International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

Method for Fragmentation of Requested Applications in File-Sharing Peer-to-Peer Networks

Volodymyr Popovskyy[1], Nabeel Oudah Mnekhir[2], Niran Sabah Jasim[3]

[1]Department of Infocommunication Systems, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

[2]Communications Engineering Department, Uruk University, Baghdad, Iraq

[3]Department of Mathematics, College of Education for pure Science/Ibn-Al-Haitham, University of Baghdad, Baghdad, Iraq

Abstract: ***The processes of request, search, selection and delivery of video content in peer-to-peer networks with the unstructured architecture, which is most consistent with heterogeneous and dynamic Internet environment, are considered in this work. The problem solution of files fragmentation requested by the consumer in a P2P network of streaming video using a discrete-continuous mathematical autoregressive model is used, which displays the state of each j-th peers on each of the k-steps of fragmentation. The solution of the problem is reduced to the procedure of dynamic programming.***

Keywords: Peer-to-peer; TV video; streaming video; fragments; differential model

1. Introduction

In recent decades, with the development of broadband communication channels and the Internet, video services have gained special popularity. At the same time, the requirements for streaming video transmitted through peer-to-peer file-sharing networks have increased. These networks have become so popular that now there are several hundred users around the world, and the technologies themselves are continuously improved, providing a high level of QoS, QoE services.

Ease of access to these networks is accompanied by a great complexity of the network technologies used that make it difficult to solve the problem of optimizing these networks. At the same time, it is possible to solve a number of optimization problems for certain processes in P2P networks.

2. General Information about Peer-to-Peer Networks

Peer-to-peer networks, depending on the architecture, are divided into structured and unstructured networks. In addition to the differences in the structure, these networks also differ in function. Therefore, in structured networks deterministic routing is used. Each peer has a local routing table used to forward messages. Unlike in unstructured networks, the architecture is random. Unstructured networks are also called self-structured, reflecting a high degree of dynamism and decentralization.

In structured networks, the focus is on managing the overlay. Structured overlay nodes jointly support routing information, according to which delivery of the ordered data is provided to the address indicated in the identification key.

Routing in structured overlays is based on keywords (keys) that support the address space. Options put (enter) and get index (get the index) are mapped to a distributed hash table (DHT). Since the address space is virtualized, peer addresses are usually assigned randomly, which allows remote peers to be enabled.

Most structured P2P networks support searching for objects based on a static key and an identifier. To do this, a tree topology overlay is used, when only one copy of each fragment is delivered to each peer.

Depending on the method of forming media content in peer-to-peer networks, a "strip-based" system is used, which is used in structured networks and "fragment-based" systems. A characteristic feature of strip-based systems is that each strip has its own individuality of distribution: the tree structure is built with the root in the source of the media data. Peers at the same time tend to be a part of each of these distribution structures in order to obtain the entire media flow.

In systems based on fragments, media content is divided into a series of fragments (of a given or arbitrary size) and the fragment is transmitted "from the repository" on request. Each fragment in the overlay has its own behavior and arriving at the peers takes place in a different order, thus there arises problem of establishing the order and use of the buffer for storing fragments. The method of forming on fragments used in unstructured networks is well coordinated with the dynamics of the Internet.

In unstructured networks for message delivery, the corresponding node only deals with neighboring nodes in the overall overlay.

The central problem in the study of unstructured overlays is the development of an efficient search method, the distribution and processing of queries, the localization of objects. Therefore, the aim of our report is to develop and optimize the method for searching and processing the claimed content.

The specifics of the functioning of peer-to-peer networks is that the information required on requests is distributed, the content is scattered over various peers. Therefore, when searching for content, it is necessary to take into account the specific network model and characteristics of the resource to which access is made. Reliable enough is the search for objects based on DHT, where keys and routing are used with respect to the peer, which has the closest key to the key of the requested object.

However, DHT management requires considerable effort due to the dynamics of the network topology. In addition, DHT does not support complex queries.

The specificity of streaming is that the delivery of content and its transportation is carried out simultaneously with its consumption. In this case, multicast is the main type of streaming, where protocols are defined to deliver video to a group of recipients at the same time, using effective strategies.

To search for the necessary resource in peer-to-peer unstructured networks based on keywords (KW), two types of request routing are used: "blind routing", when resource allocation is not taken into account, and the "routing index" method, where this allocation is taken into account.

Thus, in the search process, in addition to the specified methods of searching and delivering content, it is necessary to solve the routing task with the orientation to the current state of the content and the nature of which depends on the reliability and delay in obtaining the necessary resources, the degree of utilization of network resources, and the result in obtaining popular and rare media data.

The most common in peer-to-peer networks is "blind routing", which includes flooding and random walk methods.

The difference is under flooding, the received request is sent to all its neighbors, while for random walks one of the neighbors is selected to continue the search on request. Therefore, flooding consumes more network resources. On the other hand: coverage for random walks is significantly narrowed, which is especially true when searching for rare resources.

An alternative is the method of index routing, which uses additional information on network topologies, search history, etc. There are two types of indexes for routing queries: with a content orientation directed to a structure of a non-cyclic graph oriented to queries, when peers accumulate a story about the results of previous requests for neighbors, and these data are taken into account when selecting routing indexes.

The process of searching for keywords occurs in a relay fashion and begins with a request for a peer (the source of the request). The request can consist of one KW or several KWs. One or group of peers, receiving a request and finding the resources relevant to the request, generates a message and tries to send the result back to the source of the request. If the required resources are not found, the requesting peer redirects this request to the neighboring peer, reducing the TTL (Time To Live) value by one - the lifetime of the request. After the TTL is exhausted, the application is canceled. Each request has a unique identifier in the form of a 128-bit hash key, with which the request is addressed and an appropriate response is made.

The process of searching for a resource on demand can be decomposed into five stages:

1) Simulation of the request. The request source generates an identification code and sets the TTL value.
2) In unstructured P2P networks, unlike structured networks, the address and content stored in a particular node are not linked, while searching for the necessary information is carried out on the basis of two solutions: blind flooding and information retrieval, based on saving the routing index to obtain data on the topology and state of neighbors.
3) Forwarding of the request. The peer who received the request writes the identifier for the case of a return of the request retry, reduces the TTL and forwards the request to the neighbors.
4) Local search. When the requested resource is found, the result message is sent over the backup path to the repeater.
5) Transfer of results. The detected resource is sent to the request source according to its identifier.
6) Merging the results. The request source receives results from its neighbors and ranks them accordingly. After that, selection takes place according to the chosen criterion.

Peer-to-peer networks, being private overlay file-sharing networks; have a number of specific features relatively to a common IP network. Therefore, in the well-known file-sharing applications (Bit Torrent) the optimization of network performance is reduced to searching for the "best" peer from which the content is downloaded. We will point out other tasks that are important for improving network performance, which must be solved in parallel in real time:

- Algorithms for the scheduling of fragments, which provides the possibility of smooth playout,
- Selection of a peer with low delays,
- Ensuring interaction between the dynamics of overlay and ip-routing.

Users (peers) of P2P networks can download data portions both from the server and from each other. When exchanging buffer maps, each user gets information about what data chunks are available at a particular seed-user and what he is ready to download. Thus, at the same time there are m suggestions for downloading the necessary information. The number of these proposals may be different and limited to the appropriate protocol of the scheduler or the number of peers offering the requested information and forming a particular cluster. At the same time, the number of proposals cannot be excessively large, since the network is unnecessarily overloaded, the bandwidth decreases. Therefore, the overlay module usually limits the number of possible connections. The control module acts as a manager in relation to the scheduler and overlay, providing monitoring of video exchange and signal information. Each media stream in the network is divided into fragments, the sizes of which can be equal or different, depending on the P2P protocol. At the same time, the dynamics of the download procedure, in addition to other reasons (the number and quality of the proposed options, fragments, bandwidth limitations, download rate and upload rate, etc.) depends on how well the fragments themselves are organized.

3. Problem Statement

Monitoring should be provided to: the vector of parameters $X_L^N = (x_1, x_2, ..., x_n)$ observed on the interval $t = 1,2,...,N$ of the proposed $i = 1,2,...,m$ fragments, the points of their state change, and a matrix of conditional probability of transitions Q from state to state. The download process is a discrete-continuous vector process:

$$\vec{x}(k+1) = f(k, \vec{x}(k), \vec{u}(k)),$$

$$k \in K = \{k_1, k_2, ..., k_F\},$$

where

$\vec{x}_t^k = (x_1, x_2, ..., x_{KF})$ is a sequence of selected fragments, forming a downloadable file,

$x_{ki} = (x_{k1}, x_{k2}, ..., x_{km})$, $i = I \in \{1,2,...,m\}$ is a sample at each k-th step from the m-class of suggested fragments,

$u(k)$ is control of the selection process for fragments on each of the k-steps.

The quality and time of delivery and the duration of each of the fragments are random and different.

To ensure smooth play out, it is necessary to arrange the sequence of fragments in the buffer in accordance with how they are delivered to the consumer. In addition, there should be no pauses between the fragments and they cannot run into the neighbor. To do this, it is necessary to select at each the k-steps the fragment from the m-proposed ones, which will have a minimum delay, the maximum transmission rate, etc.

Obviously, as the mathematical model of the file being processed, the autoregression equation can be used, the coefficients of which α_i, $i = (0, m)$ are controlled by a Markov chain with m states [1,2]

$$x_t = \alpha_0(h_t) + \sum_{i=1}^{m} \alpha_i(h_t) x_{t-i} + b(h_t)\xi_t \qquad (1)$$

where

ξ_t is sampling from Gaussian white noise, $b(h_t)$ is the noise level,

$H_{iN} = h_1, h_2, ..., h_N$ is the state of the sequence controlling the parameters of the shift and the scale of the corresponding fragments.

$p(h_t / h_{t-1}) = q(h_{t-1}, h_t)$ is the conditional probability of transition from the state h_{t-1} to state h_t, elements of transition matrix $Q = \{q(h_{t-1}, h_t\}$.

It is logical to set the task of finding the optimal values of states $\hat{H}_{iN} = \hat{h}_1, \hat{h}_2, ..., \hat{h}_N$ ordering the sequence of fragments in accordance with the ordering requirements. As an optimization criterion, we choose such a set of parameters h_i, for which the minimum of the functional is ensured on the interval $t \in (1, N)$:

$$Y(H_0^N) = R_0(h_0) + \sum_{t=1}^{n} [\eta_t(h_t) + \eta(h_{t-1}, h_t)] \qquad (2),$$

Where

$R_0(h_0)$ is the initial, probable value of the zero sample of membership in the given sample class h_0^i, $i = 1,2,...,m$,

$$a_t(h_t) = \frac{1}{2b^2(h_t)} \left[x_t - \alpha_0(h_t) - \sum_{t=1}^{n} \alpha_i(h_t) x_{ti} \right]^2 - biq(h_{t-1}, h_t) \quad (3),$$

$$h(h_{t-1}, h_t) = \ln \frac{q(h_t, h_t)}{q(h_{t-1}, h_t)} = \ln \frac{p(h_t / h_t)}{p(h_t / h_{t-1})} \qquad (4)$$

Obviously, under $h_{t-1} = h_t$, the value $h(h_{t-1}, h_t) = 0$, in other cases α has an idea of the penalty function that depends on the difference and mismatch of the neighboring samples of the fragments.

The value $a_t(h_t)$ has the meaning of inconsistencies in the values of the parameters of the fragment in the section $t - n \le s \le t$ with the predicted ones.

4. Problem Solution

The solution of this optimization problem is reduced to methods of dynamic programming or integer linear programming (ILP), [3].

Assuming the Gaussian distribution law of the autoregression coefficients, the conditional apriori probability of the occurrence of $H_t^N = (h_1, h_2, ..., h_N)$ combination is determined by the transition probabilities of the Markov chain [1,3]

$$f(\chi_1^N / \chi_{-n+1}^0, H_1^N) = \prod_{t=1}^{N} f(x_t / x_{t-1}, h_t) =$$

$$\prod_{t=1}^{N} \frac{1}{b(h_t)\sqrt{2\pi}} \exp\left\{ -\frac{1}{2b^2(h_t)} \left[x_t - \phi_0(h_t) - \sum_{i=1}^{n} \phi_i(h_t) x_{t-i} \right]^2 \right\} \qquad (5)$$

It is logical to assume the Gaussian character of the probability density of the initial value $R_0(h_0)$, which can be represented in the form $d_0(h_0) = -\ln p(h_0)$, where $p(h_0)$ are the a priori probabilities of possible initial states of the Markov chain. In this case, the minimum of the functional $Y(H_0^N)$ corresponds to the maximum a posteriori probability of combinations of fragments for a given implementation, [5]

$$\hat{H}_0^N = \arg\min_{H_0^N} Y(H_0^N) = \arg\max_{H_0^N} \ln p(H_0^N / \chi_1^n)$$
$$= \arg\max_{H_0^N} [\ln p(h_0) + \ln p(H_1^N / h_0) \quad (6)$$
$$= \ln f(\chi_1^n / \chi_{n+1}, H_1^N)]$$

Here $p(H_1^N / h_0) = \prod_{t=1}^{N} q(h_{t-1}, h_t)$ is the conditional a priori probability of a combination of fragments $X_i^N = (x_1, x_2, ..., x_n)$ composed of m variants offered by different peers.

Construction of optimal fragmentation. We define a sequence of functions with allowance for a temporal discrete argument:

$$R_t(h_t) = \min_{H_0^{t-1}} \left\{ R_0(h_0) + \sum_{S=1}^{t-1} a_s(h_s) + \eta(h_{s-1}, h_s) + a_t(h_t) + \eta(h_{t-1}, h_t) \right\},$$
$$t = 1, 2, ..., N \quad (7)$$

This value $R_t(h_t)$ indicates what minimum value of the criterion $Y(H_0^N)$ can be achieved on the sequence of fragments up to $(t-1)$ including, if the state of the last fragment h_t is fixed. Obviously, the minimum value of the last value $d_N(h_N)$, $h_N = 1,2$ coincides with the minimum criterion value $Y(H_0^N)$:

$$\min R_N(h_N) = \min Y(H_0^N) \quad (8)$$

If the last fragment of the optimal sequence is H_0^N, then

$$\hat{h}_N = \arg\min R_N(h_N) \quad (9)$$

We compute successively the value vectors $R_{t+1}(h_{t+1})$ for the corresponding instants of time $t = 0,1,..., N-1$, starting with the $R_0(h_0)$ according to the rule:

$$R_{t+1}(h_{t+1}) = \min_{h_t}[d_t(h_t) + a_t(u_t) + a(h_t, h_{t+1})] \quad (10)$$

In this case, integer values

$$K_{t+1}(h_{t+1}) = \arg\min\{R_t(h_t) + a_t(u_t) + h(h_t, h_{t+1})\} \quad (11)$$

form a rectangular matrix K_i^N in which the columns N are arranged in time-ordered values $\hat{H}_0^N = (\hat{h}_0, \hat{h}_1, ..., \hat{h}_N)$. The number of rows corresponds to the values m presented for selection at each fragmentation step. The corresponding values $\hat{h}_t$ are found from the recursive formula

$$\hat{h}_t = K_{t+1}(\hat{h}_{t+1})$$

With the initial condition (9).

Thus, the evaluation of optimal values $\hat{h}_t$ is taken in the reverse order: $t = N, N-1,...,1,0$ after the calculation in the direct order of the elements - the columns of the matrix K_i^N, which contain integers mN - media fragments.

As for the problem of variants selection, from the theory of solutions is known [4] that the problem of optimal fragmentation is extremely cumbersome, refers to NP-complete and with an increase in the number m of proposed options it turns out to be problematic to obtain a solution in real time. Solutions are quite allowable with $m = 1;2;3$. It is also known from the theory of auto-selection [5] that the transition from a single choice to a two-fold one allows one to reduce the probability of an error P_{er} from $P_{er} = 10^{-2}$ up to $P_{er} = 10^{-4}$. A further increase in the multiplicity adds an ever smaller relative increase in efficiency in decision making figure 1. Therefore, in order to prevent overloading in the solution of the problem, we stop the choice by the value $m = 2$ that for the tasks of fragmentation it is quite acceptable.

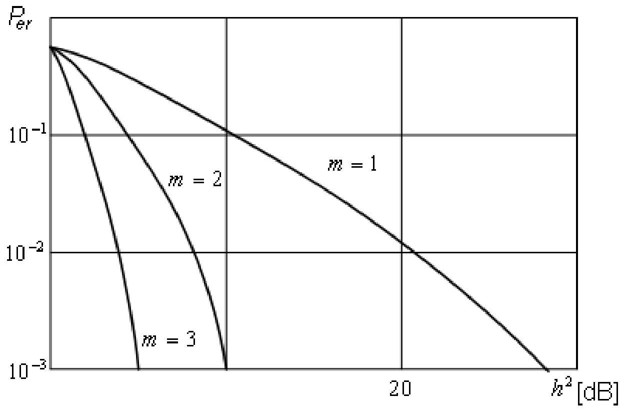

Figure 1: Dependence of the error probability P_{er} on the ratio of the signal level to the spectral noise power h^2

5. Conclusions

1) Peer-to-peer networks are among the peer-to-peer (P2P) file-sharing networks that work over the Internet. They have become very popular with consumers of video products. Networks are characterized by simplicity of access and rather complex technologies of querying, searching, delivering and transporting content simultaneously with its consumption.
2) In unstructured P2P networks, the search for the required content directly depends on the amount of memory and replication of the content of this network, which ensures its maximum performance and stability. Since in these networks, unlike structured ones, the address and content stored in a particular node are not interconnected, the search for the necessary information is carried out on the basis of two solutions: blind flooding and information search, which, on the basis of saving the routing index, allows to receive the information on topology and neighbor state. This technology also has greater scalability and reliability, which makes it suitable for heterogeneous and dynamic environment of the Internet.
3) A solution is presented for the task of fragmentation of files requested by the consumer in a P2P network of streaming video using a discrete-continuous mathematical autoregressive model that displays the process of

searching for and selecting suitable resources when downloading video content.

4) Using the developed model, it is possible to display the state of each j-th peer on each of the k-steps of fragmentation taking into account the delay in their delivery. This model allowed solving the problem of optimal construction of the procedure for finding the solution vector by sequentially selecting the requested file from m-suggested ones. The solution of the problem is reduced to the procedure of dynamic programming.
5) From the m-suggested fragments, it is recommended to select two, which will help reduce the service traffic. In this case, the probability of an erroneous solution, in accordance with the theory of auto-selection, is reduced by a value of 17-20 dB.

References

[1] IETF REC 6972: Hroblem Statement and Requirements of the Peer-To-Peer Streaming Protocol (PPSP). http://datatracker.ietf.org/doc/zfc6972/.
[2] IETF Survey of P2P Streaming Applications. http://datatracker.ietf.org/doc/draft-ietf-PPSP-survey/.
[3] Yu.V.Gaydamaka and A.K.Samulov, “Data Buffering Model in Streming P2P Networks,” WSPU, Moscow, pp.8656-8666, 2014.
[4] L.A.Kogan “Optimal Sementation of Expermental Curves on the Basis of the Dynamic Programming,” A& T, Vol.45, pp.146-156, 1988.
[5] V.Popovskij and A.Barkalov and L.Titarenko “Control and Adaptation in Telecommunication Systems,” Springer-Verlag Berlin Heidelberg, Vol.5, 2011.
[6] J.Crowcroft, E.Lua and M. A.Pias, “Survey and Comparison of Peer-to-Peer Overlay Network Schemes. – ICST,” 2004.
[7] N.Moskalets and E.O.Popvskaya “Optimization Optimization of the Total Weighted Service Time in the Per-To-Per,”.