Research Journal of Applied Sciences, Engineering and Technology 7(19): 4018-4028, 2014
DOI:10.19026/rjaset.7.763
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: November 07, 2013 Accepted: November 18, 2013 Published: May 15, 2014

Research Article

An Improved Particle Swarm Optimization Based on Deluge Approach for Enhanced Hierarchical Cache Optimization in IPTV Networks

M. Somu and N. Rengarajan
KSR College of Engineering, KSR Kalvi Nagar, Tiruchengode, Tamil Nadu 637215, India

Abstract: In recent years, IP network has been considered as a new delivery network for TV services. A majority of the telecommunication industries have used IP network to offer on-demand services and linear TV services as it can offer a two-way and high-speed communication. In order to effectively and economically utilize the IP network, caching is the technique which is usually preferred. In IPTV system, a managed network is utilized to bring out TV services, the requests of Video on Demand (VOD) objects are usually combined in a limited period intensively and user preferences are fluctuated dynamically. Furthermore, the VOD content updates often under the control of IPTV providers. In order to minimize this traffic and overall network cost, a segment of the video content is stored in caches closer to subscribers, for example, Digital Subscriber Line Access Multiplexer (DSLAM), a Central Office (CO) and Intermediate Office (IO). The major problem focused in this approach is to determine the optimal cache memory that should be assigned in order to attain maximum cost effectiveness. This approach uses an effective Grate Deluge algorithm based Particle Swarm Optimization (GDPSO) approach for attaining the optimal cache memory size which in turn minimizes the overall network cost. The analysis shows that hierarchical distributed caching can save significant network cost through the utilization of the GDPSO algorithm.

Keywords: Digital Subscriber Line Access Multiplexer (DSLAM), Grate Deluge (GD) algorithm, IPTV, PSO, Video on Demand (VOD)

INTRODUCTION

Internet Protocol Television (IPTV), in which TV channels are distributed through IP multicast, has become one of the widely used techniques (Henrik and Mats, 2010). IPTV is considered as an essential aspect in the upcoming IP convergence networks and a new promising approach for telecommunication in which they identify a saturated market in terms of the number of broadband subscribers and residential penetration ratio (Won *et al*., 2008).

A majority of telecom and broadband industries have become TV providers and distribute TV channels via multicast over their backbone network. IPTV is also considered as a development to time-shifted TV in which viewers can select to observe the programs at anytime (Henrik and Mats, 2010). The main benefits of the IPTV are its flexibility to support personalized service as it transmits the user requests to the manage center immediately and it easily distinguishes the user identity. On the other hand, all data on IPTV are encoded as a sequence of IP packets and communicated to the viewers via the residential broadband access network. This characteristic feature removes the conventional constraints of watching TV and furthermore, any Internet connectable digital devices provided with a multimedia player could be a "TV". These unique features offer more chances for generating new services on the next generation TV industry (Hsuan *et al*., 2011).

The basis for IPTV network architecture is to bring QoS guaranteed video services to the end user at the lowest network cost possible. Multicast and several of its associated protocols, is an apparent option by many telecommunications; but, it is still in an experimental phase prior to large-scale deployment in several places (Won *et al*., 2008).

In spite of the supreme effort Quality of Service (QoS) of the Internet, IPTV service providers have yet to offer better user Quality of Experience (QoE) than conventional TV technology (Prasad *et al*., 2012). Furthermore, the main factor which is to be focused in the deployment of the IPTV is the network cost.

A number of researchers have dealt with most of the technical issues of implementing IPTV service in existing or redesigned infrastructures (Agrawal *et al*., 2007). A variety of data delivery techniques, such as multicast (Imran *et al*., 2007; Smith, 2007) and Peer-to-Peer (P2P) (Sentinelli *et al*., 2007) style bartering are examined to lessen the network traffic at the backbone while preventing Quality of Experience (QoE) degradation.

Corresponding Author: M. Somu, KSR College of Engineering, KSR Kalvi Nagar, Tiruchengode, Tamil Nadu 637215, India

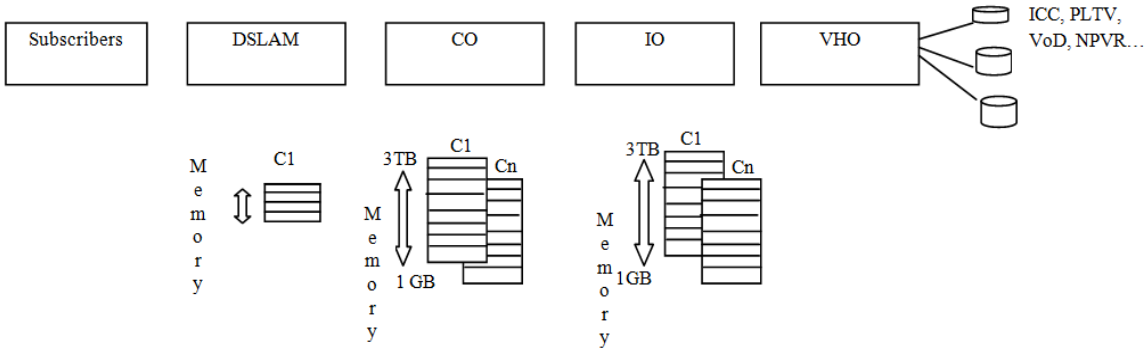


Fig. 1: Hierarchical caching in IPTV network

Caching is a well examined and promising approach for web content and video (Liu and Xu, 2004; Eager *et al*., 1999) and have been extensively used in the context of IPTV (Krogfoss *et al*., 2008).

Caching of video contents can be a prominent solution that lessens the problems of the users in IPTV's on-demand services (De Vleeschauwer and Laevens, 2009; Sarhan and Das, 2008; Almeida *et al*., 2001) Generally, as a segment of popular movies are mostly requested by several users, if certain popular video contents are cached, the load of storage servers can be lightened. Moreover, cached video contents can be streamed directly to users without any start-up delay.

In an IPTV network, Video on Demand (VoD) creates vast unicast traffic from the Video Head Office (VHO) to subscribers and, thus, necessitates added equipment resources in the network. In order to minimize this traffic (overall network cost), segment of the video content may be accumulated in caches closer to subscribers (DSLAMs, COs and/or in IOs) which is shown in Fig. 1. The main issue focused is to find the optimal size and localities of the cache memory in IPTV networks and to determine the appropriate titles and services which should be cached at the suitable locations to attain the maximum cost effectiveness (Bill *et al*., 2008).

Several of the existing video caching approaches do not dynamically cache the files based on individual client requests; instead these approaches make use of replication of segments of the videos based on a pre-estimated access pattern of each video. Practically, the request rate for a specific video may differ with time and the relative importance of the videos may differ from proxy to proxy (Anna and Michael, 2006). Hence, Hierarchical cache optimization technique is used in this study for better performance.

The relationship between cache hit rates and the popularity distribution of VoD titles is briefly examined in this approach to set the stage. Bill *et al*. (2009) described two modeling approaches for hierarchical cache optimization in IPTV networks in order to determine the optimal cache architecture using a heuristic algorithm. As an extension of the Bill *et al*. (2009) approach, this study uses an efficient intelligence algorithm for identifying optimal cache memory size and content distribution at every layer to minimize the overall network costs. Particle Swarm Optimization (PSO) approach is observed to be very effective in determining the optimal cache memory size which in turn minimizes the network cost (Somu and Rengarajan, 2012). In this approach, in order to provide better performance to the PSO, Great Deluge (GD) algorithm based PSO has been utilized for providing effective optimized results.

LITERATURE REVIEW

A more advanced feature is the use of co-operative proxy caching (Chae *et al*., 2002), where a better performance than with independent proxies can be achieved through load balancing and improved system scalability. In this case it is important to continuously keep track of cache states. Note that contrary to standard co-operative proxy caching, there is no need to switch to segments on other proxies when using co-operative proxy caching with sliding intervals. Similar peer-to-peer caching techniques have also been introduced in streaming CDNs, where whole files are stored instead of segments (Turrini and Panzieri, 2002).

WiMAX radio Resource Allocation (WRA) issue is studied in the framework of IPTV broadcasting over mobile WiMAX Multicast, Broadcast Services (MBS) channels. The main aim is to enhance the quality of services, in terms of number of subscribers served; number of IPTV channels carried and perceived video qualities of individual viewers, subject to constraints on multicast channel capacities and space-time channel quality variations. Po-Han and Yu-Hen (2011) presented an effective heuristic technique depending on the Pareto principle that obtains near-optimal results in polynomial time complexity. The simulation results show that the performance of this approach is very significant when compared with other existing heuristic algorithms.

Caching of video content facilitates minimization of bandwidth and IPTV network cost. An algorithm that optimally partitions a cache between several video services with different traffic characteristics and content sizes is described in Krogfoss *et al*. (2008). Sofman *et al*., proposed the concept of content cacheability and

introduced a fast algorithm that utilizes cache-ability to optimally partition a cache between several video services with various traffic characteristics and content sizes. The main focus of the optimization is to serve maximum (in terms of bandwidth) amount of subscribers' requests subject to constraints on cache memory and throughput.

There are various problems and issues that are to be overcome in the existing techniques present in the IPTV environment. Identifying an optimal (in terms of network cost) placement and amount of cache memory is a complex optimization problem (Somu and Rengarajan, 2012). Though most of the processors for IPTV set-tops using hardware-based built-in decoders exhibit good performance in processing high-resolution compressed video, they have drawbacks in running general purpose software as their embedded RISCs provide relatively low computing capability for minimized product costs (Xin *et al*., 2008).

METHODOLOGY

Hit rate in hierarchical networks: Hit rate is the percentage of all requests that are contented by the data in the cache. The effectiveness of the cache is illustrated by hit rate. The discrete version of hit rate, $H(n)$, denotes a segment of service requests that may be served by the n "most popular" titles stored in the cache. The continuous version of hit rate, $H(m)$, is a function of cache memory size m. Hit rate is based on the statistical characteristic features of traffic and on the efficiency of the caching algorithm to update the cache content (Vanichpun and Makowski, 2004).

Zipf-Mandelbrot (1999) (ZM) distribution (Breslau *et al*., 1999) is used in this approach, even though any alternative distribution could also be used. The ZM Probability Mass Function is described by:

$$p(k) = \frac{C}{(k+q)^{\alpha}}$$

where, C is a normalization constant, k is the rank of the object, q is the shift factor, α is a power parameter that find out the steepness of the curve. In the ideal scenario, when the caching algorithm has complete data about the statistical characteristic features of the traffic, the hit rate is equal to the cumulative popularity distribution.

In multiple services, the hit rate is based on the popularity distribution and other characteristic features of individual services as illustrated in Krogfoss *et al*. (2008). In the following, traffic symmetry is assumed for nodes at each level (i.e., the hit rate is the same at each node of every level). It is also assumed that no redundant caching, which means that if certain title is cached at a certain level (e.g., IO), this title is not cached again in downstream nodes (e.g., CO and DSLAM).

The model of "cumulative memory effect" of hierarchical caching, or "virtual" cache (De-Vleeschauwer and Laevens, 2007) is shown in Fig. 2. The "virtual" cache in any node of the tree is the actual cache at that node augmented by the caches in deeper nodes (downstream) of the tree. Video content positioned in the "virtual" cache of the node reduces the unicast traffic on the upstream link of the node (and all links added upstream right up to the root of the tree).

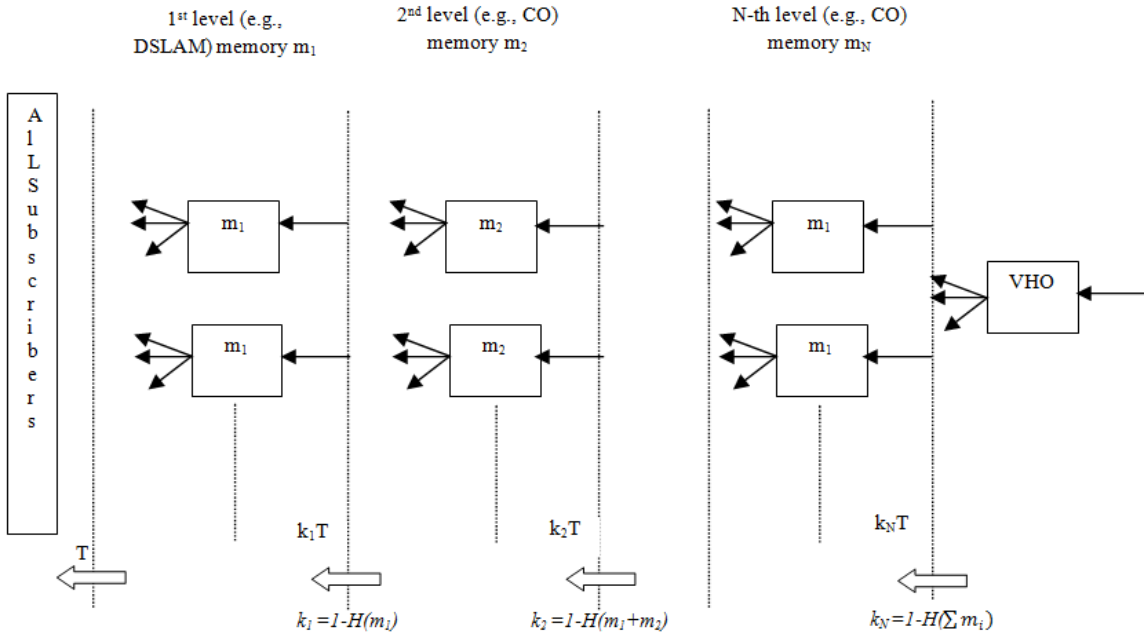


Fig. 2: Traffic flow in hierarchical network with cache memory

For instance, if the cache size per DSLAM is m_1, the cache size per service switch at CO is m_2 and the cache size per service router at IO is m_3, then the caching related traffic reduction (hit rate) at the DSLAM level is $H(m_1)$, at the CO level is $H(m_1 + m_2)$ and at the IO level is $H(m_1 + m_2 + m_3)$.

Heuristic model:

Assumptions: The cache optimization model illustrated below may be applied to any type of tree topology. But, in the following the tree is assumed to be symmetrical and network topology is defined by the following parameters:

- Number of subscribers per DSLAM
- Number of DSLAMs per to CO
- Number of COs per IO
- Number of IOs per VHO

There is an option in this model to dual-home COs, i.e., connect every CO to two IOs. It is to be observed that the COs is associated directly to the VHO in small IPTV networks and there is no IO level; the model can support this network topology as well.

One multicast and one or more unicast services are considered. Parameters of the multicast service (for busy hour) are:

- Number of offered High Definition (HD) and Standard Definition (SD) channels
- Bandwidth per HD and per SD channel
- % of multicast viewers that view HD channels
- % of Set-Top Boxes (STB) tuned to multicast channels

Parameters of every unicast service are:

- Number of titles in the service
- Average memory size per title
- Average traffic per title
- Hit rate

In this model, the cache may be positioned at any mixture of the following layers such as DSLAM, CO, or/and IO. It is to be assumed that there is one equipment shelf per DSLAM and one or several equipment shelves per CO and IO.

The cache in each location consists of one or several cache modules and each cache module engage one slot of the equivalent equipment shelf. Each cache module can accumulate a limited quantity of data (e.g., up to 3,000 GB) and can support a limited amount of traffic throughout (e.g., up to 20 Gbps). The amount of memory per cache module is a multiple of the memory granularity parameter (e.g., 100 GB). Cache cost comprises of cost per cache module and cost per unit of memory.

It is to be observed that the equipment configuration and cost structure can produce certain modularity consequences. For instance, fairly small variations in traffic volume may result in a considerable change in the number of network elements (e.g., ports, MDAs, IOs and even shelves) and, thus, results in considerable change in network cost. With more cache modules and total cache memory per shelf, titles stored in the cache would be more and the more unicast traffic requests will be served from this cache and thus, lesser resources such as bandwidth, ports, equipments, etc., will be necessary upstream from this cache location (De Vleeschauwer and Laevens, 2009). Alternatively, there are a limited number of slots in the equipment, so when more slots are utilized for cache then only fewer slots are available for ports. The main focus is to determine the optimal cache memory size and content distribution at every layer to minimize the overall network costs (i.e., transport, equipment and cache cost).

Cache optimization modes and heuristics: This approach considers three optimization modes such as Adhoc optimization, Layered optimization and Global optimization.

It is to be considered that in Adhoc optimization, the cache configuration (i.e., number of cache modules and the cache memory per shelf at every layer-DSLAM, CO, IO) is given. The main purpose of Adhoc optimization is to determine the optimal distribution of content between caches. It means that the number of titles of each service that should be cached at each layer.

Adhoc optimization also facilitates evaluation of the cost of the network without any cache. The other two modes of optimization namely Layered optimization and Global optimization facilitates the simultaneous optimization of both cache configuration and distribution of the content. These two modes of optimization are built on top of the Adhoc optimization. These two modes of optimization also permits to constrain the layers for cache deployment, e.g., DSLAM only, CO only, IO only, DSLAM and CO only, etc. This can be very valuable in scenarios where the caches can only be deployed at specific layers of the network.

It is to be noted that due to memory granularity (model's parameters) and the finite number of cache modules per shelf, there are a finite number of various cache configurations that should be considered. In case of Global optimization, all probable cache configurations are itemized and Adhoc optimization is executed for every cache configuration. The cache configuration that provides the best Adhoc optimization outcome will be the solution of the Global optimization technique. Generally, Global optimization needs long processing times. Layered optimization provides fairly good solution in lesser time.

The fundamental building block of Layered optimization is optimizing the cache for one specific layer (e.g., CO) whereas the cache configuration for the

other layers (e.g., DSLAM and IO) are kept as fixed. In Layered optimization, an ordered subset of layers is first chosen (in one particular scenario, all three layers-DSLAM, CO and IO-could be chosen). Cache optimization is carried out for the 1st chosen layer and the optimal cache configuration for this layer is fixed. Then, cache optimization is carried out for the 2nd chosen layer and so on. After cache optimization has been carried out for the last chosen layer, the process is repeated with the 1st chosen layer. This process ceases when no further improvement results from the optimization of any of the chosen layers cost. Different to Global optimization, the Layered optimization solution is a local optimum. But, in all the scenarios considered, the results of Global and Layered optimization were close or identical (Vleeschauwer *et al*., 2009). In this approach, PSO algorithm is used for optimizing the cache memory size and content distribution at every layer.

Particle swarm optimization: PSO process is initiated with a collection of random particles (solutions), N. The i[th] particle is denoted by its position as a point in S-dimensional space, where S denotes the number of variables. All through the process, each particle i observes three values namely its current position (X_i), the best position it arrived in previous cycles (P_i), its flying velocity (V_i). These three values are denoted as follows:

$$\begin{aligned} &\text{Current position } X_i = (x_{i1}, x_{i2}, \dots, x_{iS}) \\ &\text{Best previous position } P_i = (p_{i1}, p_{i2}, \dots, p_{iS}) \\ &\text{Flying velocity } V_i = (v_{i1}, v_{i2}, \dots, v_{iS}) \end{aligned} \quad (1)$$

In each time interval (cycle), the position (P_g) of the best particle (g) is computed as the best fitness of all particles.

Therefore, each particle updates its velocity V_i to get closer to the best particle g, as follows (Bill *et al*., 2009):

$$\begin{aligned} &New\ V_i = \omega \times current\ V_i + c_1 \times rand() \times \\ &(P_i - X_i) + c_2 \times Rand() \times (P_i - X_i) \end{aligned} \quad (2)$$

As such, the particle's updated position by means of the new velocity V_i becomes:

$$\begin{aligned} &New\ position\ X_i \\ &= current\ position\ X_i + New\ V_i \\ &V_{max} \geq V_i \geq -V_{max} \end{aligned} \quad (3)$$

where c_1 and c_2 denote two positive constants named learning factors (usually $c_1 = c_2 = 2$); rand ( ) and Rand ( ) denotes two random functions in the range (0, 1), V_{max} is an upper limit on the maximum change of particle velocity (Kennedy and Eberhart, 1995) and ω denotes an inertia weight employed as an enhancement proposed by Shi and Eberhart (1998) to manage the influence of the previous history of velocities on the current velocity. The operator ω balances the global search and the local search; and was introduced to minimize linearly with time from a value of 1.4-0.5. In such case, global search starts with a great weight and then decreases with time to favor local search over global search (Eberhart and Shi, 1998).

```
Begin;
Generate random population of N solutions
(particles);
For each individual i ∈ N: calculate fitness (i);
Initialize the value of the weight factor, ω;
For each particle;
Set pBest as the best position of particle i;
If fitness (i) is better than pBest;
pBest(i) = fitness (i);
End;
Set gBest as the best fitness of all particles;
For each particle;
Calculate particle velocity according to Eq. (3);
Update particle position according to Eq. (4);
End;
Update the value of the weight factor, ω;
Check if termination = true;
End;
```

Fig. 3: Pseudo code for PSO

It is to be observed that the second term in Eq. (2) denotes cognition, or the private judgment of the particle when comparing its current position to its own best position. The third term in Eq. (2), alternatively, denotes the social collaboration among the particles, which compares a particle's current position to that of the best particle (Kennedy, 1997). Moreover, in order to control the change of particles' velocities, upper and lower bounds for velocity change is limited to a user-specified value of V_{max}. Once the new position of a particle is computed using Eq. (3), the particle, then, flies towards it (Shi and Eberhart, 1998). Thus, the main parameters used in the PSO are: the population size (number of birds); number of generation cycles; the maximum change of a particle velocity V_{max} and ω. Figure 3 shows the pseudo code for the PSO.

In the original PSO, the velocity and position updating rule is given by Eq. (4) to (5):

$$\begin{aligned} &v_{id}^{t+1} = v_{id}^{t} + c_1 r_1 \\ &(pbest_{id}^{t} - x_{id}^{t}) + c_2 r_2 (gbest_{d}^{t} - x_{id}^{t}) \end{aligned} \quad (4)$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{td}^{t+1}, i = 1,2, \dots, n \quad (5)$$

where, c_1 and c_2 represent constants named acceleration coefficients. r_1 and r_2 are two independent random numbers uniformly distributed in the range of (0, 1). $v_i \in [-v_{max}, v_{max}]$ where $v_{max}$is a problem-dependent constant defined in order to clamp the

For each particle i
Randomly initialize $v_i x_i = p_i$
Evaluate $f(p_i)$
$p_g = \arg max\{f(p_i)\}$
End for
Choose WL and Up
Repeat
Each particle i
Update particle position x_i according to equation below
$v_i = x[v_i + c_1 e_1 - (p_g - x_i) + c_2 e_2 - (p_i - x_i)]$,
$x_i = x_i + v_i$
Evaluate $f(x_i)$
If $(f(x_i) > f(p_g))$
$p_i = x_i$
End if
If $((f(xi) > f(pi))(\&\&f(x_i) > WL))$
$p_g = \arg\max\{f(p_i)\}$
End if
$WL = WL + Up$
Until termination criterion reached

Fig. 4: Pseudo-code of GDPSO

excessive roaming of particles. $pbest_{id}^{t}$ is the best previous position along the d[th] dimension of particle i in iteration t (memorized by every particle); $gbest_{d}^{t}$ represents the best priorlocation among all the particles along the d[th] dimension in iteration t (memorized in a general store house) (Sajjad *et al*., 2012).

The original PSO is improved by Shi and Eberhart (1998) by modifying Eq. (6):

$$v_{id}^{t+1} + v_{id}^{t} + c_1 r_1 (pbest_{id}^{t} - x_{id}^{t}) + c_2 r_2 (gbest_{d}^{t} - x_{id}^{t}) \quad (6)$$

where w $>= 0$ is defined as inertia weight factor. The thorough experimental investigations of PSO with inertia weight have shown that a relatively large w have more global search capability while a relatively small w results in a faster convergence.

Convergence criterion of PSO: In general, convergence is a phenomenon in which a system or process reaches a stable state. For the population based optimization approach, the convergence of algorithm can be considered via individual or the entire swarm. For instance, there are two convergence definitions for Genetic Algorithm. The convergence definition of PSO was suggested by Van Den Bergh (2002) gave, which is explained below.

Provided a particle position $x(t)$ and an arbitrary position p in search space, the convergence is defined by the following Eq. (7):

$$\lim\ x(t)_{t\to\infty} = p \quad (7)$$

This definition shows that the convergence of particles is that the particle eventually stops at a certain position p in search space. By examining the trajectories of particles, Van Den Bergh (2002) suggests that all the particles are convergent to the positions of the global best solutions. This aspect is very important, as it reveals a key characteristic aspect of PSO, i.e., *gbest* is the attractor of the whole swarm. Evidently, *gbest* itself alters the algorithm runs.

If the entire particles attain the convergence, no further alteration exists and the stable state is obtained. Thus, the PSO algorithm has attained convergence. As a result, *gbest* will not alter. Thus, another convergence definition of PSO can be given which is discussed below.

Definition 2: Given that the best position of PSO in time *t* or in t[th] generation is *gbest* (*t*), *gbest** is a fixed position in search space, the convergence definition is written as Eq. (8):

$$\lim\ gbest(t)_{t\to x} = gbest^{*} \quad (8)$$

Definition 2 suggests that, if *gbest* constructed by PSO does not alter any more, then convergence is attained. If the *gbest* is the global best solution, then the algorithm achieves the global best convergence. Or else, the algorithm is stuck in local optima.

Great Deluge based PSO (GDPSO): This study presents an enhanced version of the Particle Swarm Optimization algorithm using the Great Deluge algorithm called GDPSO (GDPSO). In the general PSO, after attaining a new result, the obtained result is compared with the best solution identified so far and if it is found to be better, it will be accepted. But, in this proposed approach, the achieved solution is compared with both, the best identified solution so far and with another parameter called "Water Level" or WL. If it is better than the both, it is accepted as new solution.

Actually, there is a level of acceptance inside the PSO for new solutions and this process provides a second chance to particles to get rid of the trap, if it is trapped in the local optimum.

The proposed algorithm is fundamentally different with the basic PSO so that it tries to utilize the basic technique of Great Deluge local search in the PSO algorithm.

The basis of this technique is the PSO algorithm and some alterations have been made to the PSO. The WL parameter is used as an acceptance level and UP parameter of the great deluge algorithm is used to find out the permissible range of the answers. UP parameter is used in increasing or decreasing the WL. Novel approach has been evaluated on certain standard functions and performance of the algorithm compared with PSO standard. Test results show that the proposed algorithm extensively raises the capability of PSO to escape from the local optimum and the accuracy and the convergence rate. Pseudo-code of MPSO algorithm is shown in Fig. 4 (Sajjad *et al*., 2012).

Analytical model:

Assumptions: In order to analytically resolve the cache optimization issue, certain assumptions are to be considered. Initially, in this analysis, granularity factors are eliminated. Especially, it is to be considered that the cost of cache memory is proportional to the size of the cache and equipment cost is proportional to the amount of traffic that traverses the equipment. In particular, the equipment cost is evaluated based on the amount of traffic obtained by this equipment from top levels of network hierarchy (r-traffic) and from the quantity of traffic sent by this equipment to subordinate levels of the network hierarchy (s-traffic). For instance, the cost of a CO node may be evaluated based on:

- The amount of traffic this CO node sends to DSLAMs (or s-traffic) and the cost per unit of this traffic
- The amount of traffic that this CO node receives from IO (or r-traffic) and the cost per unit of this traffic

To compute a total network cost, a cost per unit of cache memory and a cost per unit of s-and r-traffic are defined for each level of the network hierarchy (DSLAM, CO, IO).

It is to be assumed that the tree topology structure is completely symmetric, i.e., the "fan-out" at each level is the same. Moreover, all contents are downloaded only once to the caches during off-peak time, so the network cost associated with these downloads can be ignored.

The next consideration to be made is regarding the hit rate, $H(m)$, as a function of cache memory m. It is clear that the hit rate, $H(m)$, increases with memory m. Moreover, $H(m)$ is assumed to have a continuous derivative $H'(m)$ and this derivative is decreasing (effect of diminishing returns for hit rate). Thus, $H(m)$ is strictly concave.

It is to be assumed that out of the two cache resources namely cache size and cache throughput, cache size is a limiting factor and the only resource to be considered. This assumption can be justified for the cache in DSLAMs by setting a maximum cache size that assures that traffic from the cache does not go beyond the cache throughput.

In the analytical model, only the unicast traffic is taken into consideration. Because of replication, multicast traffic is a fairly small segment of the total traffic between the VHO, IO, CO and DSLAM levels and, thus, does not make a huge influence on equipment costs at those levels.

Mathematical formulation: In the case of tree topology with K levels of hierarchy, the total network cost, $NtwkCost$, may be computed as:

$$NtwkCost(m_1, m_2, \ldots, m_k) = \sum_{k=1} N_k c_k^m m_k + Tc_{1s}^t + T\sum_{k=1}^{k}((c_{kr}^t + c_{k+1s}^t)\left(1 - H\left(\sum_{j=1}^{k} m_j\right)\right) \quad (9)$$

The following parameters are used in (9).

Decision variables:

m_k : Cache memory size per node at k-th level, $1 \le k \le K$ (GB)

T : Total amount of traffic requested by subscribers (Mbs)

$H(m)$: Hit rate as a function of cache memory m

N_k : Number of nodes at k-th level of hierarchy, $1 \le k \le K$

M_k : Maximum cache size per node at k-th level, $1 \le k \le K$, (GB)

c_k^m : Cost of cache memory at k-th level, $1 \le k \le K$ ($/GB)

c_{ks}^t and c_{kr}^t: Cost of traffic at k-th level sent to (k-1) -th (straffic) and received from (k+1) -th level (r-traffic), $1 \le k \le K+1$, ($/Mbs).

The goal is to minimize network cost subject to constraints on cache memory size:

$$NtwkCost(m_1, m_2, \ldots, m_k) \rightarrow \min$$
$$\text{such that } 0 \le m_k \le M_k, i = 1 \le k \le K$$

EXPERIMENTAL RESULTS

A large metropolitan DSL based ISP network is used in this reference scenario. A 4-level network is assumed with DSLAMs at the lowest level that are aggregated at COs by routers. In large metros there are often intermediate aggregation points, known as Intermediate Offices (IOs) that aggregate several COs. The IOs all terminate at a VHO that can be collocated with a Point of Presence (PoP). These topology assumptions are:

- The total number of DSLAMs in the network, is 9,600
- The total number of service switches in all COs, is 100
- The total number of service routers in all IOs, is 16

The following maximum storage limits per cache location are assumed:

- The maximum cache size per DSLAM is 100 GB
- The maximum cache size per service switch at CO is 12,000 GB (12TB)
- The maximum cache size per service router at IO is 24,000 GB (24TB)

The cost assumptions are:

$c_k^m, k = 1, 2, 3$, the cost of flash memory is $22/GB.

c_{1r}^t; the cost of traffic that a DSLAM receives from a CO is $1.5/Mbps.

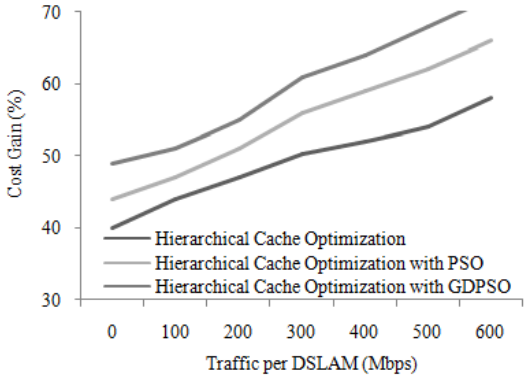


Fig. 5: Optimal cache solution for varying traffic

c_{2s}^t and c_{2r}^t; the cost of traffic that a CO sends to a DSLAM and receives from a IO, respectively, is \$2.5/Mbps.
c_{3s}^t and c_{3r}^t; the cost of traffic that a IO sends to a CO and receives from the VHO, respectively, is \$4/Mbps.

Number of the particles in the algorithms is considered 50. The amount of increasing Water Level (WL) is averaged as 0.0002 (Sajjad *et al*., 2012).

The total traffic T is varied to investigate the impact of increasing traffic on different caching solutions. Ultimately, Zipf-Mandelbrot (1999) distribution is considered for popularity with a power parameter alpha = 1 for the reference scenario. These numbers were chosen based on empirical data and industry averages; nevertheless, a variety of sensitivity analyses was done to investigate the degree to which the results and conclusions would depend on specific values of these parameters. In the following sections, all parameters (unless mentioned specifically) have values from this reference scenario.

Sensitivity to traffic variation: The modeling results of the reference scenario are shown in Fig. 5. The graph shows the optimal cache solution in terms of cost gain as traffic volume is varied.

It is clear from the graph that the cost gain of the proposed Hierarchical Cache optimization with GDPSO attains better cost gain with varying traffic per DSLAM.

According to the graph, for a traffic volume of 400 Mbps at the DSLAM, a cost gain of 52 and 59% is obtained for the Hierarchical Cache optimization approach and hierarchical cache optimization technique with PSO approach. But, the proposed hierarchical cache optimization technique with GDPSO attains a cost gain of 64%.

It is observed that the solution becomes hierarchical (as opposed to single level caching) with increase in traffic volume. As traffic volume increases, caches are first deployed at the IO, then at the CO and ultimately at the DSLAM too.

Impact of network topology: This section considers the impact of network topology on caching solutions and cost gain. The topologies of the network operators vary due to differences in loop lengths, number of COs per region and broadband technique (VDSL, ADSL, GPON, etc.). For a particular number of COs, longer loop networks necessitate distributed and smaller DSLAMs and more DSLAMs per CO, while shorter loop networks facilitate centralized and larger DSLAMs and lesser DSLAMs per CO.

From the analytical solution, consider the case in which the optimum cache has a moderate (or "non boundary") solution. m_i for each location is:

$$0 < m_i, < M_i; i = 1, 2, 3$$

The corresponding equations are (Somu and Rengarajan, 2012):

$$H'(m_1^o) = \frac{N_1 c_1^m - N_2 c_2^m}{Ts_1} \quad (10)$$

$$H'(m_1^o + m_2^o) = \frac{N_2 c_2^m - N_3 c_3^m}{Ts_2} \quad (11)$$

$$H'(m_1^o + m_2^o + m_3^o) = \frac{N_3 c_3^m}{Ts_3} \quad (12)$$

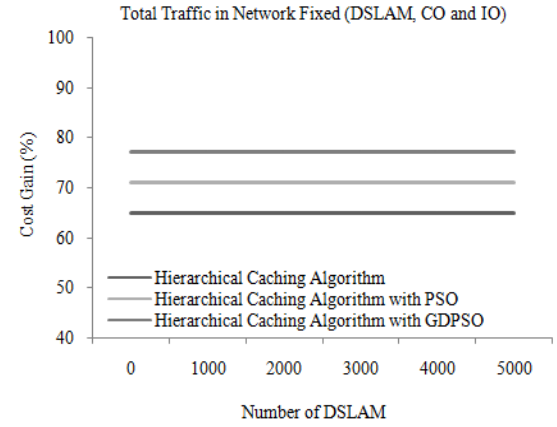
where, s_1, s_2 and s_3 are traffic cost parameters:

$$s_1 = c_{12}^t + c_{21}^t$$
$$s_2 = c_{22}^t + c_{31}^t \quad (13)$$
$$s_3 = c_{32}^t + c_{41}^t$$

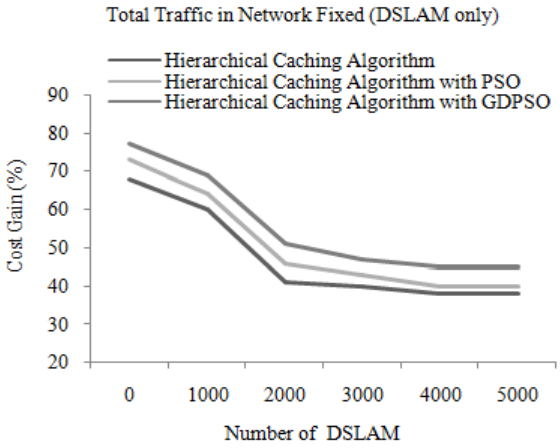
From Eq. (10), as N_1 (# of DSLAMs) increases for a given amount of traffic T the total memory at the DSLAM m_1 must decrease. From Eq. (11), it is to be observed that the total storage $m_1 + m_2$ does not depend on the number of DSLAMs (N_1), thus if N_1 increases and m_1 decreases, m_2 must increase by a sufficient amount so as to satisfy both equations. Therefore if the number of COs is fixed and the number of DSLAMs can be changed, all other things being equal, whatever storage is removed from the DSLAM will be added to the CO.

Figure 6 shows the result of varying the number of DSLAMs. Figure 6a shows that the cost gain achieved for the varying DSLAM. The cost gain achieved by the Hierarchical caching Algorithm approach and Hierarchical caching Algorithm with PSO approach is 65 and 71%, respectively. But, the proposed MPSO approach, the cost gain achieved is 77%.

In Fig. 6b, the solution is limited to DSLAM only and it is clearly observed that the savings decrease as the number of DSLAMs increase. The cost gain achieved by the proposed GDPSO approach is observed to be better than the other approaches considered.



(a)



(b)

Fig. 6: Optimal cache solution for varying topology

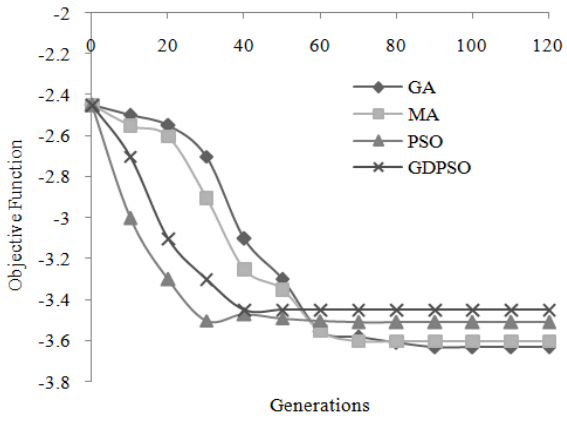


Fig. 7: Comparison of objective function

Table 1: Performance comparison of the optimization techniques

Optimization algorithms	Processing time (sec)
Genetic algorithm	16
Memtic algorithm	21
PSO	15
GDPSO	11

Performance of the optimization algorithms: The GDPSO algorithm outperformed the Genetic Algorithm, Memtic Algorithm and the PSO algorithm in attaining the optimal cache memory size in terms of the processing time.

It is observed from the Table 1 that the proposed GDPSO optimization technique takes processing time of 11 sec, where as the other optimization techniques such as GA, MA and PSO takes longer processing time such as 16, 21 and 15 sec, respectively.

Figure 7 shows the comparison of the objective function of the GA, MA, PSO and the proposed GDPSO approach. It is observed from the figure that the GDPSO converges in lesser iterations (i.e., 40 iterations) when compared with the other optimization techniques such as GA, MA and PSO. Thus the proposed GDPSO technique is very significant when compared with the other optimization approaches taken for consideration.

CONCLUSION

This study focuses on the hierarchical cache optimization in an IPTV network using swarm intelligence approach. Great Deluge based Particle Swarm Optimization (GDPSO) is presented in this study for better optimized results. In the proposed optimization approach, the range for obtained answers is defined that is the same parameter used in the GD algorithm called "water level". Amount of this range reduces or increases regarding to algorithm's property being used in terms of minimum or maximum during the time. The main aspect of the proposed GDPSO algorithm is that the particles are given a second opportunity using GD algorithm. Therefore, if a particle is trapped in the local optimum they can get rid of it very soon. Optimization algorithm is used for attaining the optimal cache memory size which in turn minimizes the overall network cost. A number of key parameters such as network topology, traffic volume, hit rate and cost are taken into consideration to compute optimal cache sizes in DSLAM, CO and IO nodes. The heuristic model considers more detailed information about equipment configuration and cost; this model is appropriate when caching architectures in a specific IPTV network is need to be optimized. But, due to multiple levels of cost modularity in the heuristic model, it is hard to identify factors that influence the solution using this approach. The analytical technique exploits a simpler cost structure and certain reasonable assumptions, which facilitates to recognize basic factors that affect this solution. A sensitivity examination based on the analytical model facilitated to calculate the influence of several parameters on the optimal cache configuration and illustrated that for several typical cases the optimal cache configuration involves caches at two (CO and IO) or all three levels of the network hierarchy.

REFERENCES

Agrawal, D., M.S. Beigi, C. Bisdikian and L. Kang-Won, 2007. Planning and managing the IPTV service deployment. Proceeding of the 10th IFIP/IEEE International Symposium on Integrated Network Management (IM '07). Munich, Germany, pp: 353-362.

Almeida, J.M., D.L. Eager and M.K. Vernon, 2001. A hybrid caching strategy for streaming media files. Proceeding of the SPIE/ACM Conference on Multimedia Computing and Networking.

Anna, S. and P. Michael, 2006. Efficient caching of video content to an architecture of proxies according to a frequency-based cache management policy. Proceedings of the 2nd International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications (AAA-IDEA '06), NY.

Bill, K., B.S. Lev and A. Anshul, 2008. Optimal cache partitioning in IPTV network. Proceedings of 11th Communications and Networking Simulation Symposium (CNS'08). Ottawa, Canada, April 14-17, pp: 79-84.

Bill, K., B.S. Lev and A. Anshul, 2009. Hierarchical cache optimization in IPTV networks. Proceeding of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB '09).

Breslau, L., P. Cao, L. Fan, G. Phillips and S. Shenker, 1999. Web caching and Zipf-like distributions: Evidence and implications. Proceedings of the 18th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '99), 1: 126-134.

Chae, Y., K. Guo, M.M. Buddhikot, S. Suri and E.W. Zegura, 2002. Silo, rainbow and caching token: Schemes for scalable, fault tolerant stream caching. IEEE J. Sel. Area Comm., 20(7): 1328-1344.

De-Vleeschauwer, D. and K. Laevens, 2007. Caching to Reduce the Peak Rate. Alcatel-lucent Internal Report. Retrieved from: www.docstoc.com/.../Particle-Swarm-Intelligence-Approach-for-Enhanc.

De Vleeschauwer, D. and K. Laevens, 2009. Performance of caching algorithms for IPTV on-demand services. IEEE T. Broadcast., 55(2): 491-501.

Eager, D., M. Ferris and M. Vernon, 1999. Optimized regional caching for on-demand data delivery. Proceedings of the Multimedia Computing and Networking (MMCN'99). San Jose, California.

Eberhart, R. and Y. Shi, 1998. Comparison between genetic algorithms and particle swarm optimization. Proceedings of the 7th Annual Conference on Evolutionary Programming. Springer, Berlin, pp: 611-618.

Henrik, A. and B. Mats, 2010. Simulation of IPTV caching strategies. Proceeding of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS). Ottawa, ON, pp: 187-193.

Hsuan, C., C. Chi-He, T. Chao-Wei and L. Chi-Shi, 2011. Window-based popularity caching for IPTV on-demand services. ISRN Commun. Network., 2011(2011): 1-11.

Imran, K., M. Mellia and M. Meo, 2007. Measurements of multicast television over IP. Proceeding of the 15th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN 2007). Princeton, NJ, pp: 170-175.

Kennedy, J., 1997. The particle swarm: Social adaptation of knowledge. Proceedings of the IEEE International Conference on Evolutionary Computation. Indianapolis, IN, 1: 303-308.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks. Perth, WA, pp: 1942-1948.

Krogfoss, B., L. Sofman and A. Agrawal, 2008. Caching architectures and optimization strategies for IPTV networks. Bell Labs Tech. J., 13: 13-28.

Liu, J. and J. Xu, 2004. Proxy caching for media streaming over the internet. IEEE Commun. Mag., 42: 88-94.

Po-Han, W. and H. Yu-Hen, 2011. Optimal layered video IPTV multicast streaming over mobile WiMAX systems. IEEE T. Multimedia, 13(6): 1395-1403.

Prasad, C., C. Prashanth, T. Gregg, H. Nathan, R. Rajiv, Y. Delei, L. Ying, X. Lixia and Y. Daoyan, 2012. Multi-resolution multimedia QoE models for IPTV applications. Int. J. Digit. Multimedia Broadcast., 2012(2012): 1-13.

Sajjad, G., P.K. Rahim, S.M. Mahdi and R.M. Mohammad, 2012. A modified PSO using great deluge algorithm for optimization. J. Basic Appl. Sci. Res., 2(2): 1362-1367.

Sarhan, N.J. and C.R. Das, 2008. Caching and scheduling in NAD-based multimedia servers. IEEE T. Parall. Distr., 15(10): 921-933.

Sentinelli, A., G. Marfia, M. Gerla, L. Kleinrock and S. Tewari, 2007. Will IPTV ride the peer-to-peer stream? [Peer-to-peer multimedia streaming]. IEEE Commun. Mag., 45(6): 86-92.

Shi, Y.H. and R.T. Eberhart, 1998. A modified particle swarm optimizer. Proceedings of the IEEE International Conference on Evolutionary Computation, Proceedings of IEEE World Congress on Computational Intelligence. Anchorage, AK, pp: 69-73.

Smith, D.E., 2007. IPTV bandwidth demand: Multicast and channel surfing. Proceeding of the 26th IEEE International Conference on Computer Communications (INFOCOM 2007). Anchorage, AK, 6-12 May, pp: 2546-2550.

Somu, M. and N. Rengarajan, 2012. Particle swarm intelligence approach for enhanced hierarchical cache optimization in IPTV networks. Eur. J. Sci. Res., 76(3): 366-378.

Turrini, D. and F. Panzieri, 2002. Using p2p techniques for content distribution internetworking: A research proposal. Proceeding of the 2nd IEEE International Conference on Peer-to-Peer Computing, pp: 171-172.

Van den Bergh, F., 2002. An analysis of particle swarm optimizers. Ph.D. Thesis, University of Pretoria, Elandspoort 357-Jr, Pretoria 0002, South Africa.

Vanichpun, S. and A.M. Makowski, 2004. Comparing strength of locality of reference-popularity, majorization and some folk theorems. Proceeding of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004), 2: 838-849.

Vleeschauwer, D.D., Z. Avramova, S. Wittevrongel and H. Brueel, 2009. Transport capacity for a catch-up television service. Proceedings of the EuroITV'09. Leuven, Belgium, pp: 161-170.

Won, J.W., J.W.K. Hong, C. Mi-Jung, H. Chan-Kyu and Y. Jae-Hyoung, 2008. Measurement of download and play and streaming IPTV traffic. IEEE Commun. Mag., 46(10): 154-16.

Xin, W., Z. Changyi, Z. Zhenyuan, L. Hong and X. Xiangyang, 2008. The design of video segmentation aided VCR support for P2P VoD systems. IEEE T. Consum. Electr., 54(2): 531-537.

Zipf-Mandelbrot, 1999. Law. Retrieved from: http://en.wikipedia.org/wiki/Zipf-Mandelbrot_law.