Behavior Analysis for WebRTC Peer-to-Peer Streaming with Dynamic Topology

3 authors:

Khalid Kahloot
Budapest University of Technology and Economics

Patrik J. Braun
Massachusetts Institute of Technology

Péter Ekler
Budapest University of Technology and Economics

Behavior Analysis for WebRTC Peer-to-Peer Streaming with Dynamic Topology

Khalid Kahloot, Patrik J. Braun & Peter Ekler
Automation and Applied Informatics Department (AUT)
Budapest University of Technology and Economics (BME)
Budapest, Hungary
{kahloot.khalid, braun.patrik, ekler.peter}@aut.bme.hu

Abstract— **Peer-to-Peer (P2P) data streaming in web browsers without the need of a third-party plugin is a preferable privacy-preserving option. However, the complexity of network is a challenge concerning theP2P topology and coding calculations. Despite the performance enhancing added by Web Real-Time Communication (WebRTC) over traditional P2P Communication, protocols overhead produce burst Traffic and the burstiness has a significant impact on the network performance. In this paper, we present a behavior analysis for burstiness and correlation for a WebRTC P2P file steaming. Then, we suggest a model of states like the Markovian chain. We evaluate some of the popular burstiness measures and present different kinds of traffic produced by two protocols WebPeer and CodedWebPeer based on P2P streaming. For modeling, we describe three main stochastic states of the peers regarding their position in the network and their stored data. We aim to contribute to better understanding of the behavior of traffic overhead caused by the protocols in static topology and dynamic joining topology. The results presented in this paper help to design P2P solutions based on WebRTC technologies.**

Keywords- Burstiness; Correlation; Peer-to-Peer (P2P); State Modeling; Markov chains.

I. INTRODUCTION

Video on demand (VoD) streaming is one of the P2P applications that consumes a big portion of the network resources and the computational power as well. In general, the P2P applications are overtaking the traffic share of the the Internet [13][14]. The concept of P2P sharing depends on chunking a file into slices, which are distributed over the peers. However, the functioning protocol should be able to adapt to the nature of the P2P system as individual peers join or leave the network stochastically. This feature of the protocol became very critical when serving time critical data, like videos.

In the recent years, there have been many notable studies concerning the performance of P2P protocol regarding video on demand streaming. For instance, many types of research perform system design, traffic measurement, performance analysis, and optimization over the BitTorrent. For example, Qiu et al. [13] provided a complete modeling and performance analysis of BitTorrent-like P2P Networks. Barbera et al. [5] standardized the processes of the BitTorrent system in the model of states to analyze the free rider phenomenon.

In our research, the key research area is to evaluate the performance of P2P WebRTC protocols. Moreover, we provide a suggestion for network performance enhancement, which shows how these protocols can evolve in the future. Studies concerning performance analysis help to understand the network traffic protocols such as WebPeer and CodedWebPeer [1]. A lot of efforts have been dedicated in this area, including real data measures, game-theoretic analysis and differential equation based macroscopic analysis. Through these efforts, it is clear now that piece and neighbor selection strategies are the two keys of efficient and scalable P2P systems. For each peer, piece and neighbor selection strategies decide which peers to upload to and which pieces to download from which service peers.

In this paper, we form a comprehensive basis for which we have a fundamental understanding of design parameters for WebPeer and CodedWebPeer protocols including performance analysis, burst and correlation measures. In addition, we reverse engineered the *WebPeer* protocol into a state modeling as Markov chain approximation. States and sub-states are modeled for CodedWebPeer and CodedWebPeer protocols. We present a comprehensive survey of analytical performance Modeling for video streaming. The results obtained not only will help to design better protocols, but also be useful for establishing a generally acceptable burstiness and correlation measures.

The layout of this paper is organized in seven sections. The following section reviews the previous related work on both modeling and performance measures. In Section 3, we introduce the *WebPeer* and *CodedWebPeer*. In addition, the components of the protocols are described to provide more understanding of their functionalities. The traffic measures are explained in Section 4. A state model is presented in Section 5. Finally, Section 6 concludes the paper and proposes further research areas.

II. RELATED WORK

In traffic analysis, a well-defined model is used study the behaviour of packet arrival and performance. Empirical or statistical studies calculate numerical approximations for network protocols. The WebPeer is a BitTorrent-like protocol, especially designed to work over WebRTC. Therefore, we first summarize previous P2P modeling studies. Then, we contrast traffic analysis, especially, burst and correlation measures.

Most of the empirical studies used modeling for simulations and data generation. Gerber et al. [16] report the behavior of Gorilla and its analysis of P2P traffic of two different regions. Sen and Wang [17] report measurement and analysis of three P2P systems, which include FastTrack, Gnutella, and DirectConnect. In both [4][16], Cisco Netflow is used to measure flow-based P2P traffic. Matei et al. in [18][19], modeled the Gnutella by using a crawler, which collected topology information and connectivity among peers. Multi-hop Carrier Sense Multiple Access (CSMA)-based was also considered for modeling. Stajkic et al. [2] proposed a novel approach based on a semi-Markov chain analysis, decoupling node and network levels. The state modlling is presented through a Semi-Markov Process.

Plenty of measures can investigate the performance of the traffic. Statistics from [24] can provide continuous measures and the analysis of Abilene network traffic. A weekly report is generated under the category of "File sharing" as it includes the majority of currently relevant P2P protocols. Cisco provides a NetFlow report as well [2]. To be more precise, we have chosen measures for burst and correlation of the traffic flow. Those measures are described in details in [20]-[22][24].

Izal et al. [8] and Pouwelse et al. [7] present measurement-based studies of BitTorrent based on tracker logs of different torrents. Gkantsidis and Rodriguez [6] present a simulation-based study of a BitTorrent-like system. They show results indicating that the download time of a BitTorrent-like system is not optimal. Other studies, e.g., [10]-[13], are concerned with the effective performance and the QoS issues of P2P systems. The key difference between previous research and our paper is that we started with traffic measures for network analysis. Then we modeled the process into functional states.

Some studies modeled the P2P traffic itself in a non-Markovian way. For instance, Gummadi et al. [15] model characteristics of P2P traffic in KaZaa as objects. They represent P2P processes as immutable, multi6media, large objects. On the other hand, Klemm et al. [9] use two Zipf distributions to model query popularity in Gnutella. We noticed that a few types of research applied Markov modeling approach over a P2P system such as Zhang et al. [3][4] who included an extended Markov chain model and insensitivity of count-down time. In addition, they provided an analysis on the trade-off between approximation gap and mixing time.

In this paper, we gathered the measures presented in P2P modeling studies [4], [15]-[17] due to the similarity of WebPeer and CodedWebPeer compared to BitTorrent. In addition, we followed the semi-Markov modeling apeared in [2] but we reverse engineered the already processes into states and substates.

III. System Environment

The functioning system is part of student research from the Automation and Applied Informatics Department (AUT) at the Budapest University of Technology and Economics (BME). Braun, Ekler and Fitzek [1] developed two versions of P2P protocol, which can establish direct browser to browser connections. Those protocols are designed to be efficient for browser-based P2P streaming. The first is called WebPeer protocol, which splits the original data into several equal sized pieces. Similar to BitTorrent's bitfield, however, it is a completely embedded in the web browser of the peer. The other protocol is called CodedWebPeer, extends the WebPeer protocol with network coding [24] capabilities. CodedWebPeer introduces new packet types for distributing Random Linear Network Coding (RLNC) coded data.

In both protocols, the P2P communication is implemented in JavaScript with WebRTC. In other words, WebRTC is still not officially standard, but several browsers already implement it. Moreover, storing huge amount of data at client-side or accessing file-system from browsers is not possible because of security reasons. In addition, the newly connected partner peer would have a higher probability of finding the missing packet, with a comparison to P2P communication using random packet dropping approach.

The target system uses network coding to enhance browser-based P2P streaming. In WebPeer and CodedWebPeer protocols, there are three kinds of entities: seeder, leecher, and tracker. The tracker is the active server, which is responsible for maintaining information about the peers and the available files for sharing. On the other hand, the leecher is a regular peer, which either sending or receiving files. Leecher carries out three basic operations; tracking, seeding and leeching.

The description above is quite similar to BitTorrent architecture. However, the tracker of WebPeer protocol does not keep track of the data amount stored at the individual peers [13].

We setup multi-variations for the selected measures. We choose to have 18 peers in the network in average, the peers could store 100%, 50%, 10% of the data. We carry out these measurements with WebPeer and CodedWebPeer protocols as well. In all cases, the peer can leech as;

- All slices of the file will be streamed from the server.
- Slices of the file will be streamed from randomly selected other peers in static topology.
- Slices of the file will be streamed for randomly selected other peers in dynamic topology, which means that peers can leave and join during the streaming.

IV. Burst and Correlation Measures

In this Section, we present a burst and correlation measuring for a collection of Ethernet traffic. We evaluated some of the popular burst and correlation measures and presented variant kinds of figures to describe traffic behavior. Our aim is to contribute to the understanding of the behavior of traffic overhead regardless of the used protocols in LAN and WAN.

The only data field we are considering is the timestamp, which represents the actual time in which current reading was recorded. First of all, we need to decide the amount of time t which we will carry out the measures over it. Obviously, the selected time is part of the total time, e.g., $t \subset T$. Therefore, if we decided to study the behavior of packet arrival for one hour of time then t will be 60. This period will be divided into sub-periods N.

Define the $P_N = \{P_1, P_2, P_3, \ldots, P_N\}$, which is a set of period times for the selected t of study period. Then for each S_i, we will calculate t_i, which is the average arrival time for this period. Define the ***arrTime,*** arrival times corresponding to each period. In addition, define the set of inter-arrival times ***interArr*** to represent the amount of time separating the packet arrivals. Define the set of lags, which represent the cumulative time of inter-arrival packets for a set of periods from X_N and the tests are:

- Stationarity test: is used to indicate the arriving rate of packets during the study period. It is good that the arriving rate or packets are stationary.
- Intensity test: is used to count the number of arrivals per unit time.
- Probably Density function: is used to describe the distribution of data over time.
- Correlation: is used to indicate how the consecutive shifted inter-arrival time is correlated.
- IDI: the index of dispersion for intervals is used to the sequence of inter-arrivals
- IDC: the index of dispersion for counts is used to the sequence of counts of arrivals in consecutive time units.

Since we have only the time stamp and the traffic size in bytes, then the inter-arrival time and the number of packets should be calculated first. The inter-arrival time is defined as the time between the start of arrival two packets. Suppose that a dataset is acquired to recording packet arrival time for T amount of time. Bustiness cause interference because when the packets get queued this will make the departures less burst. Delay is caused by packet interference. If arrivals are regular or sufficiently spaced apart, no queuing delay occurs.

Other measures are used, some of which ignore the effect of second order properties of the traffic. A first measure is the ratio of peak rate to mean rate and has the drawback of being dependent upon the interval used to measure the rate. A second measure is the coefficient of variation of the inter-arrival times. A metric considering second order property of the traffic is the IDC. In particular, given an interval of time τ, Because of the relationship in IDC includes in the numerator the effects of the correlation between the inter-arrivals.

Measures based on the first-order properties are the Peak to mean ratio (PMR), SCV and Higher Moments. For the PMR, inter-arrival time and the number of packets are correlated then graphed. The peak to mean ratio can be found when the arrays of frequencies are used as input for d2 frequency offset and tune the offset frequency and the rate of data collection in Hz.

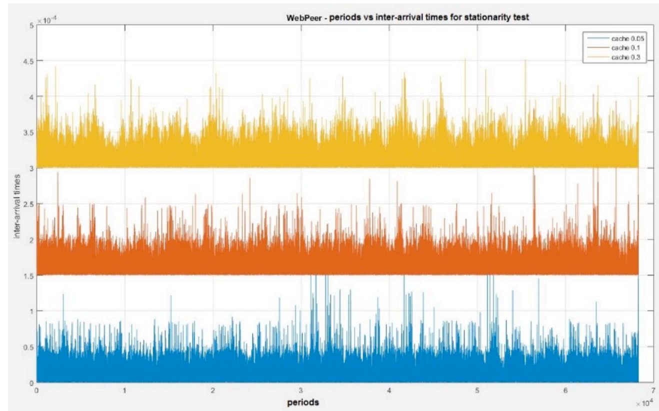

Figure 1. (a) WebPeer inter-arrival times for stationarity test

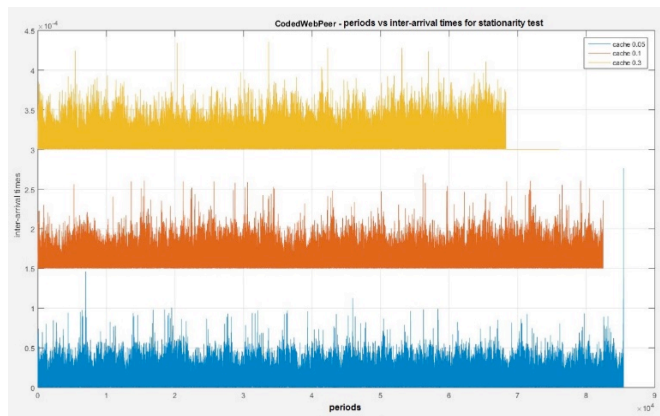

Figure 1. (b) CodedWebPeer inter-arrival times for stationarity test

Coefficient of Variation is the percentage variation in mean, standard deviation being considered as the total variation in the mean. For the the Squared coefficient of Variation (SCV), is considered more accurate to measure improved accuracy and it can be evaluated by visualization.

The moment represents the central sample moment of X specified by the positive integer order. For a given matrix, the moment characterizes the central moment of the specified order with respect to the elements of first, second, third, etc. As shown in Table I, the third moment of CodedWebPeer is much higher than the WebPeer, which indicates higher traffic.

TABLE I. FIRST ORDER MEASURES

	PMR	SCV	M3	M2	M1
WebPeer	8.8150	6.4187	1.0360	6.4053	0
CodedWebPeer	11.095	6.8053	7.34440	4.4415	0

Stationarity is time invariance of data, as shown in Fig. 1. For example, inter-arrival times in packet traffic undergoing reliability growth testing usually increases with time statistically and inter-arrival times in the network in service can be decreasing with time stationary.

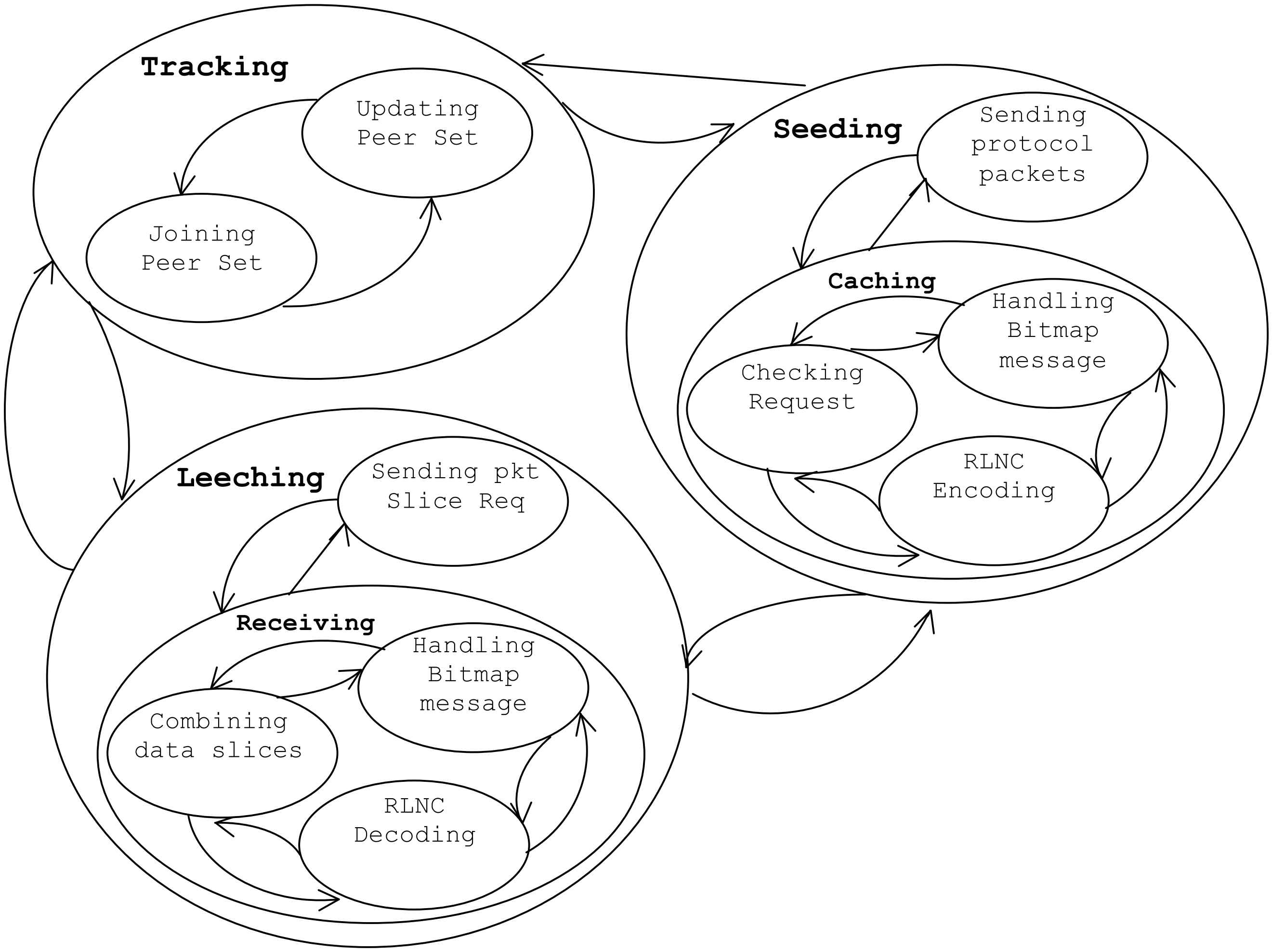

Figure 2. **State Modeling for CodedWebPeer and CodedWebPeer protocol**

If such trends do not exist, the arrival process of the traffic is stationary. The objective of stationarity tests is to determine whether the pattern of arrivals is significantly changing with time as to select appropriate models for modeling the data.

The intensity of packet arrival is the count of packet arrived in a particular point of time, as shown in Fig. 3. Using the average packet arrival count as observed by the test packets as an indicator, the intensity for active measurements is plotted.

The probability distribution function (PDF) of packet inter-arrival times should be viewed as a flat histogram to describe the scholastic process of packet traffic. The more we have independent sources of packets in the networks, the more it causes absolutely random inter-arrival times. As shown in Fig. 4, the PDF of inter-arrival times appears to be flat. When the PDF has different peak rates, this means that the capacity of connecting to the aggregate node gets lower many times. In other words, packets have been in queuing several before arriving at the aggregate node.

Correlation is used to measure the similarity of two types of traffic of different lengths. However, here we used it to measure the similarity between shifted inter-arrival times. The positive correlation will imply a low burst in the traffic. Mathematically, traffic burstiness is related to short-terms correlations between the inter-arrival times. However, there is accepted the notion of burstiness. A metric considering second order property of the traffic is the IDC. In particular, given an interval of time τ, it is an important measure of the correlation for a sequence arrival counts. The dependence among successive inter-arrival times can be expressed by means of the IDI. The IDI, also called the k-interval squared coefficient of variation sequence is defined as the sequence of indices. The limit of the IDI is an important measure to characterize the effect of an arrival process on the congestion of a queue in heavy traffic.

V. PROTOCOL MODELING

The original functionalities of WebPeer and CodedWebPeer are explained in details in [1]. The purpose of the modeling is to represent and to reshape traffic behavior from grounds up for the system environment discussed in the previous section. We aim for a synthesized behavior of P2P peers, based on empirical observations of the WebRTC video streaming. Peers transfer 'request/message' among each other. Thus, transfers are to be modeled into a functional state. Packets can flow as traffic between the states in a hierarchical way. However, there are ten sub-states included as shown in Fig. 2 and details are listed in Table II.

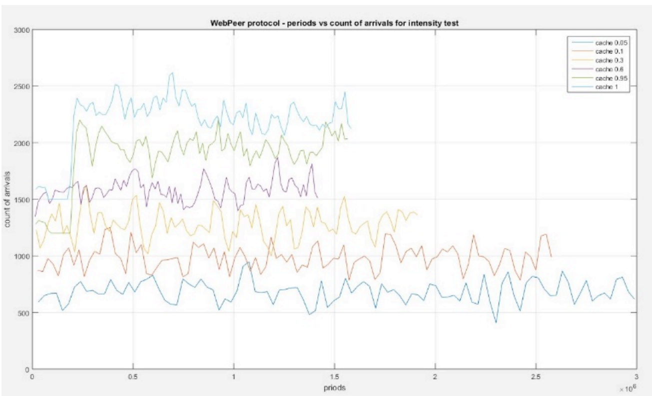

Figure.3. (a) WebPeer count of arrivals for intensity test

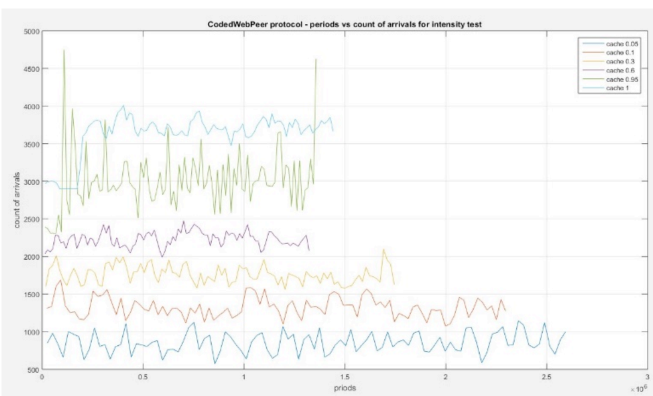

Figure 3. (b) CodedWebPeer count of arrivals for intensity test

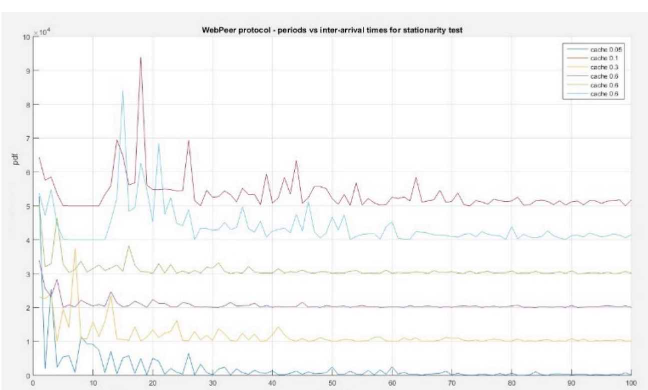

Figure 4. (a) WebPeer probability distribution function

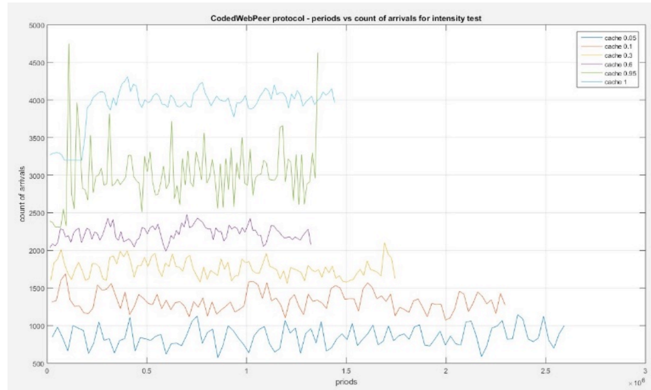

Figure 4. (b) CodedWebPeer probability distribution function

TABLE II STATES of WEBPEER & CODEDWEBPEER

State	Description
- Tracking	Describe the behavior of a Peer operates on P2P network topology
---- Updating Peer Set	Update the Set containing the Peers on the network
---- Joining Peer Set	Join a specific Set of Peers
- Seeding	Describe the behavior of a Peer seeding data
---- Sending protocol pkts	Send communication packets of peer protocols
---- Caching	Process Data to be sent on the P2P network topology
------- Handling Bitmap mg	Bitmap protocol message tells the Peer how to seed
------- Checking Request	Generating network coded packets for Slices of data
------- RLNC Encoding	Slice and encrypt data
- Leeching	Describe the behavior of a Peer performing actual video streaming
---- Sending pkt Slice Req	Send Slice of a requested packet
---- Receiving	Prepare the received data for the peer
------- Handling Bitmap mgs	Bitmap protocol message tells the Peer how to leech
------- Combining data Slices	Arrange the sliced data into real time stream
------- RLNC Decoding	Calculating and decrypting data

VI. Conclusion

We have reported a burst and correlation analysis study by investigating the most important candidate measures of burst to compare the characteristics of traffic flow and we can state that increasing the number of peers has a direct effect on the intensity and the probability density distribution of packet arrival. The IDC of the traffic of both protocols illustrates the effect the burstiness of the WebRTC streaming. IDI and IDC measures for the CodedWebPeer traffic show that RLNC can be highly inaccurate, even though it is affected by network performance as well. Both protocols are modeled into three hierarchal states with a total of ten sub-states. We concluded that both protocols have overhead, which produces burst traffic. Therefore, the numerical results of the used measures and the states in the supposed model can be used in a future work for optimization. The States will be reflected into parallel processes and the measures will be used to parametrized the algorithms.

Acknowledgment

This work was supported by the Janos Bolyai Research Fellowship of the Hungarian Academy of Sciences.

References

[1] P. J. Braun, P. Ekler and F. Fitzek, "Network coding enhanced browser based Peer-to-Peer streaming," 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, 2016, pp. 002104-002109. doi: 10.1109/SMC.2016.7844550

[2] A. Stajkic, C. Buratti and R. Verdone, "Modeling multi-hop CSMA-based networks through Semi-Markov chains," 2015 International Wireless Communications and Mobile Computing Conference (IWCMC), Dubrovnik, 2015, pp. 520-525. doi: 10.1109/IWCMC.2015.7289138

[3] H. Zhang, Z. Shao, M. Chen, and K. Ramchandran. 2011. Optimal neighbor selection in BitTorrent-like peer-to-peer networks. In Proceedings of the ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems (SIGMETRICS '11). ACM, New York, NY, USA, 141-142. DOI:10.1145/1993744.1993796

[4] S. Zhang, Z. Shao, and M. Chen, “Optimal distributed p2p streaming under node degree bounds,” in Proc. of IEEE ICNP 2010, 2010, pp. 253–262.

[5] M. Barbera, A. Lombardo, G. Schembra and M. Tribastone, "A Markov model of a freerider in a BitTorrent P2P network," GLOBECOM '05. IEEE Global Telecommunications Conference, 2005., 2005, pp. 5 pp.-. doi: 10.1109/GLOCOM.2005.1577784

[6] C. Gkantsidis and P. R. Rodriguez, "Network coding for large scale content distribution," Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies., 2005, pp. 2235-2245 vol. 4. doi: 10.1109/INFCOM.2005.1498511

[7] J.A. Pouwelse, P. Garbacki, D.H.J. Epema, and H.J. Sips, “A Measurement Study of the BitTorrent Peer-to-Peer File-Sharing System,” Technical Report PDS-2004-003, Pages 205-216, Delft University of Technology, The Netherlands, April 2004.

[8] M. Izal, G. Urvoy-Keller, E.W. Biersack, P. Felber, A. Al Hamra, and L. Garc´ es-Erice, “Dissecting BitTorrent: Five Months in a Torrent’s Lifetime,” PAM, Apr. 2004.

[9] A. Klemm, C. Lindemann, M. K. Vernon, and O. P. Waldhorst, “Characterising the query behavior in peer-to-peer file sharing systems,” in Proc. of ACM/SIGCOMM Internet Measurement Conference (IMC’04), Taormina, Sicily, Italy, Oct. 2004, pp. 55–67.

[10] M. Iguchi, M. Terada, K. Fujimura, "Managing Resource and Servent Reputation in P2P Networks", in Proc. 37th Annual Hawaii Int. Conf. on System Sciences (HICSS’04), 2004

[11] D. Hughes, I. Warren, G. Coulson, "Improving QoS for Peer-to-Peer Applications through Adaptation", in Proc. Of the 10th Int. Workshop on Future Trends in Distributed Computing Systems (FTDCS 2004), Suzhou, China, May 26-28, 2004.

[12] E. Kalyvianaki and I. Pratt, "Building adaptive peer-to-peer systems," Proceedings. Fourth International Conference on Peer-to-Peer Computing, 2004. Proceedings., 2004, pp. 268-269. doi: 10.1109/PTP.2004.1334959

[13] D. Qiu and R. Srikant, “Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks”, ACM SIGCOMM Computer Communication Review 34(4):367-378 · October 2004, Portland, Oregon, USA

[14] A. Gerber, J. Houle, H. Nguyen, M. Roughan, S. Sen, "P2P The Gorilla in the Cable", in National Cable & Telecommunications Association (NCTA) 2003 National Show, Chicago, IL, June 8-11, 2003.

[15] K. Gummadi, R. Dunn, S. Saroiu, S. Gribble, H. Levy, and J. Zahorjan, “Measurement, modeling, and analysis of a peer-to-peer file-sharing workload,” in Proc. of the 19th ACM Symposium on Operating Systems Principles (SOSP’03), Bolton Landing, NY, USA, Oct. 2003, pp. 314–329.

[16] Alexandre Gerber, Joseph Houle, Han Nguyen, Matthew Roughan, and Subhabrata Sen, "P2P The Gorilla in the Cable", National Cable & Telecommunications Association (NCTA) 2003 National Show, Chicago, IL, June8-11,2003

[17] S. Sen and J. Wang, "Analysing peer-to-peer traffic across large networks", in Proc. of ACM SIGCOMM Internet Measurement Workshop, Marseilles, France, Nov. 2002.

[18] R. Matei, A. Iamnitchi and P. Foster, "Mapping the Gnutella network," in IEEE Internet Computing, vol. 6, no. 1, pp. 50-57, Jan/Feb 2002. doi: 10.1109/4236.978369

[19] R. Matei, I. Foster, and A. lamnitchi, "Mapping the Gnutella Network Properties of Large-scale Peer-to-Peer Systems and Implications for System Design", IEEE Internet Computing Journal, vol. 6(1), pp. 80-86, 2002.

[20] D. Kouvatsos, R. Fretwell, “Batch Renewal Process: Exact Model of Traffic Correlation", High-Speed Networking for Multimedia Application, Kluwer Academic Press, pp. 285-304, 1996.

[21] R. Gusella, "Characterizing the variability of arrival processes with indexes of dispersion," in IEEE Journal on Selected Areas in Communications, vol. 9, no. 2, pp. 203-211, Feb 1991. doi: 10.1109/49.68448

[22] S. Molnar, G. Miklos, “On Burst and Correlation Structure of Teletraffic Models”, 5th IFIP Workshop on Performance Modeling and Evaluation of ATM Networks, 21-23 July 1997, West Yorkshire, U.K.

[23] D. R. Cox, P. A. W. Lewis, “The Statistical Analysis of Series of Events", Methuen & Co Ltd, 1966.

[24] Ahlswede, R., Ning Cai, Li, S.-Y.R., Yeung, R.W.: Network information flow, Information Theory, IEEE Transactions on (Volume:46, Issue: 4), pages: 1204-1216, ISSN: 0018-9448, 2000.