

P2P VoD サービスにおける 動画の人気度に基づくキャッシュ手法の提案

藤本 貴也^{†1} 遠藤 伶^{†1}
松本 敬^{†1} 重野 寛^{†2}

P2P VoD サービスでは、視聴した動画データをピアがキャッシュに保持し、サーバの代わりに配信することで、サーバの配信負荷を低減している。ピアのキャッシュ方法は一般的に FIFO が用いられる。しかし、FIFO では、不人気動画のデータもキャッシュに保持してしまうため、ピアの送信帯域を有効に活用できない。本稿では、動画の人気度を考慮することで、ピアの送信帯域を効率的に利用し、サーバの配信負荷を低減する手法 PEC を提案する。PEC では、人気動画のデータを優先してピアがキャッシュに保持し、サーバの代わりに配信するピア数を増やす。また、シミュレーション評価により、各ピアの過去視聴動画数が 2 個以上の状態における有用性を示す。

Proposal of Caching Based on Video Popularity for P2P VoD Service

TAKAYA FUJIMOTO,^{†1} REI ENDO,^{†1} KEI MATSUMOTO^{†1}
and HIROSHI SHIGENO^{†2}

In P2P VoD services, peer saves the load of delivery server by caching data of the video that the peer has watched, and delivering the video to other peers instead of the delivery server. Although FIFO is typically used as this caching algorithm, FIFO is not efficient for using upload band of peers, because peer may caches data of non-popular video. This paper propose PEC that reduces the load of delivery server by considering popularity of videos. PEC increases the number of peers that deliver the video to other peers because peer gives data of the popular videos priority over non-popular videos for caching. In addition, we evaluate the PEC by the simulation and show effectivity for situation in that each peer has watched 2 or more videos.

1. はじめに

ユーザが観たい時に様々な映像コンテンツをインターネットから視聴できる streaming サービスとして、VoD(Video-on-Demand) サービスがある。server-client 型 VoD サービスでは、動画配信は全て配信サーバが行う。そのため、同時に視聴している総ユーザ数が増加すると、サーバの台数や通信回線の帯域が多く必要となり、配信コストが高くなる。

この問題を解決する手段として P2P ネットワークを使用した P2P VoD Streaming が研究されている。P2P ネットワークでは、接続された対等な存在のコンピュータをピアと呼び、サービスをお互いに直接やりとりする事ができる。このピアとはサービスに参加する一般ユーザの PC である。P2P VoD Streaming では、ピア間で動画の配信を行える。そのため、server-client 型では配信サーバに集中していた配信負荷をピア間に分散でき、サーバの配信負荷を低減できる。一般的に P2P VoD Streaming において、動画データはある再生時間間隔で分割されて扱われる。その一つ一つをセグメントと呼ぶ。そして、多くは、視聴後のセグメントを一定サイズだけキャッシュに残し、そのキャッシュにあるセグメントを他ピアへ送信することで、ピアが配信を行う。

しかし、既存研究では、不人気の動画をピアが視聴した際、そのキャッシュが不人気動画のセグメントみになってしまう。そして、他ピアがそのキャッシュにあるセグメントを要求しなくなり、不人気動画を視聴したピアは他ピアに配信できない場合がある。そのため、既存手法ではピアの送信帯域を活用しきれない。よって、使用できていなかったピアの送信帯域を活用することで、サーバの配信トラフィック量をさらに削減する余地がある。

本稿では、動画の人気度を考慮してピアがキャッシュを行うことで、ピアの送信帯域をより多く利用し、サーバの配信トラフィック量を低減する手法 (PEC) を提案する。本提案手法では、ピアのキャッシュを記録している tracker サーバが動画の人気度を計測する。そして、ピアがセグメントをキャッシュに保持するか否かを、その動画の人気度に基づき決定する。不人気の動画を視聴しても、過去に視聴した人気の動画をキャッシュから破棄しない事で、キャッシュしたセグメントを他ピアから要求されないピアを減らす。そのため、より多くのピアの送信帯域を活用でき、サーバの配信トラフィック量を削減できる。

^{†1} 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, Keio University

^{†2} 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University

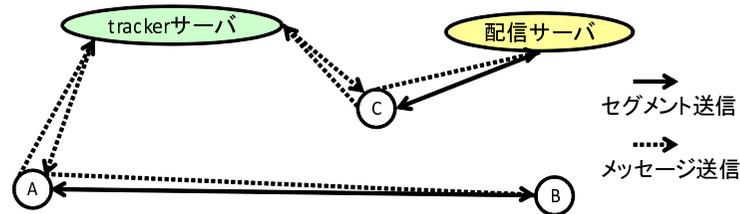


図1 P2P VoD サービス

2. 関連研究

図1にP2P VoDサービスにおける tracker を用いた手法のトポロジを示す．このサービスは配信サーバ，tracker サーバ，ピアから成り立つ．

- 配信サーバ
配信動画のセグメントを全て保持し，常にオンラインである．ピアからセグメントの要求が来た場合，そのピアに対しセグメントを送信する．
- ピア
ユーザとして動画を視聴する．視聴動画のセグメントを配信サーバ，または他ピアからダウンロードする．そして，セグメントをビデオストリームに組み立て，再生する．視聴後もセグメントはキャッシュに保持する．キャッシュにあるセグメントを他ピアから要求されれば，そのピアに対して要求されたセグメントを送信する．
- tracker サーバ
どのピアがどのセグメントを持っているのかを記録する．ピアからセグメント探索要求が来た際，そのセグメントを持つピアを伝える．もしそのセグメントをキャッシュに保持しているピアがいなければ，サーバからもらうよう伝える．

P2P VoD サービスでは，ピアが動画を一定サイズだけキャッシュに保持し，配信サーバの代わりに他ピアへ配信することで，配信サーバの配信負荷を低減する¹⁾²⁾³⁾では，ピアが視聴した動画のみをキャッシュに残し他ピアへ配信する手法を提案した．ピアは既に視聴したセグメントを典型的には FIFO(First In First Out) でキャッシュに一定サイズ保持する．

そして，視聴するセグメントを所持するピアを検索するために tracker サーバを用いる手法が一般的に用いられる⁴⁾．tracker サーバはピアの保持するキャッシュを記録し，ピアに視聴するセグメントをどの相手からダウンロードすべきか教える．この手法は，VoD Streaming に P2P ファイル共有手法である BitTorrent⁵⁾ を併用した Toast⁶⁾，BASS⁷⁾ で

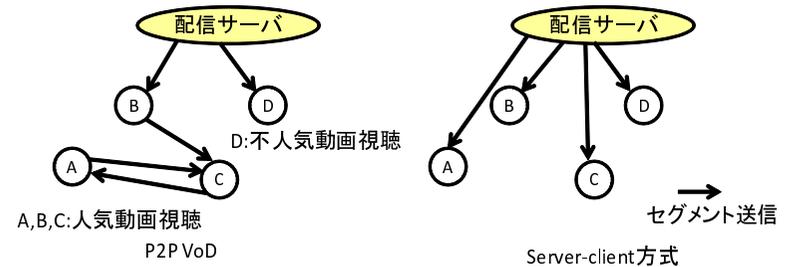


図2 P2P VoD と server-client 型 VoD

用いられる⁸⁾では現実の VoD サービスである中国の国営放送の中央放送局，CCTV で実際にログをとり，そのログに基づいたシミュレーションによりこの手法の有用性が示された．

図2に示すように，tracker を用いた手法の利点は，人気動画を視聴したピアは，そのキャッシュに人気動画のセグメントが保持されるため，人気動画を視聴する他のピアにセグメントを送信できる点である．多くのピアは人気動画を視聴するため，多くのピアの送信帯域を利用でき，その分，配信サーバの配信トラフィック量を低減できる．

しかし，この手法では，不人気の動画を視聴したピアは，他ピアに送信できない場合があり，その分ピアの送信帯域を活用できていないという問題がある．そのピアのキャッシュにあるセグメントが不人気の動画のセグメントのみになり，キャッシュにあるセグメントをどのピアも要求しないからである．以上により，セグメントを他ピアに送信できていなかった，不人気動画を視聴したピアの送信帯域を活用することで，配信サーバの配信トラフィック量をさらに削減する余地がある．

3. 提 案

本稿では，P2P VoD サービスにおいて，動画の人気度を考慮することで，ピアの送信帯域を効率的に利用し，サーバの配信トラフィック量を減らす手法 PEC(Popularity Based Efficient Caching) を提案する．動画を視聴したピアがその動画のセグメントをキャッシュに保持するか否かを，tracker サーバがその動画の人気度に基づき決定する．動画の人気度は tracker サーバが計測する．人気動画のセグメントのみをピアのキャッシュに残すことで，キャッシュしたセグメントを他ピアから要求されないピアを減らす．そのため，より多くのピアの送信帯域を活用できるので，サーバの配信トラフィック量を低減できる．

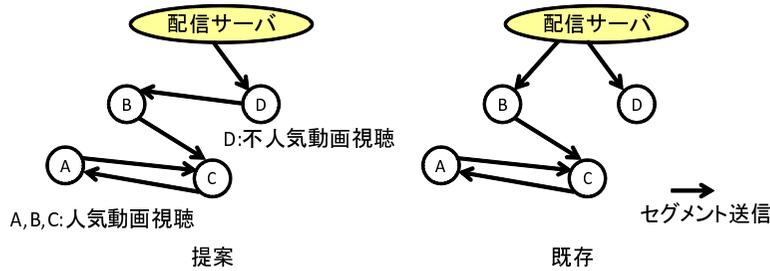


図3 セグメント送信者の例

3.1 PEC

PECでは、不人気動画視聴ピアのキャッシュにも人気動画のセグメントを残すようにする。これにより、既存手法では使用できなかった不人気動画視聴ピアの送信帯域を活用でき、配信サーバの配信トラフィック量を低減できる。

図3に、FIFOを用いた既存手法と、PECでのセグメント送信者の例を示す。既存手法では、人気動画視聴ピアであるA,B,Cは、キャッシュに人気動画のセグメントを持つため、他ピアにセグメントを送信できる。しかし、不人気動画視聴ピアであるDは、キャッシュに不人気動画のセグメントしか持っていないため、他ピアにセグメントを送信できない。

提案手法PECでは、不人気動画視聴ピアであるDも、キャッシュには人気動画のセグメントを持つ。そのため、人気動画視聴ピアであるA,B,Cに加え、ピアDも他ピアにセグメントを送信できる。そして、既存手法に比べ、不人気動画視聴ピアであるDもセグメントを送信する分、配信サーバの負荷が低くなる。

3.2 ピアによるキャッシュ方法

PECでは、人気動画のセグメントは、FIFOでキャッシュに残し、不人気動画のセグメントはキャッシュに残さない。そのため、不人気動画を視聴した場合も、過去に視聴した人気動画がキャッシュに残る。図4に、ピアAがa,b,c,d,eの順に動画を視聴した場合のピアAのキャッシュを示す。FIFOでは、不人気動画であるc,eを視聴した場合、キャッシュにある人気動画のセグメントが破棄されてしまい、不人気動画のセグメントが増えてしまう。一方、PECでは、不人気動画であるc,eを視聴しても、動画c,eのセグメントをキャッシュに保持しないため、それぞれ人気動画であるb,dのセグメントがキャッシュに残る。

図5にピアによるセグメントのキャッシュ保持手順を示す。ピアはまず、セグメント探索要求をtrackerサーバに送信する。trackerサーバは、セグメント探索要求を受信後、セグ

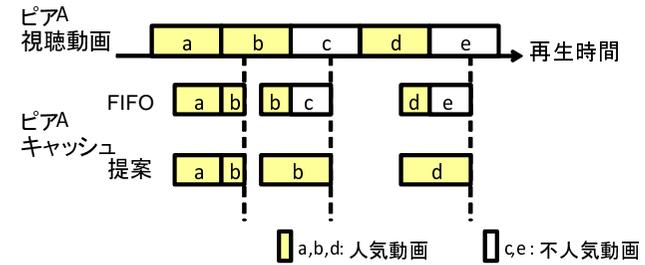


図4 ピアのキャッシュ

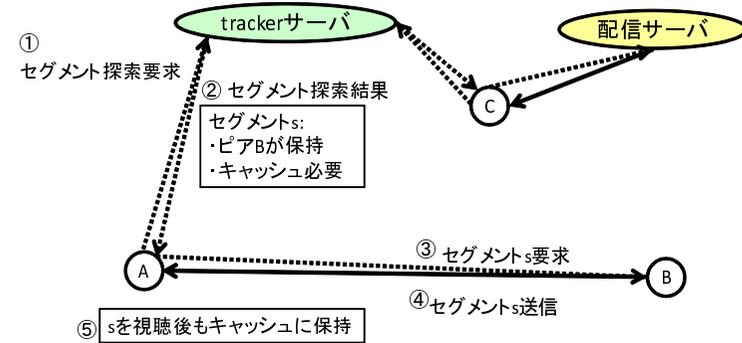


図5 セグメントのキャッシュ保持

メント探索結果を返す。セグメント探索結果では、そのセグメントをどの相手からダウンロードすべきか伝え、さらに、視聴後にキャッシュに保持すべきかも伝える。そして、ピアはtrackerサーバから指示された相手にセグメント要求を送信する。セグメント要求を受信したピアまたは配信サーバは、セグメントを送信する。セグメントをダウンロードしたピアは、そのセグメントを視聴後、trackerサーバからのセグメント探索結果に従い、キャッシュに保持または破棄する。以上により、ピアはtrackerサーバの判断に従ってセグメントをキャッシュに保持する。

ただし、初めて動画を視聴する場合、キャッシュに空きがある間は、不人気動画のセグメントもキャッシュに保持する。

3.3 trackerによる人気度計測方法

trackerサーバは一定時間ごとに動画の人気度を測り、更新する。詳細は3.4節に示す。

```

//Nv:総動画数, Ni:人気度 i 番目の動画を視聴するピア数
//N:総ピア数, Up:平均送信帯域, Vb:動画再生ビットレート
//th:人気スレッシュヨルド順位
sum = 0;
for(i = 1; i <= Nv; i++)
{
    sum += Ni
    if(sum >= N *  $\frac{U_p}{V_b}$ )
        break;
}
th = i;
    
```

図 6 人気スレッシュヨルド順位 th の算出手順

アからの通知により、どのピアがどのセグメントを持っているのかを tracker サーバは記録する。そこで、一定時間ごとに、動画毎の視聴ピア数を記録し、この視聴ピア数で動画の人気順位をつけ、動画をキャッシュに保持するか決定するための人気度スレッシュヨルド順位 th を設定する。なお、動画毎の視聴ピア数は、各動画の最初の再生時間のセグメントを要求したピア数とする。 th 位までの人気度上位の動画はキャッシュ必要動画とし、ピアに対して視聴後もキャッシュに保持するよう指示する。 th 位より人気度下位の動画はキャッシュ不要動画とし、ピアに対して視聴後はキャッシュから破棄するよう指示する。

人気度スレッシュヨルド順位 th は動画の人気度順位、視聴ピア数、ピアの平均送信帯域の関係から算出する。 th を算出するための疑似コードを図 6 に示す。

また、この th の計算式は式 1 で表せられる。

$$th = \min(v) \quad \{v \mid \sum_{i=1}^v N_i > \frac{U_p}{V_b} N\}. \quad (1)$$

ここで、人気度 i 番目の動画の視聴ピア数を N_i 、 U_p はピア平均送信帯域、 V_b は動画のビットレート、 N は同時視聴ピア数である。

全ピアがセグメントの送信を行っても $N \frac{U_p}{V_b}$ のピア数にしか、ピアによる動画配信が行えない。そこで、この全ピアの送信帯域を th 位より人気度上位の人気動画の配信に割り当てるために、 th 位より人気度上位の人気動画のみを FIFO でキャッシュに保持する。以上の

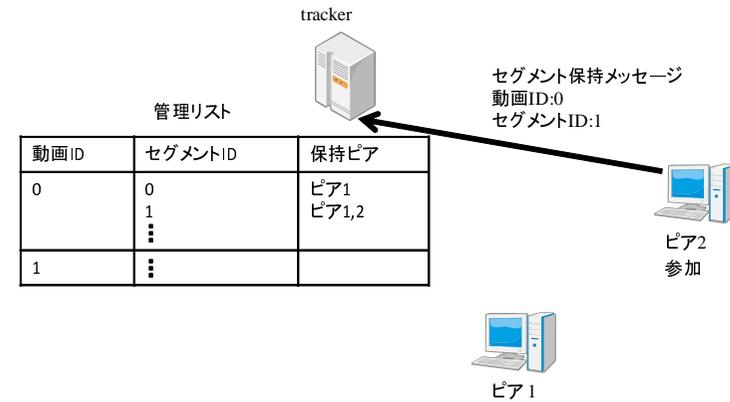


図 7 参加ピアのキャッシュの記録

ように、 th を設定することで、ピアの送信帯域を多く活用する。

3.4 動作アルゴリズム詳細

a ピア参加時の動作

ピアが参加する際に、過去に視聴し、現在もキャッシュに保持しているセグメントをセグメント保持メッセージによって tracker サーバに知らせる。図 7 に示すように、tracker サーバはセグメント保持メッセージにより、新たに参加したピアのキャッシュを記録する。この記録はセグメント管理リストにピアを追加することで成される。セグメント管理リストは以下の情報をもつリストである。

- 管理しているセグメントの動画の ID
- 管理しているセグメントの ID
- 管理しているセグメントをキャッシュに持つピアのリスト

これにより、新たに参加したピアのキャッシュにあるセグメントを他ピアが求める際、tracker サーバが新たに参加したピアを指定することができる。

また、ピアは参加時に送受信帯域通知メッセージによって、tracker サーバに自分の送信帯域と受信帯域を申告する。tracker サーバは送受信帯域通知メッセージから、ピアごとの送受信帯域を記録しておく。

b ピアによるセグメントのダウンロード時の動作

ピアのキャッシュは以下の二つのバッファから成り立つ。ピアはダウンロードしたセグメ

ントをこの二つのバッファに保持する。

- 再生バッファ

ピアがこれから視聴する再生前のセグメントを一定再生時間分だけ保持できるバッファである。キャッシュ必要動画のセグメントは再生後、送信バッファに保持した後、再生バッファからは破棄する。キャッシュ不要動画のセグメントは再生後、送信バッファに保持せずに、再生バッファからは破棄する。

- 送信バッファ

ピアが再生後のセグメントを一定サイズ保持できるバッファである。再生後のキャッシュ必要動画のセグメントを FIFO で送信バッファに保持する。

ピアは再生バッファが満たされるか、視聴動画の最後の再生時間のセグメントがダウンロードされるまで、先の再生時間のセグメントをダウンロードする。

ピアがセグメントをダウンロードするために、まずピアがセグメント探索要求を tracker サーバに送信する。tracker サーバはどのピアがどのセグメントを持っているのかをセグメント管理リストに記録している。そのため、tracker サーバはセグメント探索要求を受信したら、セグメント管理リストから目的のセグメントをキャッシュに持つピアを検索する。そして、セグメントの送信元となる相手であるセグメント送信ピアを選択する。

セグメント送信ピアの選択の際は、その目的のセグメントのセグメント管理リストに最も最近追加されたピアを優先して、選択する。その理由として、第一には、リストに最近追加されたピアは、別のピアに対するセグメント送信ピアにまだ割り当てられていない傾向にあるため、送信帯域に余裕がある可能性が高いからである。第二には、ピアのキャッシュは人気のセグメントを FIFO で保持しているため、新しくセグメント管理リストに追加されたピアの方が、長い間そのセグメントを保持している可能性が高いからである。

さらに、セグメント送信ピアを選択する際、ピア管理リストからそのピアの残りの送信帯域を確認する。もし別のピアにセグメントを送信しており、送信帯域が残っていないならば、セグメント管理リストにその次に最近追加されたピアを確認する。二番目に最近追加されたピアにもし送信帯域が残っていれば、そのピアをセグメント送信ピアとする。もし送信帯域が残っていないならば、セグメント管理リストのまたさらに次に最近追加されたピアを確認する。以下同様にして、セグメント送信ピアを決定する。もし、セグメント管理リストに送信帯域に残りがあるピアがいなければ、配信サーバをセグメントの送信元となる相手とする。

tracker サーバは、セグメント送信ピアの IP アドレスを、セグメント探索結果を送信することによりピアに伝える。もし適切なセグメント送信ピアがいなければ、配信サーバの

IP アドレスを伝える。さらに、tracker サーバはそのセグメントの動画がキャッシュ必要動画か判断し、人気度から目的のセグメントをキャッシュに保持する必要があるかもセグメント探索結果の送信により伝える。

また、tracker サーバはピア管理リストを用いて、このセグメントの送受信が行われた際の、セグメントを要求するピアの残りの受信帯域と、セグメント送信ピアの残りの送信帯域を計算し、記録しておく。常にピアの残りの送受信帯域を記録しておくことで、一つのピアに過剰にセグメントを送信させる状況を防ぐ。

tracker サーバからセグメント探索結果が返ってきたピアは、そのメッセージに従い、目的のセグメントを視聴後キャッシュに残すべきかを記録しておく。

そして、セグメント探索結果に指示されたセグメント送信ピアまたは送信元サーバにセグメント要求を送る。セグメント要求を受けたセグメント送信ピアまたは送信元サーバは、そのメッセージに指示されたセグメントを送信する。そして、ピアはセグメントをダウンロードしたら、tracker サーバにセグメント保持メッセージを送り、保持したセグメントを報告する。このメッセージにより、セグメント管理リストにピアを追加することで、tracker サーバは、どのピアがどのセグメントを持っているのか把握できる。

また、tracker サーバは各動画の最初の再生時間のセグメントを要求したピア数を、各動画の視聴ピア数として記録しておく。動画毎の視聴ピア数は一定時間間隔で更新し、3.3 節で示したアルゴリズムに従い、キャッシュ必要動画とキャッシュ不要動画を判定する。

c ピアによるセグメントの視聴後の動作

ピアはセグメントのダウンロードの際に、tracker サーバからのセグメント探索結果から、目的のセグメントを視聴後キャッシュに保持すべきかを記録している。視聴後のセグメントは、その記録に従い、キャッシュに保持する。もし、視聴後キャッシュに保持すべきであれば、送信バッファにセグメントを保持し、再生バッファからセグメントを破棄する。その際、送信バッファに FIFO でセグメントを保持するので、送信バッファに空きが無ければ、破棄されるセグメントがある。送信バッファからセグメントを破棄したら、tracker サーバにセグメント破棄メッセージを送り、セグメントを破棄したことを伝える。もし、視聴後キャッシュに保持すべきでなければ、送信バッファにセグメントを保持しなく、再生バッファからセグメントを破棄する。そして、tracker サーバにセグメント破棄メッセージを送り、破棄したセグメントを報告する。ただし、送信バッファに空きがある場合は、tracker サーバから保持すべきのセグメントと指示されていなくても、送信バッファにセグメントを保持する。この場合はセグメント破棄メッセージは送信しない。

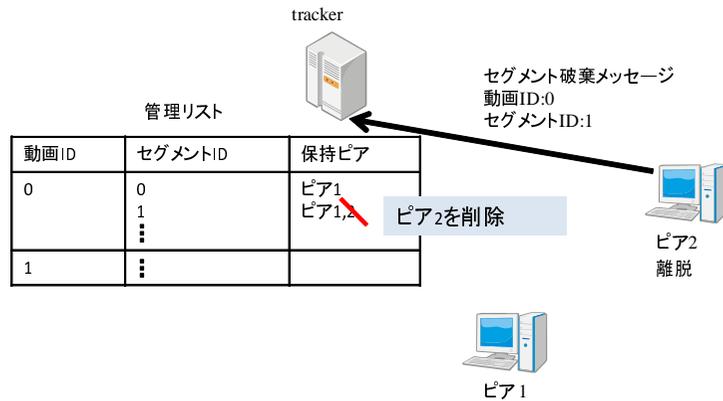


図 8 管理リスト上の離脱ピアの削除

d ピア離脱時の動作

ピア離脱後は、セグメントを送信することができなくなる。そのため、現在キャッシュに保持しているセグメントそれぞれについて、キャッシュ破棄メッセージを tracker サーバに送信する。これにより、示すように、tracker サーバはセグメント管理リストから離脱ピアを削除できるので、離脱ピアがセグメント送信ピアに選択されずにすむ。

また、ピアは離脱後も保持したキャッシュを破棄しないものとする。そのため、次に参加した際、キャッシュ保持メッセージを tracker サーバに送信することで、キャッシュに保持したセグメントの送信を可能にする。

4. シミュレーション評価

提案手法 PEC の有用性を示すために、既存手法である FIFO との比較評価を行う。

4.1 シミュレーション条件

表 1 にシミュレーションで使用した条件を示す。総動画数は、実際の VoD サービスの中国 CCTV を参考に設定したものである⁸⁾。ピアの帯域は、ADSL 回線を想定している。それぞれの VoD サービスの規模によって同時視聴ピア数は異なる。tracker サーバがピアのキャッシュを記録し、セグメント送信ピアの指定も行う P2P VoD サービスでは、tracker サーバの処理負荷を考え、同時視聴者数数千人以上の大規模な VoD サービスは想定していない。ここでは 500 人規模の VoD サービスを想定するが、大規模な VoD サービスでは、

500 人規模で分割して tracker サーバを複数配置すれば対応可能である。tracker サーバ統計更新間隔を 100 秒としたのは、4.4 節で述べる実験の結果により判断した。

表 1 シミュレーション条件

同時視聴ピア数	500
総動画数	5000 個
セグメントサイズ	1sec
ピアのキャッシュサイズ	180 セグメント
ピア upload 帯域	512kbps
ピア download 帯域	1536kbps
動画 bitrate	768kbps
動画の長さ	180sec
tracker サーバ統計更新間隔	100sec
シミュレーション時間	7200sec
シミュレーション回数	10 回

動画の長さを 180 秒としたので、ピア毎に再生時間を分散させるため、最初の 180 秒間でランダムに分散して全ピアが視聴を開始した。その後 100 秒間隔で人気度を判定した。全ピアが参加した 180 秒後から 180 秒毎にサーバ負荷の平均値を算出した。各ピアは動画視聴終了後、別の動画を視聴開始する。表 1 にその他用いたパラメータを示す。またシミュレーションは 7200 秒間を 10 回行い、その平均値を算出した⁸⁾⁹⁾によると、実際に使用されている VoD サービスである CCTV, MSN Video では動画の人気度は Zipf の法則にほぼ従うため、本シミュレーション実験でも Zipf の法則により動画の人気度を設定した。

4.2 ピアの過去視聴動画数の変化による影響

図 9 にピアの過去視聴動画数の変化させたときの平均サーバ配信トラフィック量の結果を示す。横軸はピアの過去に視聴した動画数、縦軸は平均サーバ配信トラフィック量を示す。

ピアの過去視聴動画数が 1 以上あれば、提案手法の方が低いサーバ配信量を達成できている。過去視聴動画数が 1 以上であると、ピアは 2 つ目の動画を視聴しているため、FIFO と提案手法でキャッシュに保持するセグメントに差が出始める。過去に視聴した要求度の高い動画を、キャッシュから消さない方法が有効である事が分かる。

また、過去視聴動画数が 2 個以上で、提案手法はサーバの配信量にほぼ変化がなくなり、FIFO に比べ、平均すると、約 27Mbps のサーバ配信トラフィック量を低下を達成した。これは、FIFO の約 12% にあたる。よって、提案手法は、FIFO に比べ、同じサーバ配信量で、さらに約 60 人分大きい規模に拡張できる。

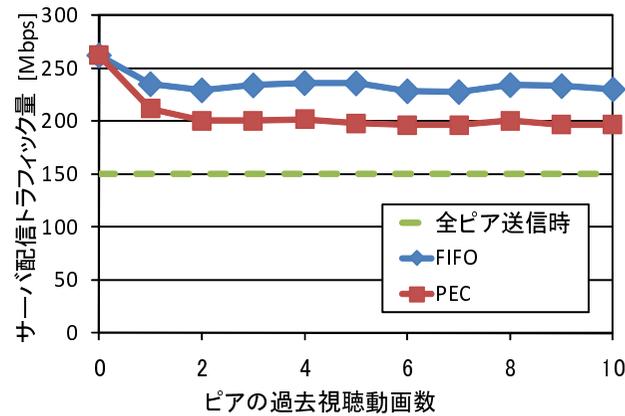


図 9 ピアの過去視聴動画数と平均サーバ配信トラフィック量の関係

全ピアの送信帯域活用時は、約 77Mbps の低下が見込めたのに対し、PEC では 27Mbps 低下できた。キャッシュ必要動画でも、人気スレッシュホールド順位が th 位付近の動画は視聴ピア数が 0 である場合があり、キャッシュ必要動画をキャッシュに保持していても、他ピアからキャッシュにあるセグメントを送信できない場合があるからである。また、あるピアのキャッシュの中で希少性の高いセグメントに要求があった場合でも、他のセグメントを既に送信していて送信帯域に余裕がなく、ピアから送信できない場合もあるからである。

4.3 人気度スレッシュホールド順位 th の変化による影響

図 10 に人気度スレッシュホールド順位 th を変化させたときのサーバ配信トラフィック量を示す。横軸は人気度スレッシュホールド順位 th を対数目盛で表示している。縦軸はサーバ配信トラフィック量である。

ピア数 300 以上では、 th を 240 としたときのサーバ配信トラフィック量が最も低いことがわかる。本提案手法のアルゴリズムでの th の設定方法でも、動画の人気度が既知であれば、240 になるので、ピア数 300 以上で、提案手法の算出方法が有効であることが確かめられた。

また、 th が総動画数の 5000 であるときは、既存手法の FIFO と同じ状況となる。そのため、本提案手法で動画の要求度が既知であれば、既存手法の FIFO に比べ、12% のサーバ配信トラフィック量低下を達成できることを確認した。

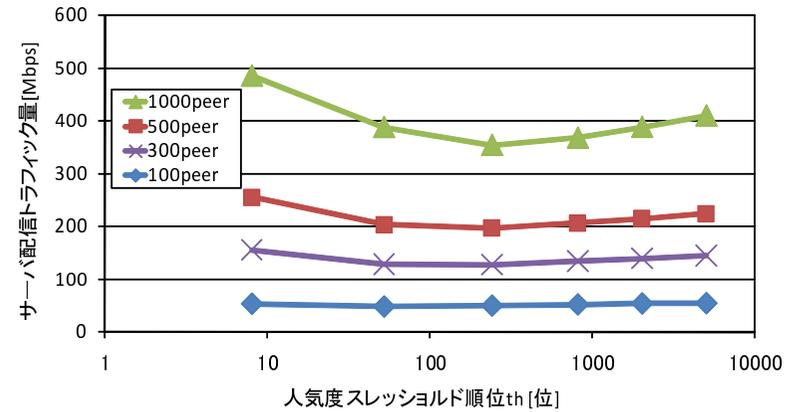


図 10 人気度スレッシュホールド順位 th とサーバ配信トラフィック量の関係

4.4 人気度更新間隔の変化による影響

図 11 に人気度更新間隔を変化させたときのサーバ配信トラフィック量を示す。100 秒間隔のとき最も低いサーバ配信トラフィック量を達成した。100 秒より短い場合、更新間隔が短い方がサーバ配信トラフィック量が高くなるのは、短期間ではピアからの動画要求数が少なく、正確に人気度を測れないためである。100 秒より長い場合、更新間隔が長い方がサーバ配信トラフィック量が高くなるのは、人気度の一時的な変化についていけないためである。動画の人気度の設定は基本的に変化させてはいないが、シミュレーションでは、乱数を発生させ、設定した確率に従ってピアが動画を選択するので、短い期間では、動画の人気度は変化していると考えられる。

5. おわりに

VoD サービスにおけるサーバの低配信コストを目的とする既存の P2P VoD では、不人気の動画を視聴するピアの送信帯域を活用できないという問題点があった。そこで、tracker サーバが動画の人気度を測り、不人気の動画はピアのキャッシュに保持しない手法 PEC を提案した。PEC では視聴セグメントをキャッシュに保持するか否かの判断は、tracker サーバが判断する。tracker サーバは、全ピアの送信帯域で人気度何位までの動画を配信できる帯域があるか算出し、人気度スレッシュホールド順位 th を設定する。このスレッシュホールド順位 th によって、視聴セグメントをキャッシュに保持するか判断することで、より多くのピアが

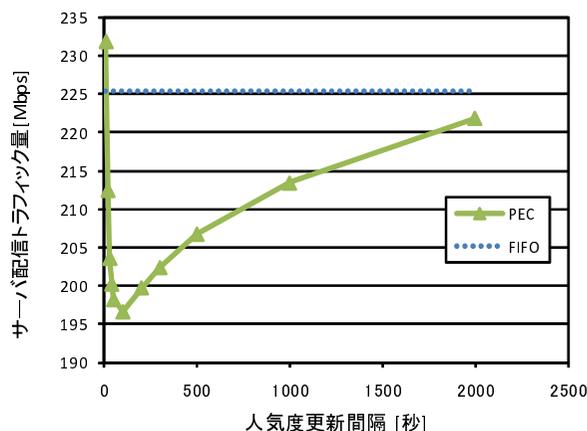


図 11 人気度更新間隔とサーバ配信トラフィック量の関係

キャッシュにあるセグメントを他ピアに送信できる。

また PEC を、シミュレーションにより既存手法との比較評価を行った。その結果、PEC は、ピア数 300 以上で、ピアの過去視聴動画数が 2 個以上のとき、既存手法に比べ 12% のサーバ配信トラフィック量の低減した。配信コスト低下を目的とする P2PVOD サービスにおいて、配信サーバの最大配信トラフィック量の低減を達成したことから、本提案手法の有用性を示した。

6. 謝 辞

本研究の一部はグローバル COE プログラム「アクセス空間支援基盤技術の高度国際連携」により行われました。

参 考 文 献

- 1) Y. Shan and S. Kalyanaraman. Hybrid video downloading/streaming over peer-to-peer networks. In *in Proc. IEEE International Conf. on Multimedia & Expo (ICME)*, 2003.
- 2) A. Sharma, A. Bestavros, and I. Matta. dpam: A distributed prefetching protocol

for scalable asynchronous multicast in p2p systems. In *in Proc. IEEE INFOCOM*, 2005.

- 3) Y. Cui, B. Li, and K. Nahrstedt. ostream: Asynchronous streaming multicast in application-layer overlay networks. *IEEE Journal on Selected Areas in Communications*, 2004.
- 4) Vijay S. Pai, YungRyn Choe, Jagadeesh Dyaberi, Derek Schuff, and Karthik Kannan. Peer-to-peer video on demand: Challenges and solutions. In *IEEE International Conference on Multimedia & Expo (ICME)*, 2009.
- 5) A. Bharambe, C. Herley, and V. Padmanabhan. Analyzing and improving a bittorrent network's performance mechanisms. In *in Proc. IEEE INFOCOM*, 2006.
- 6) YungRyn Choe, Derek L. Schuff, Jagadeesh M. Dyaberi, and Vijay S. Pai. Analyzing and improving a bittorrent network's performance mechanisms. *ACM MM '07*, 2007.
- 7) Chris Dana, D. Li, D. Harrison, and C.-N. Chuah. Bass: Bittorrent assisted streaming system for video-on-demand. In *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2005.
- 8) Jian-Guang Luo, Qian Zhang, Yun Tang, and Shi-Qiang Yang. A trace-driven approach to evaluate the scalability of p2p-based video-on-demand service. In *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS*, 2009.
- 9) C. Huang, J. Li, and K. W. Ross. Can internet video-on-demand be profitable. In *Proc. ACM SIGCOMM '07*, 2007.