INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Peer-to-Peer IPTV Services

Daniel A. G. Manzato
Nelson L. S. da Fonseca

Technical Report - IC-12-17 - Relatório Técnico

June - 2012 - Junho

Peer-to-Peer IPTV Services

Daniel A. G. Manzato   Nelson L. S. da Fonseca*

Abstract

IPTV promises to be within the spectrum of services offered in the future Internet. In cooperative IPTV, clients' resources are typically used to build a scalable system to distribute TV content. One of the major challenges of this approach is to reach the same quality of service of traditional television and commercial IPTV by employing only best effort network layer services. This paper proposes a novel architecture based on P2P networking for cooperative IPTV. Challenges are discussed and some solutions are proposed as part of the architecture. The aim is to make this type of system more competitive with traditional television and commercial IPTV.

1 Introduction

The development of architectures and softwares which hide technical details from users enabled the wide spread use of voice over IP (VoIP) services. Simultaneously, television over IP (IPTV) services have emerged as an alternative delivering system to traditional cable and broadcast systems. However, IPTV technology needs to mature, since its visual quality is still poor when compared to that of traditional systems. Regulatory issues have also prevented IPTV a broader deployment. However, they have brought newcomers to the market such as traditional telephony companies, which offer IPTV services in triple play packets.

IPTV architectures can be either private or public. In private ones, also known as commercial IPTV, the size of the network limits the population served [1]. In public architectures, also known as cooperative IPTV, the Internet provides the infrastructure, enabling a global population to have access to services [1]. Commercial IPTV is normally provided together with Internet broadband access and telephony over IP (triple play services), creating a unique business opportunity which capitalizes on customer fidelity [1].

Cooperative IPTV employs open architectures and takes advantage of clients' resources. Achieving the same quality of service of traditional television and commercial IPTV by adopting best effort

*Institute of Computing, State University of Campinas, Campinas, SP, Brazil, 13089-971. This research was partially sponsored by CNPq.

This technical report is a pre-print of the paper D. A. G. Manzato and N. L. S. da Fonseca, "Peer-to-peer iptv services", In Proc. of the IEEE GLOBECOM Workshops, pp. 1-6, 2008, DOI: 10.1109/GLOCOMW.2008.ECP.29.

service support is the major challenge of this approach [1]. P2P networking is the most intuitive paradigm for cooperative IPTV, since it promotes scalability by using clients' resources. In existing P2P systems designed to deliver video streams in real time, called P2PTV, the video streams are typically TV channels from all over the world [2].

In this paper, a novel architecture based on P2P networking and designed for cooperative IPTV is proposed. Although P2P IPTV is feasible in current Internet, it is still in its infancy [3]. In this paper, common challenges of these systems are discussed and some solutions are incorporated in the proposed architecture. The aim is to make cooperative IPTV more competitive with traditional television and commercial IPTV, narrowing the existing gap of quality between them.

The rest of this paper is organized as follows. In Section 2, common challenges of P2PTV systems are discussed. In Section 3, a novel architecture for cooperative IPTV is described and compared to other architectures and solutions. Details of components, connections and features are also provided. Last, in Section 4, some conclusions are drawn.

2 Challenges of P2PTV Systems

In this section, challenges of P2P IPTV systems are discussed. Issues are addressed in the light of making this technology competitive with traditional television.

2.1 Short Start-up Delay

The PPLive system [4] stores tens of seconds of video frames in buffers before playback [3]. In [5], it was reported that a new peer may spend 10 to 15 seconds to join a P2P overlay and take another 10 to 15 seconds to launch the media player and store video frames in buffers. Although the employment of buffers helps ammeliorating the problems of peer failure and bandwidth fluctuation, the long start-up delays hinder user experience, specially when fast browsing through different channels is desirable. In fact, users of traditional television enjoy the possibility of fast channel switching [3]. Thus, start-up delays need to be shortened from tens of seconds to just a few seconds in P2P IPTV systems [3]. Although redundant data transfer and network coding can ammeliorate this problem, they increase traffic [3]. Moreover, while network coding in P2P file sharing has already produced positive results [6, 7], its employment in P2P media streaming has yet to be better understood [5].

2.2 Need for Incentive Mechanisms

Statistics of P2P network usage point out that only a minority of peers cooperate altruistically in these systems [8]. In [9], it was shown that in some P2P media streaming systems some peers are requested to contribute 10 to 35 times more uploading bandwidth than the downloading bandwidth they consume [9]. Given such demands, it is natural that peers tend not to cooperate without enforcement. Therefore, the employment of effective incentive mechanisms is necessary in systems which depend on user cooperation. In P2P IPTV, this is specially important due to the large bandwidth requirements created by the simultaneous transmission of multiple channels. Although several incentive schemes have been proposed in the literature, the design of a light-weight and scalable incentive mechanism which can be incorporated into video broadcast applications remains as an open problem [5].

2.3 Need for Some Dedicated Infrastructure

It is common in P2P media streaming systems that users be connected by DSLs and cable modems and, consequently, have less outgoing bandwidth than incoming bandwidth. Moreover, the outgoing bandwidth can be smaller than the video playback rate. As a result, more provisioned peers typically cooperate more than they consume, acting as amplifying nodes which compensate the resource deficit caused by clients less provisioned [3,5]. In IPTV systems, resource requirements are usually greater due to the existence of multiple channels. An interesting question is whether or not more provisioned peers (and their ISPs) will have the capability and incentive to continue providing the additional upload traffic [3]. Thus, new emerging P2P IPTV architectures should consider not only an efficient incentive mechanism, but also some level of dedicated infrastructure to compensate resource deficit [3].

2.4 Support for Flash Crowds and High Churns

Flash crowd is a common situation in P2P media streaming systems. It occurs when the arrival rates increase rapidly and remain high for a short period of time, typically due to the broadcast of a popular event [5,10]. The first difficulty under flash crowds is the admission of new peers without degenerating the quality of service for already connected peers. The second difficulty is to have fast repair of delivering structures when a high number of peers depart, so that service interruption is avoided. This situation deteriorates when peers disconnect right after having been accepted since they do not obtain the desired quality of service.

When high churn occurs the system needs to adapt itself continuously to cope with high arrival and departure rates. In IPTV systems, high churns are aggravated by frequent channel switching. The usage of multiple description coding and multiple distribution trees for promoting redundancy in data and network paths can ammeliorate the effects of churns [11–13]. Some works also consider overlays in mesh instead of multiple trees [4,14]. In addition, stimulating longer session times can diminish the consequences of peer disconnections [15,16]. Although all these solutions have been already investigated, the design of a P2P video broadcast system which is robust to extreme peer dynamics still needs to be conceived [5].

2.5 Support for Client Heterogeneity

Another difficulty in P2P media streaming systems is the diversity of access technologies. A stream delivered at a single rate can overwhelm less provisioned peers as well as furnish low visual quality to more provisioned peers. Thus, there is a clear need for mechanisms to adapt to different client bandwidths. Coding videos at multiple rates in parallel at the server for delivering the appropriate stream to each client is one solution to the problem [17]. Moreover, scalable coding techniques such as layered coding [18,19] and, more recently, multiple description coding can also ammeliorate the problem [11–13]. Although these two approaches are effective for video streaming services, most of media players can only support single description coding [5]. Furthermore, progress need to be made in order to diminish the processing and bandwidth overheads of these new coding techniques [5].

2.6 Synchronization of Client Playback

In [3], it was reported large playback lags for the PPLive system, which make some peers watch video frames in a channel minutes behind others. This is particularly undesirable in live broadcasts with user interaction. In order to ammeliorate this problem, better overlay strategies and more efficient scheduling schemes for video chunks can be employed [3]. Among the challenges, there are the prevention of multiple overlay edges from traversing the same physical link and the optimization of communications between end users that traverse other end systems (which increases latency) [5].

2.7 Better NAT and Firewall Traversal

In [3], it was reported the existence of multiple occurrences of private IP addresses in the PPLive system, which is an indication of several peers behind NAT servers. Moreover, it is pointed out in [5] that Internet environments can have over 50% of nodes behind NATs and firewalls. The reachability of peers depends on factors such as the transport protocol utilized, the type of NAT and firewall employed as well as whether or not peers are behind the same network. Depending on the scenario, some peers can not be reachable at all, i.e., their bandwidth will not be used by the system. Thus, the design of efficient schemes for NAT and firewall traversal is a relevant matter to be considered [3, 5].

3 A Novel Architecture for IPTV

In this section, a novel P2P architecture for IPTV is proposed to address the challenges mentioned before.

3.1 Overview

The architecture proposed in this paper employs multiple distribution trees and multiple description coding, as in CoopNet [12], SplitStream [11] and Chunkyspread [20]. Other systems such as CoolStreaming [14] and PPLive [4] employ the mesh-pull approach to increase reliability and explore more efficiently the outgoing bandwidth of peers, since in tree overlays interruptions occur when an internal node disconnects. Besides that, tree overlays do not explore the bandwidth of leaf nodes, which do not admit children under themselves [21]. However, when *multiple* distribution trees are employed, such disadvantages do not exist, since all peers are interior nodes in at least one tree. Moreover, when a subset of trees suffers disruptions, the continuity of the video broadcast is ensured by the other trees, causing only a momentary drop in the perceived quality.

The main reason for adopting multiple distribution trees instead of mesh networks is to reduce the start-up delay, given that the data-driven scheduling cost is eliminated. The employment of an auxiliary structure in mesh to optimize the main structure in trees is an interesting approach when there is a single distribution tree [22]. However, the benefits of such approach when there are multiple distribution trees are still unknown, demanding further investigation. Therefore, the proposed architecture adopts only multiple distribution trees.

Multiple distribution trees demand maintenance of explicit delivering structures. The Chunkyspread system [20], which also employs multiple distribution trees, strives to compensate these costs by designing simple and scalable algorithms for the construction and optimization of trees. In this paper, the proposed architecture tries to compensate the same costs by employing multiple servers, either dedicated or temporary, in a way that the tree management overhead is shared among themselves and communication costs are localized.

In order to compensate the resource deficit caused by less provisioned peers, the proposed architecture includes dedicated servers, enabling not only the transmission of large bandwidth streams, but also the coexistence of multiple channels. The proposed architecture differs from CoopNet, SplitStream and Chunkyspread, which include only data seed servers, without any dedicated intermediary layer. Another difference from those systems is the adoption of multiple stream providers, which makes the proposed architecture closer to a full IPTV service.

To provide full IPTV services, the proposed architecture features a scheme for channel switching with short start-up delays. This scheme takes advantage of multiple description coding and is based on redundant data transfer, organization of the multiple distribution trees and management of multiple buffers in stream media players. This is a distinct aspect of the proposed architecture.

Given the increasing demand for bandwidth created by high quality streams [23, 24] and the existence of multiple channels, the proposed architecture features a specific incentive mechanism, which aims at increasing cooperation as well as stimulating peers to act as temporary servers. In this scheme, temporary and dedicated servers share the tree management overhead and the responsibility to deliver TV content. The incentive mechanism is also a unique feature of this architecture.

As CoopNet, SplitStream and Chunkyspread, the proposed architecture addresses flash crowds and high churns by employing multiple distribution trees and multiple description coding. The proposed architecture addresses client heterogeneity by employing layered multiple description coding, as the last release of CoopNet [13]. With that, less provisioned peers receive only the descriptions of the base layer, while more provisioned peers receive, in addition, the descriptions of enhancement layers. Client heterogeneity can also be treated by the incentive mechanism together with bandwidth adaptation [15, 16]. It is also possible to stimulate longer session times to ammeliorate the effects of peer disconnections. Such approach also helps handling flash crowds and high churns.

The proposed architecture does not address the synchronization of client playback. Another challenge not covered is better NAT/firewall traversal. It is known that the NAT traversal scheme employed by Joost [25] is based on the STUN protocol [26], which lets peers behind NATs discover their public address, the type of NAT and the public ports associated to the local ports.

3.2 Detailed Description

The proposed architecture is composed by four layers. Such layering takes into account the volume of traffic transmitted among components (Figure 1).

3.2.1 First Layer

Stream providers (SPs) reside in the first layer; they can be compared to broadcast stations in traditional television. An SP can produce live TV content, audio streams and pre-stored content. The functionality of this layer is to support multiple channel content generation, making the proposed architecture closer to a full IPTV service rather than a simple multicast P2P media streaming.

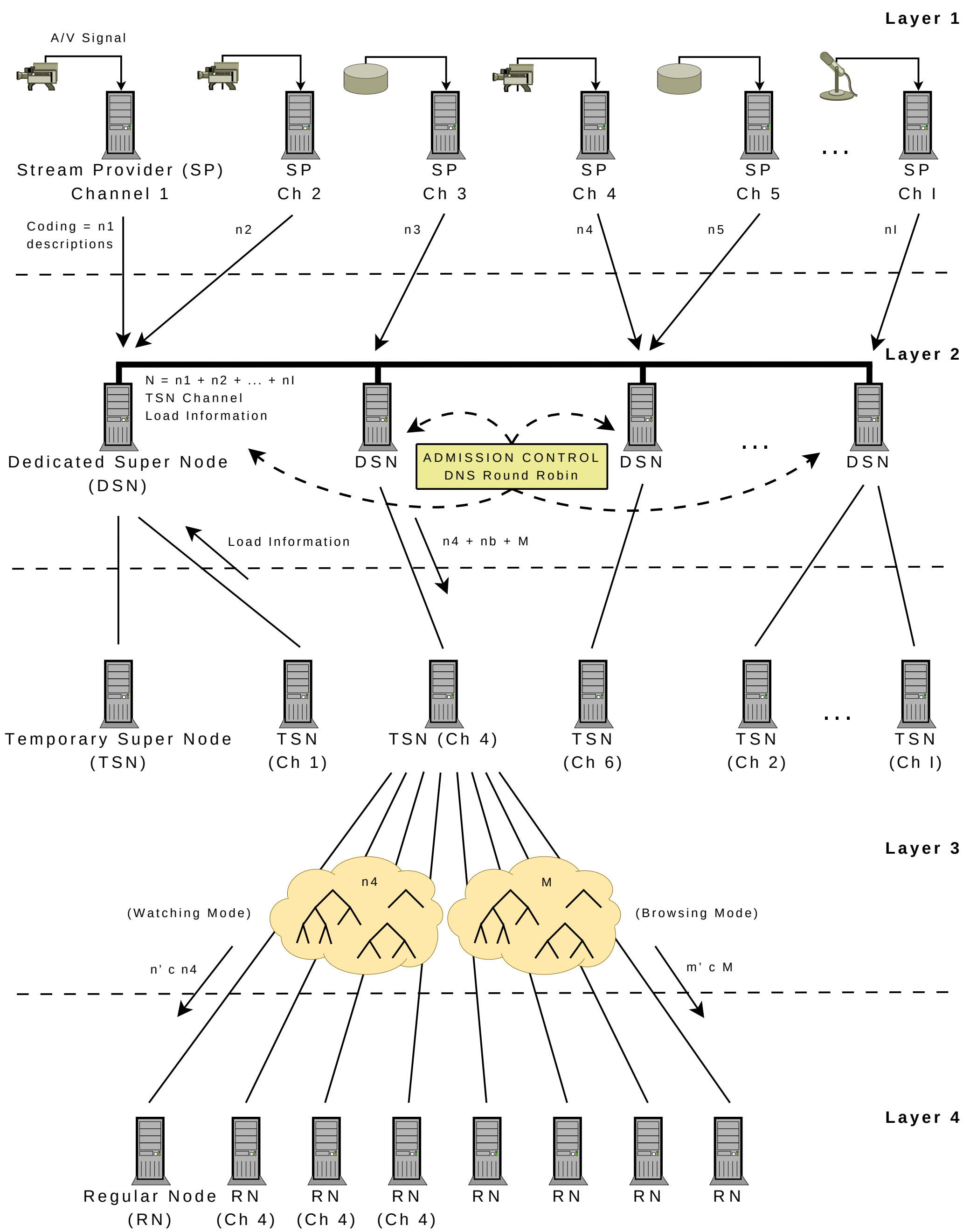

Figure 1: Proposed Architecture for Peer-to-Peer IPTV.

The stream generated by an SP is coded into layered multiple description coding. Multiple description coding have been employed for various purposes such as promoting redundancy in data and network paths [11, 12], adaptation to peer heterogeneity [13, 15] and implementation of incentive mechanisms [15, 27]. The proposed architecture also employs multiple description coding to facilitate browsing through multiple channels. Although the number of descriptions can vary for each channel, their bandwidth must be the same after coding. Thus, a higher quality channel will have a greater number of descriptions than a lower quality one.

3.2.2 Second Layer

In this layer, SPs are connected to dedicated super nodes (DSNs). The functionality of this layer is: (1) to make available at each DSN all existing TV channels, (2) to provide dedicated infrastructure to compensate the resource deficit caused by less provisioned peers and (3) to coordinate the admission of new peers in the system.

The main difficulty in this layer is the distribution of channels among all DSNs. Ideally, to reduce the number of hops, direct connections among all DSNs could be provisioned. As a result, each DSN should have an outgoing bandwidth of at least the bandwidth for the content produced by all its SPs times the number of remaining DSNs. Similarly, each DSN should have an incoming bandwidth of at least the bandwidth for the content produced by all SPs in the system. Such approach is only feasible for a small number of DSNs and channels. Alternatively, a solution based on application multicast can be employed. However, this has the shortcoming of increasing the number of hops and, hence, the start-up delay.

3.2.3 Third Layer

The components of the third and fourth layers are the participating peers. In the third layer, there are temporary super nodes (TSNs), which are more provisioned peers. The functionality of this layer is to select multiple TSNs that jointly with the DSNs will share the responsibility of distributing TV content to all peers in the system. In particular, TSNs also share the burden of managing multiple distribution trees.

While DSNs have locally available all the existing TV channels, TSNs have available only the channel being forwarded and the channel being watched. Additionally, TSNs receive samples of the non-forwarded channels in order to implement the scheme for channel switching described next. Suppose that a TSN is forwarding Channel a and watching Channel b. Such TSN receives the descriptions $na + nb + M$, where na is all the descriptions of Channel a, nb is all the descriptions of Channel b and M is the set of all initial descriptions of the non-forwarded channels.

3.2.4 Fourth layer

In this layer, there are regular nodes (RNs), which are peers: (1) less provisioned; (2) not willing to cooperate more than the necessary; and (3) more provisioned and willing to cooperate, but who have not been selected by the system due to the existence of better suited peers to become TSNs. The functionality of this layer is to bring together all peers that are not acting as TSNs, but that are forwarding descriptions in the existing trees. Depending on the channel selection of a particular RN, it stays admitted under a certain TSN, based on which TSNs are forwarding that channel.

RNs are admitted by TSNs similarly as in the CoopNet system. Each RN is admitted as an interior node in just one tree and as a leaf node in the remaining trees. In this way, trees become more diverse and hence robustness is increased [11]. Differently from CoopNet, there are two sets of trees to implement the scheme for channel switching: one for delivering the descriptions of the selected channel and the other for delivering the initial descriptions of the non-selected channels. As an RN is admitted into only one set of trees at a time, no additional bandwidth is required to implement such scheme.

Each RN can receive only a subset $n' \subset ni$ of the n descriptions of a given selected channel i, in case its incoming bandwidth is not sufficient to accommodate all ni descriptions. Similarly, each RN can receive only a subset $m' \subset M$ of the initial descriptions of the non-selected channels. In this way, peer heterogeneity can be handled by not admitting RNs into all distribution trees.

3.3 Scheme for Channel Switching

RNs are always admitted by TSNs. An RN can be in two states: “watching mode” or “browsing mode”. In the former, the peer is admitted into the first set of trees, which delivers the descriptions of the selected channel. In the latter, the peer is admitted into the second set of tress, which delivers the initial descriptions of the non-selected channels.

When an RN is watching a channel and decides to check the programs in exhibition on other channels, it leaves the state “watching mode” and enters into the state “browsing mode”, migrating from the first to the second set of trees. The message overhead for this operation is limited to the scope of a TSN, not overloading the components of the upper layers. The start-up delay for the new stream is bounded by the period of time necessary for migrating to the other set of trees and for filling up the playback engine buffers, since TSNs receive, continuously, the initial descriptions of the non-selected channels.

One enhancement is to let the playback engine manage multiple buffers for different streams simultaneously, i.e., while receiving and playing a channel, it also receives the remaining channels, storing these other streams in buffers in a first in first out policy. As long as the aggregated bandwidth is the same for both set of trees, the memory capacity required for this buffer arrangement is equivalent to that of all descriptions being received for the same channel. This approach makes channel switching faster, limiting the start-up delay to the change from the state “watching mode” to the state “browsing mode”. However, if an RN’s incoming bandwidth is not sufficient to receive all initial descriptions delivered in the second set of trees, then a paging or swapping scheme is necessary.

When an RN is browsing through channels and selects a specific one, it leaves the state “browsing mode” and enters into the state “watching mode”. In this case, it must be admitted by a new TSN which forwards the descriptions of the selected channel. A major challenge is to smooth the synchronization of the stream from the old TSN with that from the new TSN, until the old TSN can be dismissed.

TSNs can freely change their channel selection without affecting the RNs admitted under themselves. They do not have to be admitted into two different sets of trees, since they receive all the descriptions through the same connection with the admitting DSN. They just switch the playback engine’s buffers to load either the initial descriptions of the non-selected channels or the descriptions of the selected channel at a time. Since TSNs always receive all the initial descriptions of the non-selected channels, no paging scheme is necessary to accommodate subsets of channels.

3.4 Admission Control

All DSNs are registered in the DNS in a circular way. When a peer wants to connect to the system, it first sends a query to the DNS, which returns the address of one existing DSN. When contacted, the DSN first checks whether the peer has the capabilities to become a TSN and whether or not he or she is willing to do that. If so, the peer is admitted as a TSN of a channel with deficit of TSNs. The admitting DSN is not necessarily the one returned by the DNS; instead, the least loaded DSN is contacted, based on load information exchanged among all DSNs.

The peer is admitted as an RN when: (1) it does not have the capabilities to become a TSN or (2) the user does not want to become a TSN or (3) there is more TSNs in the system than the necessary. In this case, the DSN initially contacted determines the best TSN to admit the new node, based on load information. As illustrated in Figure 1, TSNs send their load information to DSNs. In addition to their own load information, DSNs distribute the ones received from the TSNs, as well as the information of which TSNs are forwarding each existing channel.

The DSN initially contacted returns to the new peer the TSN chosen. The new peer then contacts the TSN, requesting an admission operation. The new peer is then admitted in the second set of trees to receive the initial descriptions of the non-selected channels, entering into the state “browsing mode”. If the new peer knew a channel selection a priori, then the DSN initially contacted could have selected a TSN which was forwarding the same channel. In this case, the new peer would be admitted into the first set of trees, entering into the state “watching mode”.

3.5 Incentive Scheme

The incentive mechanism of the proposed architecture is credit-based; credits result from peers’ cooperation and they are stored in bank accounts. By allowing both cooperation and remuneration to occur at different moments, some benefits can be achieved, such as: (1) the system can take advantage of peers which are offline and decided to continue cooperating to accumulate credits; (2) peers using their outgoing bandwidth for other applications can obtain a video without cooperating, as long as they have enough credits to pay for their immediate consumption; (3) less provisioned peers can obtain, by using their credits, a stream with higher quality than the one they would obtain in a tit-for-tat scheme; (4) more provisioned peers can keep cooperating continuously more than the necessary to accumulate credits so that they can be reverted into later compensation, at the end of a stipulated period.

The last mentioned benefit can be compared to the Internet advertisement business model, in which affiliate websites earn revenue for displaying pre-determined advertisements, making the publishing platform broader [28]. Similarly, part of the revenue of this platform could be transfered to peers which cooperate as TSNs. This could reduce the infrastructure needed, since more provisioned peers would be hired dynamically to help in the distribution of videos.

In this way, it becomes necessary to classify TSN cooperators so that fairness can be promoted as well as to provide the system a means to select the best peers. The system should consider two capability-related metrics, namely throughput and delay, and two behavioral metrics, namely session mean time and abrupt disconnection. Such classification scheme is not necessary for RNs, which only have their credits computed periodically.

Following this rationality, the incentive scheme of the proposed architecture is comprised by the following components [29]:

- *Banking:* All peers have a bank account to store the credits resulting from the excess of cooperation. The credits accumulated can be used later to compensate low or no cooperation for a period of time. For the TSNs, such credits can also be reverted in revenue at the end of a stipulated period.
- *Barter Trade:* It ensures a balance between consumption and cooperation while a peer is in the system. When the cooperation is greater than consumption, the excess is stored in a bank account. The consumption can only be greater than the cooperation when a peer has accumulated credits. Debit is accounted at every unit of time, according to the deficit measured.
- *Community (Reputation):* It is employed only for TSNs, to classify cooperators according to their throughput, delay, session mean time and abrupt disconnection. It promotes fairness and provides the system a means to select the best cooperators.

4 Conclusion

In this paper, a novel architecture based on P2P networking and designed for cooperative IPTV was proposed. Common challenges were discussed and some solutions for them were proposed as part of the architecture. The aim of this work is to make this type of system more competitive with traditional television and commercial IPTV, narrowing the existing gap of quality between them and, consequently, increasing the popularity of cooperative IPTV. The architecture is currently under development and it is our hope to have the evaluation of its efficacy in a near future.

References

[1] C. V. Albuquerque, T. Proença, and E. Oliveira, "Tvoip: Tv sobre ip arquiteturas para transmissão em larga escala," in *Proceedings of the 24th Brazilian Symposium on Computer Networks (SBRC 2006)*, May 2006.

[2] "P2ptv," http://en.wikipedia.org/wiki/P2PTV, Aug. 2008, (Access Date).

[3] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, "A measurement study of a large-scale p2p iptv system," *To appear in IEEE Transactions on Multimedia*, Nov. 2007.

[4] "Pplive," http://www.pplive.com/en/index.html, Aug. 2008, (Access Date).

[5] J. Liu, S. G. Rao, B. Li, and H. Zhang, "Opportunities and challenges of peer-to-peer internet video broadcast," *To appear in Proceedings of the IEEE, Special Issue on Recent Advances in Distributed Multimedia Communications*, 2007.

[6] C. Gkantsidis and P. Rodriguez, "Network coding for large scale content distribution," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies IEEE (INFOCOM 2005)*, vol. 4, Mar. 2005, pp. 2235–2245.

[7] C. Gkantsidis, J. Miller, and P. Rodriguez, "Comprehensive view of a live network coding p2p system," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement (IMC '06)*. New York, NY, USA: ACM, 2006, pp. 177–188.

[8] E. Adar and B. A. Huberman, "Free riding on gnutella," *First Monday*, vol. 5, no. 10, Oct. 2000.

[9] S. Ali, A. Mathur, and H. Zhang, "Measurement of commercial peer-to-peer live video streaming," in *First Workshop on Recent Advances in Peer-to-Peer Streaming*, Aug. 2006.

[10] K. Sripanidkulchai, A. Ganjam, B. Maggs, and H. Zhang, "The feasibility of supporting large-scale live streaming applications with dynamic application end-points," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 107–120, 2004.

[11] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth multicast in a cooperative environment," in *Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP'03)*. ACM Press, Oct. 2003, pp. 298–313.

[12] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," in *Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP)*. IEEE Computer Society, Nov. 2003, pp. 16–27.

[13] ——, "Supporting heterogeneity and congestion control in peer-to-peer multicast streaming," in *Proceedings of the Third International Workshop on Peer-to-Peer Systems (IPTPS)*. Springer, Feb. 2004, pp. 54–63.

[14] X. Zhang, J. Liu, B. Li, and T. S. P. Yum, "Donet/coolstreaming: A data-driven overlay network for live media streaming," in *IEEE INFOCOM*, vol. 3, Mar. 2005, pp. 2102–2111.

[15] D. A. G. Manzato and N. L. S. da Fonseca, "An incentive mechanism for peer-to-peer networks with media streaming," in *Proceedings of the 49th IEEE Global Telecommunication Conference (GLOBECOM 2006)*, Nov. 2006, pp. 1–6.

[16] ——, "Incentive mechanism for the coopnet network," *Peer-to-Peer Networking and Applications (P2PNA)*, vol. 1, no. 1, pp. 29–44, Mar. 2008.

[17] Y. hua Chu, S. G. Rao, and H. Zhang, "A case for end system multicast," in *Proceedings of the 2000 ACM SIGMETRICS international conference on Measurement and modeling of computer systems (SIGMETRICS '00)*. New York, NY, USA: ACM, 2000, pp. 1–12.

[18] Y. Cui and K. Nahrstedt, "Layered peer-to-peer streaming," in *Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video (NOSSDAV '03)*. New York, NY, USA: ACM, 2003, pp. 162–171.

[19] J. Liu, B. Li, and Y.-Q. Zhang, "Adaptive video multicast over the internet," *IEEE MultiMedia*, vol. 10, no. 1, pp. 22–33, 2003.

[20] V. Venkataraman, P. Francis, and J. Calandrino, "Chunkyspread: Multitree unstructured peer to peer multicast," in *Proceedings of the Fifth International Workshop on Peer-to-Peer Systems (IPTPS '06)*, Feb. 2006.

[21] N. Magharei, R. Rejaie, and Y. Guo, "Mesh or multiple-tree: A comparative study of live p2p streaming approaches," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, May 2007, pp. 1424–1432.

[22] F. Wang, Y. Xiong, and J. Liu, "mtreebone: A hybrid tree/mesh overlay for application-layer live video multicast," in *Proceedings of the 27th International Conference on Distributed Computing Systems (ICDCS '07)*. Washington, DC, USA: IEEE Computer Society, 2007, p. 49.

[23] N. L. S. da Fonseca and R. de A. Façanha, "The look-ahead-maximize-batch batching policy," *IEEE Transactions on Multimedia*, vol. 4, no. 1, pp. 114–120, 2002.

[24] N. L. S. da Fonseca and H. K. S. Rubinsztejn, "Dimensioning the capacity of true video-on-demand servers," *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 932–941, oct 2005.

[25] "Joost," http://www.joost.com, Aug. 2008, (Access Date).

[26] J. Rosenberg, J. Weinberger, C. Huitema, and R. Mahy, "STUN - simple traversal of user datagram protocol (UDP) through network address translators (NATs)," RFC 3489, IETF, Mar. 2003.

[27] Y. hua Chu, J. Chuang, and H. Zhang, "A case for taxation in peer-to-peer streaming broadcast," in *Proceedings of the ACM SIGCOMM workshop on Practice and theory of incentives in networked systems (PINS '04)*. New York, NY, USA: ACM Press, 2004, pp. 205–212.

[28] "Google advertising programs," http://www.google.com/ads, Aug. 2008, (Access Date).

[29] P. Obreiter and J. Nimis, "A taxonomy of incentive patterns – the design space of incentives for cooperation," in *Proceedings of the Second International Workshop on Agents and Peer-to-Peer Computing (AP2PC'03), Springer LNCS 2872*, Jul. 2003, pp. 89–100.