

推薦論文

強化学習を用いたMPEG-DASHにおける 映像品質制御手法の実験と評価

生出 真人^{1,a)} 阿部 亨^{1,2,b)} 菅沼 拓夫^{1,2,c)}

受付日 2019年9月5日, 採録日 2020年3月13日

概要: 映像をリアルタイムに配信するライブストリーミングサービスは、様々な特性の映像を対象に、多様な状況のネットワークを通じて、多種の異なった性能の受信端末を用いて、幅広く利用されている。既存のライブストリーミングサービスでは、主に利用可能なネットワーク資源の状況に基づいて受信端末へ配信する映像の品質を制御しているため、配信映像の品質が、利用している受信端末の計算機資源に適したものであるとは限らない。そこで本研究では、映像の特性、利用可能なネットワーク資源、計算機資源と、それらに応じて受信端末でサービス利用者へ配信される映像品質の関係を強化学習することで、ネットワークの状況だけでなく受信端末の性能にも適応した品質の映像を配信するための映像品質制御手法を提案する。本稿では、MPEG-DASHを対象とした映像品質制御手法の設計と実装を行い、実機を用いた実験から、提案手法を用いることで各資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であることを確認した。

キーワード: ライブストリーミングサービス, MPEG-DASH, 携帯端末, 映像品質制御, 強化学習

Experiment and Evaluation of Video Quality Control Method for MPEG-DASH Using Reinforcement Learning

MAKOTO OIDE^{1,a)} TORU ABE^{1,2,b)} TAKUO SUGANUMA^{1,2,c)}

Received: September 5, 2019, Accepted: March 13, 2020

Abstract: Live streaming services that deliver video in real-time are widely used on networks with various conditions and receivers with different performances, for videos with variety of characteristics. Since the existing live streaming service controls the video quality delivered to the user device considering available network resource status, the quality of the delivered video is not always adopted to the receiver's computational resource status. This study propose a reinforcement learning scheme to learn the relationship between video characteristics, available network resources, computational resources, and video quality delivered to the users. This scheme can provide adopted video quality in accordance with not only the network status but also the performance of the user device. In this paper, we show design and implementation of a video quality control method for MPEG-DASH. Also some experimental results show that our proposed method can provide high-quality video to users while reducing wasteful consumption of resources.

Keywords: live streaming service, MPEG-DASH, mobile device, video quality control, reinforcement learning

¹ 東北大学大学院情報科学研究科
Graduate School of Information Sciences, Tohoku University,
Sendai, Miyagi 980-8579, Japan

² 東北大学サイバーサイエンスセンター
Cyberscience Center, Tohoku University, Sendai, Miyagi
980-8578, Japan

a) medio@ci.cc.tohoku.ac.jp

b) beto@tohoku.ac.jp

c) suganuma@tohoku.ac.jp

1. はじめに

情報端末の高性能化と無線ネットワーク網の発展とともに、インターネットを用いた様々なサービスが種々の環

本稿の内容は2018年11月の第26回マルチメディア通信と分散処理ワークショップで報告され、同プログラム委員長により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である。

境で利用されるようになった。近年、サービスの高機能化により、個々のサービスで生じるネットワークトラフィックが増加したことに加え、サービスを手軽に利用可能な端末の台数が急増したため、インターネット上のトラフィックが爆発的に増加している [1], [2]。この結果、ネットワーク回線の混雑により、利用者に提供されるサービスの品質低下が懸念されており、各サービス利用者に対し、用途に応じた必要十分なレベルでネットワーク資源を割り当てる手法が求められている [3]。

現在、ネットワークトラフィックの大多数は映像配信によるもので占められている。特に、映像をリアルタイムに配信するライブストリーミングサービスでは、地上波の放送映像と 4K 映像を切り換えての視聴が可能な配信 [4] や、スポーツや音楽のイベントを VR 映像として視聴可能な配信 [5] 等、トラフィックの増加をとまなうサービスの高機能化が進められている。スマートフォンやタブレット端末等の携帯端末の急速な普及と相まって、ライブストリーミングサービスによるトラフィックは 2016 年から 2021 年の 5 年間で 5 倍に増加すると予測されている [2]。そこで本研究では、インターネット上のトラフィック量に大きな影響を与えるライブストリーミングサービスを対象としたネットワーク資源の効果的な割り当て手法に着目する。

映像をリアルタイムに配信するライブストリーミングサービスの実現手法として、適応型ライブストリーミングが多く採用されている [6], [7], [8]。本研究では、適応型ライブストリーミングサービスの実現手法として、Moving Picture Experts Group (MPEG) が標準化した Dynamic Adaptive Streaming over HTTP (MPEG-DASH) [9] を対象とする。MPEG-DASH は、送信端末が送信する映像をエンコーダが圧縮符号化してセグメントファイルに分割 (セグメント化) して受信端末に配信する。受信端末は、セグメントの受信に要した時間から利用可能なネットワーク資源を推定することにより、配信サーバに対して、ネットワーク資源の変動に対応した映像品質のセグメントを要求する。

このように、MPEG-DASH はネットワークの状況、すなわち利用可能なネットワーク資源を考慮した品質制御手法である [9], [10]。そのため、各々の端末性能に応じた適切な映像品質を提供し、サービス利用者が満足するサービス品質を提供することは困難である。これに対し既存研究では、映像の特性や再生バッファの占有率を考慮することで、サービス利用者の体感品質を向上させる方法 [11], [12], [13], [14] や、利用者が体感する品質である Quality Of Experience (QoE) を改善するため機械学習に基づく品質制御手法を採用する方法 [15], [16] 等が提案されている。しかし、これらの手法では配信された高品質な映像の復号・再生に起因する計算機資源への影響は十分に考慮されておらず、受信端末の性能、すなわち利用可能な計算機資源が乏しい端末

での高品質なサービス提供は困難である。

そこで本研究では、ネットワークの状況および受信端末の性能に適応したライブストリーミングサービスの実現を目的とする。しかし、ライブストリーミングサービスは種々の環境で利用され、使用する端末や配信映像の特性も様々であるため、ネットワークおよび受信端末の状態を事前に把握することは困難である。したがって、あらかじめ定義した状態に対して映像品質を決定するルール型推論のような品質制御手法では、受信端末の性能・状況に適応した適切な映像品質を決定できるとは限らない。

本研究ではこの課題を解決するため、強化学習を用いた映像品質制御手法を提案する。具体的には、映像の特性、利用可能なネットワーク資源、計算機資源と、それに応じた利用者へ配信される映像品質の関係を強化学習することで、ネットワークの状況だけでなく受信端末の性能にも適応した映像品質を決定する資源適応型映像品質制御機構を提案する。本稿では、本制御機構の提案と設計について述べる。また、実装したプロトタイプを用いた実験から、提案手法を用いることで資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であることを確認する。

本稿の構成は以下のとおりである。2 章では、MPEG-DASH の特性、既存研究および本研究で解決する課題を説明する。3 章では、課題を解決するための、ネットワークの状況と受信端末の性能に適応した映像品質を決定する資源適応型映像品質制御機構を提案する。4 章では、提案機構を実現するための手法を設計する。5 章では、プロトタイプシステムを用いて行った実験から得られた結果より、提案手法の有効性を議論する。6 章では、本稿をまとめ、今後の課題を明らかにする。

2. 関連研究と課題

2.1 関連研究

MPEG-DASH における映像品質制御では、セグメントのデータ量と転送時間から、利用可能な帯域を受信端末が推定し、その推定値に適応した品質 (ビットレート) のセグメントを配信サーバに対して要求することが一般的である [10]。

帯域以外のネットワーク資源を考慮した手法として、再生バッファの占有率に基づいた手法がある [11], [12]。これらの手法では、再生バッファに下限と上限の閾値を設定し、下限の閾値を下回った場合は映像の品質を劣化させ、上限の閾値を越えた場合は映像の品質を向上させる。これにより、再生バッファが枯渇することで生じる配信映像の再生中断回数を削減している。

セグメントの特性を考慮した手法として、文献 [13], [14] では、映像が可変ビットレートで圧縮符号化されていることに着目している。利用可能な帯域と各セグメントのビットレートに基づいたネットワークの使用帯域を予測するこ

とで、帯域のみを考慮した場合と比較して高精度な予測が可能となり、より高品質なセグメントを受信端末に提供している。

機械学習を用いた手法では、ネットワーク資源とセグメントの特性を考慮した学習モデルを構築する [15], [16]. 受信端末の状態とセグメントの品質の関係性を学習することで、受信端末の状態に応じた適切な品質のセグメントを提供している。これにより、利用者が体感する映像品質 (QoE) を改善している。

2.2 課題

以上のように既存手法では、ネットワーク資源 (帯域, バッファ占有率) やセグメントの特性 (ビットレート) を考慮することで、サービス利用者の QoE を向上させた。しかし、計算機資源が考慮されていないため、ネットワーク資源に余裕がある環境においても映像の再生が中断してしまう可能性がある。高精細・大容量な映像配信のためには、映像の符号化効率を高めるための映像圧縮手法が用いられるため、これらの復号には、高い処理性能を備えた計算資源が必要となる。そのため、ライブストリーミングサービスにおいてネットワーク資源に加え計算機資源を考慮した映像品質制御手法が求められる。

受信端末となるスマートフォンやタブレット端末等の携帯端末が備えることが可能な計算機資源およびネットワーク資源 (以降、資源) には制限があり、各端末で利用可能な資源量は様々である。さらに、配信映像はその用途により映像の特性や求められる品質が異なるため、これらすべてのあらゆる状況に適応した映像品質制御の実現が不可欠となる。ルール型推論による映像品質制御では、あらかじめ利用可能な資源に対応した映像品質を定義する必要があるため、映像の特性とあらゆる資源の変動に適応した品質を決定できるとは限らない。

そのため、資源およびセグメントの特性を包括的に考慮するためには、機械学習を使用したサービス品質制御手法が効果的である。しかし、機械学習を使用した既存の研究では、学習済みのモデルを使用するため、受信端末が多様化した場合および配信映像の種類や環境の変化にともなうセグメントの特性に変化が生じた場合に初めから学習をやり直す必要がある。そのため、多様な受信端末の状況・配信環境に適用できない可能性がある。

3. 強化学習を用いた資源適応型映像品質制御機構の提案

3.1 提案の概要

前章で述べた課題を解決するため本研究では、サービス利用時のネットワークの状況 (利用可能なネットワーク資源) および受信端末の性能 (利用可能な計算機資源) に適応した映像品質を決定・制御するための映像品質制御手法

を提案する。具体的には、強化学習を用いた資源適応型映像品質制御機構を提案し、課題の解決を図る。

資源適応型映像品質制御機構 (以降、品質制御機構) は、映像の特性、利用可能なネットワーク資源、計算機資源と、利用者へ配信される映像品質の関係を強化学習することで、ネットワークの状況だけでなく受信端末の性能にも適応した映像品質を決定する。教師データが不要な強化学習を用いることで、新たな映像の特性や、端末性能を持つ受信端末が追加された場合においても逐次的に学習し、あらゆる状況に適応した映像品質が決定可能となる。また、一般的な MPEG-DASH とは異なり、品質制御機構を配信サーバに設置することで、多数の受信端末の状態の把握が可能となり、より効率的な学習が可能となる。なお、本研究では受信端末の利用者に対して高品質なサービス提供を行うため、送信端末および配信サーバの負荷は考慮しないものとする。

3.2 品質制御機構の構成

図 1 に品質制御機構の構成を示す。品質制御機構は、映像の特性を解析するデータ解析機構、映像の特性からセグメントの長さを決定するセグメント長決定機構、受信端末で利用可能な資源に適応した映像品質を選択する資源学習機構から構成される。各機構の概要を以下に示す。

データ解析機構

送信端末が送信する映像を入力として、その特性を解析する。具体的には、映像を構成する複数フレーム単位でビットレート等の特性を解析する。これにより、映像の特性を考慮した映像品質を決定する。

セグメント長決定機構

データ解析機構で解析した映像の特性を入力として、セグメントの長さ (セグメント長) を決定する。セグメント長は映像再生時の受信端末の資源に影響を与えるため [17], [18], セグメント長を制御することにより、受信端末で利用可能な資源を考慮した映像品質を決定する。

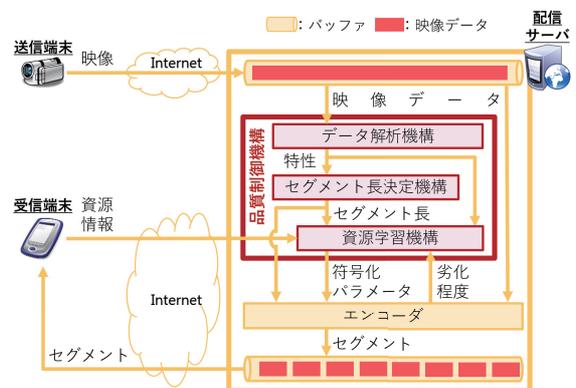


図 1 品質制御機構の構成

Fig. 1 Overview of our proposal.

資源学習機構

資源学習機構は、データ解析機構で解析した映像の特性とセグメント長決定機構で決定したセグメント長、受信端末が送信する資源情報、エンコーダから獲得する圧縮符号化後の映像品質の劣化程度を入力として強化学習し、エンコーダが映像を圧縮符号化する際に設定する符号化パラメータを選択する。これにより、映像の特性、ネットワークの状況、受信端末の性能、圧縮符号化後の映像品質の劣化程度を総合的に考慮し、資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信するための映像品質を決定する。

本稿では、資源学習機構に焦点を当てるため、次章では資源学習機構の詳細設計について述べる。

4. 設計

4.1 資源学習機構の設計

資源学習機構は、データ解析機構で判定した映像の特性 $char$ とセグメント長決定機構で決定したセグメント長 len_{seg} 、受信端末の資源情報 R 、圧縮符号化後の映像の劣化程度 D に基づき、強化学習 [19] を用いて符号化パラメータ a を導出する。本稿では、 a は映像解像度、フレームレート、画質で構成されるパラメータの集合とする。強化学習の枠組みに基づく資源学習機構の動作設計を以下に示す。なお、学習を行う主体をエージェント (Ag) と呼ぶ。

(step 1) 受信端末が送信する計算機資源集合 R_{CP} 、ネットワーク資源集合 R_{NW} 、データ解析機構とセグメント長決定機構で解析、決定した映像の性質集合 $C = \{char, len_{seg}\}$ から、Ag は受信端末の状態 s を観測する。 s の全体集合 S を以下に定義する。

$$S = \{s_i \mid s_i = \langle R_{CP_i}, R_{NW_i}, C_i \rangle; i = 1, 2, \dots, N_i\} \quad (1)$$

ここで、 N_i は受信端末の総数である。また、計算機資源集合 R_{CP} は以下のとおりである。

$$R_{CP} = \{cp \mid cp = \langle cpu, mem \rangle\} \quad (2)$$

ここで、 $cpu \in CPU$ は CPU 使用率、 $mem \in MEM$ はメモリ使用率である。符号化された映像の復号に必要な資源として CPU 使用率とメモリ使用率を考慮する。

ネットワーク資源集合 R_{NW} は以下とする。

$$R_{NW} = \{nw \mid nw = \langle buf, bwt \rangle\} \quad (3)$$

ここで、 $buf \in BUF$ は再生バッファの占有率、 bwt は受信端末の使用帯域である。再生バッファの占有率と使用帯域からネットワーク資源を推定する。これにより、利用可能なネットワーク資源の急激な変動に関して、映像の品質制御により対処すべきかどうかを判断する。

(step 2) Ag は、観測した受信端末の状態 s から、受信

Algorithm 1 Resource Learner based on Q-learning

- 1: Initialize Q values $Q(s, a)$ for all state-action pairs to 0.0.
- 2: **repeat**
- 3: Get resources ($R_{CP_i,t}$ and $R_{NW_i,t}$) from the Receiver i at time t .
- 4: Observe a state $s_{i,t} \in S$.
- 5: Determine an encoding parameter $a_t \in A$ according to the policy π .
- 6: Send a_t to the Encoder.
- 7: Get resources ($\Delta R_{CP_i,t+\tau}$ and $\Delta R_{NW_i,t+\tau}$) from the Receiver at time $t + \tau$.
- 8: Get a quality degradation $\Delta D_{i,t+\tau}$ from the Encoder.
- 9: Calculate a reward $r_{i,t}$ according to Eq. (5).
- 10: Observe the next state $s_{i,t+\tau}$.
- 11: Update the Q value $Q(s_{i,t}, a_t)$ according to Eq. (6).
- 12: Transition to next time $t \leftarrow t + \tau$.
- 13: **until** the streaming service is finished.

端末の消費資源量を抑え、かつ、サービス利用者の体感品質が向上するような符号化パラメータ a を方策 π に基づき選択する。 a の全体集合 A を以下に定義する。

$$A = \{a_j \mid a_j = \langle rsl, fps, crf \rangle; j = 1, 2, \dots, N_j\} \quad (4)$$

ここで、 $rsl \in RSL$ は映像解像度、 $fps \in FPS$ はフレームレート、 $crf \in CRF$ は映像を圧縮符号化する際に指定する画質の程度 Constant Rate Factor (CRF) である。また、 N_j は符号化パラメータの総数である。

(step 3) step 2 で決定した a に基づき受信端末の状態に与えた影響として Ag に報酬 r を与える。 r は、計算機資源への負荷 ΔR_{CP} 、ネットワーク資源への負荷 ΔR_{NW} 、圧縮符号化後の映像品質の劣化程度 ΔD を用いて以下の式で求める。

$$r = -\Delta R_{CP} - \Delta R_{NW} - \Delta D \quad (5)$$

このとき、 ΔR_{CP} 、 ΔR_{NW} が増加すると、それぞれ計算機資源、ネットワーク資源の消費量が大きくなるため、 r は小さい値を設定する。同様に、 ΔD が増加した場合、圧縮符号化後の映像の品質劣化が大きく、サービスの利用者に対して与えるサービス品質が劣化するため、 r は小さい値を設定する。

以上の動作により、Ag は r が最大となるような a を選択していくことで、映像の特性、受信端末の利用可能な資源、圧縮符号化後の映像品質の劣化程度を総合的に考慮した a を選択可能となる。

4.2 Q 学習を適用した資源学習機構の設計

本稿では、強化学習アルゴリズムとして Q 学習 [20] を使用する。Q 学習は、Ag が符号化パラメータ a を決定する際に、長期的に得られる報酬である Q 値を推定し、Q 値を最大化するような a を決定するアルゴリズムである。

Algorithm 1 に Q 学習に基づく資源学習機構の疑似コー

ドを示す. 初めに, すべての Q 値 $Q(s, a)$ を 0 に初期化する (1 行目). サービス提供中, Ag は時刻 t における受信端末 i の計算機資源 $R_{CPi,t}$ とネットワーク資源 $R_{NW_i,t}$ から, 受信端末 i の状態 $s_{i,t} \in S$ を観測する (3 行目, 4 行目). Ag は観測した $s_{i,t}$ に対応した映像品質 $a_t \in A$ を方策 π に基づき決定し, エンコーダに送信する (5 行目, 6 行目). このとき, π は ε -greedy アルゴリズムに基づく決定方法とする. 具体的には, 確率 ε で受信端末が利用可能な帯域に適應した a_t を, 確率 $(1 - \varepsilon)$ で Q 値が最大となる $s_{i,t}$ に対応した a_t を選択する.

次に, 受信端末の資源情報である $\Delta R_{CPi,t+\tau}$, $\Delta R_{NW_i,t+\tau}$ とエンコーダから獲得する圧縮符号化後の映像品質の劣化程度 $\Delta D_{i,t+\tau}$ から式 (5) に基づき報酬 $r_{i,t}$ を求める (7 行目~9 行目). そして, 状態 $s_{i,t+\tau}$ を再度観測し, 以下の式で Q 値 $Q_{i,t}$ ($= Q(s_{i,t}, a_t)$) を更新する (10 行目, 11 行目).

$$Q_{i,t} \leftarrow Q_{i,t} + \alpha \{ r_{i,t} + \gamma \max_{a' \in A} Q(s_{i,t+\tau}, a') - Q_{i,t} \} \quad (6)$$

ここで, α は学習率, γ は割引率であり, ともに範囲 (0.0, 1.0] の実数である. 一般に, α は Q 値の更新の程度を制御し, 0.0 に近い値を用いる. γ は次の状態における Q 値を割引き, 1.0 に近い値を用いる. また, a' は $s_{i,t+\tau}$ において Q 値が最大となる符号化パラメータである.

最後に, t を $t + \tau$ に更新する (12 行目).

以上の流れをサービス提供中繰り返す (13 行目) ことにより, 映像の特性と利用可能な受信端末の資源に適應した高品質な映像品質を決定する.

5. 実験と評価

5.1 実験の概要

前章の設計に基づき品質制御機構を実装し, 実装したプロトタイプを用いて品質制御機構の有効性を確認する実験を行った. 実験では, 品質制御機構が受信端末で利用可能な資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であることを確認するため, 利用可能な帯域のみで符号化パラメータを選択する従来の MPEG-DASH の単純手法と, 品質制御機構を用いて符号化パラメータを選択する提案手法を比較した.

図 2 に実験環境を示す. 実験では送信端末を使用せず, DASH サーバ内に配置した映像を受信端末に配信した. 受

信端末は 2 機のスマートフォンを使用し, Wi-Fi を用いて配信サーバへ接続した. 使用したスマートフォンと配信サーバの性能を表 1 に示す. 受信端末の利用可能な帯域は, 端末で実際に多様なネットワークサービスを利用した際の実効値に基づき決定した. 配信映像は, 3DCG アニメーション映像 *Big Buck Bunny* (BBB) [21] とスポーツ映像 *Football* [22] を使用した. 表 2 に配信映像の概要を示す. 3DCG アニメーション映像はフレーム間の動きが小さく, 圧縮効率が高いため [23], 圧縮効率の低いサッカー等のスポーツ映像との実験結果を比較することにより, 品質制御機構の動作変化を確認する.

表 3 に実験で使用した状態 S のパラメータ設定を示す. 計算機資源端末 R_{CP} とネットワーク資源 R_{NW} のパラメータに関しては, 映像再生中の端末資源の値に最も近い値を選択する. なお, 映像再生において CPU 使用率の変動は大きいと 5 段階, メモリ使用率の変動は小さいと 3 段階で選択する. 再生バッファの占有率は, バッファが枯渇状態, 正常状態, 潤沢状態として 3 段階とした. データ解析機構が解析する映像の特性 $char$ は, 映像のファイルサイズに応じて決定し, セグメント長決定機構が決定するセグメント長 len_{seg} は, 一般に DASH で使用されている値を使用した [14], [24].

また, 学習周期 τ は 1 秒, 学習率 α , 割引率 γ , 確率 ε はそれぞれ 0.1, 0.9, 0.1 とした. ライブストリーミングサービスを利用する不安定な環境を想定し, 再生バッファは 5 秒, 初期バッファリング時間は 1 秒に設定した.

評価指標は, 画質評価指標 Structural SIMilarity (SSIM) [25] の平均 $SSIM_{avg}$, 映像再生中の平均フレームレート FPS_{avg} , 映像再生プロセスの平均 CPU 使用率 CPU_{avg} , 累計ドロップフレーム数 $DROP_{total}$ とした. SSIM は, 値が 1.0 に近いほど元映像からの劣化が少ないとされる指標である.

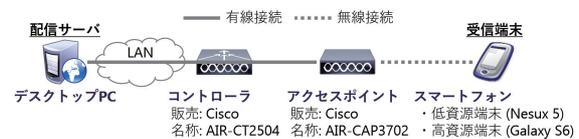


図 2 実験環境

Fig. 2 Experimental environment.

表 1 実験で使用した端末性能

Table 1 Device specifications used in experiments.

	配信サーバ	受信端末	
	デスクトップ PC	低資源端末 (Nexus 5)	高資源端末 (Galaxy S6)
OS	Windows 10 Pro	Android 6.0.1	Android 7.0
CPU	Core i7-6700@3.4 (GHz)	Snapdragon 800	Exynos 7420
メモリ	8 (GB)	2 (GB)	3 (GB)
ネットワーク	IEEE 802.3ab	IEEE 802.11n	IEEE 802.11ac
利用可能な帯域	N/A	28 (Mbps)	80 (Mbps)

表 2 実験で使用した配信映像

Table 2 Streaming data used in experiments.

	<i>Big Buck Bunny</i>	<i>Football</i>
ファイル名	bbb_sunflower_2160p_60fps_normal.mp4	Samsung_UHD_Soccer_Barcelona_Atletico_Madrid.ts
符号化形式	H.264/AVC	H.265/HEVC
サイズ <i>size</i>	657.4 (MB)	1,298.2 (MB)
再生時間 <i>len</i>	634 (s)	232 (s)
動画解像度	2160p	2160p
フレームレート	60 (fps)	50 (fps)

表 3 実験で使用した状態 *S*

Table 3 State *S* used in experiments.

		パラメータ
R_{CP}	<i>CPU</i>	{0, 25, 50, 75, 100} (%)
	<i>MEM</i>	{0, 50, 100} (%)
R_{NW}	<i>BUF</i>	{0, 50, 100} (%)
I_{DAT}	<i>char</i>	<i>size</i> × 8 / <i>len</i> (Mbps)
	<i>len_{seg}</i>	2 (s)

表 4 実験 1 で使用した符号化パラメータ *A*

Table 4 Encoding parameters used in experiment 1.

		パラメータ
<i>CRF</i>		{18, 23, 28}
<i>FPS</i>		{60, 30, 24, 20} (fps)
<i>RSL</i>		{2160p, 1080p, 720p, 360p, 180p}

計算機資源への負荷 ΔR_{CP} は以下の式で求めた。

$$\Delta R_{CP} = \Delta cpu + \Delta mem \quad (7)$$

ここで、 Δcpu 、 Δmem は映像を再生するアプリケーションの CPU 使用率とメモリ使用率である。また、確率 ε で利用可能な帯域 *bwt* を考慮した符号化パラメータ *a* を選択するため、ネットワーク資源への負荷 ΔR_{NW} はバッファ占有率 Δbuf のみ考慮した。

符号化後の映像品質の劣化程度 ΔD は、選択した符号化パラメータ a_t 中の fps_t ($\in FPS$) に対して受信端末が配信映像を 1 秒間で画面に表示したフレーム数の割合 Δfps 、 fps_t に対して受信端末が配信映像を 1 秒間で画面に表示せずドロップしたフレーム数の割合 $\Delta drop$ 、再生中の配信映像の SSIM である $\Delta ssim$ を用いて以下の式で求めた。

$$\Delta D = \Delta fps + \Delta drop + \Delta ssim \quad (8)$$

5.2 実験 1: 3DCG アニメーション映像を用いた実験

5.2.1 実験 1 の概要

配信映像として 3DCG アニメーション映像 *BBB* を使用した実験 1 の手順は以下のとおりである。

- (1) エンコーダは配信映像を圧縮符号化およびセグメント化し、すべての品質のセグメント (= $|A|$ 個) を生成する。獲得した圧縮符号化後の SSIM $\Delta ssim$ の時系列データは資源学習機構に送信する。
- (2) 受信端末は配信サーバが送信するセグメントを再生する。このとき、配信サーバは品質制御を行わずに、一定の品質のセグメントを送信する。これをすべての品質の数だけ (= $|A|$ 回) 実施し、獲得した資源情報 ΔR_{CP} 、 ΔR_{NW} の時系列データは品質制御機構に送信する。

- (3) 獲得した $\Delta ssim$ 、 ΔR_{CP} 、 ΔR_{NW} の時系列データを入力として資源学習機構が Q 学習し、符号化パラメータを選択する。これを配信映像の再生時間 *len* だけ繰り返す。

実験手順 (3) において、配信映像の再生時間に応じて利用可能な帯域 *bwt* と CPU 使用率 *cpu* を制限することにより、受信端末で利用可能な資源に適応した符号化パラメータを決定可能であることを確認する。具体的に資源の制限時は、資源学習機構の入力値を制限値に置き換えて学習する。*bwt* は制限時に表 1 に示す利用可能な帯域から 3G ネットワークの最大受信速度である 14.4 Mbps に置き換え、利用可能な帯域の変動を確認する。*cpu* は制限時に 100% に置き換え、映像再生プロセスの CPU 使用率の変動を確認する。映像の経過時間に応じて制限した *bwt* と *cpu* の区間は以下のとおりである。

- (3-1) 50 秒後から 100 秒間、利用可能な帯域を制限
- (3-2) 200 秒後から 100 秒間、受信端末の CPU 使用率を増加
- (3-3) 350 秒後から 100 秒間、利用可能な帯域を制限し、CPU 使用率を増加

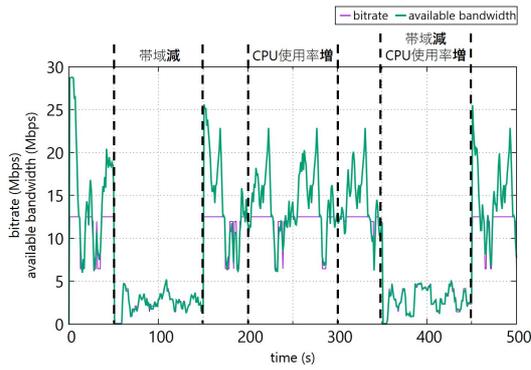
以上のように、資源学習機構に仮想的なパラメータを入力することで、受信端末で利用可能な資源に制限がある状態を再現した。

表 4 に実験 1 で使用した符号化パラメータ *A* のパラメータ設定を示す。資源学習機構は、*CRF*、*FPS*、*RSL* の各要素を組み合わせた符号化パラメータを決定する。

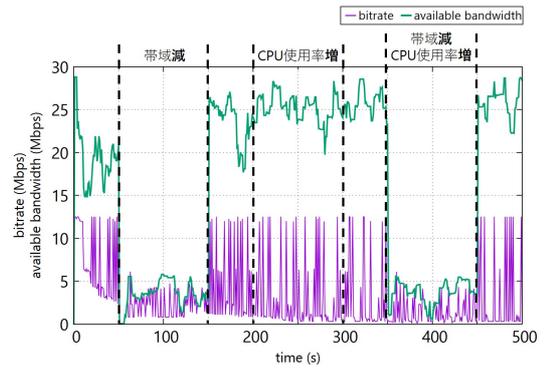
5.2.2 実験 1 の結果

図 3 に低資源端末、図 4 に高資源端末を用いた場合の利用可能な帯域と CPU 使用率の変動を示す。各図の横軸は時間 (time) である。

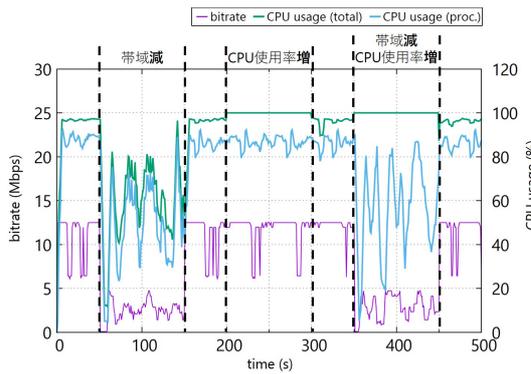
両図の (a) および (b) の縦軸は資源学習機構が選択した符号化パラメータに対応するビットレート (bitrate)、受



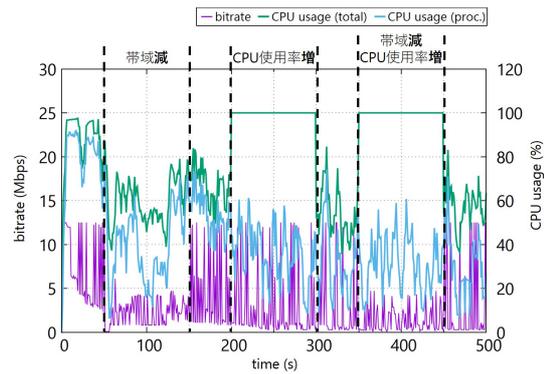
(a) 利用可能な帯域の変動 (単純手法)



(b) 利用可能な帯域の変動 (提案手法)



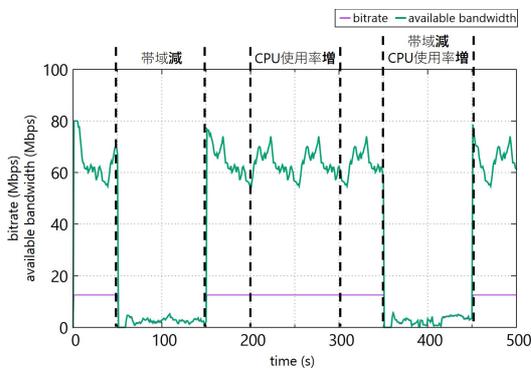
(c) CPU 使用率の変動 (単純手法)



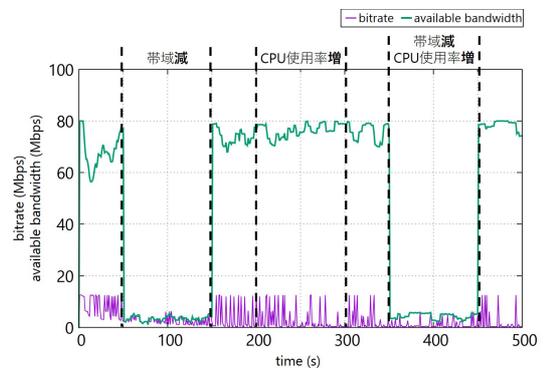
(d) CPU 使用率の変動 (提案手法)

図 3 実験 1 の結果 (低資源端末使用時の利用可能な帯域と CPU 使用率の変動)

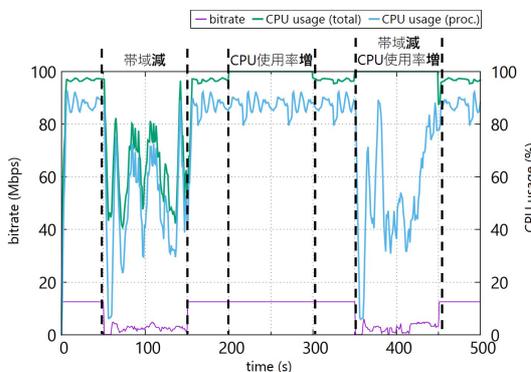
Fig. 3 Experiment 1 results using the low-resource device.



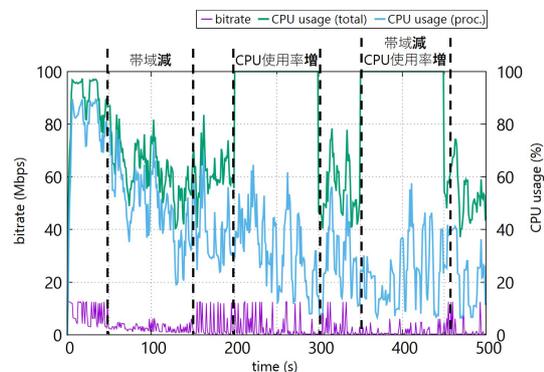
(a) 利用可能な帯域の変動 (単純手法)



(b) 利用可能な帯域の変動 (提案手法)



(c) CPU 使用率の変動 (単純手法)



(d) CPU 使用率の変動 (提案手法)

図 4 実験 1 の結果 (高資源端末使用時の利用可能な帯域と CPU 使用率の変動)

Fig. 4 Experiment 1 results using the high-resource device.

信端末で利用可能な帯域 (available bandwidth) である。単純手法は、受信端末で利用可能な帯域や CPU 使用率が変動しても、帯域に適応した符号化パラメータを決定し続けた。これに対し提案手法は、受信端末で利用可能な資源が変動した場合は利用可能な帯域に余裕を持たせるように低品質な符号化パラメータを決定するようになった。

利用可能な帯域を制限し、CPU 使用率が増加した実験手順 (3-3) 時において、単純手法を用いた低資源端末では、帯域は平均 0.054 Mbps (0.188%)、最大で 0.421 Mbps (1.462%) 余裕が生じた。提案手法では、平均 2.565 Mbps (8.905%)、最大で 23.018 Mbps (79.923%) の余裕が発生した。単純手法を用いた高資源端末では、0.490 Mbps (0.612%)、最大で 44.542 Mbps (55.677%)、提案手法では、平均 4.100 Mbps (5.124%)、最大で 76.382 Mbps (95.478%) の余裕が生じた。

両図の (c) および (d) の縦の第一軸は資源学習機構が選択した符号化パラメータに対応するビットレート (bitrate)、縦の第二軸は CPU 使用率 (CPU usage) である。単純手法は、受信端末の CPU 使用率が増加しても映像再生プロセスの CPU 使用率は高い状態が見られ、高品質な符号化パラメータを決定し続けた。これに対し提案手法は、受信端末の CPU 使用率が増加した場合に受信端末がセグメントを再生する負荷を軽減するために低品質な符号化パラメータを決定するようになった。

実験手順 (3-3) 時において、単純手法を用いた低資源端末では、CPU 使用率は平均 43.450%、最大で 94.897% 余裕が生じた。提案手法では、平均 68.677%、最大で 90.571% の余裕が発生した。単純手法を用いた高資源端末では、平均 46.715%、最大で 94.080%、提案手法では、平均 76.861%、最大で 93.380% の余裕が生じた。

表 5 に評価指標を用いて単純手法と提案手法を比較した結果を示す。低資源端末における単純手法の平均 SSIM は 0.101% 減少したが、平均フレームレートは 122.525% 増加した。累計フレームドロップ数は 66.932%、平均 CPU 使用率は 45.257% 減少した。また、提案手法の平均 SSIM は 0.202% 減少したが、平均フレームレートは 92.408% 増加した。累計フレームドロップ数は 63.329%、平均 CPU 使用率は 47.605% 減少したことを確認した。

5.3 実験 2: スポーツ映像を用いた実験

5.3.1 実験 2 の概要

配信映像としてスポーツ映像、具体的にサッカー映像 *Football* を使用した実験 2 の手順は実験 1 と同様であるが、映像の再生時間が異なるため、実験手順 (3) において、映像の経過時間に応じて制限した利用可能な帯域 *bwt* と CPU 使用率 *cpu* の区間は以下とした。

(3-1) 25 秒後から 50 秒間、利用可能な帯域を制限し、CPU 使用率を増加

(3-2) 100 秒後から 25 秒間、受信端末の CPU 使用率を

表 5 実験 1 の結果 (評価指標を用いた比較)

Table 5 Experiment 1 results using evaluation metrics.

(a) 低資源端末			
	単純手法	提案手法	増減率 (%)
SSIM _{avg}	0.990	0.989	-0.101
FPS _{avg} (frames/sec.)	9.375	20.862	122.525
DROP _{total} (frames)	7028	2324	-66.932
CPU _{avg} (%)	72.192	39.520	-45.257
(b) 高資源端末			
	単純手法	提案手法	増減率 (%)
SSIM _{avg}	0.990	0.988	-0.202
FPS _{avg} (frames/sec.)	9.912	19.072	92.408
DROP _{total} (frames)	7660	2809	-63.329
CPU _{avg} (%)	71.980	37.714	-47.605

表 6 実験 2 で使用した符号化パラメータ A

Table 6 Encoding parameters used in experiment 2.

	パラメータ
CRF	{18, 23, 28}
FPS	{60, 25, 20} (fps)
RSL	{2160p, 1080p, 720p, 360p, 180p}

増加

(3-3) 150 秒後から 25 秒間、利用可能な帯域を制限

表 6 に実験 2 で使用した符号化パラメータ A のパラメータ設定を示す。

5.3.2 実験 2 の結果

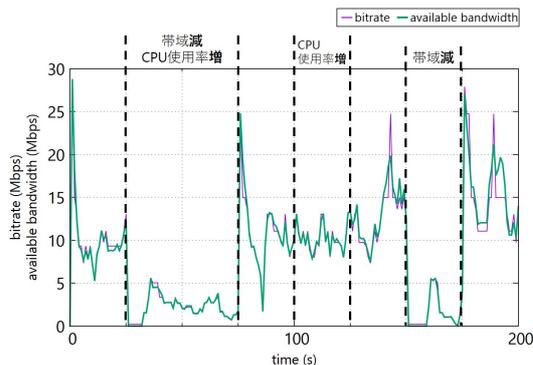
図 5 に低資源端末、図 6 に高資源端末を用いた場合の利用可能な帯域と CPU 使用率の変動を示す。各図の縦軸と横軸は実験 1 と同一である。

実験 2 の結果は実験 1 の結果と同様の傾向となった。具体的には、両図の (a) および (b) より、単純手法は帯域に適応した符号化パラメータを決定し続けたのに対し、提案手法は利用可能な帯域に余裕を持たせるようになった。

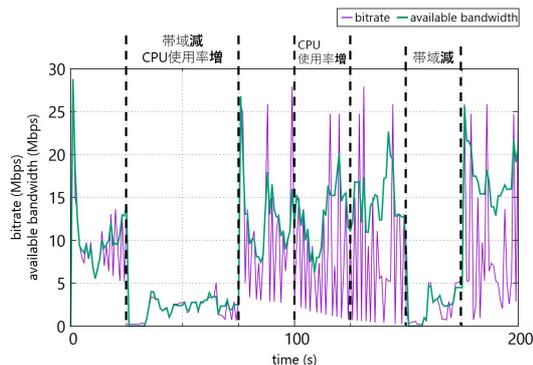
利用可能な帯域を制限し、CPU 使用率が増加した実験手順 (3-1) 時において、単純手法を用いた低資源端末では、帯域は平均 0.010 Mbps (0.036%)、最大で 0.134 Mbps (0.466%) 余裕が生じた。提案手法では、平均 0.600 Mbps (2.082%)、最大で 6.704 Mbps (23.277%) の余裕が発生した。単純手法を用いた高資源端末では、0.009 Mbps (0.012%)、最大で 0.134 Mbps (0.168%)、提案手法では、平均 1.766 Mbps (2.220%)、最大で 36.086 Mbps (45.107%) の余裕が生じた。

両図の (c) および (d) より、単純手法は高品質な符号化パラメータを決定し続けたのに対し、提案手法は受信端末の CPU 使用率が増加した場合に受信端末が映像を再生する負荷を軽減するようになった。

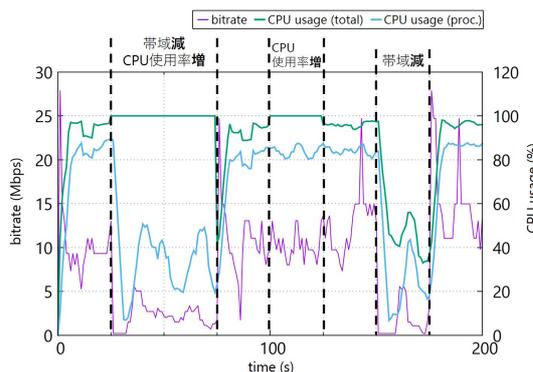
実験手順 (3-1) 時において、単純手法を用いた低資源端末では、CPU 使用率は平均 22.005%、最大で 39.281% 余裕が生じた。提案手法では、平均 19.592%、最大で 30.337% の



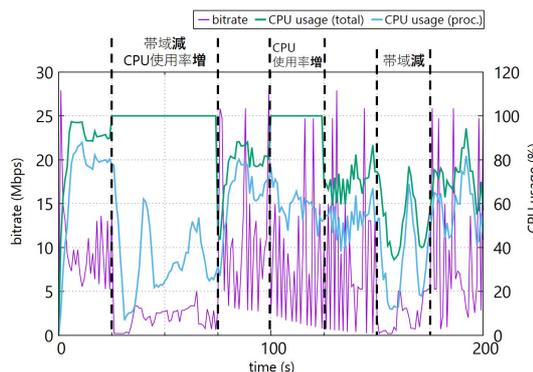
(a) 利用可能な帯域の変動 (単純手法)



(b) 利用可能な帯域の変動 (提案手法)



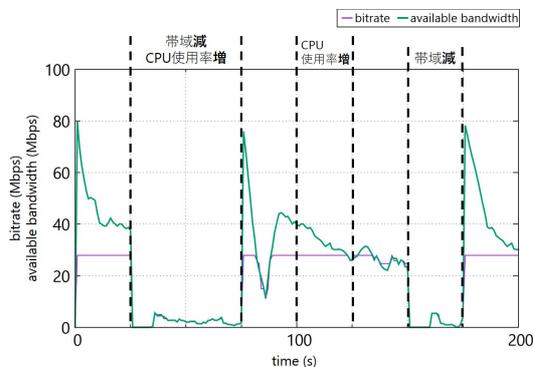
(c) CPU 使用率の変動 (単純手法)



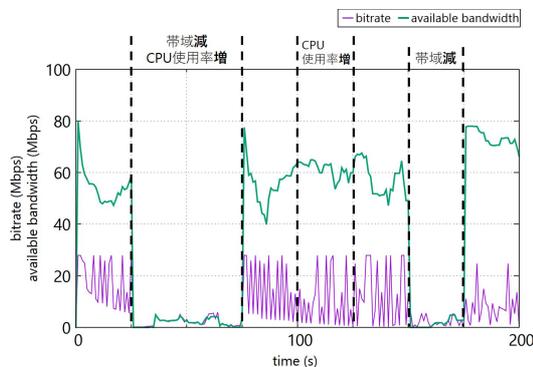
(d) CPU 使用率の変動 (提案手法)

図 5 実験 2 の結果 (低資源端末使用時の利用可能な帯域と CPU 使用率の変動)

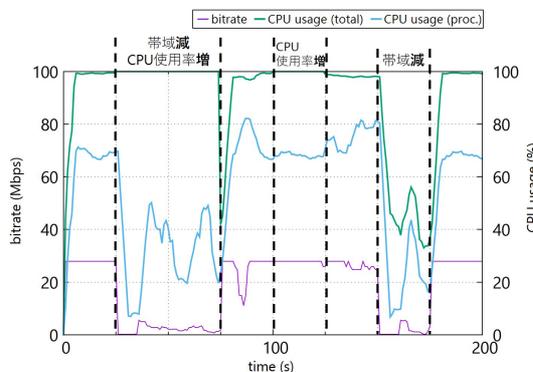
Fig. 5 Experiment 2 results using the low-resource device.



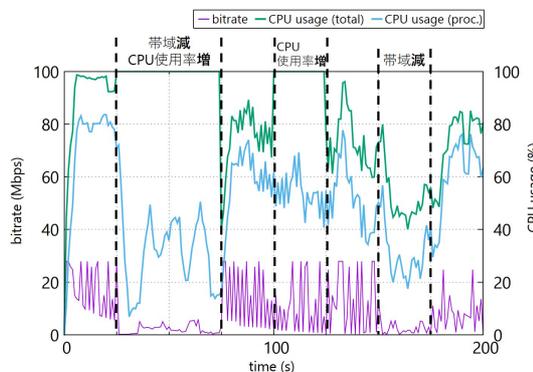
(a) 利用可能な帯域の変動 (単純手法)



(b) 利用可能な帯域の変動 (提案手法)



(c) CPU 使用率の変動 (単純手法)



(d) CPU 使用率の変動 (提案手法)

図 6 実験 2 の結果 (高資源端末使用時の利用可能な帯域と CPU 使用率の変動)

Fig. 6 Experiment 2 results using the high-resource device.

表 7 実験 2 の結果 (評価指標を用いた比較)

Table 7 Experiment 2 results using evaluation metrics.

(a) 低資源端末			
	単純手法	提案手法	増減率 (%)
SSIM _{avg}	0.978	0.976	-0.204
FPS _{avg} (frames/sec.)	10.841	18.104	67.003
DROP _{total} (frames)	1796	1116	-37.862
CPU _{avg} (%)	61.530	50.792	-17.452
(b) 高資源端末			
	単純手法	提案手法	増減率 (%)
SSIM _{avg}	0.980	0.977	-0.306
FPS _{avg} (frames/sec.)	10.000	17.846	78.458
DROP _{total} (frames)	599	751	25.376
CPU _{avg} (%)	53.293	48.546	-8.907

余裕が発生した。単純手法を用いた高資源端末では、平均 22.349%、最大で 39.281%、提案手法では、平均 21.801%、最大で 26.138%の余裕が生じた。

表 7 に評価指標を用いて単純手法と提案手法を比較した結果を示す。低資源端末における単純手法の平均 SSIM は 0.204%減少したが、平均フレームレートは 67.003%増加した。累計フレームドロップ数は 37.862%、平均 CPU 使用率は 17.452%減少した。また、提案手法の平均 SSIM は 0.306%減少したが、平均フレームレートは 78.458%増加した。累計フレームドロップ数は 25.376%増加したが、平均 CPU 使用率は 8.907%減少したことを確認した。

5.4 考察

実験 1 および実験 2 の結果より、提案手法は利用可能な帯域が減少した際には、利用可能な帯域に余裕を持たせるためにデータ量が小さいセグメントの符号化パラメータを徐々に決定するようになり、総 CPU 使用率が増加した際には、配信映像の再生負荷を軽減するため低品質なセグメントの符号化パラメータを徐々に決定するようになった。すなわち、利用可能な資源を制限した際には低ビットレートの符号化パラメータを徐々に決定するようになった。これは、学習の初期段階で低ビットレートの符号化パラメータを選択することでエージェントが得られる報酬が大きくなる (消費資源量が少ない、圧縮符号化後の映像品質の劣化が少ない) ことを学習したためである。なお、符号化後の映像品質の劣化程度 ΔD を求める式 (8) より、フレームドロップが発生していない場合は、SSIM が大きく、フレームレートが小さい映像が選択されることがある。フレームレートの変動による利用者の体感品質 (QoE) への影響は大きい [26], [27]、式 (8) の各項に優先度を設定することや、 Δfps で決定した符号化パラメータ中のフレームレート値の大きさを考慮することにより、フレームレートを重視した学習が可能になる。

評価指標を用いて比較した実験結果より、提案手法は単純手法よりも SSIM が減少し画質の劣化が見られたが、平均フレームレートは向上した。さらに、累計フレームドロップ数と平均 CPU 使用率は減少し、消費資源量を削減した。単純手法は平均 CPU 使用率と累計フレームドロップ数が大きいことから、配信映像の再生時に頻繁にバッファリングが発生し、スムーズな再生ができなかった一方で、提案手法は、平均 CPU 使用率と累計フレームドロップ数が小さいため、配信映像の再生に問題は生じなかった。また、低ビットレートの符号化パラメータを決定したことで SSIM が減少した。しかし、その減少率は非常に小さく、平均フレームレートは増加、累計フレームドロップ数は減少しているため、利用者が体感する映像品質は向上していると考えられる。

以上のことから、提案手法を用いることにより、映像の特性、受信端末で利用可能な資源、符号化後の映像品質の劣化程度との関係を強化学習した提案手法を用いることで、受信端末の資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であることを確認した。消費資源量を抑えることにより、ネットワークの通信量を削減可能であるだけでなく、CPU 使用率を抑えることで長時間サービスを利用可能となる。しかし、スポーツ映像を用いた実験 2 において高資源端末用いた場合で累計フレームドロップ数が増加した。これは、高性能端末において学習が効率的に行っていないためだと考えられるので、今後学習アルゴリズムについては改良が必要になる。具体的な改良策として 2 つ考えられる。

1 つ目は、強化学習における符号化パラメータの決定方針である方策 π の改良である。実験では、利用可能な帯域に適応した品質を選択する確率 ϵ の値を固定した ϵ -greedy を使用した。この場合、学習が進んだ状況下においても利用可能な帯域に適応した品質を選択することになるため、利用可能な端末資源に適さない過少/過剰な品質を決定する可能性がある。そのため、学習の進行状況に応じて ϵ の値を小さくしていくことにより、Q 値が大きい品質 (すなわち、利用可能な端末資源に適した品質) を選択するよう制御する必要がある。

2 つ目の改良策は、Q 学習を発展させた深層強化学習 [28] を使用することである。本稿で設計した Q 学習は、入力値を離散化する必要がある。入力値を細かく離散化した場合、学習すべき Q 値が爆発的に増大し、非常に多くの学習ステップが必要となる。強化学習と深層学習を組み合わせることで Q 値の近似関数を得る深層強化学習では連続値を入力することが可能である。そのため、より高精度で柔軟な品質制御が実現可能であると考えられる。

また、実際のライブストリーミングサービスに提案手法を適用する際は、受信端末の頻繁な状態の変化への対処を行う必要がある。これは、頻繁な品質の変動は受信端末へ

の影響が大きいためである。したがって、受信端末の資源の変動に生じてても一定時間は品質を変動させない制御が必要である。さらに、資源学習機構とエンコーダを複数用意することで、受信端末ごとの学習、セグメント化が可能になるため、より利用者指向のサービス提供が可能になると考える。

6. おわりに

本研究では、適応型ライブストリーミングサービスの実現手法の1つである、Moving Picture Experts Group (MPEG) が標準化した Dynamic Adaptive Streaming over HTTP (MPEG-DASH) を対象とし、ネットワークの状況および受信端末の性能に適応したライブストリーミングサービスを実現するため、強化学習を用いた映像品質制御手法を提案した。具体的には、映像の特性、利用可能なネットワーク資源、計算機資源と、それに応じた利用者へ配信される映像品質の関係を強化学習することで、映像をリアルタイムに配信するライブストリーミングサービスを利用する様々な環境に適応した映像品質を決定する資源適応映像品質制御機構を提案した。実験結果から、提案手法を用いることで資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であることを確認した。

今後は、データ解析機構とセグメント長決定機構の詳細設計を行い、これらの機構が配信映像と受信端末で利用可能な資源に応じて適切な映像の特性とセグメント長を決定可能であることを確認する。さらに、品質制御機構を実際のライブストリーミングサービスに適用し、実端末を用いた実験により、提案手法の効果を検証する。

参考文献

- [1] 総務省：平成 29 年版情報通信白書，総務省（オンライン），入手先（<http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/pdf/n2100000.pdf>）（参照 2019-09-05）。
- [2] Cisco Systems, Inc.: Cisco Visual Networking Index: Forecast and Methodology, 2016–2021, Cisco Systems, Inc. (online), available from (<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf>) (accessed 2019-09-05).
- [3] Kesidis, G., Urgaonkar, B., Nasiriani, N. and Wang, C.: Neutrality in Future Public Clouds: Implications and Challenges, *8th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 2016)*, pp.1–6, USENIX Association (2016).
- [4] 伊藤正史：現行地上波で HD/4K サイマル視聴サービスを実現するハイブリッドキャストコンテンツ（4K ランドスケープ），映像情報メディア学会誌，Vol.71, No.4, pp.J125–J130 (2017).
- [5] KDDI 株式会社：『EAFF E-1 サッカー選手権 2017 決勝大会』で国内初となるスタジアムでの VR 観戦など新しいサッカー観戦体験！「au Future Stadium」, KDDI 株式会社（オンライン），入手先（<https://www.au.com/information/topic/mobile/2017-082/>）（参照 2019-09-05）。
- [6] YouTube: Ten years of YouTube video tech in ten videos (online), available from (<https://youtube-eng.googleblog.com/2015/05/ten-years-of-youtube-video-tech-in-ten.html>) (accessed 2019-09-05).
- [7] Park, A. and Watson, M.: The Netflix Tech Blog: HTML5 Video at Netflix, Netflix (online), available from (<http://techblog.netflix.com/2013/04/html5-video-at-netflix.html>) (accessed 2019-09-05).
- [8] 馬場美由紀：AbemaTV がリニア型配信で「MPEG-DASH」をサポートした理由と、その使い方は？, CodeIQ（オンライン），入手先（<http://getnews.jp/archives/1973058>）（参照 2019-09-05）。
- [9] ISO/IEC: Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats (2012).
- [10] Li, Z., Zhu, X., Gahm, J., Pan, R., Hu, H., Begen, A.C. and Oran, D.: Probe and Adapt: Rate Adaptation for HTTP Video Streaming At Scale, *IEEE Journal on Selected Areas in Communications*, Vol.32, No.4, pp.719–733 (online), DOI: 10.1109/JSAC.2014.140405 (2014).
- [11] Huang, T., Johari, R., McKeown, N., Trunnell, M. and Watson, M.: A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service, *ACM Conference on Special Interest Group on Data Communication (SIGCOMM 2014)*, pp.187–198, ACM Press (2014).
- [12] Juluri, P., Tamarapalli, V. and Medhi, D.: QoE Management in DASH Systems Using the Segment Aware Rate Adaptation Algorithm, *IEEE/IFIP Network Operations and Management Symposium (NOMS 2016)*, pp.129–136, IEEE (2016).
- [13] Lee, S., Lee, E. and Lee, H.: Quality adaptation scheme for improving QoE of MPEG DASH, *International Conference on Information and Communication Technology Convergence (ICTC 2016)*, pp.368–370, IEEE (2016).
- [14] Kim, M., Park, J. and Chung, K.: Content-aware rate adaptation scheme to improve stability in HTTP Adaptive Streaming, *International Conference on Information Networking (ICOIN 2017)*, pp.401–405, IEEE (2017).
- [15] De Filippo De Grazia, M., Zucchetto, D., Testolin, A., Zanella, A., Zorzi, M. and Zorzi, M.: QoE Multi-Stage Machine Learning for Dynamic Video Streaming, *IEEE Trans. Cognitive Communications and Networking*, Vol.4, No.1, pp.146–161 (online), DOI: 10.1109/TCCN.2017.2784449 (2018).
- [16] Gadaleta, M., Chiariotti, F., Rossi, M. and Zanella, A.: D-DASH: A Deep Q-Learning Framework for DASH Video Streaming, *IEEE Trans. Cognitive Communications and Networking*, Vol.3, No.4, pp.703–718 (online), DOI: 10.1109/TCCN.2017.2755007 (2017).
- [17] Petrangeli, S., Bouten, N., Dejonghe, E., Famaey, J., Leroux, P. and De Turck, F.: Design and Evaluation of a DASH-compliant Second Screen Video Player for Live Events in Mobile Scenarios, *IFIP/IEEE International Symposium on Integrated Network Management (IM 2015)*, pp.894–897, IEEE (2015).
- [18] Hassan, Y.M., Helmy, A. and Rehan, M.M.: Effect of Varying Segment Size on DASH Streaming Quality for Mobile User, *International Conference on Engineering and Technology (ICET 2014)*, pp.1–4, IEEE (2014).
- [19] 牧野貴樹，澁谷長史，白川真一ほか：これからの強化学習，森北出版 (2016)。
- [20] Watkins, C.J.C.H. and Dayan, P.: Q-Learning, *Ma-*

- chine Learning*, Vol.8, No.3, pp.279–292 (online), DOI: 10.1007/BF00992698 (1992).
- [21] Blender Foundation: Big Buck Bunny, Blender Foundation (online), available from (<https://peach.blender.org/>) (accessed 2019-09-05).
- [22] Samsung: Barcelona vs Atletico Madrid 4K Football Sample Footage, Samsung (online), available from (<http://4ksamples.com/barcelona-vs-atletico-madrid-4k-football-sample-footage/>) (accessed 2019-09-05).
- [23] Srinivasan, A. and Rohini, G.: Performance based algorithms for video bit transmissions, *Cognitive Systems Research*, Vol.56, pp.179–191 (online), DOI: 10.1016/j.cogsys.2018.10.019 (2019).
- [24] Ansarul Haque, M. and Delwar Hossain, M.: Technology survey of wireless communication for in-vehicle applications, *The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)*, pp.1–7, IEEE (online), DOI: 10.1109/SKIMA.2014.7083523 (2014).
- [25] Wang, Z., Bovik, A., Sheikh, H. and Simoncelli, E.: Image Quality Assessment: From Error Visibility to Structural Similarity, *IEEE Trans. Image Processing*, Vol.13, No.4, pp.600–612 (online), DOI: 10.1109/TIP.2003.819861 (2004).
- [26] Ben Youssef, Y., Mellouk, A., Afif, M. and Tabbane, S.: Video Quality Assessment Based on Statistical Selection Approach for QoE Factors Dependency, *IEEE Global Communications Conference (GLOBECOM 2016)*, pp.1–6, IEEE (online), DOI: 10.1109/GLOCOM.2016.7842375 (2016).
- [27] Zimmer, T., Hohlfeld, O., Abboud, O. and Hossfeld, T.: Impact of frame rate and resolution on objective QoE metrics, *2nd International Workshop on Quality of Multimedia Experience (QoMEX 2010)*, pp.29–34, IEEE (online), DOI: 10.1109/QOMEX.2010.5518277 (2010).
- [28] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S. and Hassabis, D.: Human-level control through deep reinforcement learning, *Nature*, Vol.518, No.7540, pp.529–533 (online), DOI: 10.1038/nature14236 (2015).

推薦文

本研究は、映像をリアルタイムに配信するライブストリーミングサービスにおいて、映像の特性、利用可能なネットワーク資源、計算機資源と、受信端末でサービス利用者へ配信される映像品質の関係を強化学習することで、ネットワークの状況だけでなく受信端末の性能にも適応した品質の映像を配信するための映像品質制御手法を提案しており、資源の無駄な消費を抑えつつ高品質な映像を利用者へ配信可能であり、有用な研究と考える。以上の理由により、本稿を推薦いたします。

(第26回マルチメディア通信と分散処理ワークショップ
プログラム委員長 中沢 実)



生出 真人 (学生会員)

2018年3月東北大学大学院情報科学研究科修士課程修了。現在、同博士課程在学中。エージェント指向コンピュータリング、マルチメディア通信の研究開発に従事。



阿部 亨 (正会員)

1990年3月東北大学大学院工学研究科博士後期課程単位取得退学。工学博士。東北大学情報処理教育センター助手、北陸先端科学技術大学院大学情報科学研究科助教授を経て、現在、東北大学サイバーサイエンスセンター准教授。知識処理、パターン認識の研究開発に従事。



菅沼 拓夫 (正会員)

1997年3月千葉工業大学大学院博士後期課程修了。工学博士。同年東北大学電気通信研究所助手。同大学助教授、准教授を経て、2010年から東北大学サイバーサイエンスセンター教授。エージェント指向コンピュータリング、共生コンピュータリング、新世代ネットワーク管理等の研究開発に従事。電子情報通信学会、IEEE各会員。