017). Delay tolerant network for disaster information transmission in challenged network environment. *IEICE Transactions on Communications*, 100(1), 11–16. doi:10.1587/transcom.2016CQ10002
- Singhal, C., De, S., Trestian, R., & Muntean, G.-M. (2014). Joint Optimization of User-Experience and Energy-Efficiency in Wireless Multimedia Broadcast. *IEEE Transactions on Mobile Computing*, 13(7), 1522–1535. doi:10.1109/TMC.2013.138
- Sobin, C. C., Raychoudhury, V., Marfia, G., & Singla, A. (2016). A survey of routing and data dissemination in delay tolerant networks. *Journal of Network and Computer Applications*, 67, 128–146. doi:10.1016/j.jnca.2016.01.002
- Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2005, August). Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking* (pp. 252-259). ACM. doi:10.1145/1080139.1080143
- Stockhammer, T. (2011, February). Dynamic adaptive streaming over HTTP-- standards and design principles. In *Proceedings of the second annual ACM conference on Multimedia systems* (pp. 133-144). ACM. doi:10.1145/1943552.1943572

Thakur, A. (2020, February). An adaptive system for transmission of user-generated content over delay tolerant networks using scalable video coding. In *Proceedings of the Third International Conference on Communications and Cyber-Physical Engineering (ICCCE – 2020)*. Academic Press.

Trestian, R., Comsa, I., & Tuysuz, M. F. (2018). Seamless Multimedia Delivery Within a Heterogeneous Wireless Networks Environment: Are We There Yet? *IEEE Communications Surveys and Tutorials*, 20(2), 945–977. doi:10.1109/COMST.2018.2789722

Trono, E. M., Arakawa, Y., Tamai, M., & Yasumoto, K. (2015, January). Dtn mapex: Disaster area mapping through distributed computing over a delay tolerant network. In *Proceedings of the 2015 Eighth International Conference on Mobile Computing and Ubiquitous Networking (ICMU)* (pp. 179-184). IEEE. doi:10.1109/ICMU.2015.7061063

Unanue, I., Urteaga, I., Husemann, R., Del Ser, J., Roesler, V., Rodríguez, A., & Sánchez, P. (2011). A tutorial on H. 264/SVC scalable video coding and its tradeoff between quality, coding efficiency and performance. *Recent Advances on Video Coding*, 13, 9–15.

Wu, H., & Ma, H. (2014, June). Opportunistic routing for live video streaming in vehicular ad hoc networks. In *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014* (pp. 1-3). IEEE. doi:10.1109/WoWMoM.2014.6919002

## ENDNOTES

- <sup>1</sup> For RWP with 25 nodes, two of the simulation runs (out of 50) did not have any successfully delivery at the destination. The two runs have been skipped while computing overheads and CBO.

*Abhishek Thakur is a graduate from University of Roorkee (now IIT-Roorkee), and started his professional career with Wipro Global R&D in 1997, where he worked on Non-Stop Unix platforms and network management products for Alcatel DSLAMs. He juggled many roles while working for Synchronous Inc., a small company that made products for RF and optical transmission systems. He is pleased about his contribution to embedded software in optical amplifiers which were OEM'ed to Cisco as well as other organizations. After Motorola acquired Synchronous Inc., he worked in the network management team for broadband access networks. Later on, as part of Motorola India, he contributed to multiple projects in Cable/IP-TV domain. He also worked at Microsoft IDC in projects spanning mobile devices and virtualization. He joined BITS Pilani (Hyderabad Campus), in 2011 where he taught multiple subjects and researched in the realm of wireless networks and multimedia systems. Since December 2019, he has been working at IDRBT. Besides teaching, he has been actively contributing to the center for affordable technologies, IBFX (IDRBT Bank FinTech eXchange), Blockchain project, and the 5G Use case lab for BFSI. His passion is to simplify technology for rural enablement and is actively pursuing solutions for furthering Financial Inclusion.*