

映像情報の同期によるネットワーク内一次記憶の削減に関する研究

研究代表者 中里秀則 早稲田大学基幹理工学部情報通信学科 教授

1 はじめに

インターネットの利用目的は、ユーザ間の通信や資源の共有から情報コンテンツの取得に変化してきており、特に数年後にはインターネット上を流れる情報の80%を映像コンテンツが占めることが予想されている[1]。このようなインターネット利用状況の変化の中で、コンテンツ指向ネットワーク(ICN)が注目されている[2]。ICNでは、ネットワーク内のルータでコンテンツを一時的に記憶(キャッシュ)することが可能である。同一コンテンツを何度もコンテンツを保持するサーバから取得する代わりに、そのコンテンツをキャッシュしたルータから配送することにより、ネットワークやサーバの負荷を軽減することが期待されている。ここで、ネットワークやサーバの負荷が有効に軽減されるためには、要求されたコンテンツがルータにキャッシュされているように、キャッシュを制御することが重要である。

ところがビデオコンテンツは、コンテンツ当たりの情報量が多いため、要求されるコンテンツが一時記憶に存在するように制御することが容易ではない。ルータに装備することができるキャッシュ容量には限度があるため、アクセスされるコンテンツのすべてをルータで記憶させることはできない。できるだけ多くの視聴者に対して、キャッシュされたコンテンツから要求を満たせるようにするためには、要求の多いコンテンツに対してより多くのキャッシュを割り当てるとともに、各コンテンツに対して割り当てるキャッシュ量をできるだけ少なくし、より多くのコンテンツをキャッシュできるようにすることが重要である。

そこで、本研究では、同一ビデオコンテンツの視聴者が、近いタイミングでコンテンツデータを要求するように視聴者のビデオ再生速度を調整し、できるだけ短い時間の間に、多くの視聴者に対して同一コンテンツデータの送目が終わるようにすることにより、コンテンツ毎のキャッシュ割当量を削減し、効率的に一時記憶を活用できるようにする手法について検討を行った。本研究の成果は、これから増加が予想されるビデオ配信において、ビデオ配信サーバやネットワークの負荷を軽減し、コストを抑えながら、快適なビデオ視聴をネットワークで提供することに貢献する。

2 ビデオコンテンツのキャッシュ

2-1 制御対象ビデオコンテンツの特定と割り当てキャッシュ量

本研究ではICNの一つの実装であるNamed Data Networking (NDN)を仮定している。NDNでは、Interestパケットでコンテンツを要求し、Dataパケットでコンテンツが要求者に送り届けられる。一般に、ビデオコンテンツ(ビデオタイトル)は、複数のビデオセグメントからなり、このビデオセグメントを順番に連続して取得することによりビデオ再生が行われる。ここでは、NDNによって連続してビデオセグメントを取得するものとし、一つのInterestパケットに対して、Dataパケットで返送されるのはビデオセグメントであるものとする。

NDNでは、コンテンツ名をInterestパケットに記述することによって、コンテンツを要求するが、コンテンツ名は階層的な名前が使われる。あるビデオタイトルは、その階層的な名前のプレフィクスとして指定され、その後、各ビデオセグメントを示すシーケンス番号を付けることによって特定のビデオセグメントを指定する。例えば `ndn://waseda/video/frozen/7` というビデオセグメントの名前では、“`//waseda/video/frozen`”というプレフィクス部分でビデオタイトルが指定され、そのビデオタイトルの中の“7”番目のセグメントということを示している。

NDNのInterestパケットには、コンテンツの要求者を示す情報が含まれておらず、視聴者を特定できない。また、同一のコンテンツに対する複数の要求は、ルータで集約が行われ、ただ一つのInterestパケットのみがサーバに向けて転送される。そのため、何人の視聴者が同一コンテンツを視聴しているのかも、途中のルータでは把握することができない。したがって、視聴者のビデオ再生速度の調整は視聴者を区別することなく、コンテンツ名のプレフィクスによりビデオコンテンツの要求を管理することにより、再生速度を変化させるビデオセグメント群を特定する。

ネットワークの中には多くのビデオタイトルが存在することが予想され、かつそのすべてのタイトルが頻繁に要求される訳ではない。一般に、ビデオタイトルの人気の分布は Zipf 則に沿うといわれている。つまり、少数の人気タイトルに多くの要求が集中することになる。このことから、すべてのビデオタイトルに同じ量のキャッシュを割り当てることは非効率であり、キャッシュが機能しなくなる。そこで、コンテンツのアクセス頻度に応じて、割り当てるキャッシュサイズを変化させることを提案した。各ルータでビデオタイトル i に割り当てるキャッシュサイズは、各ルータでビデオタイトル i の要求頻度 Rec_i を観測し、 Rec_i の全タイトルの要求頻度の和 Rec に占める割合

$$Req_i = Rec_i / Rec \quad (1)$$

に応じてキャッシュを割り当てる。すなわち、 Rcs_i をビデオタイトル i に割り当てられたキャッシュ量とすると、

$$Rcs_i = K \times Req_i \quad (2)$$

である。ここで K は当該ルータにあるキャッシュの総量である。割り当てたキャッシュサイズによって、複数のビデオセグメントを保持することができるビデオタイトルのみが再生速度制御対象のビデオタイトルとなる。

2-2 ビデオコンテンツのキャッシュ方式

上記のように一定量のキャッシュをビデオタイトルに割り当てるために、以下のキャッシュアルゴリズムを適用する。

まず、 Req_i を求めるために、各ルータではビデオタイトル i に属するビデオセグメントに対する Interest パケット数 Rec_i をカウントする。 Rec はすべてのビデオタイトル i に対する Interest パケット数 Rec_i の総合計であるので、 Rec_i をカウントすることにより、式(1)により Req_i を求めることができる。

ルータでは、ビデオタイトル i が占有するキャッシュ量 Rcs_i を管理する。ルータがビデオタイトル i に属するビデオセグメントを含む Data パケットを受け取ったときに、以下の三つの可能性があり、それぞれ以下に示す動作をする。

- $Rcs_i > K \times Req_i$ の場合： この場合、ビデオタイトル i が必要以上にキャッシュを占有していることになる。よって、ビデオタイトル i が占有するキャッシュの内、2ビデオセグメント分の領域を解放し、その内、1セグメント分に受信した Data パケットに含まれるビデオセグメントを格納する。
- $Rcs_i = K \times Req_i$ の場合： この場合、ビデオタイトル i は適切なキャッシュ量を使用していることになる。したがって、ビデオタイトル i が占有するキャッシュの内、1セグメント分を解放し、解放したキャッシュに受信した Data パケットに含まれるビデオセグメントを格納する。
- $Rcs_i < K \times Req_i$ の場合： この場合、ビデオタイトル i が占有しているキャッシュ量は適切な量よりも不足している。そのため、ビデオタイトル i 以外のビデオタイトルが占有するキャッシュの1セグメント分を解放し、受信した Data パケットのコンテンツを格納する。

以上のアルゴリズムにより、ビデオタイトル毎のキャッシュ割り当て量を調整することができる。なお、上記記述の内、どのビデオセグメントを解放するかについては、次節で説明する。

2.3 キャッシュから解放するビデオセグメントの選択方法

既にルータにキャッシュされている、ビデオタイトル i のビデオセグメント j が、将来アクセスされる可能性を示す評価値を E_{ij} とし、ルータはキャッシュされているすべてのビデオセグメントについて E_{ij} を管理する。 E_{ij} は以下のように設定する。

- ビデオタイトル i に属するビデオセグメント j をルータ内のキャッシュに格納するとき、 E_{ij} をその時の現在時刻 t に設定する。
- ルータは、ビデオタイトル i に属するビデオセグメント k ($k > j$)を要求する Interest パケットを受信したとき、 E_{ij} を以下のように更新する。

$$E_{ij} = \max\{E_{ij}, t + (k - j) \times \Delta t\}$$

ここで、 t は Interest パケットを受け取った現在時刻、 Δt は一つのビデオセグメントの再生に要する時間である。

以上で求めた E_{ij} を用いて、前節で説明したアルゴリズムにおける、解放するビデオセグメントの決定を行う。

- $Rcs_i > K \times Req_i$ の場合： 解放する二つのビデオセグメントはビデオタイトル i に属するビデオセグメ

ントの中で、 E_{ij} が最も小さい2つのビデオセグメントである。

- $Rcs_i = K \times Req_i$ の場合： 解放するビデオセグメントはビデオタイトル*i*に属するビデオセグメントの中で、 E_{ij} が最も小さいビデオセグメントである。
- $Rcs_i > K \times Req_i$ の場合： 解放するビデオセグメントはビデオタイトル*i*以外のビデオセグメント*k*の中で、 E_{kj} が最も小さいビデオセグメントである。

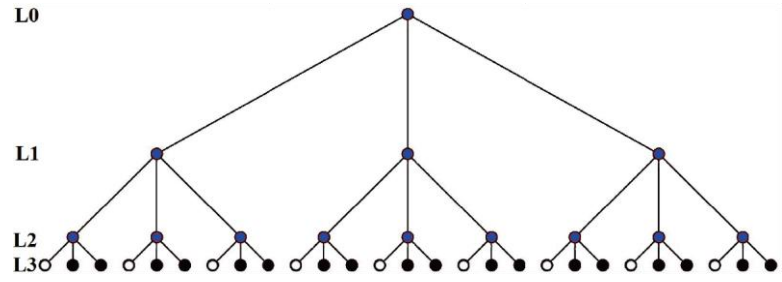


図 1 シミュレーションのトポロジ

2.4 キャッシュ方式の評価

前節で提案したキャッシュ方式(TLP-TTH)についてシミュレーションによる評価を行った。シミュレーションは図 1 のような高さ 4 の完全 3 分木のトポロジのネットワークで行った。ビデオサーバはこの木の葉の部分に均等に 9 台準備をした。図中では白丸のノードで示している。この 9 台のビデオサーバに 25 の異なるビデオタイトルをランダムに配置した。各ビデオファイルは 1,000 のビデオセグメントからなる。視聴者は、ビデオサーバが配置されているノード以外の葉のノードに 100 人いるものとした。視聴者がどのビデオタイトルを視聴するかは、 $\alpha = 1.2$ の Zipf 分布に従うものとした。視聴者は平均到着間隔 1 秒のポアソン到着でビデオの視聴を開始する。視聴者は、ビデオセグメント 1 から順番にビデオタイトルの最後まで視聴

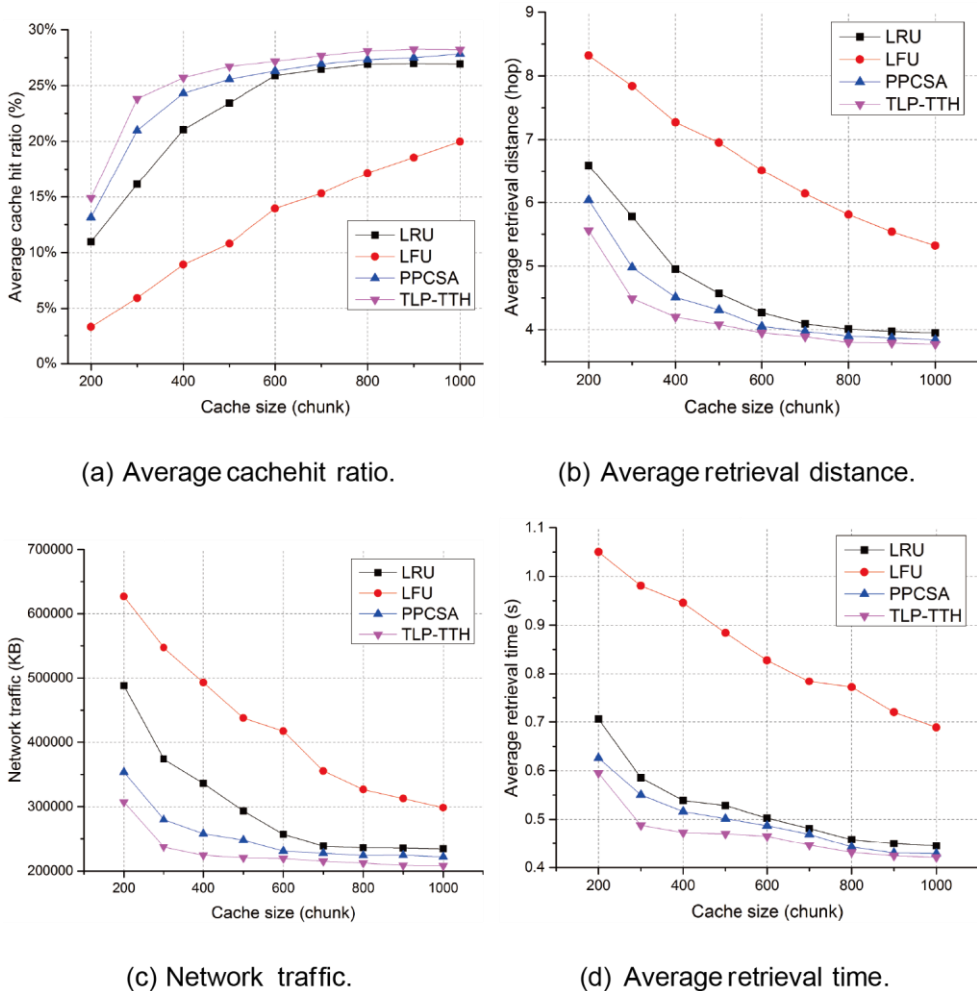


図 2 シミュレーション結果

する。以上の条件でルータのキャッシュサイズを 200 ビデオセグメントから 1,000 ビデオセグメント分まで変化させてシミュレーションを行った。

評価は以下の 4 つの評価基準により行った。

1. 平均キャッシュヒット率
2. 平均コンテンツ取得距離：視聴者からビデオサーバまでの往復ホップ数
3. ネットワークトラヒック：ルータが転送したデータの総量
4. 平均コンテンツ取得時間：平均の Interest パケット送出から Data パケット受信までにかかった時間
また比較対象として、Least Recently Used (LRU)、Least Frequently Used (LFU)、および PPCSA [3] についても評価を行った。

シミュレーション結果は図 2 のようになった。平均ヒット率 (図 2(a)) では、提案の TLP-TTH は、それ以外の中で最もヒット率が高い PPCSA を約 3% 上回っている。一方、PPCSA は最も一般に使われる LRU よりも約 5% ヒット率が高い。したがって、提案の TLP-TTH は一般に使われる LRU よりも 8% ヒット率が向上したことになる。平均コンテンツ取得距離 (図 2(b)) に関しては、TLP-TTH では PPCSA よりも 24% 取得距離が減少している。一般の LRU に比べると約 30% の取得距離の減少となっている。この結果から、TLP-TTH はキャッシュをより有効に活用しており、視聴者は、より近いルータからビデオセグメントを取得できているといえる。

TLP-TTH は、ネットワークトラヒックや平均コンテンツ取得時間に関しても、他の比較対象アルゴリズムに比べて減少することができている (図 2(c) と (d))。ネットワークトラヒックについて、TLP-TTH は PPCSA に比べて 10%、LRU に比べて 24% それぞれ減少させることに成功している。平均コンテンツ取得時間においては、TLP-TTH は PPCSA に比べて 4% 以上、LRU に比べて 10% 以上、それぞれ減少させている。

高い平均キャッシュヒット率により平均コンテンツ取得距離が減少し、平均コンテンツ取得距離の抑制がルータ間の不必要なネットワークトラヒックを減少させ、また平均コンテンツ取得時間も抑制することができた。結論として、特にルータのキャッシュ容量が限られている場合、TLP-TTH は消費するネットワーク資源を抑え、またコンテンツ取得にかかる時間も減少させることができることが示された。

図 3 は図 1 のネットワークトポロジにおけるコアネットワーク、アクセスネットワークそれぞれでの TLP-TTH のパフォーマンスを示している。ここで、コアネットワークは図 1 の L0 と L1、アクセスネットワークは図 1 の L2 と L3 とした。図 3 から、TLP-TTH はコアネットワークとアクセスネットワークの両方で平均キャッシュヒット率が上昇している。特に、PPCSA がコアネットワークで LRU に比べてヒット率が減少しているのと対照的に、TLP-TTH ではコアネットワークに於いても LRU に比べて平均キャッシュヒット率を上昇させることに成功している。

ここでは図 1 の完全三分木のトポロジについてのみ評価結果を照会したが、他のトポロジについてもほぼ同様の結果を確認している。

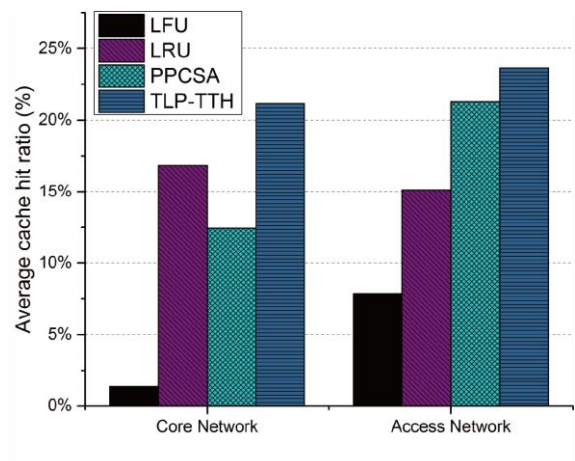


図 3 異なるネットワークドメインでの平均キャッシュヒット率

3. ビデオ再生周期の調整アルゴリズム

3.1 調整の概要

ビデオコンテンツを視聴する場合、視聴者は連続したビデオセグメントを Interest パケットにより要求し、その連続したセグメントが Data パケットとして返送され、ルータでキャッシュされることになる。何人かの視聴者が、類似した時刻に視聴を開始したとすると、最初にアクセスした視聴者の要求によりキャッシュされたビデオセグメントが、できるだけ多くの後続の視聴者によって利用されるのが好ましいが、一方で、他のビデオタイトルとキャッシュを分け合って使うため、一つのビデオタイトルが占有するキャッシュ量は、前節で説明したように限定される。例えば、図 4 はビデオ 1 の連続したセグメントがキャッシュに一時保存されている状態を示している。このキャッシュされたビデオセグメントを視聴する視聴者がグループを構成し、このグループのメンバがキャッシュされたビデオセグメントを共用する。このグループの中では、最初に視聴し始めた視聴者は、すでにセグメント 9 を見ており、そのグループの中で最後に見始めた視聴者はセグメント 1 を見ている。このように、キャッシュにビデオセグメントを保存することにより、このグループの視聴者に対して、サーバへのアクセスは一度だけすれば済むことになり、サーバ負荷およびネットワーク負荷を低減できるとともに、視聴者にとっても応答よくビデオを見ることができるようになる。

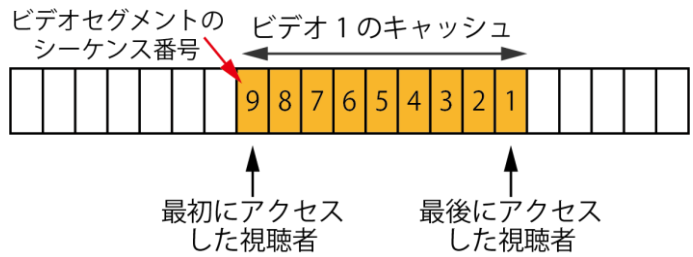


図 4 使用中のキャッシュ状態

時間の経過とともに、最初にアクセスした視聴者は、さらにビデオセグメントを要求し、新たに到着したビデオセグメントをキャッシュするために、前節で説明したキャッシュアルゴリズムにより、キャッシュされたビデオセグメントは図 5 のように更新される。

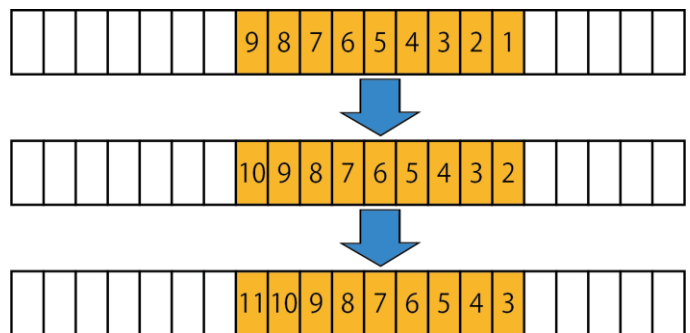


図 5 キャッシュ状態の遷移

ここで、最初にアクセスした視聴者のアクセス周期を長く、最後にアクセスした視聴者のアクセス周期を短くするように操作をすると、視聴者のアクセスタイミングが近寄っていくことになり (図 6)、結果として、この視聴者グループに対して使用するキャッシュを削減できることになる。

ビデオ再生において、視聴者が感知できない程度に再生速度を変更することができるとされている。再生速度はビデオセグメントの要求周期とみることができ、要求周期を、視聴者が感知できない範囲内で変更することによってキャッシュ量の削減を行う。

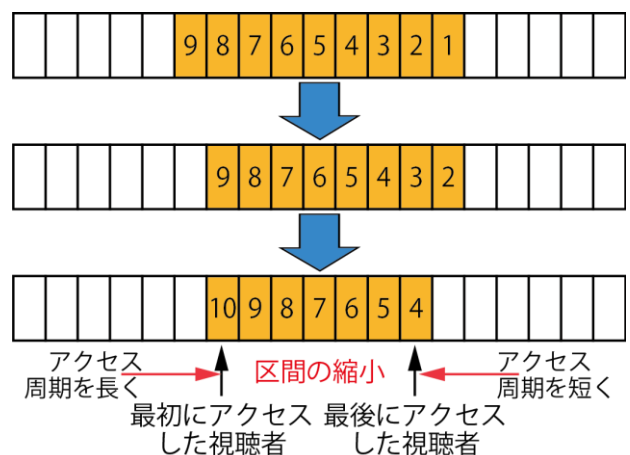


図 6 キャッシュの縮小

再生速度の変更を続けると、最終的にはグループ内のすべての視聴者が同期して同じビデオセグメントを要求するようになるため、キャッシュ量が 0 となり、この視聴者グループに対してのキャッシュは行われなくなる。この状態のとき、このグループに属する視聴者からの Interest パケットは、途中の NDN ルータによって集約され、一つの Interest パケットのみがビデオサーバに配送される。ビデオサーバから送信された Data パケットは、NDN ルータの機能により途中でコピーされ、要求しているすべての視聴者に同一の Data パケットが配送される。

3.2 調整アルゴリズム

まず、ビデオタイトル*i*のビデオセグメントを要求する視聴者の中から、 $S_{min} \leq S \leq S_{max}$ を満足するシーケンス番号*S*のビデオセグメントを要求する視聴者によりグループを構成する。 S_{min} と S_{max} は以下で与えられる。

$$S_{max} = S_0 + \frac{t - t_0}{t_f}$$

$$S_{min} = S_{max} - Rcs_i$$

ここで、 t_0 はビデオタイトル*i*の最初のビデオセグメントを要求する Interest パケットの到着時刻、 S_0 はその時に要求されたビデオセグメントの番号、 t は現在時刻、 t_f はビデオセグメントの要求周期とする。また、 Rcs_i は2節で説明した、ビデオタイトル*i*に割り当てられたキャッシュ量である。

$S_{min} \leq S < (S_{min} + S_{max})/2$ を満たすビデオセグメント*S*の要求に対しては、要求周期を $0.95 \times t_f$ してビデオ再生を早め、一方、 $(S_{min} + S_{max})/2 < S \leq S_{max}$ を満たすビデオセグメント*S*の要求に対しては、要求周期を $1.05 \times t_f$ として再生を遅らせる。これにより、キャッシュするビデオセグメントの範囲を抑制する。

$S_{min} \leq S \leq S_{max}$ の範囲外のシーケンス番号*S*のビデオセグメントを要求する視聴者については、別のグループを構成することになる。

3.3 調整アルゴリズムの評価実験

ビデオ再生周期調整アルゴリズムについてのシミュレーションによる評価実験を行った。

無限長のコンテンツをセグメントに切り分け、0から順番にシーケンス番号を割り当てる。視聴者はアクセスを開始するとシーケンス番号0のセグメントから順にビデオセグメントを要求する。1パケットには一つのビデオセグメントが含まれる。簡単化のためビデオセグメントの大きさは一定とする。

視聴者のビデオセグメント要求周期は、前節のアルゴリズムにしたがって、通常の95%、100%、105%の3段階に切替られる。この実験では、それぞれ95、100、105 Interest パケット/sの周期で Interest パケットを送信する。(それぞれ、低速モード、等速モード、高速モードと呼ぶ。) 視聴者は Data パケットを受け取ると、Data パケットに付与された周期変更の指示を読み取り、ビデオ再生周期、すなわち Interest パケットの発生周期を変更する。また、この実験ではルータのキャッシュ容量は100ビデオセグメント分とした。

図7のようにルータを介して二人の視聴者が同一ビデオタイトル視聴のためにビデオサーバにアクセスする構成で実験を行った。視聴者-ルータ間、ルータ-ビデオサーバ間の遅延はそれぞれ20msとした。視聴者1が Interest パケットを送信し始めてから*g*秒後に視聴者2が Interest パケットを送信し始める。両視聴者が Interest パケットを送信する間隔はともに通常時100パケット/sである。またグループを決定する Rcs_i については、ここ

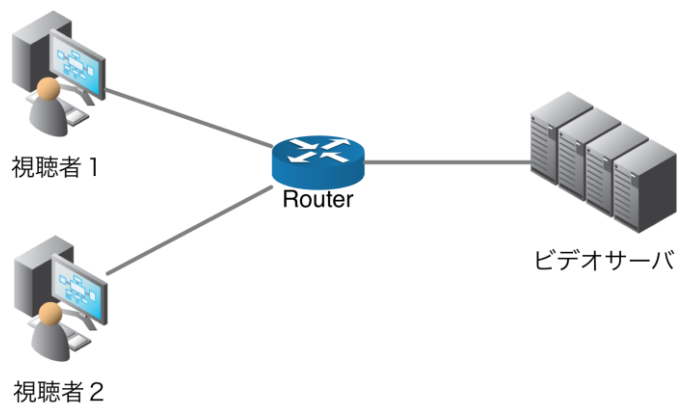


図7 評価実験

それぞれ、低速モード、等速モード、高速モードと呼ぶ。) 視聴者は Data パケットを受け取ると、Data パケットに付与された周期変更の指示を読み取り、ビデオ再生周期、すなわち Interest パケットの発生周期を変更する。また、この実験ではルータのキャッシュ容量は100ビデオセグメント分とした。

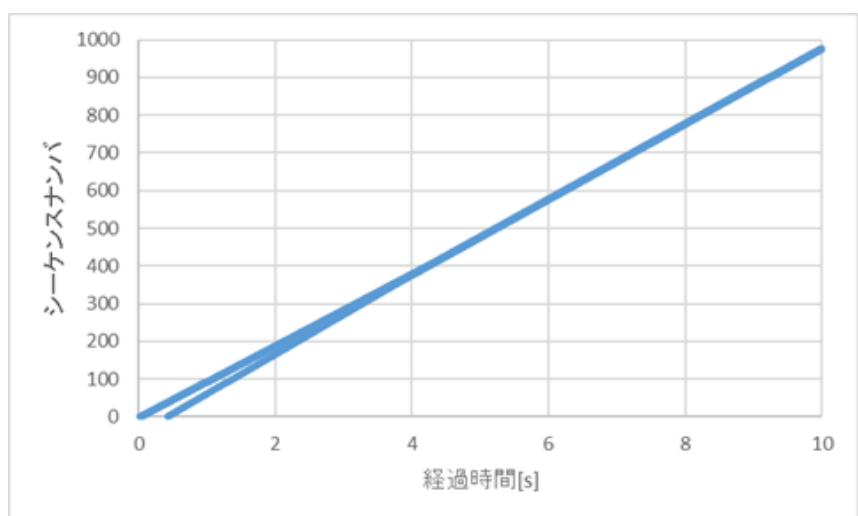


図8 $g = 0.4$ のときの経過時間と参照シーケンス番号

では 45 として実験を行った。

視聴者間のアクセス開始時間差 g が 0.4 秒の場合の参照シーケンスナンバと経過時間の分布は図 8 のようになった。視聴者 1 は最初に低速モードとなり、4.2603 秒で等速モードとなった。視聴者 2 は最初に高速モードとなり、3.7000 秒で等速モードとなった。二人の視聴者が共に等速モードになるとルータのデータ送信タイミングは視聴者 1 への Data パケットが先に返送されるが、視聴者 1 と 2 の間で 0.0066 秒の差を保ちながら送信が続く。視聴者 1 への Data パケットのホップ数は 2、視聴者 2 への Data パケットのホップ数は 1 となった。

視聴者間のアクセス開始時間差 g が 0.5 秒の場合の参照シーケンスナンバと経過時間の分布は図 9 のようになった。視聴者 1、視聴者 2 共に、最初は低速モードとなった。これにより視聴者 1 と視聴者 2 が別々のグループに分類されたことがわかる。視聴者 1 は 4.2603 秒、視聴者 2 は 5.8113 秒でそれぞれ等速モードとなった。二人の視聴者が等速モードになると、視聴者 1 と視聴者 2 の Data パケットの送信間隔は 0.551 秒を保ちながら送信が継続される。Data パケットのホップ数はどちらの視聴者に対しても 2 となった。

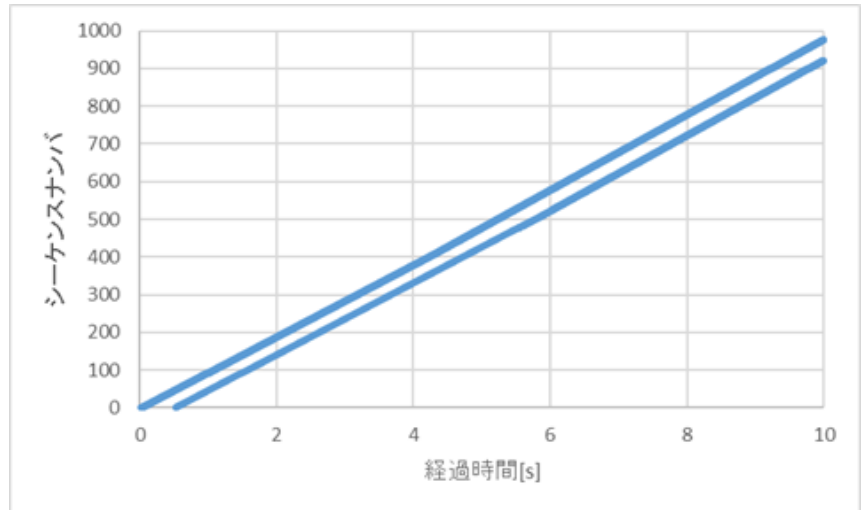


図 9 $g = 0.5$ のときの経過時間と参照シーケンス番号

二人の視聴者が同じグループに分類された場合は、アクセス間隔を揃えることができ、ルータの使用中のキャッシュを削減することができた。また、アクセス間隔が長くルータキャッシュが既に削除されている場合は、別グループとしアクセスタイミングを揃えることはなかった。

4. ビデオ周期変更指示の仕組み

前節で提案したビデオ再生周期調整アルゴリズムは、各ルータで実装するアルゴリズムであるが、実際にビデオ再生周期を調整するのは視聴者の端末であり、周期変更の指示をルータから視聴者端末まで伝達し、視聴者端末でビデオ再生やビデオセグメント要求周期を調整する必要がある。このとき、視聴者端末での対応の仕方により二つの場合が考えられる。一つは、視聴者端末に何も手を加えず、通常の視聴者端末でビデオ再生周期を変更する方法であり、もう一つは、視聴者端末に専用アプリケーションを準備し、ルータからの指示を判別し、ビデオ再生とビデオセグメント要求周期をアプリケーションが仕組みの一部となって変更する方法である。

4.1 視聴者端末に手を加えない場合

視聴者端末に手を加えずにビデオセグメント要求周期を変更させる仕組みとして、ビデオ再生速度を変化させることが考えられる。再生速度を速くすればセグメント要求周期は短くなり、遅くすればビデオセグメント要求周期が長くなる。ただし、この場合の再生速度の変更は、視聴者端末が自律的に変更するのではなく、送られて来る情報を基に変更されるようにしなければならないため、通常の、例えば Real Time Streaming Protocol を使ったビデオ再生の制御のような手法は使うことができない。

ここでは、一つの可能性として、ビデオ情報に手を加えることによってビデオ再生のアプリケーションを騙し、再生速度を変更させる方法について検討した。

ビデオ情報としては、動画ファイルの mp4 フォーマットを仮定した。視聴者からビデオサーバにビデオセグメントを要求する Interest パケットが送られ、ビデオサーバがルータを介して視聴者にビデオセグメントを Data パケットとして配信する。その際に、通過する途中のルータで Data パケットのペイロード部分を書き換えることにより、再生速度をルータで操作するシステムを実現する。

mp4のフォーマットでは、atomと呼ばれる情報の単位が階層的に組み合わされた構造になっていて、この中の幾つかのatomに再生速度に係わる情報が記述される。具体的には、まずビデオセグメントの再生時間を変更するためにmovie header atomのDurationを再生周期の調整に合わせて変更する。ビデオの再生時間および再生速度を変更するためにビデオトラックのtrack atomに含まれるedit list atomのTrack durationおよびMedia rate、オーディオの再生時間および再生速度を変更するためにオーディオトラックのedit list atomのTrack durationおよびMedia rateを変更する。以上の情報の変更により、ビデオ再生速度を変化できることを確認した。その結果、Interestパケットの送出間隔が変化することを期待することができる。

ただし、この手法によるビデオ再生速度の変化は、ビデオ再生ソフトウェアに依存する部分があり、すべての視聴者端末に対応できる訳ではないこともわかった。また、この手法によって視聴者端末に再生速度の調整を指示した場合、複数のルータによってビデオ再生周期調整アルゴリズムが適用されると、調整が累積し、再生速度が早くあるいは遅くなりすぎてしまう欠点があり、ルータ間で適用を調整する必要がある。

4.2 専用アプリケーションを視聴者端末に用意する場合

この場合、Dataパケットに、3.2節で提案したビデオ再生周期調整アルゴリズムでの判定結果を指示するためのフィールドを設けて、視聴者端末に再生周期の調整を指示することができる。ビデオ再生周期調整アルゴリズムによる制御は、低速モード、等速モード、高速モードの3値なので、調整を指示する再生速度調整フィールドは2ビットあれば十分である。この調整フィールドの指示を受けて、視聴者端末では、0%および±5%の固定の変動を再生速度およびビデオセグメント要求周期に与える。

この方式の場合、再生周期の調整は特定のフィールドで示されるため、複数ルータによる調整アルゴリズムの適用を阻止することが可能である。複数ルータを経由する経路によってビデオセグメントを要求するInterestパケットが配送される場合、各ルータではビデオ再生周期調整アルゴリズムが実行され、ビデオセグメントのシーケンス番号により視聴者からの要求がグルーピングされる。グループ内のシーケンス番号の位置により、ビデオ再生周期の調整が判断されるが、固定のフィールドで指示がなされ、また固定の変動幅のみ許容されているため、過度の周期の調整は起こらない。また、ビデオ再生周期調整アルゴリズムの判定結果をDataパケットに書き込もうとしたときに、もし上流のルータで既にビデオ再生周期調整アルゴリズムが適用されていると、再生周期調整フィールドに値が設定されており、上流で調整が行われていることをルータで検知することができる。その検知の結果として、そのルータでの当該ビデオタイトルのキャッシュを停止し、キャッシュを他のビデオタイトルに振り分けることが可能になる。また、この手法によって、上流ルータから順次キャッシュの削減を行い、ネットワーク全体としてのさらなるキャッシュの有効活用が可能になる。

5. まとめ

本研究では、ネットワーク内のルータに一時記憶（キャッシュ）を装備したネットワーク、特にコンテンツ指向ネットワークの一実装であるNamed Data Networkingにおいて、複数の視聴者が同一ビデオコンテンツをアクセスする場合の、キャッシュの有効利用方法について検討を加えた。

まず、ビデオタイトル毎の人気度（アクセス頻度）を基にしたキャッシュサイズの割当方法について検討し、TLP-TTHというキャッシュアルゴリズムを提案した。TLP-TTHではビデオタイトルにそのアクセス頻度に応じて一定のキャッシュサイズを割り当てる。ビデオのような連続したコンテンツ取得が行われる場合に、TLP-TTHが他のキャッシュ方式より有効に限られたサイズのキャッシュを活用できることを示した。

次に、TLP-TTHが決定するビデオタイトル毎のキャッシュサイズを基準として、同一ビデオタイトルにアクセスする視聴者のビデオ再生周期を変動させることによって必要とするキャッシュ量を抑制する方式について検討した。TLP-TTHによって決まるキャッシュサイズにより視聴者をグループ化し、グループ内で再生周期を調整し、その割当サイズを縮小する方式を提案した。この方式により、視聴時間の経過とともに使用するキャッシュ量を減少させ、最終的にはコンテンツ指向ネットワークが提供するマルチキャスト機能により、キャッシュを使わずに複数視聴者にサービスできるようになることを確認した。

また、再生周期の視聴者端末への指定方法についても、ビデオアプリケーションの変更を必要としない方式と、専用ビデオアプリケーションを使用する場合について検討した。複数ルータで再生周期調整アルゴリズムを実行した場合、ビデオアプリケーションを変更しない手法の場合、再生周期の変動が大きくなり、問

題が発生する。一方、専用アプリケーションにより、再生周期を固定量変動させる方式では、複数ルータで再生周期調整アルゴリズムを実施しても、過大な再生周期の調整にならないようにすることができるとともに、さらなるキャッシュ利用の効率化の可能性を示した。

本研究の成果は、これから増加が予想されるビデオ配信において、ビデオ配信サーバやネットワークの負荷を軽減し、コストを抑えながら、快適なビデオ視聴をネットワークで提供することに貢献する。

【参考文献】

- [1] Cisco Systems. Cisco visual networking index: Forecast and trends, 2017–2022. [Online]. Available <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.pdf>, 2019. Accessed: 2019-6-23.
- [2] 山本幹. コンテンツオリエンテッドネットワーク. 電子情報通信学会論文誌 B, Vol. J96-B, No. 6, pp. 589–604, 2013.
- [3] H. Li, H. Nakazato, A. Detti, and N.B. Melazzi, “Popularity proportional cache size allocation policy for video delivery on CCN,” Proc. 2015 European Conference on Networks and Communications (EuCNC), pp.434–438, 2015.

〈発表資料〉

題 名	掲載誌・学会名等	発表年月
LCD-Based Probabilistic Caching for Information Centric Networking	電子情報通信学会通信方式研究会	2017年8月
Research on content distribution over ICN	電子情報通信学会総合大会	2017年9月
Request Expectation Index Based Cache Replacement Algorithm for Streaming Content Delivery over ICN	Future Internet	2017年11月
ルータによる動画ファイル再生速度制御方式	電子情報通信学会総合大会	2018年3月
キャッシュ削減手法の提案	電子情報通信学会総合大会	2019年3月
NDN におけるネットワーク遅延とキャッシュ容量に基づくキャッシュ方式.	電子情報通信学会総合大会	2019年3月