An economic replica placement mechanism for streaming content distribution in Hybrid CDN-P2P networks

Mehran Garmehi [a,*], Morteza Analoui [a], Mukaddim Pathan [b], Rajkumar Buyya [c]

[a] School of Computer Engineering, Iran University of Science and Technology, Iran
[b] Media Distribution, Telstra Corporation Limited, Australia
[c] CLOUDS Lab, The University of Melbourne, Australia

A B S T R A C T

Hybrid CDN-P2P networks blend Content Delivery Networks (CDN) and Peer-to-Peer (P2P) networks to overcome their shortcomings. Replica placement in these networks is still an open problem. Hierarchical structure of these networks makes it inefficient to use available replica placement strategies specialized to CDN or P2P network domains. In this work, we introduce a novel replication strategy which enables traditional CDNs to offer Hybrid CDN-P2P streaming content delivery services. The proposed solution relies on economic model of the Hybrid CDN and employs a dynamic mechanism to optimize the number and places of replicas for P2P service. In particular, processing and communication complexities of the proposed solution are studied and its performance is analyzed in a typical Hybrid CDN-P2P network in comparison to related strategies.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Hybrid CDN-P2P networks (HCDN) blend Content Delivery Network (CDN) and Peer-to-Peer (P2P) technologies. They benefit from the advantages of both architectures. CDNs benefit from the ease of management and are able to guarantee high Quality of Service (QoS) [1], but compared to P2P networks, they have higher costs and lower scalability [2]. P2P networks have low running costs and are more scalable, but it is not easy to guarantee their QoS and their management is more challenging [3]. Hybrid CDN-P2P networks combine these technologies to offer low priced high quality services [4].

In HCDNs, users can receive the service in either of Client–Server (CS) or Peer-to-Peer (P2P) approaches. In P2P mode, users need to contribute to content delivery using their upload bandwidth. In order to incentivize the end-users to contribute to P2P content delivery, HCDNs offer P2P service with higher quality compared with CS service [3]. Therefore, for streaming content distribution, an HCDN needs to replicate the stream with different qualities for CS and P2P end-users. In this work, by introducing a novel Replica Placement (RP) mechanism, a traditional CDN has been enabled to offer hybrid streaming content delivery services.

It is assumed that the CDN employs its own replica placement strategy for distribution of the content in CS mode. Based on this assumption, a dynamic replica placement mechanism is designed to replicate the streaming content in P2P mode. This RP mechanism requires cooperation of edge servers and relies on communication of a few small messages to determine the best location of replicas for higher quality content distribution in P2P mode.

HCDNs receive the content from content providers and deliver it to end-users either in CS or P2P modes. They charge the content providers based on the delivered traffic and the Service Level Agreement (SLA), similar to traditional CDNs. In HCDNs, higher QoS in P2P mode encourages users to contribute in P2P content delivery using their upload bandwidth. Therefore, in P2P content delivery approach, the HCDN spends less bandwidth to deliver the content to end-users. As a result, in the SLA between the content provider and HCDN provider, price of content delivery in P2P mode must be kept less than traditional CS service. This is necessary to prevent the HCDN from reducing to a pure P2P service provider and to convince the content provider to request for the hybrid content delivery service. HCDN spends different amount of resources for CS and P2P end-users and charges the content provider differently for CS and P2P services. Therefore, in order to select the best number and places of replicas of P2P content, not only the content delivery costs and popularity of the content (like traditional CDNs) but also revenue of the HCDN must be taken into account.

---

* Corresponding author. Tel.: +98 9123372605; fax: +98 5842244071.
  E-mail address: Garmehi@iust.ac.ir (M. Garmehi).

CDNs, P2P networks and consequently HCDNs consist of thousands of servers and peers. This raises concern over the resource consumption of the RP mechanism. Moreover, high expected availability makes it mandatory to implement the RP as a distributed algorithm. The RP problem in this domain has many varying parameters like popularity of the content, prices of bandwidth and storage, and structure of the network. Therefore, the solution must consider many effective parameters but it must not spend much of valuable network bandwidth for transferring these parameters to and from the agents. It is also important to consume the minimum processing resources to solve the problem.

Existing studies of replica placement in HCDNs [6,7], are confined to replication of static content and employ non-cooperative pull-based approach. In this work, we introduce a cooperative push-based replication strategy that extends the use of the HCDN to streaming content distribution, which is called Hybrid Replica Placement (HPR). To answer the resource consumption concerns, the economic model of the system is introduced. Then a hierarchical tree-based cooperative replica placement is proposed based on the Economic Mechanism Design Theory [5]. Experiments show that using HPR, it is possible to find the optimum placement of replicas. The major contribution of this paper is proposing an economic RP mechanism for replication of streaming content on a traditional CDN to offer Hybrid CDN-P2P streaming content delivery service.

The rest of the paper is organized as follows. We explore related works and compare the problem with similar problems in Section 2. In Section 3, the economic model of the HCDN is introduced. Section 4 is devoted to formal definition of the economic replica placement problem. In Section 5 the economic push based cooperative replica placement mechanism is introduced. The solution is studied through simulation and its outcome is compared with some other strategies in Section 6. Practical implications are discussed in Section 7. Finally the paper is concluded in Section 8.

2. Background and related work

Effectiveness of combining CDNs and P2P networks has been covered in earlier literature [8,9]. In [10] the cost savings in Hybrid CDN-P2P architecture is studied. A survey of different architectures and models of Hybrid CDN-P2P networks is reported in [11]. In some works, e.g. [12,13,8], the architecture of the HCDN is inspired from P2P systems and the CDN part is employed to fill the missing blocks of the playback buffer. Authors in [14] have tried to introduce a resource scheduling mechanism for these networks. In the literature, there is also another class of HCDN architecture that can be deployed as a commercial Hybrid CDN-P2P network. They can extend existing commercial CDNs and are able to tolerate the fact that some end-users might not prefer to install the special hybrid content distribution application on their machines [15]. This architecture is able to serve the end-users directly; therefore it is not mandatory for end-users to install the application software. However, end-users equipped with this special software are able to receive the content with higher quality [9]. Livesky [15] can be referred as one of these architectures and also as the first successful deployment of Hybrid CDN-P2P networks. It extends a traditional CDN to a Hybrid CDN-P2P network and has succeeded to serve 10 million concurrent requests for streaming videos. Accordingly, the architecture introduced in this network was adopted in the present study.

CDNs host the third-party content for high performance delivery of any digital content, including static files, dynamic content, streaming media (e.g. audio, real time video), and different content services (e.g. e-commerce service) [1]. Among these categories, HCDNs are most suited for distribution of streaming content,

where it is possible to offer the end-users a higher level of quality and benefit from their contribution [12]. It can be observed that most of the works in the field of HCDNs are concentrated on distribution of streaming content [11].

Replica Placement (RP) and Request Routing (RR) are two key mechanisms affecting performance of CDNs [16]. Most commercial CDNs rely on DNS-based request routing mechanism [1]. Consequently, in order to optimize the performance and economics of content distribution, the RP strategy must be designed considering that the RR mechanism has a predefined DNS-based behavior. Despite the fact that there has been a significant body of research on replica placement in traditional CDNs [1] and P2P networks [3], replica placement in Hybrid CDN-P2P networks is studied just in [6,7]. Both of these works are confined to placement of replicas of static content while in this work we employ the HCDN for distribution of streaming content.

In CDNs, content replication is performed using cooperative push-based, non-cooperative pull-based or cooperative pull-based approaches [1]. In the non-cooperative pull-based approach, requests of clients are redirected to their closest edge server and edge servers pull content from the origin server in case of cache misses. Many commercial CDNs use this approach [16]. This approach leads to high traffic on the origin server and seems to be inefficient for replication of high quality streaming content in a hybrid CDN environment. The cooperative pull-based approach differs from the non-cooperative approach in the sense that edge servers cooperate with each other to get the requested content in case of a cache miss. This approach is employed mostly by academic CDNs like Coral [17]. This approach also seems to be inefficient in HCDNs, since churn rate of P2P end-users makes it necessary to run the replication algorithm when a new P2P end-user appears. Cooperative push-based approach relies on the prefetching of content to the edge servers. Content is pushed to the edge servers from the origin, and edge servers cooperate to reduce replication and update cost [1]. In the literature, push-based cooperative RP algorithms have been used successfully in many domains from Data-Grids [18], and P2P networks [19] to CDNs [1]. This approach is compatible with our problem and we have introduced our solution based on this approach. Our solution has two key differences with aforementioned works; first, instead of trying to minimize replication costs, it considers both revenue and costs of the system and maximizes the net profit of the HCDN. Second, in order to preserve compatibility with the CDN environment, it cooperates with the existing replica placement of the CDN and considers current CS content delivery service.

Several works can be found in the literature demonstrating the effectiveness of different replication strategies for the content. Under cooperative push-based approach, greedy-global heuristic algorithm is reported to be efficient for making replication decision among cooperating edge servers [20]. Kangasharju et al. [20] have employed four heuristics, namely random, popularity, greedy-single, and greedy-global for replication of the content and in this work we compare outcome of the proposed solutions with this set of strategies.

Thanks to the high potential of Economic Mechanism Design Theory [5], we have employed it to introduce the problem formally and design an efficient mechanism considering resource consumption. Implementation of privacy preserving mechanisms demands minimum of communication resources [5]. Economic mechanisms are classified as direct and indirect. In direct mechanisms, all the decisions are made based on private information of each agent plus information which is transferred to it by other agents. These mechanisms demand less processing resources [5]. In order to guarantee that minimum communication and processing resources are spent in the implementation; our solution is based on a direct and privacy preserving mechanism.

3. Economic model of the HCDN

Table 1 summarizes the notations used in this paper to model economics and performance of streaming content distribution in HCDN. Fig. 1 illustrates architecture of a Hybrid CDN-P2P network employed to define the replica placement problem. In these networks, the content provider signs an SLA with the HCDN provider and supplies the content i having two different qualities ($BW_{CS,i}$ and $BW_{P2P,i}$) to distribution servers of the HCDN. The revenue in the contracted SLA_i for distribution of content i between a commercial HCDN provider and content provider can be quantified as follows:

$$R_{SLAi} = P_{CS,i} \cdot BW_{CS,i} \cdot N_{totalCS,i} + P_{P2P,i} \cdot BW_{P2P,i} \cdot N_{totalP2P,i}, \quad (1)$$

where R_{SLAi} determines the revenue of the Hybrid CDN provider according to SLA_i. $N_{totalCS,i}$ and $N_{totalP2P,i}$ represent total number of users served with content i using CS and P2P approaches, respectively. $P_{CS,i}$ and $P_{P2P,i}$ denote contracted prices of bandwidth in CS and P2P modes. Bandwidth of content i in CS and P2P modes are indicated by $BW_{CS,i}$ and $BW_{P2P,i}$, respectively. In order to keep P2P content distribution economic and also prevent the HCDN from becoming a pure CDN, it is assumed that $P_{CS,i} > P_{P2P,i} > 0$. Additionally, $BW_{CS,i} \ll BW_{P2P,i}$ guarantees that the users have enough incentives to contribute to P2P content delivery. The cost function of content distribution at the edge servers in Hybrid CDN-P2P network for SLA_i can be represented as:

$$Cost_{SLAi} = DistCost_i + RepCost_i$$
$$= \sum_{ES \in Replicas_i} (DistCost_{ES,i} + RepCost_{ES,i}), \quad (2)$$

where the set $Replicas_i$ denotes all the edge servers containing a replica of the content i. $DistCost_i$ and $RepCost_i$ indicate costs spent by HCDN for distribution and replication of the content i. $DistCost_{ES,i}$

and $RepCost_{ES,i}$ denote content distribution and replication cost on edge server ES for content i.

In this system, the RR mechanism redirects end-users requesting the content i to the suitable edge servers containing replicas of i. End-users have the option to request content i in either of CS or P2P modes. Therefore, each edge server receives $Req_{ES,CS,i}$ and $Req_{ES,P2P,i}$ requests for the stream in CS and P2P modes, respectively. Users interested to receive the higher quality content in P2P mode need to have their machines equipped with the special software and in practice [6] it is not likely to have the software installed on all the receiving machines. Therefore, an edge server even with high number of P2P receivers cannot stop to serve the content in CS mode [15]. When the number of requests on an edge server is small, all the requests are served in CS mode but whenever the number of P2P enabled end-users exceeds the bootstrapping threshold, P2P content delivery begins [14]. Therefore, we have:

$$Rep_{P2P,i} \subset Rep_{CS,i} = Replicas_i, \quad (3)$$

where $Rep_{P2P,i}$ and $Rep_{CS,i}$ represent the set of edge servers containing replicas of high quality content (P2P mode) and base quality content (CS mode). In HCDNs, P2P content delivery becomes promising when the number of end-users interested to receive the content in P2P mode increases beyond a bootstrapping threshold, T_B. In other words, when the number of end-users requesting to receive the content in P2P mode is below the bootstrapping threshold or when the edge server does not contain a replica of the requested content for P2P content delivery, all the end-users will be served with the base quality CS content. Therefore, we have:

$$\begin{cases} N_{P2P,ES,i} = 0 \text{ and } N_{CS,ES,i} = Req_{CS,ES,i} + Req_{P2P,ES,i} & \text{if } (ES \notin Rep_{P2P,i} \text{ or } N_{P2P,ES,i} < T_B) \\ N_{P2P,ES,i} = Req_{P2P,ES,i} \text{ and } N_{CS,ES,i} = Req_{CS,ES,i} & \text{if } (ES \in Rep_{P2P,i} \text{ and } N_{P2P,ES,i} > T_B) \end{cases}, \quad (4)$$

Table 1
Summary of notations.

Notation	Description
R_{SLAi}	Revenue of the Hybrid CDN provider based on SLA_i
$N_{totalCS,i}$	Total number of users served with content i using CS approach
$N_{totalP2P,i}$	Total number of users served with content i using P2P approach
$P_{CS,i}$	Price of bandwidth in CS mode for content i
$P_{P2P,i}$	Price of bandwidth in P2P mode for content i
$BW_{CS,i}$	Bandwidth of content i in CS mode
$BW_{P2P,i}$	Bandwidth of content i in P2P mode
$DistCost_i$	Costs spent by HCDN for distribution of the content i
$RepCost_i$	Costs spent by HCDN for replication of the content i
$DistCost_{ES,i}$	Costs spent by ES for distribution of the content i
$RepCost_{ES,i}$	Costs spent by ES for replication of the content i
$Req_{ES,CS,i}$	Number of end users requesting to receive the stream of content i in CS mode from ES
$Req_{ES,P2P,i}$	Number of end users requesting to receive the stream of content i in P2P mode from ES
ES	An edge server of the HCDN
$N_{CS,ES,i}$	Number of end users receiving content i in CS mode from edge server ES
$N_{P2P,ES,i}$	Number of end users receiving content i in P2P mode from edge server ES
$Replicas_i$	Set of edge servers containing replicas of the content i
$Rep_{CS,i}$	Set of edge servers containing replicas of stream of the content i in CS mode
$Rep_{P2P,i}$	Set of edge servers containing replicas of stream of the content i in P2P mode
T_B	Bootstrapping threshold of P2P content distribution
$BW_{ES,i}$	Total bandwidth consumption for distribution of content i at the edge server ES
$BW_{CS,ES,i}$	Bandwidth spent for distribution of the content i in CS mode at the edge server ES
$BW_{P2P,ES,i}$	Bandwidth spent for distribution of the content i in P2P mode at the edge server ES
$\alpha_{ES,i}$	Effectiveness factor of distribution of content i in P2P mode at edge server ES
BWP_{ES}	Price of bandwidth at edge server ES
$TransCost_{CS,ES,i}$	Replica transmission cost for stream of content i for distribution in CS mode from the source server to ES
$TransCost_{P2P,ES,i}$	Replica transmission cost for stream of content i for distribution in P2P mode from the source server to ES
$StrCost_{CS,ES,i}$	Buffering cost for stream of content i for distribution in CS mode at ES
$StrCost_{P2P,ES,i}$	Buffering cost for stream of content i for distribution in P2P mode at ES
$PBW_{Source,ES}$	Price of network bandwidth in path from $Source$ to ES
$PSTR_{ES}$	Price of storage at ES
$Size_{P2P,i}$	Size of the content for distribution in P2P mode
$NetProfit_{SLAi}$	Profit or utility of the Hybrid CDN-P2P network due to SLA_i
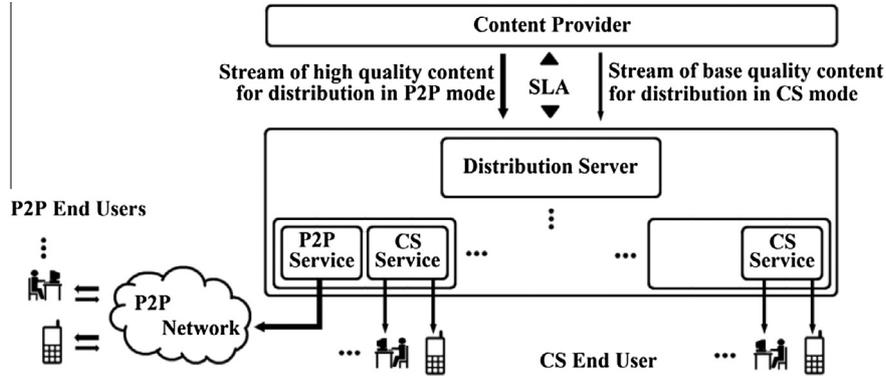
Fig. 1. Architecture of the Hybrid CDN-P2P network.

where $Req_{P2P,ES,i}$ and $Req_{CS,ES,i}$ denote the number of end-users requesting to receive content i from the edge server ES in P2P and CS mode, respectively. $N_{CS,ES,i}$ and $N_{P2P,ES,i}$ indicate the number of end-users receiving content i in CS and P2P modes from ES, respectively. T_B represents the bootstrapping threshold of P2P content distribution.

The bandwidth consumption of an edge server to serve CS and P2P end-users receiving content i is expressed as follows:

$$BW_{ES,i} = BW_{CS,ES,i} + BW_{P2P,ES,i}$$
$$= BW_{CS,i} \cdot N_{CS,ES,i} + (1 - \alpha_{ES,i}) \cdot BW_{P2P,i} \cdot N_{P2P,ES,i}, \qquad (5)$$

where $BW_{ES,i}$, $BW_{CS,ES,i}$ and $BW_{P2P,ES,i}$ symbolize the total bandwidth consumption, the bandwidth spent for CS end-users ($N_{CS,ES,i}$) and the bandwidth spent to serve P2P end-users ($N_{P2P,ES,i}$) at this edge server, respectively. $\alpha_{ES,i}$ represents the effectiveness factor of P2P content delivery for content i at this edge server. The prices of bandwidth at different edge servers are neither equal nor constant in practice and are represented by BWP_{ES}. So, content distribution cost is quantified by Eq. (6) and denoted by $DistCost_i$.

$$DistCost_i = \sum_{ES \in Replicas_i} DistCost_{ES,i} = \sum_{ES \in Replicas_i} (BW_{ES,i} \cdot BWPrice_{ES}), \qquad (6)$$

Replication cost consists of two parts: replica transmission cost denoted by $TransCost_{ES,i}$ and buffering (storage cost) symbolized by $StrCost_{ES,i}$. Therefore, the replication cost for content i can be quantified by:

$$RepCost_i = \sum_{ES \in Replicas_i} RepCost_{ES,i}$$
$$= \sum_{ES \in Rep_{CS,i}} RepCost_{ES,CS,i} + \sum_{ES \in Rep_{P2P,i}} RepCost_{ES,P2P,i}$$
$$= \sum_{ES \in Rep_{CS,i}} (TransCost_{ES,CS,i} + StrCost_{ES,CS,i})$$
$$+ \sum_{ES \in Rep_{P2P,i}} (TransCost_{ES,P2P,i} + StrCost_{ES,P2P,i}). \qquad (7)$$

4. Optimal replication of P2P content over HCDN

We aim to add an RP mechanism to a traditional CDN and enable it to perform as an HCDN. Therefore, it is assumed that the set $Rep_{CS,i}$ is determined by the existing RP mechanism of the CDN and it is considered as an input variable of our problem. We focus on optimizing economics of the system considering $Rep_{P2P,i}$ as optimization variable. Consequently, we have:

$$RepCost_{ES,P2P,i} = TransCost_{ES,P2P,i} + StrCost_{ES,P2P,i}$$
$$= BW_{P2P,i} \cdot PBW_{Source,ES} + Size_{P2P,i} \cdot PSTR_{ES}, \qquad (8)$$

where $BW_{P2P,i}$ and $Size_{P2P,i}$ denote the bandwidth of content i in P2P mode and its required storage space, respectively. $PBW_{ES,Source}$

represents the price of bandwidth for transferring the stream of content for distribution in P2P mode from the source (cooperating edge server containing the replica of the content i in P2P mode) to ES, and $PSTR_{ES}$ symbolizes the price of storage at ES. Therefore, the objective function, net-profit of the HCDN, based on Eqs. (1)–(3), (5)–(8) can be quantified as:

$$NetProfit_{SLAi} = R_{SLAi} - Cost_{SLAi}. \qquad (9)$$

In order to maximize $NetProfit_{SLAi}$, the RP mechanism must determine $Rep_{P2P,i}$. All the other parameters of the problem including the set of replicas for CS content, the population of requests, bootstrapping threshold, storage costs and communication costs are not controllable in this phase; therefore, the decision problem to put replicas of P2P content is defined as follows:

$$Rep_{P2P,i} = \underset{rep_{P2P} \in P(Rep_{CS,i})}{argmax} (NetProfit_{SLAi}(rep_{P2P})), \qquad (10)$$

where $P(Rep_{CS,i})$ denotes the power set of $Rep_{CS,i}$. Since $|P(Rep_{CS,i})| = 2^{(|RepCS,i|)}$, finding the optimum replication set for the stream in P2P mode for small replica sets is trivial. However, considering the fact that popular CDNs, e.g. Akamai, benefit from thousands of edge servers [1], it can be concluded that it is not possible to rely on an exhaustive search solution to find the optimum set of replicas for the content in Hybrid mode.

5. The proposed mechanism (HRP)

In order to design the RP mechanism of the HCDN (HRP), the Economic Mechanism Design Theory is employed. In this section, economic model of the problem which is called Hybrid Replica Placement Model (HRPM) is formalized mathematically. Subsequently, HRP is introduced formally using the popular notations of the Economic Mechanism Design Theory [5]. The major advantages of HRPM over popular cost models used in the literature of content distribution are twofold. First, it computes the complex net profit function of hybrid content delivery system instead of raw cost function of replica placement considered in traditional CDNs [1]. Second, it embeds all the constraints and goals of the problem in a single dimensional objective function suitable for efficient mechanism design [5].

5.1. Mathematical model (HRPM)

As seen in Eq. (3), the underlying network for the replication of $Rep_{P2P,i}$ consists of an overlay network composed of edge servers that contain replicas of the content in CS mode. The first responsibility of this overlay network is to disseminate the base quality content from origin servers to replica servers for CS service, $Rep_{CS,i}$, through multicasting. In order to keep the problem tractable, we make a technical assumption about the overlay network.

We assume that the overlay multicast network employed for distribution of replicas for P2P content distribution is equal to the multicasting infrastructure composed of $Rep_{CS,i}$. This technical assumption confines the source of transferring the P2P content for cooperative replication to ancestor servers in this multicasting infrastructure. Relying on this hierarchical infrastructure, it is possible to quantify the net profit of HCDN for distribution of content i in a sub-tree rooted at ES recursively as:

$$NetProfit(ES,h,i) = Max \begin{cases} \sum_{child \in ES\_child} NetProfit(child, h+1, i) \text{ (case : a No replica at } ES) \\ + R_{PureCDN,ES,i} - RepCost_{CS,ES,i} - DistCost_{PureCDN,ES,i}, \\ \sum_{child \in ES\_child} NetProfit(child, 1, i) \text{ (case : b 1 replica at } ES) \\ + R_{HCDN,ES,i} - RepCost_{CS,ES,i} - RepCost_{P2P,ES} - DistCost_{HCDN,ES,i} \end{cases}, \quad (11)$$

where $NetProfit(ES,h,i)$ denotes the maximum net profit of content distribution for the sub-tree rooted at ES when the first replica of the high quality content is located at h steps higher than ES. In this equation, when it is not economic to have a replica of high quality content at ES, (case a in Eq. (11) and Fig. 2), all the child sub-trees of ES denoted by ES_child have to fetch the content from $h + 1$ steps upper. In this case, all the requests redirected to ES are served in the CS mode. Thus, the revenue of ES is denoted with $R_{PureCDN,ES,i}$ and is therefore calculated like that of a pure CDN server. In this case, the replication cost is paid only for the CS content. Consequently, the distribution cost is calculated multiplying bandwidth of content i in CS mode by price of the bandwidth at ES, and total number of requests redirected to ES, ($DistCost_{PureCDN,ES,i} = BW_{CS,i} \cdot PBW_{ES} \cdot (Req_{ES,CS,i} + Req_{ES,P2P,i})$).

When it is economic and possible to have a replica at ES, (case b in Eq. (11) and Fig. 2), all the child sub-trees receive the high quality P2P content replicated at ES from one step upper; however, in addition to the replication cost for base quality content, ES has to pay for the replication of high quality instance of content i in P2P mode ($RepCost_{P2P,ES,i}$). In this case, ES acts as a hybrid content distribution server. Therefore, the revenue of edge server, ES, is calculated as an HCDN server using Eq. (1). Based on Eq. (11) and knowing that the maximum depth of the multicasting trees in IP networks is limited, it is possible to find the maximum net profit for every multicasting infrastructure, finding the maximum net profit for the root server ($NetProfit(Root,0,i)$), recursively.

5.2. Mechanism design

In order to employ Eq. (11) efficiently and design a resourceful replication mechanism, Economic Mechanism Design Theory is employed. In this approach, notations illustrated in Fig. 3 are used to introduce problems and mechanisms formally [5].

Relying on this theory the problem is defined as follows:
Given:

- a class Θ of environments (here all the replica placement problems in this domain with parameters introduced in Table 1),
- an outcome space Z (single dimension profit function of the HCDN based on Eq. (1)),
- and a goal function F (Eq. (10)).

Requested:
An economic mechanism $\pi = (M, \mu, h)$ that realizes the problem (in this case maximizes the net profit of HCDN)

- where M represents the message space used to solve the problem and yield equilibrium,
- μ symbolizes the function used for generating the group equilibrium messages,
- and h denotes the function which determines the outcome.

The problem is solved and the best mechanism is chosen if and only if:

$$\forall \theta \in \Theta \quad F(\theta) = h(\mu(\theta)). \quad (12)$$

In this problem, the outcome function, h, equals the total net profit of the HCDN, $NetProfit(Root,0,i)$. Based on Eq. (11), every edge server needs to receive the tabular function $NetProfit$ from its direct child sub-trees. Therefore, M includes messages containing the profit function and μ consists of messages carrying the values of $NetProfit$ function between each node and its parent. Agents employed to implement the mechanism include all the members of $Rep_{CS,i}$. In order to find the exact value of net profit for the HCDN and evaluate Eq. (11) for all the edge servers, it is needed to traverse the multicasting tree in a Reverse Beneath First Search (RBFS) order and calculate the $NetProfit$ function recursively. After the maximum of $NetProfit(Root,0,i)$ is found, in order to determine the replication set for P2P content, it is needed to traverse the multicasting tree with Beneath First Search (BFS) order. Moreover, the selected value of $NetProfit$ function at each node determines which nodes are to replicate the P2P content. Fig. 4 illustrates the flow of messages among the agents in this mechanism.

For the first phase (RBFS), since the parent node by itself requests the successors to calculate and transfer their $NetProfit$ function, it is not necessary for the messages to contain the address of the edge server, ES, and the ID of the content, i. Therefore, the profit function in direct implementation of the solution can be expressed by at least a single dimension data matrix with a maximum length of H (*maximum height of the tree*). For the second phase (BFS), the parent node needs to declare the selected value of $NetProfit$ vector calculated previously by the child node which maximizes its net profit. Therefore, in this phase, messages contain just one number. Since in Eq. (11) each edge server just relies on its own information and the value of net profit function calculated by its direct child sub trees, it can be concluded that this solution is a direct mechanism.

5.3. Implementation

HRP is implemented by using a custom written simulation program. The simulation program is written in C++ programming language, it is compiled with Borland C++ version 5.02, and it is run on a Microsoft Windows 7 platform. The physical machine used to implement the mechanism had four 2.4 GHz processing cores and 8 GB of RAM.

In order to simulate the HCDN Replica Placement mechanism (HRP), a utility maximizing agent is assumed to be present at every node of the multicasting network ($Rep_{CS,i}$). Pseudo code 1 introduces the simplified implementation of HRP mechanism in C++ object oriented environment. Agents (ES objects) are organized in a bidirectional tree and calculate the tabular function $NetProfit(h,i)$, cooperatively. The mechanism consists of two recursive traversals of the multicasting tree implemented using $Find\_Max\_Net\_Profit()$ and $Find\_Rep\_Set()$ functions.

In Pseudo code 1, the $main()$ function initializes the simulation program by fetching the multicasting infrastructure and all the parameters from source files to the memory. Next, it calls the $Find\_Max\_Net\_Profit()$ recursive function at the root of the multicasting tree to find the overall maximum net profit. After finding the maximum net profit, the main function passes it to the $Find\_Rep\_Set()$ function at the root node. Calling this recursive function causes all the replicas to be placed on their appropriate edge servers belonging to the multicasting tree. Finally, results are written back to report files for analysis.

The $Find\_Max\_Net\_Profit(h,i)$ function calculates the maximum net profit of the sub-tree rooted at ES when the first replica of the content i is assumed to be at h steps higher. In order to avoid

complexities of RBFS, this function employs a pre-order traversal of the sub-tree. Therefore, first, through a *for* loop, *Find_Max_Net_Profit*(*h* + 1,*i*) is called for all the child sub-trees of the current node. After returning from recursive function calls, *Find_Max_Net_Profit*(…) considers two cases of having *0* or *1* replicas at this node. Then, the maximum of these two cases is stored and returned as the maximum expected net profit of the sub-tree rooted at *ES*.

Pseudo code 1. Simplified codes of implementation of HRP in C++

```
class ES;
 {
 private:
 set ES * Child;  //Set of all child nodes
 ES * Parent;    //Parent node of this object
...
 public:
 ststic float NPRep[H],NPnoRep[H],NPMax[H];   // 3 arrays for storing values of Net profit with 0 or 1 replicas and maximum value of net profit at this sub tree
...                                          //Maximum lengths of these arrays are equal with th maximum height of the multicasting tree denoted by H

 float Find_MAX_Net_Profit(int h, int i)                        // The recursive member function for calculation of maximum net profit in each sub tree.
  {
    for every child_node ∈ Child                               //Calculate the net profit assuming that there is no replica at this node.
      {
        NPnoRep+= child_node->Find_Net_Profit(h+1,i);          // Net profit of all child sub-trees when there is no replica at this node.
      }
    NPnoRep[h]+=(Req(CS,i)+ Req(P2P,i))*BW(CS,i)*P(CS,i);      // Revenue of this node in pure CS mode.
    NPnoRep[h]-=Rep_Cost(CS,i);                                // Replication cost for CS content.
    NPnoRep[h]-=(Req(CS,i)+ Req(P2P,i))*BW(CS,i)*BWPrice;      // Content distribution cost in CS mode.

    for every child_node ∈ Child                               // Calculation of the net profit assuming that there exists a replica at this node.
     {
        NPRep[h]+= child_node->Find_Net_Profit(1,i);           //Net profit of all child sub-trees when there is a replica at this node.
     }
    NPRep[h]+=(Req(CS,i))*BW(CS,i)*P(CS,i) + Req(P2P,i)* BW(P2P,i)*P(P2P,i);   // Revenue of this node in hybrid mode.
    NPRep[h]-= ( Rep_Cost(CS,i) + Rep_Cost(P2P,i) );           // Replication cost for both CS and P2P content.
    NPRep[h]-=( Req(CS,i) + Req(P2P,i)*α )*BW(CS,i)*BWPrice(node);   // Content distribution cost in hybrid mode.

   NPMax[h]= max(NPnoRep[h],NPRep[h]);                         // Find and store the maximum possible value for net profit.
   return(NPMax[h]);
  }

 void Find_Rep_Set(int h,int i, float Max_net_profit);  // After traversal of the multicasting tree, this function decides on places of replicas
  {                                                       // based on the final value of maximum net profit, recursively.
    if (Max_net_profit ==NPRep[h])                        // If the maximum net profit equals with net profit when there is one replica at this node...
     {
       RepP2P +=this;                                     // Insert this node in set of replicas of P2P content.
       for every child_node ∈ Child                       // For all the child sun trees, consider the case that they receive their replica from this node
         {
           child_node->Find_Rep_Set(1,i, child_node-> NPMAX[1])
         }
     }
    else
     {
      for every child_node ∈ Child
       {
         child_node->Find_Rep_Set(h+1,i, child_node-> NPMAX[h+1])
       }
     }
  }

static set RepP2P ={};
ES *root
main()
 {
  initialize();                                           // ... read the source file, construct the multicasting tree and set the values...
...
  float Max_net_profit=root->Find_MAX_Net_Profit(0,i);    // find the maximum net profit for the HCDN
  root->Find_Rep_Set(0,i,Max_net_profit);                 // find the locations of replicas for the maximum value of net profit
  report(Replication_set);

 }
```
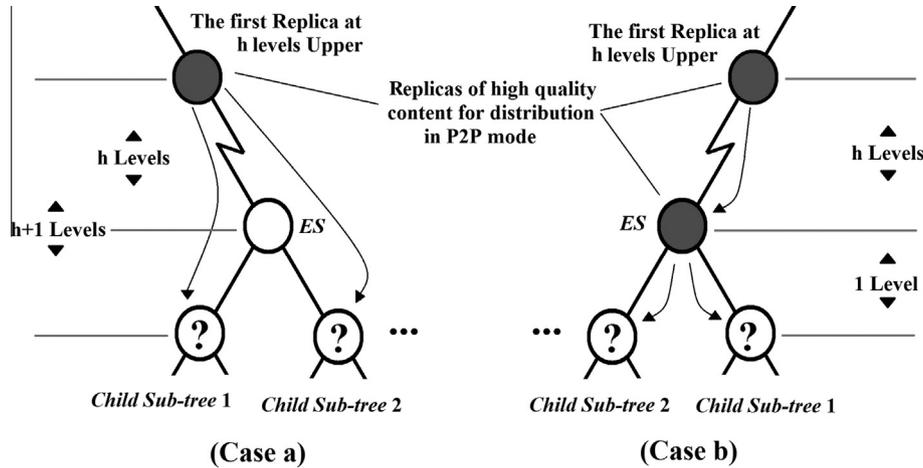
Fig. 2. Two possible cases in calculation of net profit and distribution of replicas.
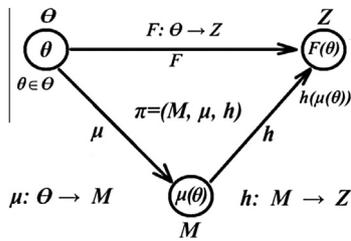
Fig. 3. Notations used in mechanism design.
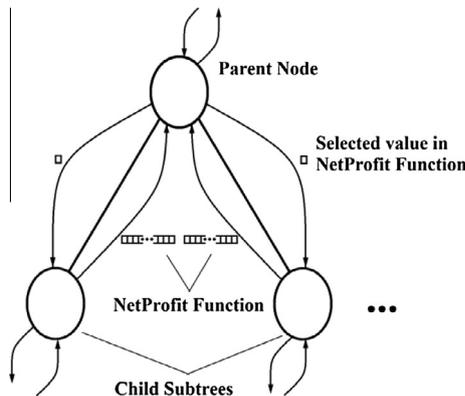
Fig. 4. Flow of messages in the mechanism.

The *Find_Rep_Set*(*h,i,Max_net_profit*) calculates appropriate replication set when the first replica of the content *i* exists at *h* levels upper and it is expected to gain *Max_net_profit* from the hybrid content delivery at this sub-tree. For the first time, the *main*() calls this function. As a result, all the other functions on nodes of multicasting tree are called recursively. As it can be seen in Pseudo code 1, in order to avoid complexities of BFS, *Find_Rep_Set*(*h,i,Max_net_profit*) relies on a post-order traversal. It starts with deciding whether to place a replica or not based on the expected net profit. Based on this decision, the set of replicas is updated. Then, through the recursive function calls this process continues until leaves of the multicasting tree.

5.4. Resource consumption

Based on Pseudo code 1, the maximum depth of recursion to calculate the maximum net-profit and the best replication set is

H (height of the multicasting tree, approximated by *Log*(*n*), where $n = |Rep_{CS,i}|$). Therefore, at each node, *ES*, the recursive function demands $O((Log(n))^2)$ of processing resources. This process also requires $O(Log(n))$ of memory and $O(Log(n))$ of communication at each node to store and transmit the net profit matrix. Where, *n* represents number of nodes in multicasting tree. Since all the processing, memory and communication demands of the HRP on every node of the HCDN scales below linear, ($O(n)$), it can be concluded that, the distributed implementation of the HRP, is scalable. Therefore, it is capable of being used in large scale HCDNs composed of thousands of servers and serving millions of end-users.

In centralized implementation, aggregated complexity of HRP is of $O(n(Log(n)^2)$, $O(n(Log(n))$ and $O(n(Log(n))$ for processing, memory and communication resources, respectively. Centralized implementation of HRP scales sub-quadratic. Therefore even in simulation environment where all the calculations are performed in a single machine, results could be achieved within a reasonably short time.

6. Experimental evaluation

HRP is evaluated in two ways. Relying on a simulation test-bed, performance of HRP is studied. Additionally, to perform a comparative study on the performance and economics of HRP, five other replica placement strategies are implemented. These replication strategies include Greedy Global (GG), Single Greedy (SG), Hot-Spot (HS), Random (RND) and pure CDN. GG algorithm is conceived to yield near optimal outcome. SG solution differs from GG only in that SG just calculates the sorted list of replication candidates once. In other words, it does not consider the effect of previous replicas on the place of next replicas. HS approach devotes replicas to servers with higher number of requests in P2P mode. Outcomes of RND approach demonstrate the importance of replication strategy. In addition to these strategies, performance and economics of a pure CDN environment is also simulated and reported.

6.1. Technical assumptions

Although the HRP mechanism is capable of solving the problem in any environment, maximum effort has been made by authors to set the parameters of experiments as real as possible. Some parameters of the environment (e.g. price of bandwidth at edge servers, price of storage and trace of requests) are set using real log files of a large-scale traditional CDN operating in Australia during the month of May 2013. Trace of requests are extracted from log files

of request routing engine of the CDN, scaled and applied to the HCDN. Price of bandwidth for CS service is extracted from the SLA of the CDN and in order to keep the SLA incentive compatible, the price of bandwidth for P2P service is assumed as 0.8 of CS service bandwidth price. A typical 1000 s video with bandwidth of 512 Kbps is considered as the CS service. Bandwidth of P2P content is assumed as double of CS service (1 Mbps). From our previous study on request routing and resource allocation in HCDNs [21], α (effectiveness ratio of P2P content distribution) is assumed to be 0.165. Also based on this study it is assumed that 16% of end-users request to receive the content only in CS mode.

In order to keep the experiments comparable, churn rate of end-users are not considered. In other words, the users enter the system in an accumulative manner to enable us to study the behavior of HRP, increasing the load without worrying about the churning rate. Since we are about to study the performance of the replica placement of HCDN in P2P mode, it is assumed that all the 1000 edge servers and 1 origin server contain replicas of the content in CS mode statically. This assumption excludes effects of replica placement of CS content from our study. In order to obtain a realistic medium size multicasting overlay network, as the replica distribution network, we have generated an internet-like weighted graph with 1001 nodes using IGEN [22]. We have generated the multicasting tree using our multicasting mechanism [23,24] on the top of this graph. Table 2 summarizes key parameters of the experiment.

Except HRP, other replication strategies need to have the number of replicas determined prior to their decision making phase. In addition, in order to keep the comparisons fare, it is necessary to compare results with the same number of replicas. Therefore, at the first experiment applying the load of requests, we have determined the optimum number of replicas using HRP for every number of concurrent requests and in the second experiment, each of replica placement strategies are fed with this number of replicas.

6.2. Performance evaluation of HRP

In this experiment, HRP is applied as the replication strategy in a typical HCDN business for distribution of a streaming video content. With assumptions described previously, the load on the HCDN is increased until 1 million requests. Fig. 5 illustrates number of replicas for HRP, increasing the load of requests on HCDN. When total number of requests is small, number of replicas for P2P content is zero. At this point for some edge servers, number of content requests in P2P mode is below the bootstrapping threshold of P2P content delivery. At some other servers with enough number of requests, high replica distribution cost prevents the edge server to replicate the content and start hybrid content delivery. When number of requests increases, some edge servers with low replication costs which receive high number of requests for P2P content, find it economic to replicate the P2P content. After this point, further increase in the load of system results in a sharp increase in the number of replicas. On one hand, more and more edge servers find the number of requests that they receive for P2P content higher than T_B. On the other hand, in the multicasting tree, replication of the P2P content on even a handful number of edge servers decreases the replica transmission cost ($TransCost_{P2P,ES,i}$) for other servers dramatically. After this sharp increase in the number of replicas, just servers with very high price of storage remain outside of $Rep_{P2P,i}$. With future increase in the number of concurrent end-users, these servers smoothly join to the replication set $Rep_{P2P,i}$. As it can be obviously seen, when number of requests is increased significantly (1 million) $Rep_{P2P,i}$ includes majority of edge servers (90.5%).

Fig. 6 represents number of requests for CS and P2P content submitted to the HCDN (denoted by Req_{CS} and Req_{P2P}) and number of CS and P2P end-users receiving service from HCDN (denoted by N_{CS} and N_{P2P}) when HRP approach is applied for different total number of requests. As it can be obviously seen, same as Fig. 5 when number of replicas increases, number of end-users served in P2P mode increases and gets closer to the number of requests for the content in P2P mode.

As illustrated in Fig. 7, when the number of requests increases, total bandwidth contribution of end-users (symbolized by *TBCU*) increases. As a result, the bandwidth consumption of the HCDN (represented with *TBSE*), is increased very slowly. Total bandwidth delivered to end-users is summation of bandwidth investment of HCDN and contribution of end-users. Based on Fig. 7, when number of requests is small, total bandwidth delivered to end-users (*TBDE*) is equal to the bandwidth consumption of HCDN. When the number of concurrent requests at some edge servers increases enough and edge servers begin to replicate the P2P content, *TBDE* experiences a sharp increase. Further increase of load boosts *TBDE* near linearly.

Figs. 8–10 represent economics of an HCDN employing HRP with a pure CDN under the same load conditions. Fig. 8 shows that although, in the hybrid mode aggregated bandwidth delivered to the end-users is much higher than a pure CDN, costs spent by HCDN does not grow fast. When the load of HCDN becomes significant, its cost function crosses the cost function of the pure CDN. This observation confirms economic effectiveness of the HRP and hybrid architecture. Fig. 9 illustrates revenue of the HCDN in hybrid and pure CDN content delivery modes. Revenue of the HCDN is equal to the cost function of content provider. It is observed that in the hybrid mode, when the number of requests is significant and majority of end-users receive the content with double quality, content provider pays much less than twice of the content delivery cost in pure CDN mode. For instance when 1 million concurrent requests is submitted to the system, based on Fig. 6, nearly 80% of end-users receive double quality content and HCDN delivers approximately 80% more traffic. However, based on Fig. 9, content provider pays only 57% more. Fig. 10 compares Net Profit of HCDN (*NP_HCDN*) and pure CDN (*NP_CDN*) strategies. As it can be obviously seen, the HCDN architecture employing HRP strategy is able to increase the net profit significantly.

6.3. Economics of HRP vs. other strategies

Figs. 11–14 compare economics of HCDN employing various replica placement strategies. These graphs illustrate revenue, replication cost, content distribution cost and net profit of the HCDN with different replica placement approaches.

Fig. 11 compares replication cost for different strategies. Among the replication strategies HS and RND have most replication cost because these two strategies do not consider replication cost. The pure CDN strategy does not invest on replication of P2P content. Therefore, its replication cost remains constant and equals to the replication cost for CS content. Other strategies, (HRP, GG and SG) have comparable replication costs. Among these strategies, HRP has the least replication cost. The replication cost for this strategy has high similarity with the curve illustrated in Fig. 5, for optimum number of replicas.

Fig. 12 illustrates the content distribution cost for the studied strategies. The pure CDN approach serves all the requests in CS mode. Therefore, it has the maximum content distribution cost. The RND approach puts replicas by chance, therefore it has the second worst content distribution cost. HRP, HS, SG and GG approaches have approximately the same content distribution cost. Since the HS approach does not pay attention to the replication costs, it succeeds to serve the maximum number of requests

Table 2
Summary of parameters used in experiments.

Parameter	Value	Comments
$BW_{CS,i}$	0.5 Mbps	Bit-rate of an average quality video
$BW_{P2P,i}$	1 Mbps	Bit-rate of a high quality video
T	1000 s	Length of video stream
$P_{CS,i}$	2.5 \$/Mbps	Monthly price of traditional CDN service in the SLA
$P_{P2P,i}$	$0.8 * P_{CS,i}$	To keep the SLA incentive compatible, the price of CS service is assumed to be more than P2P service
$BWPrice_{ES}$	[8,12] \$/Gbps	Monthly price of network bandwidth at different edge servers
$PSTR$	[8,12] \$/GB	Monthly price of storage space at different edge servers
A	0.165	Effectiveness factor of P2P content distribution
B	0.16	Ratio of CS requests to total number of requests $R_{CS}/(R_{CS} + R_{P2P})$
H	9	Maximum depth of multicasting tree
$|Rep_{CS,i}|=n$	1000	Total number of edge servers containing replicas of CS content
T_B	[65, 165]	Bootstrapping threshold for P2P content delivery
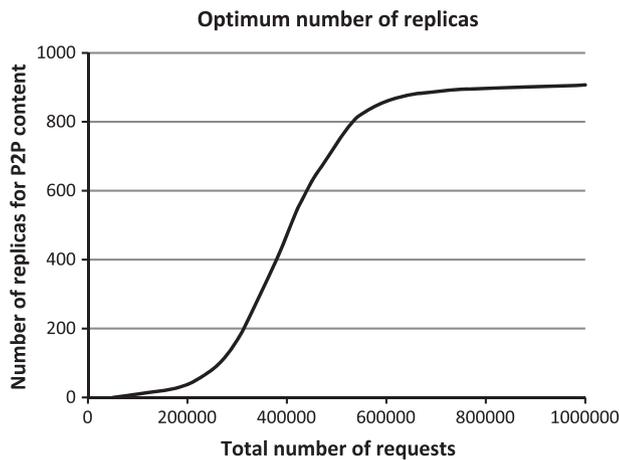
Fig. 5. Number of replicas in HRP increasing number of requests.
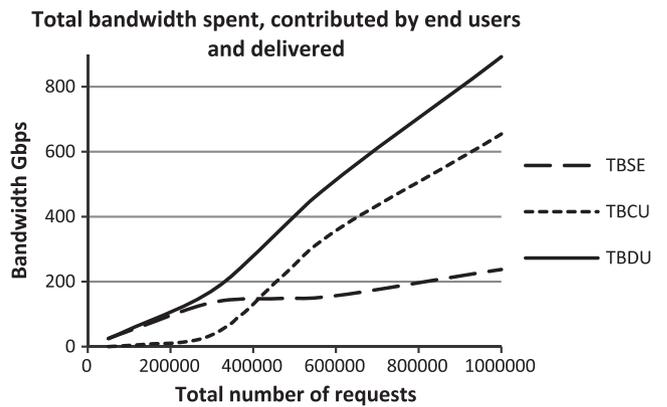
Fig. 7. Performance metrics of HCDN with HRP, total bandwidth spent by edge servers (TBSE), total bandwidth contribution of users (TBCU) and total bandwidth delivered to users.
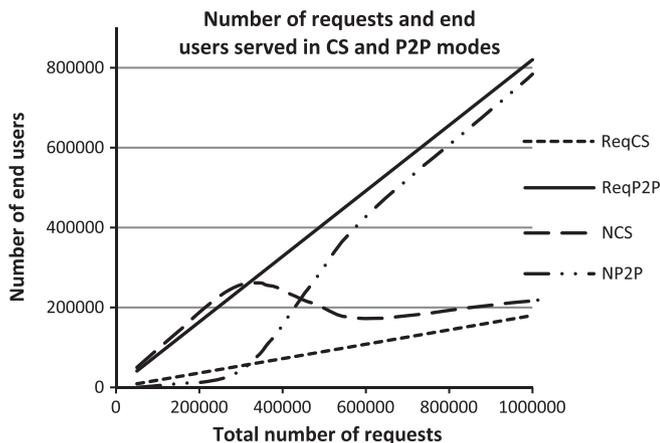
Fig. 6. Number of requests and end users served in CS and P2P modes with HRP approach, increasing number of requests.
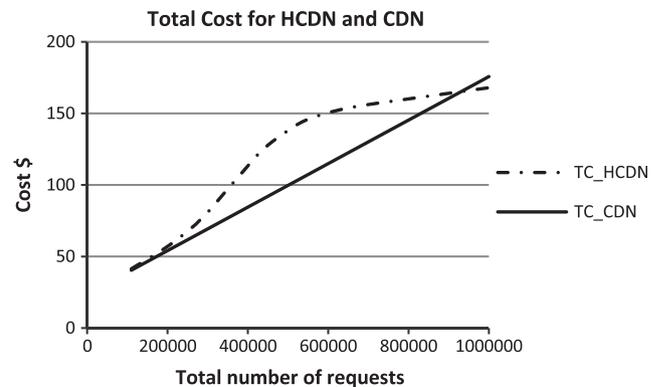
Fig. 8. Total cost for HCDN with HRP replication strategy (TC_HCDN) and total cost in pure CDN mode (TC_CDN).

in P2P mode. Therefore, this approach yields the least content distribution cost.

As it is illustrated in Fig. 13, the revenue with application of the RND approach is increased linearly. HS approach produces the most revenue and the HRP approach is pretty close to that of HS approach. Revenue of RND approach is far from other strategies.

In Fig. 14, the net profit of the HCDN with application of various replication strategies is reported. As it is expected, the pure CDN approach has a linearly increasing net profit. Compared to other strategies, RND approach has less net profit. Moreover, for a

considerably wide range of total number of requests it has the least outcome. HRP yields the maximum net profit. Comparison of the net profit generated by HS approach with the net profit generated with greedy approaches (GG and SG) shows that in this system the outcome of greedy algorithms are not as close as expected to the optimal outcome (HRP). In other words, a simple heuristic like hot spot (HS) can produce better results.

7. Practical implications

The performance evaluation of the HCDN architecture with HRP replica placement strategy reveals some features of the system. Experiments verify that in this architecture when a light load of
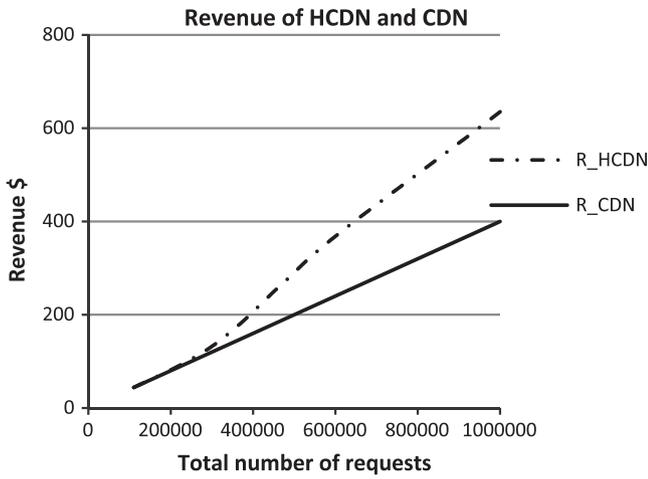
Fig. 9. Revenue of HCDN (R_CDN) and revenue of the system in pure CDN mode (RCDN).

Fig. 10. Net profit of the HCDN (NP_HCDN) with HRP replication strategy and net profit of the system in pure CDN (NP_CDN) operation mode.

Fig. 11. Replication cost for different strategies increasing the number of requests.

Fig. 12. Content distribution cost for different strategies increasing the number of requests.
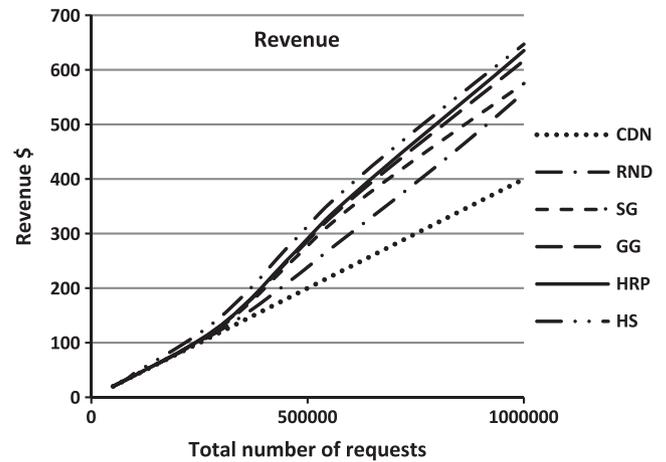
Fig. 13. Revenue of HCDN for different strategies increasing the number of requests.
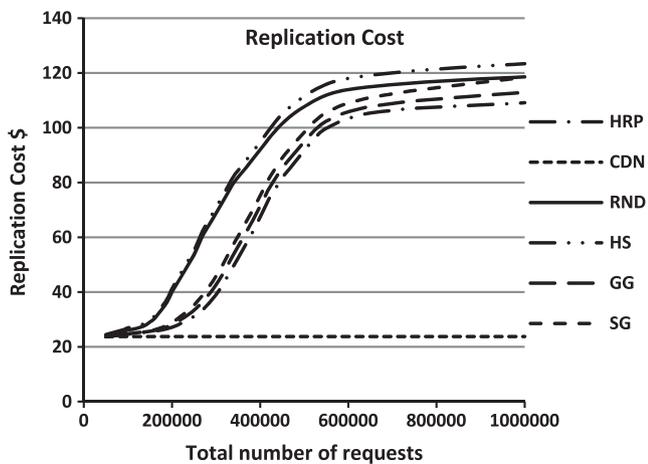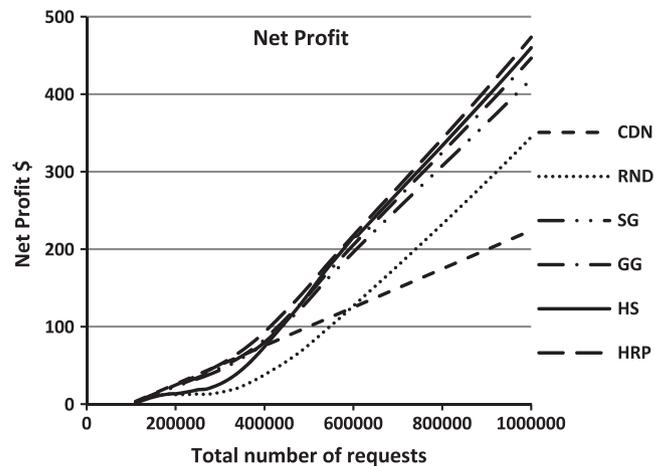
Fig. 14. Net profit of HCDN for different strategies increasing the number of requests.

requests is submitted to the system, performance and economics of the content distribution system is similar to a traditional CDN. When the load of the system increases significantly, advantages of hybrid content delivery becomes visible. Edge servers begin to replicate P2P content through HRP strategically. As a result of the contribution of end-users in P2P content delivery and investment

of HCDN for replication of P2P content, net profit of the HCDN provider improves significantly. At the same time, the content provider benefits from the hybrid content delivery by spending less for delivered traffic. In this case a big majority of the end-users enjoy high QoS. Therefore, HRP is able to guarantee incentive com-

patibility among the HCDN provider, content provider and end-users.

The comparison of the cost function in hybrid and pure CDN content delivery modes, illustrated in Fig. 8, shows that the HCDN architecture with HRP strategy has succeeded to exploit resources in a more scalable manner. In other words, considering the same level of available communication resources, HCDN architecture is able to tolerate flash-crowds better. Therefore, the HCDN provider can expect less SLA violations compared to the traditional CDNs.

Comparative study of HRP vs. other strategies shows that since HRP gives the best place and number of replicas on the given Hybrid CDN-P2P network, it is able to yield the maximum net profit. Unlike HRP, greedy algorithm makes decisions for putting k replicas step by step. This mechanism has near optimal results but as it puts replicas one by one, its replication set may differ from the optimum. Results obtained from SG approach represent that in the replica placement process it is necessary to consider the impact of existing replicas on decisions to place next replicas. Despite the simplicity of the HS approach, it has a poor outcome due to ignoring the replication cost, especially when the number of requests is small. When the number of requests is significantly high, the HS approach outperforms greedy approaches. Thus, it can be concluded that in some circumstances, the performance of the greedy approach is worse than a simple heuristic like HS.

8. Conclusion and future works

In this paper, we have presented an economic replica placement mechanism designed for streaming content distribution in HCDN environment. The proposed mechanism makes it possible to offer Hybrid CDN-P2P streaming content delivery service using a traditional CDN. In order to design the mechanism, economic model of the system (HRPM) has been introduced and a recursive hierarchical push-based cooperative replica placement mechanism is proposed. The mechanism has been formalized based on the Economic Mechanism Design Theory. We studied performance of the solution through simulation and compared its outcome with popular replication strategies.

Experiments are conducted on the traces obtained from a large-scale CDN working in Australia. These studies show that HRP is able to produce applicable results with all the positive features of Hybrid CDN-P2P architecture. Experiments verify that through HRP replication strategy and HCDN architecture it is possible to increase the net profit of a CDN provider, decrease prices of service for content provider and serve majority of end-users with high quality stream at the same time. Through the experiments HRP is compared with some popular replication strategies and it is shown that HRP replication strategy has a great positive impact on the economics of content distribution in HCDNs.

In order to reduce the complexities of implementation, we considered a fixed DNS-based request routing and a fixed contribution level of P2P end-users. Therefore, integration of HRP mechanism with dynamic and differentiated request routing and resource allocation mechanisms of HCDN remains as a future work. Cloud computing infrastructures are increasingly attracting the attention of research community to mediate the high deploying costs of CDNs. Another extension to this work should consider the integration of cloud based CDNs with P2P networks.

Acknowledgement

References

[1] Mukaddim Pathan, Buyya Rajkumar, Athena Vakali, Content delivery networks: state of the art, insights, and imperatives, Content Delivery Networks, Springer, Berlin, Heidelberg, 2008. pp. 3–32.

[2] Imsook Ha, Steven S. Wildman, Johannes M. Bauer, P2P, CDNs, and hybrid networks: the economics of Internet video distribution, Int. Telecommun. Policy Rev. 17 (4) (2010) 1–22.

[3] Heather Yu, John Buford, Mursalin Akon, Handbook of Peer-to-Peer Networking, vol. 1, Springer, Heidelberg, 2010.

[4] Jiang Hai, Jun Li, Zhongcheng Li, Xiangyu Bai, Performance evaluation of content distribution in hybrid CDN-P2P network, in: IEEE Future Generation Communication and Networking, FGCN08, 2008, pp. 188–193.

[5] Leonid Hurwicz, Stanley Reiter, Designing Economic Mechanisms, Cambridge University Press, 2006.

[6] Wang Zhan, et al., A k-coordinated decentralized replica placement algorithm for the ring-based CDN-P2P architecture, in: 2010 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2010.

[7] Fatemeh Keymasi Khalaji, Morteza Analoui, Replica placement algorithms in hybrid CDN-P2P architectures, in: 2012 Sixth International Symposium on Telecommunications (IST), IEEE, 2012.

[8] F. Bronzino, et al., An adaptive hybrid CDN/P2P solution for content delivery networks, in: 2012 IEEE Visual Communications and Image Processing (VCIP), IEEE, 2012.

[9] Hao, Xuening Liu, Tongyu Zhan, Vyas Sekar, Feng Qiu, Chuang Lin, Hui Zhang, Bo Li Yin, Livesky: enhancing cdn with p2p, in: ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), vol. 6, no. 3, 2010, pp. 1–16.

[10] Cheng Huang, Angela Wang, Jin Li, Keith W. Ross, Understanding hybrid CDN-P2P: why limelight needs its own red swoosh, in: ACM 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video, 2008, pp. 75–80.

[11] ZhiHui Lu, Ye Wang, Yang Richard Yang, An analysis and comparison of CDN-P2P-hybrid content delivery system and model, J. Commun. 7 (3) (2012) 232–245.

[12] ZhiHui Lu, XiaoHong Gao, SiJia Huang, Yi Huang, Scalable and reliable live streaming service through coordinating CDN and P2P, in: 2011 IEEE 17th International Conference on In Parallel and Distributed Systems (ICPADS), IEEE, 2011, pp. 581–588.

[13] Bogdan Florescu, Mugurel Ionut Andreica, Towards a Peer-assisted content delivery architecture, in: Proceedings of the 18th International Conference on Control Systems and Computer Science (CSCS) (ISSN: 2066–4451), vol. 2, 2011.

[14] Hu Chao, Ming Chen, Changyou Xing, Bo Xu, EUE principle of resource scheduling for live streaming systems underlying CDN-P2P hybrid architecture, Peer-to-Peer Network. Appl. 5 (4) (2012) 312–322.

[15] Hao Yin, et al., Design and deployment of a hybrid CDN-P2P system for live video streaming: experiences with LiveSky, in: 17th ACM International Conference on Multimedia, 2009, pp. 25–34.

[16] George Pallis, Athena Vakali, Insight and perspectives for content delivery networks, Commun. ACM 49 (1) (2006) 101–106.

[17] J. Freedman Michael, Eric Freudenthal, David Mazires, Democratizing content publication with coral, in: Proc. 1st USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI'04), 2004.

[18] Mohammad Shorfuzzaman, Peter Graham, Rasit Eskicioglu, QoS-aware distributed replica placement in hierarchical data grids, in: IEEE International Conference on Advanced Information Networking and Applications (AINA), IEEE, 2011.

[19] Rajasekhar Sathish, Bin Rong, Kwong Lai Yuen, Ibrahim Khalil, Zahir Tari, Load sharing in peer-to-peer networks using dynamic replication, in: 20th International Conference on Advanced Information Networking and Applications, 2006, pp. 1011–1016.

[20] Jussi Kangasharju, James Roberts, Keith W. Ross, Object replication strategies in content distribution networks, Comput. Commun. 25 (4) (2002) 376–383.

[21] Mehran Garmehi, Morteza Analuoi, Mukkadim Pathan, Rajkumar Buyya, An economic mechanism for request routing and resource allocation in Hybrid CDN-P2P, Technical report, Computer Networks Research Lab (CNRLAB), Iran University of Science and Technology (IUST), Tehran, Iran (August 2013).

[22] Quoitin Bruno, Virginie Van den Schrieck, Pierre François, Olivie Bonaventure, IGen: generation of router-level internet topologies through network design heuristics, in: (ITC21) 21st International In Teletraffic Congress, 2009, pp. 1–8.

[23] Mehran Garmehi, Morteza Analoui, An economical mechanism for multicasting of content among servers of Hybrid CDN-P2P networks, in: 2011 International Conference for In Internet Technology and Secured Transactions (ICITST), IEEE, 2011, pp. 566–571.

[24] Mehran Garmehi, Morteza Analoui, A distributed mechanism for economic management of transmission infrastructure in hybrid CDN-P2P networks, J. Econ. Comput. Econ. Cybernet. Stud. Res. (ECECSR) No. 3/2014 to be appeared.