Peer-Assisted Content Distribution Aided by Video Popularity Evolution Model

Mohamad Dikshie Fauzie* Achmad Husni Thamrin* Jun Murai† *Graduate School of Media and Governance †Faculty of Environment and Information Studies
Keio University, 252-0882 Kanagawa, Japan
dikshie@sfc.wide.ad.jp husni@ai3.net jun@wide.ad.jp

***Abstract*—In this paper, we present peer-assisted CDN model, the peer-to-peer networks that can be used to help CDN to deliver streaming media. Our peer-assisted CDN model is aided by Youtube VoD views popularity model. We have comparatively evaluated our system through trace-driven simulations with synthetic workloads. Our results show that our model gives lower number of replicas while maintaining same number of peers contribution compare to previous work. We also do the significance to the number of replicas using the Kolmogorov-Smirnov statistic on two samples and we find our results are significant.**

***Index Terms*—Internet Video on Demand (VoD), P2P, CDN, and Caching.**

I. Introduction

Streaming content, especially video, represents a significant fraction of the traffic volume on the Internet, and it has become a standard practice to deliver this type of content using Content Delivery Networks (CDNs) such as Akamai and Limelight for better scaling and quality of experience for the end users. For example, YouTube uses Google cache and MTV uses Akamai in their operations.

With the spread of broadband Internet access at a reasonable flat monthly rate, users are connected to the Internet 24 hours a day and they can download and share multimedia content. P2P (peer to peer) applications are also widely deployed. In China, P2P is very popular; we see many P2P applications from China such as PPLive, PPStream, UUSe, Xunlei, etc. [1]. Some news broadcasters also rely on P2P technology to deliver popular live events. For example, CNN uses the Octoshape [2] solution that enables their broadcast to scale and offer good video quality as the number of users increases.

From the Internet provider point of view, the presence of so many always-on users suggests that it is possible to delegate a portion of computing, storage and networking tasks to the users, thus creating P2P networks where users can share files and multimedia content. Starting from file sharing protocols, P2P architectures have evolved toward video on demand and support for live events.

Alternatively, video contents can be efficiently distributed on services offered by managed network architectures and CDN companies. The major issues of CDN are high deployment cost and good but not unlimited scalability in the long term. Given the complementary features of P2P and CDN, in recent years some hybrid solutions have been proposed and applied to the operational of CDN [3]–[5] to take the best of both approaches. In Peer assisted CDN, users can download content from CDN nodes from or other users or peers. A user may cache the content after download to serve requests from other users. Due to the complexity of the behavior of peers, the process should be done in the home gateway user where the ISP can control it.

In this work, we will revisit Guo et al., [6] work's PROP as basis to evaluate of the peer-assisted CDN and propose an improvement the model for the PROP. Internet video on demand (VoD) is on rising todays e.g. Youtube. We will take Youtube as an example of Internet VoD service model. In Youtube service model, we can get data such as (1) the time when a video is uploaded and (2) number of access or number of view. In a seminal work, Borghol et al., [7] use above information to estimate when a video become very popular. Moreover, Borghol et al., [7] classify a video popularity become three phases: before-peak phase, at-peak phase, and after-peak phase. We will use estimation of a video popularity phases for helping PROP. We will explain about video popularity in sect.III. A twofold of our contributions as follows: (1) We use the idea of VoD view popularity model to aid the PROP model. To our knowledge, the combination of PROP model and VoD view popularity model is the first. (2) From simulation-based experiments, we find that peer contributions become higher than the PROP model.

Our paper presentation as follows: (1) we describe related work in sect.II; (2) we explain detail of Youtube popularity evolution model in sect.III; (3) we explain the caching strategy for CDN and peer in sect.IV; (4) we explain our simulation design, simulator, and its evaluation in sect.V. Finally, we present our conclusions in section VI.

II. Related Work

Content Distribution Networks with peer assist have been successfully deployed on the Internet, such as Akamai [8], [3] and LiveSky [9]. The authors of [8] examine the risks and benefits of peer-assisted content distribution in Akamai and measure the effectiveness of its peer-assisted approach. The authors of [3] conclude from two real world traces that hybrid CDN-P2P can significantly reduce the cost of content distribution and can scale to cope with the exponential growth of Internet video content. Yin et al. [9] described commercial operation of a peer-assisted CDN in China. LiveSky solved several challenges in the system design, such as dynamic resource scaling of P2P, low startup latency, ease of P2P integration with the existing CDN infrastructure, and network friendliness and upload fairness in the P2P operation. Xu et al. [10] used game-theory to show the right cooperative profit distribution of P2P can help the ISP to maximize the utility. Their model can easily be implemented in the context of current Internet economic settlements. Misra et al. [11] also mentioned the importance of P2P architecture to support content delivery networks. The authors use cooperative game theory to formulate simple compensation rules for users who run P2P to support content delivery networks.

The idea of telco- or ISP-managed CDN has been proposed in recent years. The complexity of the CDN business encourages telcos and ISPs to manage their own CDN, rather than allow others to run CDNs on their networks. It has been shown that it is cost effective [12] [13]. Kamiyama et al. [14] proposed optimally ISP operated CDN. Kamiyama et al. mentioned that, in order to deliver large and rich Internet content to users, ISPs need to put their CDNs in data centers. The locations are limited while the storage is large, making this solution effective; using optimum placement algorithm based on real ISP network topologies. The authors found that inserting a CDN into an ISP's ladder-type network is effective in reducing the hop count, thus reduce total link cost. Based on the author definition: Ladder-type network is a network with a maximum degree under 10. Cisco has initiated an effort to connect telco- or ISP-managed CDNs to each other, to form a CDN federation [12] using open standards [15]. They argue that the current CDN architecture is not close enough to the users and ISPs can fill this position.

The idea of utilizing the user's computation power to support ISP operation is not new. The Figaro project [16] proposed the residential gateway as an integrator of different networks and services, becoming an Internet-wide distributed content management for a proposed future Internet architecture [16]. Cha et al., [17] performed trace analysis and found that an IPTV architecture powered by P2P can handle a much larger number of channels, with lower demand for infrastructure compared to IP multicast. Jiang et al. [18] proposed scalable and adaptive content replication and request routing for CDN servers located in users' home gateways. Maki et al., [19] propose traffic engineering for peer-assisted CDN to control the behavior of clients, and present a solution for optimizing the selection of content files. Mathieu et al., [20] are using data gathered from France telecom network to calculate reduction of network load if customers are employed as peer-assisted content delivery.

Guo et al., [6] work's PROP is closest with our work. PROP uses local system (local counter) to calculate the segment popularity in peer-assisted proxy system. PROP uses popularity for proxy cache replacement strategy. In peer side, the author use utility function for cache replacement strategy. A utility function assigns numerical value to outcomes, in such a way that outcomes with higher utility are always preferred of outcomes with lower utilities. The utility function is also function from popularity. While the authors successfully show that the results are very good, the peer-assisted system behavior over time is not explain because the author focus

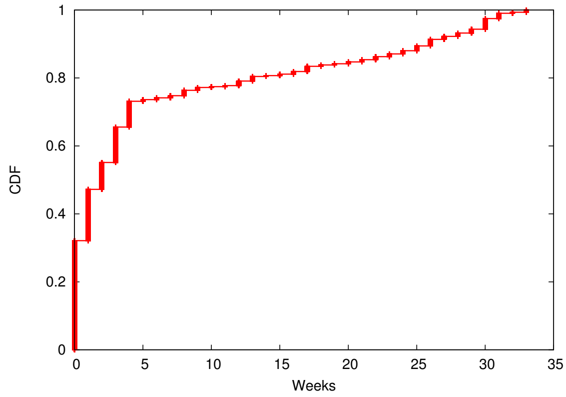


Fig. 1: Time to peak empirical distribution [7].

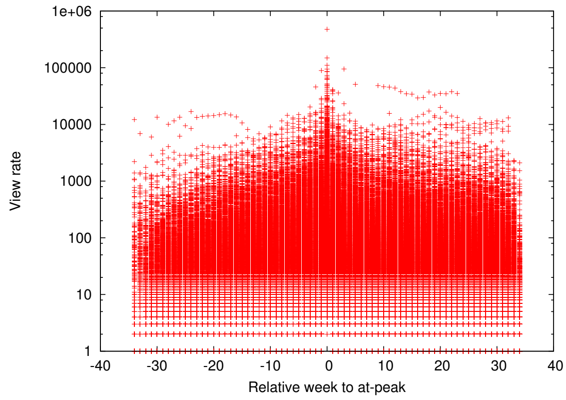


Fig. 2: View rate distribution versus week relative to at-peak phase week for every video, where y-axis in log scale. Every point lies in negative x-axis mean view rate of every video in before-peak phase. Every point lies in x-axis= 0 mean view rate of every video at-peak phase. Every point lies in positive x-axis mean view rate of every video in after-peak phase.

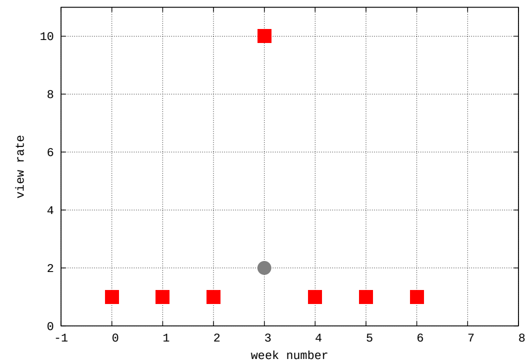
on properties such as proxy cache size variations and peer cache size variations. The explanation of the optimal number of replicas is not also clear because unavailable information when the snapshot is taken. In our work, we complement Guo et al., [6] work with VoD viewing popularity evolution model and describe the behavior of the peer-assisted CDN over the time.

III. Characterizing Internet VoD Popularity

As we mentioned earlier, we will use Youtube service model as an example of Internet VoD. The objective of estimating Internet VoD popularity phase is to get popularity state of a requested video whether a video is before-peak popularity phase or at peak popularity phase, or after-peak popularity phase. We use Youtube as an example of VoD service where we get Youtube content popularity datasets from Borghol et al., [7]. The datasets consists the measurement of 29000 videos view count and the time when videos are uploaded during 36 weeks. Borghol et al., [7] classify three phases of a video popularity: before-peak, at-peak, and after-peak. The authors define time-

view rate	1	1	1	10	1	1	1
relative week to peak	-3	-2	-1	0	1	2	3
week #	0	1	2	3	4	5	6

(a) Transformation of view rate distribution. We add week number and make it as x-axis, View rate as y-axis, and relative week to peak as z-axis.



(b) 2D visualization of view rate distribution after transformation where x-axis is week number, y-axis is view rate.

Fig. 3: Transformation of view rate distribution dataset and 2D visualization of view rate distribution.

to-peak for a video as its age (time since upload) at which its weekly viewing rate is the highest during measurement (from the first week until end of measurement). The time-to-peak distribution is shown in fig.1. Figure 1 shows Borghol et al., [7] work that around three-quarters of a large fraction videos peak within the first six weeks since their upload and beyond six weeks we have uniform distribution thus the time-to-peak is exponentially distributed mixture with uniform distribution. To estimate the rate parameter of exponential part of time-to-peak distribution, we use Maximum Likelihood Estimation (MLE) [21]. Using MLE method, we can get exponential parameter $\lambda = 0.59$. Because we know the peak time (at-peak phase) of every video, we can also know before-peak phase and after-phase of every video. For detail we refer the readers to [7].

To reveal data distribution of view rate for every video, we plot view rate versus week where we shift week of view rate at-peak phase to zero. Therefore we can get view rate distribution relative to at-peak week as shown in fig. 2

How we estimate the video popularity phase is shown in fig.3. In fig. 3a we have view rate (y-axis) and relative week to peak (x-axis) which is view rate distribution versus week relative to at-peak phase. We transform these numbers by adding week number and make it as x-axis, view rate as y-axis, and relative week to peak as z-axis fig. 3a. This transformation is shown in fig. 3b denote as box points. Assume there is a peer requests a video, we want to estimate what is the phase of that video. Is the video in at-peak phase, before-phase, or after phase. We can estimate that video phase by averaging relative week to peak numbers (the points at z-axis) of the nearest point from datasets. If the average value less than 0 we estimate the video is at before-peak phase, if the average value equal to 0 we estimate the video is at at-peak phase, and if the average value more than 0 we estimate the video is at after-peak phase.

For example: there is a peer that requests a video where the position of video is in fourth week with the last week view rate $vr = 2$ (we can get as this data from CDN) shown in fig. 3b denote as circle. In this case, the nearest points are the point at third week $(2, 1, -1)$ and the point at fifth week $(4, 1, 1)$. By averaging the points at z-axis of the nearest points $(-1 + 1)/2 = 0$, we can get estimate that video is in at-peak phase.

IV. System Description

In our work, we use Youtube VoD view model to aid our work that based from PROP. The Youtube VoD view model will be used in peer-caching strategy side to exploits the video popularity while caching strategy in CDN side is out of scope.

Since we can estimate before-peak phase, at-peak phase, and after-peak phase of video, we modified the original utility function from PROP by adding a $z(t)$ factor as follows:

$$u = \frac{(f(p) - f(p_{min}))(f(p_{max}) - f(p))}{r^{\alpha+\beta}} + z(t) \tag{1}$$

$\frac{(f(p)-f(p_{min}))(f(p_{max})-f(p))}{r^{\alpha+\beta}}$ is the utility function from PROP and $z(t)$ is the z factor from our model. $z(t)$ is a function from current time in simulation.

$$z(t) = \begin{cases} 0.15 & \text{if phase estimation is before-peak} \\ 0.47 & \text{if phase estimation is at-peak} \\ 0.38 & \text{if phase estimation is after-peak} \end{cases}$$

Following [6], we can calculate p as follows:

$$p = min\left(\frac{n_i^r}{t_i^r - t_a^i}, \frac{1}{t_{cur} - t_i^r}\right) \tag{2}$$

Where n_i^r is number of requested video, t_i^r is last time the video is requested, t_a^i is the uploaded time of the video, and t_{cur} is the current time. $z(t)$ is proportion of view count that we get from Youtube datasets. To able to track the simulation, we use default value from PROP thus we refer the readers to [6]. We choose the video with the smallest u value as the candidate to be replaced when a peer's cache capacity is full. In PROP's utility function, the difference between very popular videos and unpopular video is very difficult to differentiate. For unpopular video $f(p)$ will be very close to $f(p_{min})$ thus $f(p) - f(p_{min})$ will be very close to 0 then utility function become very small. For very popular video $f(p)$ will be very

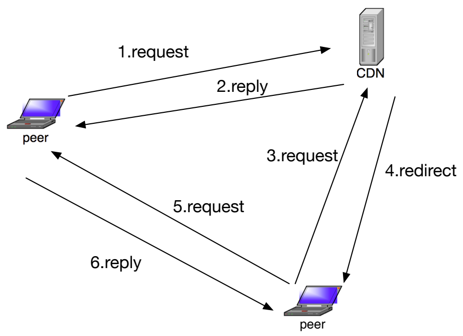


Fig. 4: Peer interaction in simulator. When a peer requests a video, it always goes to a CDN server (step 1). The CDN provides the videos to the peer (step 2). If there is another peer request same video, that request will go to CDN (step 3). A CDN will check its record to see if there is some peers cache that requested video. If there is some peers cache that requested video, a CDN will reply with redirect message that asking a peer to download requested video from other peer (step 4). If there s no peers have requested video, a CDN will serve the video. A peer then can request the video to other peer and get the video (step 5 and step 6).

close to $f(p_{max})$ thus $f(p_{max}) - f(p)$ will be very close to 0 then utility function become very small. Linear addition of $z(t)$ factor can help to differentiate the value of utility function.

V. Evaluation

In order to evaluate the proposed peer-caching strategy using estimation of before-peak, at-peak, and after-peak information from Youtube VoD view model, we have to compare our model to PROP model. We evaluate three metrics, which are peer contribution to delivery contents during simulation, access frequency of cache during simulation, and number of replicas. Peer contribution metric related to byte-hit-ratio. Byte-hit-ratio is defined as the total bytes contents served by peers normalized by the total bytes of video all peers and CDN consume. It means more peer contributions, more byte-hit-ratio because peer can get content from another peers. Access frequency of cache reflects the storage utilization. More access means more peer storage utilization. Number of replicas is also related to peer storage utilization. However, too many replicas will waste the storage resources. To evaluate these metrics, we developed a peer-assisted CDN simulator.

A. Simulation Design

An event driven simulator is developed using Python for this purpose. In fig.4, we describe the process of a peer that requests a video in simulator, which derived from PROP. A peer and a CDN are implemented in object oriented-model inside the simulator. In short, a peer always requests to CDN then CDN will decided if a requested video is available in other peers or not. If a requested video is available in other peers, CDN will redirect the request to other peers. If a requested video is not available CDN will serve the request.

1) Catalog Generator: The goal for catalog generator is to create a catalog video that consist video-id, time when a video is uploaded, a video size, view count terminus, and progress of videos popularity like Youtube service model. We assume that a video is uploaded to server following Poisson process with mean rate $\lambda = 1$ thus we can get the time when a video is uploaded. The view count terminus for every video is assigned randomly uniform from Youtube datasets and video size for every video is assigned randomly uniform between 1 and 200MB. Finally, we have a catalog that consists of: video-id, time when a video is uploaded, view count terminus, and video size.

2) Peer Request Generator: In catalog generator, we assume peer request a video to CDN following Poisson process with a mean rate $\lambda = 1$ [22]. There are three scenarios for peer request: First, a peer chooses a video that has popularity following Youtube. The objective of the first scenario, we want to see the peer requests effect to peer-assisted CDN when the request following Youtube popularity. Second, a peer chooses a video that has popularity following Youtube but we shift the request four weeks later. The objective of the second scenario, we want to see the peer requests effect to peer-assisted CDN when the request from peers are lag four weeks after popular in Youtube. Third, a peer chooses a video that has popularity following zipf distribution with rate= 0.9 [23] thus a peer choose a video that its popularity does not follow Youtube popularity. The objective of the third scenario, we want to see the peer request effect to peer-assisted CDN when the requests from peers are totally different from Youtube's videos popularity.

3) Simulation Parameters and Scenarios: The simulation parameters are follows:

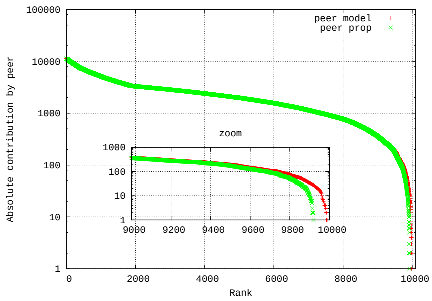
- Length: 360 days.
- Video size: uniform random between 1MB and 200MB.
- Peer capacity: 500MB.
- CDN capacity: 10000MB.
- Number of peers: 100000.
- Number of videos: 10000.

Finally, we compare our results to original PROP [6] implementation.

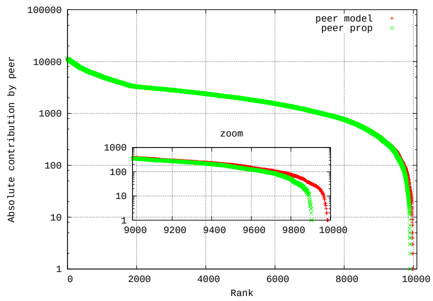
B. Result and Discussion

Figure 5a, 5b, and 5c show the absolute peer contribution to deliver videos compared between model and prop. Figure 5a and fig.5b show same pattern. The peers give more contribution in the tail while in the third scenario the peer contribution is mostly same between model and PROP. A peers can give more contribution because a video has longer duration than other videos in a peer's cache thus other peer's requests are served by the peer. A video has longer duration than other videos in peer's cache because that a video has bigger utility function than other videos for example a video that will enter the cache.

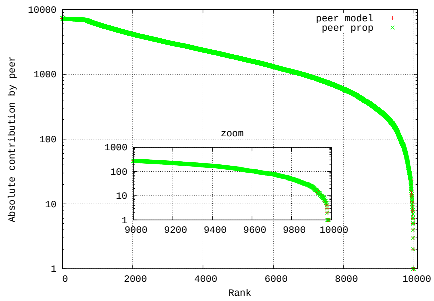
Figure 6a, 6b, and 6a show the number of videos replicas available in system when a peer requests a video. As we can see from all figures, the model gives us lower number of replicas than PROP. The model gives lower number of replicas



(a) Absolute of contribution of peer for the first scenario where y-axis in log-scale.

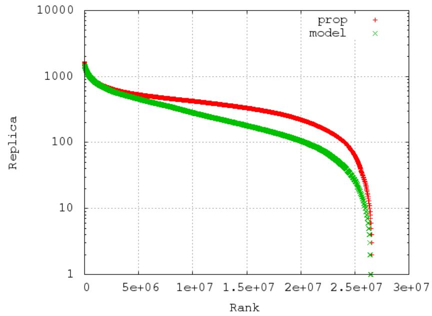


(b) Absolute of contribution of peer for the second scenario where y-axis in log-scale.

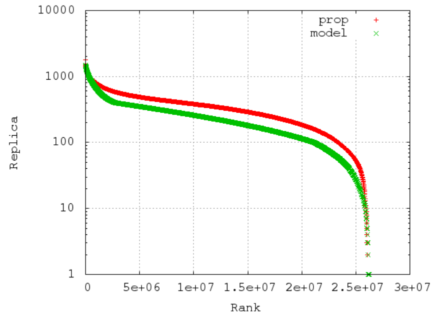


(c) Absolute contribution of peers for the third scenario where y-axis in log-scale.

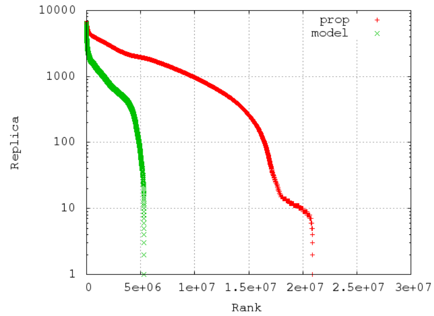
Fig. 5: Peer contributions compared between model and PROP.



(a) Number of a video replicas when a peer request a video for the first scenario where y axis in log-scale.

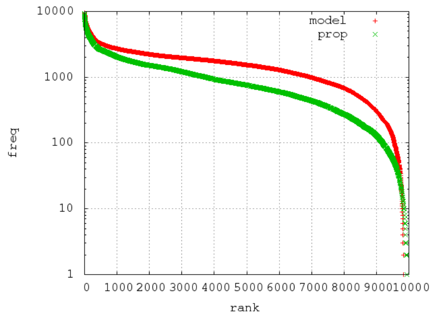


(b) Number of a video replicas when a peer request a video for the second scenario where y axis in log-scale.

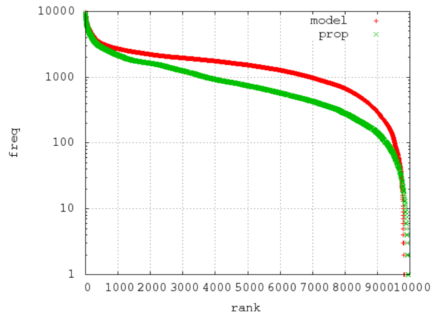


(c) Number of a video replicas when a peer request a video for the third scenario where y axis in log-scale.

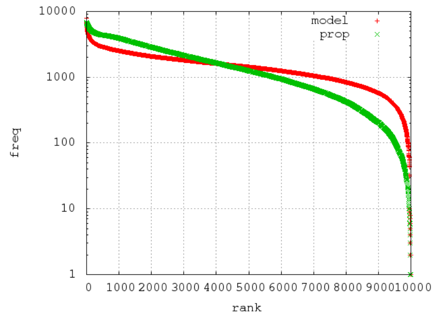
Fig. 6: Comparison of available replicas between model and prop when a peer requests a video.



(a) Frequency a video in peers for the first scenario.



(b) Frequency a video in peers for the second scenario.

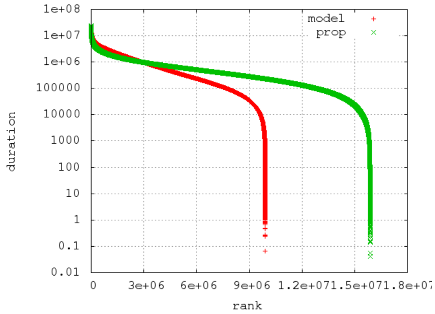


(c) Frequency a video in peers for the third scenario.

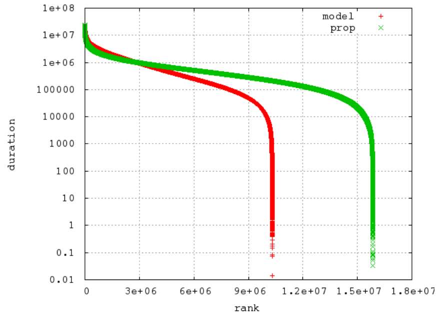
Fig. 7: Frequency a video in peers compared between model and prop.

than PROP because when a peer requests a video, that peer is not cached the video. We can see the proportion of cached and not-cached event in table.I. We also present detail of the video phase breakdown in table.II. In model, not-cached events take around 65% from all events and majority of video phase is after-peak for both cached events and not-cached events. Because the majority of video phase is after-peak for both cached events and not-cached events, In PROP, cached events take around 52% from all events for the first scenario and the second scenario, while for the third scenario is 67.7%. In model not-cached events are bigger than PROP, means peers do not cached the videos thus we get lower replicas number than PROP.

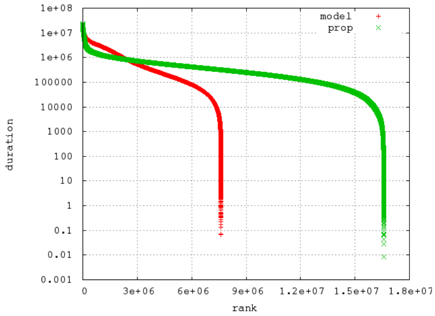
Denote u_{dl} is the minimum utility function for a video inside the cache and u_{ms} is utility function for a video that will enter the cache, p_{dl} is the popularity for a video inside the cache and p_{ms} is the popularity for a video that will enter the cache. In order a requested video is cached by a peer, the



(a) Cache duration in peers for the first scenario.



(b) Cache duration in peers for the second scenario.



(c) Cache duration in peers for the third scenario.

Fig. 8: Duration compared between model and prop.

TABLE I: Percentage of Cached events and Not-Cached events in Model and PROP

Scenario	Type	Cached	Not-Cached
Scenario 1	Model	33.5%	66.5%
	PROP	52%	48%
Scenario 2	Model	34.8%	65.2%
	PROP	52.6%	47.4%
Scenario 3	Model	32.4%	67.6%
	PROP	67.7%	32.3%

TABLE II: Percentage of Video Phase for Model in cached and not-cached events

Scenario	Type/Events	Before-Peak	At-Peak	After-Peak
Scenario 1	Model-Cached	8.2%	1.2%	24.1%
	Model-Not-Cached	11.2%	0.8%	54.5%
Scenario 2	Model-Cached	6.2%	1.2%	29.8%
	Model-Not-Cached	5.2%	0.8%	56.8%
Scenario 3	Model-Cached	8.0%	1.8%	22.7%
	Model-Not-Cached	15.1%	0.8%	51.6%

utility function for u_{dl} must be lower than the utility function for u_{ms}.

$$u_{dl} < u_{ms} \tag{3}$$

$$\frac{(f(p_{dl}) - f(p_{min}))(f(p_{max}) - f(p_{dl}))}{r_{dl}^{\alpha+\beta}} + z_{dl} < \frac{(f(p_{ms}) - f(p_{min}))(f(p_{max}) - f(p_{ms}))}{r_{ms}^{\alpha+\beta}} + z_{ms} \tag{4}$$

We assume that numbers of replicas are same, thus:

$$(f(p_{dl}) - f(p_{min}))(f(p_{max}) - f(p_{dl})) - (f(p_{ms}) - f(p_{min}))(f(p_{max}) - f(p_{ms})) < z_{ms} - z_{dl} \tag{5}$$

Since p_{min} and p_{max} are same for u_{dl} and u_{ms}, we can arrange the equation become:

$$f(p_{ms}) - f(p_{dl}) > z_{dl} - z_{ms} \tag{6}$$

As we know from table.II that the majority of a requested video is after-peak phase and a requested video phase that is at-peak phase is very small portion, then we can see that $z_{dl} - z_{ms}$ term will be in negative term if z_{dl} is before-peak phase or 0 if z_{dl} is after-peak phase. If $z_{dl} - z_{ms} = 0$ then it is same with PROP. Since the not-cached events happen in when a requested video phase after-peak phase, we can get that $f(p_{ms}) - f(p_{dl} < 0$. For the same situation and we compare to the PROP, the probability of u_{ms} less than u_{dl} in the model is less than PROP. Therefore, we can see in the model that the events when a peer does not cache a video are more often than PROP.

Figure 7a, 7b, and 7c show the frequency of a video stay in peers compared between model and PROP. As all figure show the model has higher frequency than PROP to stay in peers except for the beginning rank of data where the model has same frequency with prop in first and second scenario. In the third scenario, in the beginning rank of data the model has lower frequency than PROP, then around rank 1000 the model has higher frequency than prop until the end of data. The frequency a video stay in a video can also be viewed in fig 8a, 8b, and 8c, where in the model some videos have longer cache duration than PROP, while others have shorter cache duration than PROP. Thus, we can see the relationship between cache duration and frequency a video stays in peers.

VI. Conclusion and Future Work

This paper presents a scheme for peer-to-peer network can help CDN to deliver the content over the Internet. We show that by introducing z factor to utility function we can maintain same peer contribution while reducing number of replicas. We found that there are no much different between the first scenario, the second scenario and the third scenario in peer contribution to deliver a video. We found that in the all scenarios, the model gives lower replicas than PROP. This is because in the model, we found that not-cached events are higher than cached events, more specifically, the probability of utility function a requested video in model is lower than PROP. Therefore, in the model the numbers of available replicas are lower than PROP. We also did the significance test to the number of replicas using the Kolmogorov-Smirnov statistic on two samples and we find that for all scenarios the p-values are less than 1% thus the results are significant.

Some areas of improvement that we have identified for future are: The energy trade off this peer-assisted CDN architecture in order to know how much energy saving by ISP and how much increase of energy at users home gateway side in this architecture since we have higher peer contribution. More numerical experiments for other zipf shape parameters.

ACKNOWLEDGMENT

The authors would like to thank Internet research laboratory member at Keio University and anonymous reviewers.

REFERENCES

[1] L. Vu, I. Gupta, K. Nahrstedt, and J. Liang, "Understanding overlay characteristics of a large-scale peer-to-peer iptv system," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 6, no. 4, pp. 31:1–31:24, Nov. 2010. [Online]. Available: http://doi.acm.org/10.1145/1865106.1865115

[2] Octoshape, "Octoshape," http://www.octoshape.com/cnn-com-using-octoshapes-p2p-for-live-feed/.

[3] C. Huang, A. Wang, J. Li, and K. W. Ross, "Understanding hybrid cdn-p2p: why limelight needs its own red swoosh," in *Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, ser. NOSSDAV '08. New York, NY, USA: ACM, 2008, pp. 75–80. [Online]. Available: http://doi.acm.org/10.1145/1496046.1496064

[4] H. Jiang, J. Li, Z. Li, and J. Liu, "Efficient hierarchical content distribution using p2p technology," in *Networks, 2008. ICON 2008. 16th IEEE International Conference on*, dec. 2008, pp. 1 –6.

[5] H. Yin, X. Liu, T. Zhan, V. Sekar, F. Qiu, C. Lin, H. Zhang, and B. Li, "Design and deployment of a hybrid cdn-p2p system for live video streaming: experiences with livesky," in *Proceedings of the 17th ACM international conference on Multimedia*, ser. MM '09. New York, NY, USA: ACM, 2009, pp. 25–34. [Online]. Available: http://doi.acm.org/10.1145/1631272.1631279

[6] L. Guo, S. Chen, and X. Zhang, "Design and evaluation of a scalable and reliable p2p assisted proxy for on-demand streaming media delivery," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 18, no. 5, pp. 669–682, May 2006.

[7] Y. Borghol, S. Mitra, S. Ardon, N. Carlsson, D. Eager, and A. Mahanti, "Characterizing and modelling popularity of user-generated videos," *Perform. Eval.*, vol. 68, no. 11, pp. 1037–1055, Nov. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.peva.2011.07.008

[8] M. Zhao, P. Aditya, A. Chen, Y. Lin, A. Haeberlen, P. Druschel, B. Maggs, B. Wishon, and M. Ponec, "Peer-assisted content distribution in akamai netsession," in *Proceedings of the 2013 Conference on Internet Measurement Conference*, ser. IMC '13. New York, NY, USA: ACM, 2013, pp. 31–42. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504752

[9] H. Yin, X. Liu, T. Zhan, V. Sekar, F. Qiu, C. Lin, H. Zhang, and B. Li, "Livesky: Enhancing cdn with p2p," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 6, no. 3, pp. 16:1–16:19, Aug. 2010. [Online]. Available: http://doi.acm.org/10.1145/1823746.1823750

[10] K. Xu, Y. Zhong, and H. He, "Can p2p technology benefit eyeball isps? a cooperative profit distribution answer," *CoRR*, vol. abs/1212.4915, 2012.

[11] V. Misra, S. Ioannidis, A. Chaintreau, and L. Massoulié, "Incentivizing peer-assisted services: a fluid shapley value approach," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 215–226, Jun. 2010. [Online]. Available: http://doi.acm.org/10.1145/1811099.1811064

[12] Cisco, "Cdn federation," 10 2012, available on http://www.cisco.com/web/about/ac79/docs/sp/CDN-Federation_Phase-2-Pilot.pdf.

[13] W. B. Norton, *The Internet Peering Playbook: Connecting to the Core of the Internet*, 2nd ed. DrPeering Press, December 2012.

[14] N. KAMIYAMA, T. MORI, R. KAWAHARA, and H. HASEGAWA, "Optimally designing isp-operated cdn," *IEICE Transactions on Communications*, vol. E96.B, no. 3, pp. 790–801, March 2013.

[15] IETF, "Cdn interconnect ietf working group," 2013, available on https://datatracker.ietf.org/wg/cdni/. [Online]. Available: https://datatracker.ietf.org/wg/cdni/

[16] "Figaro project," http://www.ict-figaro.eu/, 2012.

[17] M. Cha, P. Rodriguez, S. Moon, and J. Crowcroft, "On next-generation telco-managed p2p tv architectures," in *Proceedings of the 7th international conference on Peer-to-peer systems*, ser. IPTPS'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 5–5. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855641.1855646

[18] W. Jiang, S. Ioannidis, L. Massoulié, and F. Picconi, "Orchestrating massively distributed cdns," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, ser. CoNEXT '12. New York, NY, USA: ACM, 2012, pp. 133–144. [Online]. Available: http://doi.acm.org/10.1145/2413176.2413193

[19] N. MAKI, T. NISHIO, R. SHINKUMA, T. MORI, N. KAMIYAMA, R. KAWAHARA, and T. TAKAHASHI, "Traffic engineering of peer-assisted content delivery network with content-oriented incentive mechanism," *IEICE Transactions on Information and Systems*, vol. E95.D, no. 12, pp. 2860–2869, December 2012.

[20] B. Mathieu and Y. Levene, "Impact of ftth deployment on live streaming delivery systems," in *Computers and Communications (ISCC), 2012 IEEE Symposium on*, july 2012, pp. 000 259 –000 264.

[21] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.

[22] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Characteristics of youtube network traffic at a campus network - measurements, models, and implications," *Comput. Netw.*, vol. 53, no. 4, pp. 501–514, Mar. 2009. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2008.09.022

[23] F. Guillemin, T. Houdoin, and S. Moteau, "Volatility of youtube content in orange networks and consequences," in *Communications (ICC), 2013 IEEE International Conference on*, June 2013, pp. 2381–2385.