

# Information Centric Networkingにおける 人気度の収集と通知を用いたキャッシング手法

神本 崇史<sup>1,a)</sup> 佐藤 和也<sup>1</sup> 重野 寛<sup>1</sup>

受付日 2016年5月23日, 採録日 2016年11月1日

**概要:** Information Centric Networking (ICN) においてキャッシュ活用のためのネットワーク負荷が小さい非協調型キャッシング手法は, ドメイン内で比較的サーバに近い上流ルータのキャッシュが活用されにくく, コンテンツの重複保存によってキャッシュ活用能力が低下する. 本論文では, ICN において各ルータが収集したコンテンツの人気度と, 上流からのキャッシュ情報を利用してキャッシュを活用するキャッシング手法 CaSTA を提案する. CaSTA では各ルータがコンテンツの人気度を集計することで, 上流ルータにおいて大域的に人気のあるコンテンツを把握し, パケットを用いて上流のキャッシュ情報を下流に通知する. これによって, 上流ルータのキャッシュを積極的に活用すると同時にコンテンツの重複保存を回避し, キャッシュ活用能力の向上を目指す. また, シミュレーションによる CaSTA と他の非協調型キャッシング手法との比較評価を行い, CaSTA の有用性を確認した.

**キーワード:** Information Centric Networking, キャッシング手法, キャッシュヒット率, サーバ負荷

## Caching Scheme with Popularity Collection and Caching Announcement on Information Centric Networking

TAKASHI KAMIMOTO<sup>1,a)</sup> KAZUYA SATO<sup>1</sup> HIROSHI SHIGENO<sup>1</sup>

Received: May 23, 2016, Accepted: November 1, 2016

**Abstract:** On Information Centric Networking (ICN), non-cooperative caching schemes can utilize caches with low additional traffic. However, it is difficult for them to utilize caches of upstream routers, and they make duplicated caches. As a result, they degrade performance of utilizing caches on the network. In this paper, we propose cache shifting scheme using top-down announcement for ICN, called CaSTA. In CaSTA, routers collect popularity of contents by using standard packets and enable upstream routers to store globally popular contents. Moreover, upstream routers announce cache information to downstream routers. In this way, CaSTA utilizes caches of upstream routers and avoid duplicated caches. We compare our proposal, CaSTA, by comparing it with some non-cooperative caching schemes, and confirm the effectiveness of CaSTA.

**Keywords:** Information Centric Networking, non-cooperative caching scheme, cache hit ratio, server load

### 1. はじめに

Information Centric Networking (ICN) [1], [2] は, コンテンツ取得を行う際に通信を行う相手が誰かを意識しないコンテンツ指向のネットワークとして近年注目を集めている. 現在利用されている Internet Protocol (IP) ネットワークは通信相手の情報を用いるロケーション指向のネッ

トワークであり, ユーザがコンテンツ取得を行う際はコンテンツの名前 (コンテンツ名) から相手の情報を取得する必要がある. 一方の ICN では, ユーザだけでなくネットワーク内においてもコンテンツ名によって通信を行うことができるため, 通信相手を意識しないユーザの利用行動に適していると指摘されている. さらに, すべてのルータがキャッシュを保有し, コンテンツの複製を大規模に分散できる. 以下では, コンテンツとその複製を区別せず, 単にコンテンツと表現する. 一般的に単位時間あたりの要求回数が多きコンテンツ, つまり人気のあるコンテンツがキャッ

<sup>1</sup> 慶應義塾大学大学院理工学研究科  
Graduate School of Science and Technology, Keio University,  
Yokohama, Kanagawa 223–8522, Japan

a) kamimoto@mos.ics.keio.ac.jp

シユに保存される。キャッシュの活用によって、ユーザは人気のあるコンテンツをサーバ以外のルータから取得することが可能となり、効率的なコンテンツ取得を行うことができる。

キャッシュの活用能力はキャッシング手法に左右される。ルータ間でキャッシュ活用のための特別な情報交換(協調)を行わない非協調型キャッシング手法は、キャッシュ活用のためのネットワーク負荷が小さい。しかし、あるネットワークドメイン内で人気のあるコンテンツがユーザ近くの下流ルータに保存されるため、比較的サーバに近い上流ルータのキャッシュが活用されにくい。さらに、大域的に人気のあるコンテンツが複数のルータに重複して保存され、ドメイン内でキャッシュに保存できるコンテンツの種類数が減少する。ここで、大域的に人気のあるコンテンツとは、あるネットワークドメイン内全体で人気のあるコンテンツのことであり、ドメイン内の一部から人気のあるコンテンツを局所的に人気のあるコンテンツと呼ぶ。コンテンツの重複保存は、ドメインにおけるキャッシュ活用能力低下を引き起こす。

本論文では、ICNにおいて各ルータが収集したコンテンツの人気度と、上流からのキャッシュ情報を利用してキャッシュを活用するキャッシング手法 CaSTA (Cache Shifting scheme using Top-down Announcement for Information Centric Networking) を提案する。CaSTA では各ルータがコンテンツの人気度を集計することで、上流ルータにおいて大域的に人気のあるコンテンツを把握できる仕組みを導入する。さらに、パケットを用いて上流のキャッシュ情報を下流に通知することで、上流で優先的にキャッシュへの保存を行い、上流ルータのキャッシュを積極的に活用すると同時にコンテンツの重複保存を回避する。これを利用して、ネットワークへの負荷増大を抑制しつつ、ドメインにおけるキャッシュ活用能力の向上を目的とする。

以下本論文では、2章でICNや関連のキャッシング手法について紹介し、3章で提案手法 CaSTA について詳細に説明する。4章では提案手法 CaSTA の有用性を検証するためのシミュレーション実験とその結果について議論し、5章で本論文のまとめとする。

## 2. 関連研究

本章では、ICN とその関連のキャッシング手法について述べ、その問題点について指摘する。

### 2.1 Information Centric Networking

ICN [1], [2] はコンテンツ指向のネットワークとして注目されており、ユーザだけでなくネットワーク内においてもコンテンツ名によって通信を行う。ユーザやサーバなどのノードは IP アドレスのような識別子を持たず、コンテンツ名が通信を行ううえで重要な識別子となる。ユーザがコ

ンテンツを要求すると、要求パケット (Interest) は対応するコンテンツを保有するサーバの方向へ転送され、サーバは応答パケット (Data) を返信する。Data にはユーザが要求したコンテンツが含まれており、ユーザはパケット単位でコンテンツを要求する。

各ルータはパケット転送を行うために、Forwarding Information Base (FIB), Pending Interest Table (PIT), Content Store (CS) の3つの要素を持つ。FIB は Interest の転送を行うための表であり、PIT は Data を転送する際に参照する表である。ノードが識別子を持たない ICN において、各ルータが Interest を受け取った方向に Data を転送し、同じ経路を逆順にたどることでユーザにコンテンツを転送する。CS はコンテンツの複製を保存するキャッシュであり、各ルータはキャッシング手法に従ってキャッシュの活用を行う。Interest の要求するコンテンツが CS 内に存在した場合、ルータは Interest を転送せずに破棄し、自身が Data を生成して返信する (キャッシュヒット)。ルータにおいてキャッシュヒットすると、ルータは PIT や FIB の参照を行う必要がなく、処理負荷が軽減される。この CS を各ルータが保有することでコンテンツを大規模に分散させることが可能となり、ユーザはサーバだけではなくルータからもコンテンツを取得できる。

### 2.2 キャッシング手法

ネットワークのキャッシュ活用能力を左右するキャッシング手法について、それを構成する3つの要素とキャッシング手法の分類に関して詳細に説明する。

#### 2.2.1 キャッシング手法の3要素

キャッシング手法は大きく Interest 転送方式、キャッシュ決定方式、キャッシュ置換方式の3つから構成される [3]。Interest 転送方式はキャッシュを受動的に活用するか、能動的に活用するかを決定する。通常の ICN ではユーザ-サーバ間経路上に要求するコンテンツが保存されていた場合にキャッシュを活用できるため、受動的なキャッシュ活用を行う。一方で、Nearest Routing Replica [3] や Breadcrumbs [4] は能動的なキャッシュ活用を行う Interest 転送方式を導入したキャッシング手法であり、コンテンツを保有するルータの方向へ Interest を誘導する。

キャッシュ決定方式は、ルータがあるコンテンツをキャッシュに保存する候補とするかどうか決定する方式であり、通常の ICN ではすべてのコンテンツを候補とする。Leave Cache Down (LCD) [5] はキャッシュ決定方式に着目したキャッシング手法であり、隣接ルータの CS から返信されたコンテンツのみを保存候補とする。

キャッシュが一杯になった際に、保存候補のコンテンツとキャッシュ内のどのコンテンツを入れ替えるかを決定するアルゴリズムがキャッシュ置換方式である。代表的なキャッシュ置換方式には最も古いコンテンツから置換す

る Least Recently Used (LRU) や単位時間あたりの要求頻度が最も低いコンテンツから置換する Least Frequently Used (LFU) がある。Perfect-LFU [6] はルータを通過したすべてのコンテンツ要求に関する要求頻度の記録を行う LFU 方式を採用しており、CS 内に存在するコンテンツに関する要求頻度の記録を行う一般的な LFU とは異なる。

### 2.2.2 キャッシング手法の分類

キャッシング手法は、ルータ間でキャッシュの活用のための特別な情報交換（協調）を行う協調型キャッシング手法と、協調を行わない非協調型キャッシング手法に分類される。協調型キャッシング手法 [3] はキャッシュの活用能力で優れているが、協調のための機構を新たに導入する必要があり、それによりネットワークへの負荷が増加する。非協調型キャッシング手法 [4], [7] は各ルータが独自に判断を行うことでネットワークへの負荷増大を抑制し、ネットワーク拡張性の面で優れている。

ネットワークへの負荷増大は ICN においても問題視されており [8], キャッシング手法を利用して負荷分散を行う手法も研究されている [9]。そこで、本論文ではネットワークへの負荷を増加させず、ネットワーク拡張性に優れた非協調型キャッシング手法に着目し、議論を行う。

## 2.3 ローカル性と非協調型キャッシング手法の課題

非協調型キャッシング手法において、ユーザ近くの下流ルータのキャッシュは一般的に自身の近くに存在するユーザの人気度を大きく反映したコンテンツを保存する。ルータのキャッシュが自身の近くの人気度を反映する性質をローカル性と呼ぶ [10]。非協調型キャッシング手法のようにローカル性が強く、ネットワークのエッジにあたる下流ルータにおいてキャッシュが活用される手法を、本論文では PopularEdge 型キャッシング手法と呼ぶことにする。

PopularEdge 型キャッシング手法では、あるネットワークドメイン内で大域的に人気のあるコンテンツが下流ルータに保存される。これによって、下流ルータで保存されなかった比較的人気のないコンテンツ要求が比較的サーバに近い上流ルータにおいて集中し、上流ルータのキャッシュ活用能力が低下する。さらに、大域的に人気のあるコンテンツが複数のルータに重複して保存され、ドメイン内でキャッシュに保存できるコンテンツの種類数が減少する。コンテンツの人気は地域によって異なるという指摘がされており [11], [12], 50%の動画コンテンツは視聴率の70%が単一の地域に集中し、コンテンツ要求の90%はネットワーク上の10%のコンテンツに集約していると報告されている [6]。このような人気の偏りの存在にかかわらず、PopularEdge 型キャッシング手法では大域的に人気のあるコンテンツが重複保存され、局所的に人気のあるコンテンツが保存されにくくなることでドメインにおけるキャッシュ活用能力が低下する。

## 3. CaSTA の提案

本章では、ICN において各ルータが収集したコンテンツの人気度と、上流からのキャッシュ情報を利用してキャッシュを活用するキャッシング手法 CaSTA (Cache Shifting scheme using Top-down Announcement for Information Centric Networking) を提案する。

### 3.1 CaSTA の概要

CaSTA はコンテンツの人気度やパケットの情報によってドメイン内における上流ルータが大域的に人気のあるコンテンツの保存を行い、ドメイン全体のキャッシュ活用能力を向上させるキャッシング手法である。よって、CaSTA はすべてのルータが同じ管理者の下で動作するドメイン内で利用されることを想定する。上流ルータが人気のあるコンテンツをキャッシュする非協調型キャッシング手法を PopularCore 型キャッシング手法と呼ぶことにする。CaSTA の目的は以下の2点にある。

- 上流ルータにおけるキャッシュを活用する。
- キャッシュ活用のためにネットワークへの負荷が増大することを抑制する。

上流ルータにおいてキャッシュヒットを誘導することで、通信が集中する上流ルータにおける処理負荷の軽減も期待できる。上記2点を達成するために、CaSTA ではルータやパケットによって新たな情報管理を行い、キャッシング手法を構成する3つの要素すべてに変更を加えることで PopularCore 型キャッシングを実現する。CaSTA におけるそれぞれの要素について、次節より詳細に述べる。

### 3.2 人気度収集のための Interest 転送方式

CaSTA において、各ルータはコンテンツの要求回数を記録し、他ルータと独立して人気度の把握を行う。各ルータは人気度把握のための要素として、Access Frequency Table (AFT) を新たに保有し、自身の AFT にコンテンツ名、単位時間あたりの要求回数 (アクセスカウンタ)、Cached フラグから構成される AFT エントリを記録する。ここで、Cached フラグとは、そのコンテンツを自身の CS に保存しているかを示す値である。Algorithm 1 に Interest 転送と AFT 更新の流れを示す。Interest が到着するごとに AFT を更新することによって、そのルータの位置におけるコンテンツの人気度を集計することが可能となる。

CaSTA の目的の1つに上流ルータにおけるキャッシュを活用することがある。しかし、下流ルータにおいてキャッシュヒットが発生した場合、その Interest は上流に転送されず、破棄される。その結果、上流ルータは人気のあるコンテンツの人気度を集計することができない。この課題を解消するために、CaSTA では Interest に *Satisfied-bit* を新たに導入し、すべての Interest をサーバまで転送する。

**Algorithm 1** Interest Forwarding and AFT update algorithm

**Input:**  $I$ , a newly arrived Interest  
**Variables:**  $CN$ , a content name indicated in  $I$   
 $E_{CN}$ , an AFT entry of  $CN$

- 1: Get  $CN$  from  $I$
- 2: **if** Exist  $E_{CN}$  in AFT **then**
- 3:   Increment  $E_{CN}.AccessCounter$
- 4:   **if** Satisfy  $I$  because of CS **then**
- 5:     Return the content to user
- 6:     Set  $I.Satisfied-bit \leftarrow True$
- 7:   **end if**
- 8: **else**
- 9:   Create  $E_{CN}$  in AFT
- 10:   Set  $E_{CN}.AccessCounter \leftarrow 1$
- 11: **end if**
- 12: Forward  $I$  to an original content server

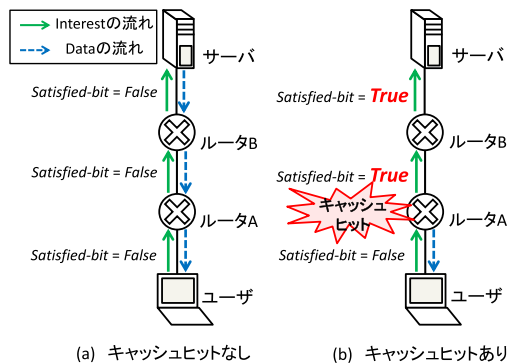


図 1 Interest 転送と Satisfied-bit

Fig. 1 Interest Forwarding and Satisfied-bit.

Satisfied-bit とは、その Interest に対して Data が返信されたかどうかを示す値である。

図 1 に Interest 転送と Satisfied-bit の関係を示す。ユーザによって Interest が生成されたとき、Satisfied-bit は False である。サーバまでの経路上でキャッシュヒットした場合は、Satisfied-bit を True に書き換えたうえで上流に転送され、ルータで Interest が破棄されることはない。サーバは Satisfied-bit が False のときに Data を返信し、True のときは経路上のルータにおいてすでに Data が返信されているため、サーバは Data を返信しない。このように、Satisfied-bit を導入することによって Interest が必ずサーバへ転送されると同時に、通常の ICN と同様にユーザから最も近い位置にあるキャッシュから Data が返信される。キャッシュ活用のための特別な情報交換を行うことなく、上流ルータはキャッシュヒットしたコンテンツの情報も把握し、より大域的な人気度の把握を行うことができる。

**3.3** キャッシュ通知によるキャッシュ決定方式

各ルータにおいて把握した人気度を用いて、大域的に人気のあるコンテンツを上流ルータで保存できるようにする。このとき、ユーザ-サーバ間で大域的に人気のあるコ

ンテンツが重複保存されると、ドメイン全体としてキャッシュに保存できるコンテンツの種類数が減り、キャッシュ活用能力が低下する。そのため、ユーザ-サーバ間でコンテンツの重複保存を回避する必要がある。これを実現するために、CaSTA では Data に Cached-bit を新たに導入する。Cached-bit は、Data を受け取ったルータより上流でその Data に含まれるコンテンツをキャッシュに保存しているルータが存在するかどうかを示す。Data が生成されたとき、Cached-bit は False である。Data を転送する際に、そのコンテンツを新たに保存した、あるいはすでに保存していたルータが Cached-bit を True に書き換える。Cached-bit が False の場合、ルータはそのコンテンツをキャッシュに保存する候補とし、3.4 節で説明する CaSTA のキャッシュ置換方式に従う。一方で Cached-bit が True の場合、その Data を受け取ったルータより上流にそのコンテンツをすでにキャッシュに保存しているルータが存在している。よって、Cached-bit が True の Data を受け取ったルータはそのコンテンツをキャッシュに保存する候補とせず、自身のキャッシュにすでに保存されている場合には CS 内から削除する。これによって、重複保存を回避し、他のコンテンツを保存するためにキャッシュを活用できる。CS 容量に空きがある場合は、Cached-bit の値にかかわらず受け取ったコンテンツをキャッシュに保存する候補とする。

**3.4** AFT を参照したキャッシュ置換方式

CaSTA におけるキャッシュ置換方式は、各ルータが AFT をもとに人気のあるコンテンツを保存するように動作する。上流ルータは大域的な人気度の把握を行い、大域的に人気のあるコンテンツを保存する。一方で、下流ルータは局所的な人気度の把握を行い、上流ルータと重複しないように局所的に人気のあるコンテンツを保存する。このように、各ルータが収集したコンテンツの人気度と、通知された上流のキャッシュ情報を総合的に判断することで上流ルータにおけるキャッシュの活用とドメイン内でキャッシュに保存できるコンテンツの種類数の増加を実現する。

Algorithm 2 に Data 転送とキャッシュ保存の流れを示す。キャッシュ容量に空きがある場合、キャッシュに保存することでキャッシュ活用を行うことができるため、Cached-bit を True にしてそのコンテンツをキャッシュに保存する。キャッシュ容量に空きがない場合、保存候補としたコンテンツの AFT エントリと CS 内にすでに保存されている他のコンテンツの AFT エントリを参照する。保存候補としたコンテンツのアクセスカウンタが CS 内に保存されているコンテンツの最小アクセスカウンタより大きい場合、最小アクセスカウンタのコンテンツを CS 内から削除し、保存候補としたコンテンツを保存する。

Data はサーバからユーザに向かって転送されるため、上流ルータほど優先的にコンテンツの保存を行う。図 2 に

**Algorithm 2** Data Forwarding and caching algorithm

**Input:**  $D$ , a newly arrived Data

**Variables:**  $CN$ , a content name indicated in  $D$   
 $E_{CN}$ , an AFT entry of  $CN$

- 1: Get  $CN$  from  $D$
- 2: **if** Exist free space in CS **then**
- 3:   Set  $D.Cached-bit \leftarrow True$
- 4:   Store the content in CS
- 5: **else**
- 6:   **if** {  $D.Cached-bit = False$  and  $E_{CN}.AccessCounter > \min(\text{Other AFT entries for contents stored in CS})$  } **then**
- 7:     Set  $D.Cached-bit \leftarrow True$
- 8:     Store the content in CS
- 9:   **end if**
- 10: **end if**
- 11: Forward  $D$  to users

**表 1** シミュレーションパラメータ  
**Table 1** Basic simulation parameters.

シミュレーション時間	1,800 sec
コンテンツサイズ	1,024 byte
CS 容量	10 [コンテンツ/router]
ユーザの要求頻度	1, 10 /sec
コンテンツの種類数	1,000
Zipf の法則 $\alpha$	0.0, 0.7, 1.0
Zipf の法則 $q$	0.7

ルとして ndnSIM 1.0 [14] を用いた. その他のシミュレーションパラメータを表 1 に示す. ルータは主にリンク帯域の制約を受けてパケット処理を行い, 処理能力はモデルには含まない.

ユーザは Zipf の法則 [6] に従ってコンテンツを要求する. Zipf の法則において,  $k$  番目のコンテンツが要求される確率  $P_N(k)$  は以下の数式で表される.

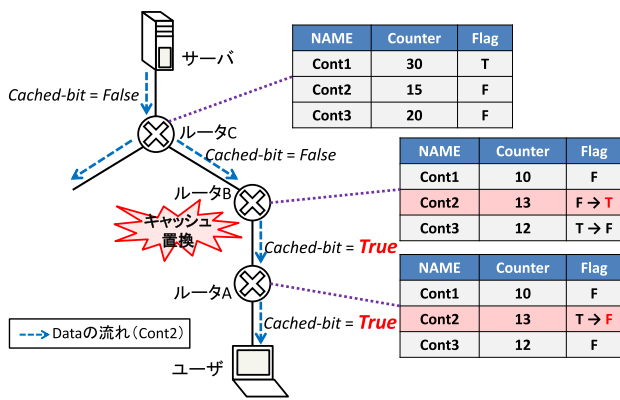
$$P_N(k) = \frac{1/(k+q)^\alpha}{\sum_{i=1}^N \frac{1}{(i+q)^\alpha}} \quad (1)$$

ここで,  $N$  はコンテンツの種類数を表し,  $\alpha$  と  $q$  は分散を制御するパラメータである.

大域的な人気のあるコンテンツと局所的に人気のあるコンテンツをシミュレーションで再現するため, ユーザの要求モデルを 2 種類用意する. 1,000 種類のコンテンツに 1 から 1000 までの ID を付与し, 一方のモデルにおいて, ユーザは ID が小さいコンテンツほど高確率で要求する. もう一方のモデルにおいて, ユーザは ID が大きいコンテンツほど高確率で要求する. シミュレーションにおいて, 前者に従うユーザは後者に従うユーザより多くなるように設定し, 前者を大域モデル, 後者を局所モデルと呼ぶことにする. 各ルータは自身に到達するユーザの人気度の和によって保存するコンテンツを決定するため, 局所モデルを大域モデルと正反対に設定することで, 最もキャッシュヒットしにくい環境を再現する. 両要求モデルにおける Zipf の法則の  $\alpha$  と  $q$  はそれぞれ同じ値に設定する.

式 (1) より, Zipf 分布による人気度の偏りへの寄与は  $q$  の変化に比べて  $\alpha$  の変化が大きい. そこで, Zipf 分布による人気度の偏りに対する各手法の基本的な性能を調べるため, まず人気度の偏りが標準的な値として  $\alpha = 0.7$  [14] の結果について議論する. その後, Zipf 分布による人気度の偏りが無い値として  $\alpha = 0.0$  を用いた場合の結果と, 人気度の偏りが極端な値として  $\alpha = 1.0$  の場合における結果について議論する.

また, シミュレーションには 5 階層の二分木トポロジとより現実的なトポロジの AS1239 [15] の 2 つを用いる. 図 3 に二分木トポロジの概形を示し, 表 2 に各トポロジの詳細な設定を示す. 二分木トポロジを用いたシミュレーションでは, 1 つのサーバに対して単一ドメイン内にある



※NAME:コンテンツ名, Counter:アクセスカウンタ, Flag:Cachedフラグ

図 2 Data 転送と *Cached-bit*

Fig. 2 Data Forwarding and *Cached-bit*.

Data 転送と *Cached-bit* の関連を示す. 図 2 のルータ B に着目すると, Cont2 というコンテンツを含む Data が到着したとき, *Cached-bit* は *False* であり, Cont2 のアクセスカウンタが最も大きい. よって, ルータ B は Cont2 を新たに保存し, *Cached-bit* を *True* にする. 次に, すでに Cont2 を保存していたルータ A に着目すると, Cont2 の Data に含まれる *Cached-bit* が *True* であるので, CS 内から Cont2 を削除する. このように, 上流ルータが優先的にキャッシュへの保存を行い, 重複保存を回避することでドメイン全体としてキャッシュに保存されたコンテンツ種類数を増加させ, キャッシュ活用能力が向上する.

**4. シミュレーション評価**

CaSTA の有用性を確認するために, シミュレーション評価を行った. 本章ではシミュレーション環境とその結果を述べ, 考察を行う.

**4.1 シミュレーション環境**

シミュレータとして ns-3.20 [13] を利用し, ICN モジュー

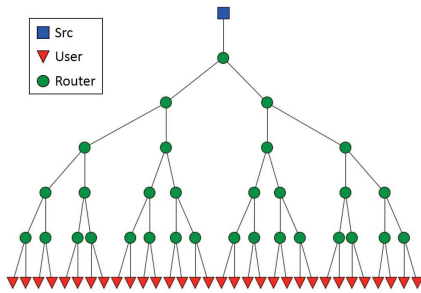


図 3 5 階層二分木トポロジ

Fig. 3 Five level binary tree topology.

表 2 各トポロジの詳細

Table 2 Simulation parameters of each topology.

トポロジ	二分木	AS1239
サーバ数	1	10
ルータ数	31	390
ユーザ数	32	200
回線容量 (ユーザー-ルータ間)	1 Gbps	1 Mbps
回線容量 (ルーター-ルータ間)	1 Gbps	2-7 Mbps
回線容量 (ルーター-サーバ間)	1 Gbps	2-7 Mbps
回線容量 (サーバー-サーバ間)	—	7 Mbps
伝播遅延 (ユーザー-ルータ間)	2 ms	10-70 ms
伝播遅延 (ルーター-ルータ間)	2 ms	5-10 ms
伝播遅延 (ルーター-サーバ間)	10 ms	5-10 ms
伝播遅延 (サーバー-サーバ間)	—	10 ms

複数のユーザが要求を行い、単純なトポロジにおける上流ルータと下流ルータの動作の違いを確認する。AS1239 は AT&T のネットワークを模したトポロジであり、より現実的で複雑な環境下における CaSTA の有用性を検証する。各トポロジのリンク帯域に合わせ、二分木トポロジを用いたシミュレーションにおいて、ユーザは毎秒 10 パケットの要求を行い、AS1239 トポロジを用いたシミュレーションにおいては毎秒 1 パケットの要求を行う。

CaSTA の有用性を確認するため、通常の ICN にキャッシュ置換方式として LFU を用いた手法と Perfect-LFU の 2 手法と比較評価を行った。LFU は ICN において一般的な PopularEdge 型キャッシング手法であり、Perfect-LFU は理想的な PopularEdge 型キャッシング手法である。なお、Perfect-LFU と提案手法 CaSTA はそれぞれキャッシングのためにルータにおいて付加的な記録の必要があるが、今回の評価ではその記憶容量は制限しない。

#### 4.2 評価項目

評価項目として、以下の 5 つを用いた。

- 平均取得時間：ユーザが Interest を生成してから対応するコンテンツを取得するまでの平均時間。
- コンテンツ種類数：1 秒ごとに算出したドメイン全体で重複なくキャッシュに保存されているコンテンツ種類数の平均値。

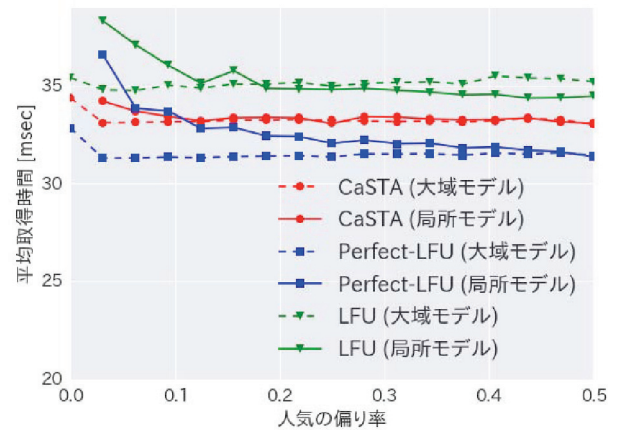


図 4 平均取得時間の比較

Fig. 4 Average data acquisition time on binary tree topology.

- キャッシュヒット率：ユーザごとのキャッシュヒットした割合の平均値。
- サーバ負荷：ドメイン全体で生成された Interest 数に対するサーバから返信した Data の割合。
- トラフィック量：ネットワーク内における 1 秒あたりの合計トラフィック量。

#### 4.3 平均取得時間

図 4 に二分木トポロジにおける人気の偏り率に対する平均取得時間の結果を示す。人気の偏り率とは、全ユーザに対して局所モデルに従うユーザの比率のことである。人気の偏り率が 0 に近いほど人気の偏りが大きく、0.5 に近いほど偏りが小さい。

図 4 より、LFU は大域モデルと局所モデルのどちらにおいても取得時間が最も長い。CaSTA は Perfect-LFU に比べて取得時間が大きいですが、人気の偏り率にかかわらず大域モデルと局所モデルの差が最も小さい。Perfect-LFU は最も取得時間が短いですが、人気の偏り率が小さいほど両モデル間の差が大きい。

Perfect-LFU が CaSTA より取得時間が小さい理由として、Perfect-LFU は PopularEdge 型のキャッシングを行い、人気のあるコンテンツが下流ルータに保存されていることが考えられる。そのため、大域モデルのユーザはコンテンツを早く取得できるが、局所モデルのユーザはキャッシュからコンテンツを取得しにくくなり、人気の偏り率が小さいほど両モデル間に差が生じる。CaSTA は PopularCore 型のキャッシング手法であるため、人気のあるコンテンツが上流ルータに保存され、大域モデルに従うユーザのコンテンツ取得時間が大きくなる。一方で、局所的に人気のあるコンテンツが下流ルータに保存される機会が増加するため、局所モデルのユーザのコンテンツ取得時間が短縮され、両モデル間の差が小さくなったと考えられる。

4.4 コンテンツ種類数とキャッシュヒット率

図 5 に二分木トポロジにおける人気の偏り率に対するコンテンツ種類数の結果を、図 6 にキャッシュヒット率の結果を示す。図 5 より、LFU は Perfect-LFU の 1.5 倍から 2 倍近くの種類が保存されており、CaSTA も LFU と同程度の種類数を実現していることが分かる。図 6 より、大域モデルと局所モデルのどちらにおいても CaSTA が最も高いキャッシュヒット率を実現していることが分かる。LFU のキャッシュヒット率は人気の偏り率にかかわらず、両モデルともに 10%程度と最も低い値になっている。また、表 3 に人気の偏り率が約 0.1 の場合における木の深さごとの平均キャッシュヒット率を示す。こ

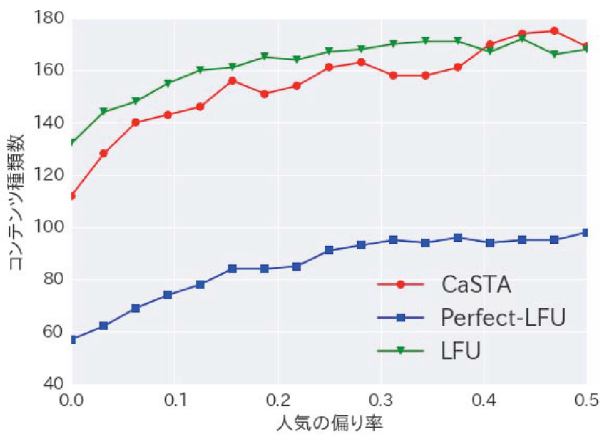


図 5 コンテンツ種類数の比較

Fig. 5 Average varieties of cached contents on binary tree topology.

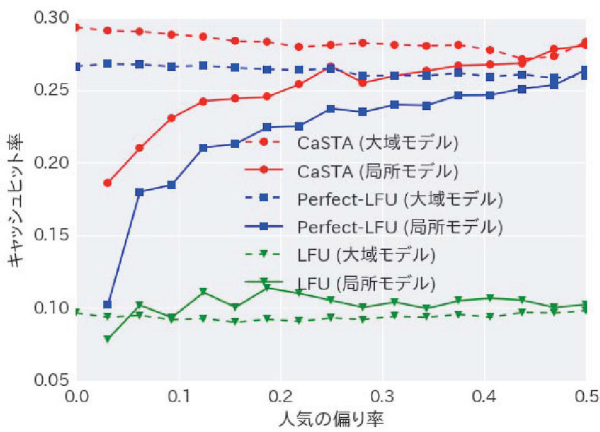


図 6 キャッシュヒット率の比較

Fig. 6 Average cache hit ratio on binary tree topology.

表 3 木の深さごとのキャッシュヒット率

Table 3 Average cache hit ratio on each level.

深さ	1	2	3	4	5
CaSTA	0.133	0.0554	0.0398	0.0309	0.0240
Perfect-LFU	0.0215	0.0248	0.0324	0.0510	0.138
LFU	0.0188	0.0171	0.0283	0.0398	0.0920

で、深さ 1 とはドメイン内における最上流ルータの平均キャッシュヒット率を表し、深さ 5 とはユーザに隣接したルータの平均キャッシュヒット率の合計を表す。表 3 より、PopularCore 型キャッシング手法である CaSTA では深さ 1 におけるキャッシュヒット率が最も高くなった一方で、PopularEdge 型キャッシング手法である Perfect-LFU と LFU は深さ 5 におけるキャッシュヒット率が最も高いことが分かる。また、CaSTA の下流ルータにおけるキャッシュヒット率は PopularEdge 型キャッシング手法の上流ルータに比べて高かった。

これらの結果から、CaSTA はより多種類のコンテンツをキャッシュに保存し、保存されたコンテンツが活用されていることが分かる。つまり、キャッシュヒットする可能性があるコンテンツを幅広くキャッシュに保存することに成功している。これは、木の深さごとのキャッシュヒット率の結果から分るとおり、上流ルータに大域的に人気のあるコンテンツを保存することで、下流ルータに局所的に人気のあるコンテンツを保存する機会を増加させる PopularCore 型キャッシングが機能していることが理由として考えられる。一方で、上流ルータにおけるパケット処理が増加していることから、提案手法による上流ルータの負荷増大が考えられる。この上流ルータにおける負荷増大は対策を行う必要があるが、今回の提案手法には含まず、今後の検討課題とする。LFU ではコンテンツ種類数が多いにもかかわらず、キャッシュヒット率が低く、保存されているコンテンツの大部分が活用されていないということが分かる。Perfect-LFU はコンテンツごとのキャッシュヒット率が高くなっているが、大域的に人気のあるコンテンツ以外はキャッシュに保存されにくく、経路上にコンテンツが複数保存されていることが推測できる。

4.5 サーバ負荷

図 7 は二分木トポロジにおける人気の偏り率に対する

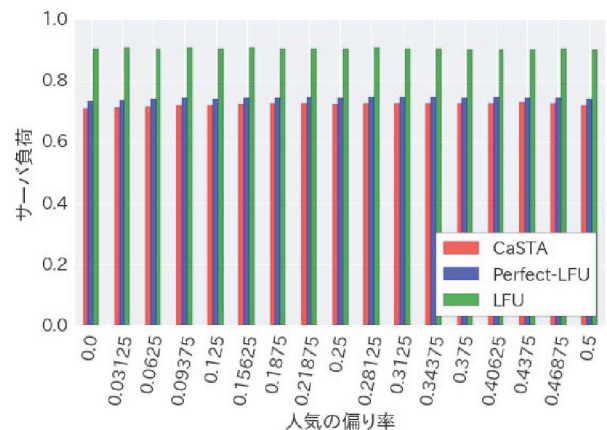


図 7 サーバ負荷の比較

Fig. 7 Average traffic load at server on binary tree topology.

サーバ負荷の結果を示す。図 7 より、CaSTA は Perfect-LFU と同程度のサーバ負荷を実現していることが分かる。一方で、LFU は CaSTA や Perfect-LFU に比べてサーバ負荷が大きい。

この結果と 4.4 節の結果より、CaSTA は多種類のコンテンツをキャッシュに保存してキャッシュヒット率を向上させた結果、サーバが Data を返信する回数を低減させ、サーバ負荷を軽減できていることが分かる。ここで、CaSTA は Interest を必ずサーバまで転送するが、サーバ近くの回線容量に対する Interest の通信負荷やサーバにおける処理負荷は、コンテンツを含む Data の通信負荷や処理負荷に比べて十分に小さい。よって、サーバが返信した Data の割合が小さくなることは、サーバ負荷が軽減できているといえる。

#### 4.6 現実的なトポロジへの適応

AS1239 を用いてより現実的なトポロジにおける CaSTA の有用性を検証する。この評価では、コンテンツ要求の 90% がネットワーク上の 10% のコンテンツに集約しているという報告に基づき [6]、人気の偏り率を約 0.1 に設定した場合の結果を示す。二分木トポロジと AS1239 における CaSTA と Perfect-LFU の結果を表 4、表 5 にそれぞれまとめた。これらの表より、平均取得時間とトラフィック量に関しては Perfect-LFU が小さく、CaSTA が相対的に大きくなっていることが分かる。一方で、キャッシュヒット率やサーバ負荷の評価に関してはどちらのトポロジにおいても CaSTA が有効に働いていることが分かる。

この結果より、CaSTA はキャッシュヒット率やサーバ負荷の結果で Perfect-LFU より優れ、現実的なトポロジにおいても高いキャッシュ活用能力をドメインに提供できることが確認された。これは、上流において大域的に人気のあるコンテンツを保存することでキャッシュの活用を行う

表 4 二分木トポロジにおける結果

Table 4 Quantitative evaluation on binary tree topology.

手法 モデル	CaSTA		Perfect-LFU	
	大域	局所	大域	局所
平均取得時間	33.2 ms	33.4 ms	31.4 ms	33.7 ms
キャッシュヒット率	0.289	0.231	0.266	0.185
サーバ負荷	0.717		0.741	
トラフィック量	10.8 Mbps		8.73 Mbps	

表 5 AS1239 における結果

Table 5 Quantitative evaluation on AS1239 topology.

手法 モデル	CaSTA		Perfect-LFU	
	大域	局所	大域	局所
平均取得時間	202 ms	154 ms	187 ms	146 ms
キャッシュヒット率	0.181	0.126	0.143	0.101
サーバ負荷	0.828		0.864	
トラフィック量	4.02 Mbps		3.39 Mbps	

と同時に、ドメイン内における重複保存を回避する CaSTA の仕組みが有効に働いたことが理由として考えられる。一方で、平均取得時間に関しては複雑で現実的なトポロジにおいて性能が低下することが確認された。これは、AS1239 は下流ルータより上流ルータの数が多いため、上流ルータのキャッシュを活用する CaSTA では大域的に人気のあるコンテンツが分散され、取得に時間がかかるようになったと考えられる。

トラフィック量はどちらのトポロジにおいても CaSTA が大きくなることが確認された。一般に、Data のサイズは Interest のサイズよりはるかに大きいため、ネットワーク内のトラフィック量は Data のトラフィック量に大きく左右される。CaSTA において、ドメイン内のキャッシュ活用能力が向上してドメインとサーバ間のトラフィック量を削減する一方で、下流ルータにおけるキャッシュヒット率が低下することでドメイン内のトラフィック量が増加するというトレードオフが存在する。今回用いた両トポロジは上流ルータとサーバ間が短距離リンクであるが、このリンクが長距離リンクの場合にはサーバからコンテンツ取得を行う比率が高くなることでコンテンツ取得時間が大きくなることが考えられる。つまり、上流ルータとサーバ間が長距離であれば、ドメイン内のキャッシュ活用能力の向上が期待できる提案手法を用いる利点が強くなる。以上より、CaSTA を AS1239 で用いたシミュレーションでは平均取得時間やトラフィック量で性能の低下がみられるものの、ドメイン全体としてのキャッシュ活用能力を向上させ、サーバ負荷を軽減できることが確認された。

#### 4.7 人気度の偏りへの適応

本節では、Zipf 分布による人気度の偏りを変化させた場合の二分木トポロジにおける結果について議論する。 $\alpha = 0.0$  の一様分布の場合における CaSTA と Perfect-LFU の結果を表 6 に示し、 $\alpha = 1.0$  の偏りが大きい場合における結果を表 7 に示す。ここで、一様分布の場合には大域モデルと局所モデルに差がなく、ともに同じ分布となる。

表 6 より、一様分布の場合にはどちらの手法にも差がなく、平均取得時間やサーバ負荷、トラフィック量が増加し、キャッシュヒット率が低下することが分かる。表 7 より、Zipf 分布による人気度の偏りが大きい場合には Perfect-

表 6 二分木トポロジにおける  $\alpha = 0.0$  の結果

Table 6 Quantitative evaluation on binary tree topology ( $\alpha = 0.0$ ).

手法	CaSTA	Perfect-LFU
平均取得時間	38.8 ms	38.9 ms
キャッシュヒット率	0.0497	0.0429
サーバ負荷	0.950	0.957
トラフィック量	10.9 Mbps	10.6 Mbps



表 7 二分木トポロジにおける  $\alpha = 1.0$  の結果Table 7 Quantitative evaluation on binary tree topology ( $\alpha = 1.0$ ).

手法 モデル	CaSTA		Perfect-LFU	
	大域	局所	大域	局所
平均取得時間	27.8 ms	27.2 ms	22.5 ms	26.1 ms
キャッシュヒット率	0.533	0.468	0.532	0.399
サーバ負荷	0.474		0.480	
トラフィック量	10.4 Mbps		5.94 Mbps	

LFU のキャッシュヒット率が大幅に上昇し、平均取得時間やトラフィック量の減少が確認できる。一方、CaSTA においてはトラフィック量がほとんど変化しなかったものの、キャッシュヒット率が上昇し、平均取得時間やトラフィック量の減少が確認された。

Zipf の法則による人気度の偏りが大きくなると、人気のあるコンテンツにより要求が集中するようになる。Popular-Edge 型である Perfect-LFU では人気のあるコンテンツを保存している下流ルータにおいてキャッシュヒットする確率が上昇し、PopularCore 型である CaSTA は上流ルータにおいてキャッシュヒットする確率が上昇する。よって、Zipf の法則による人気度の偏りが大きい場合には、CaSTA と Perfect-LFU の性能差が小さくなるものの、どちらの手法もキャッシング活用能力が上昇する。このことから、Zipf の法則による人気度の偏りが大きい場合には CaSTA を含めた 3 手法の性能が低下するが、偏りが大きくなった場合にも CaSTA は有効であることが確認された。

## 5. おわりに

本論文では、ICN において各ルータが収集したコンテンツの人気度と、上流からのキャッシュ情報を利用してキャッシュを活用する PopularCore 型キャッシング手法 CaSTA を提案した。CaSTA は大域的に人気のあるコンテンツを上流ルータで優先的に保存し、ドメイン内における重複保存を回避することでドメインのキャッシュ活用能力向上を実現する。

CaSTA の有用性を検証するため、シミュレーションによる比較評価を行った。二分木トポロジにおける結果より、CaSTA はキャッシュ活用によるサーバへの負荷増大を抑え、局所モデルに従うユーザに対して大域モデルに従うユーザと同程度の取得時間を提供した。また、LFU と同程度のコンテンツ種類数を保存し、Perfect-LFU と同程度のキャッシュヒット率を維持したことからドメイン全体のキャッシュ活用能力向上が確認できた。さらに、AS1239 における結果から、より現実的で複雑なトポロジにおいても PopularEdge 型キャッシング手法に比べてキャッシュ活用能力向上を実現し、サーバへの負荷を軽減できた。以上より、キャッシュ活用能力向上やサーバ負荷軽減の点で CaSTA の有用性を確認できた。

謝辞 本研究は JSPS 科研費 16H02811 の助成を受けたものです。

## 参考文献

- [1] Jacobson, V., Smetters, D.K., Thornton, J.D., Plass, M.F., Briggs, N.H. and Braynard, R.L.: Networking Named Content, *Proc. 5th International Conference on Emerging Networking Experiments and Technologies, CoNEXT '09*, pp.1–12 (2009).
- [2] Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K., Crowley, P., Papadopoulos, C., Wang, L. and Zhang, B.: Named Data Networking, *ACM SIGCOMM Computer Communication Review (CCR)*, Vol.44, No.3, pp.66–73 (2014).
- [3] Rossini, G. and Rossi, D.: Coupling Caching and Forwarding: Benefits, Analysis, and Implementation, *The 1st International Conference on Information-centric Networking (ICN)*, pp.127–136 (2014).
- [4] Rosensweig, E. and Kurose, J.: Breadcrumbs: Efficient, Best-Effort Content Location in Cache Networks, *INFOCOM 2009*, pp.2631–2635, IEEE (2009).
- [5] Laoutaris, N., Syntila, S. and Stavrakakis, I.: Meta algorithms for hierarchical Web caches, *2004 IEEE International Conference on Performance, Computing, and Communications*, pp.445–452 (2004).
- [6] Breslau, L., Cao, P., Fan, L., Phillips, G. and Shenker, S.: Web caching and Zipf-like distributions: Evidence and implications, *INFOCOM '99, Proc. 18th Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol.1, pp.126–134 (1999).
- [7] Hu, X. and Gong, J.: Distributed in-network cooperative caching, *2012 IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pp.735–740 (2012).
- [8] Umeda, S., Kamimoto, T., Ohata, Y. and Shigeno, H.: Interest Flow Control Method Based on User Reputation and Content Name Prefixes in Named Data Networking, *The 2015 IEEE International Symposium on Recent Advances of Trust, Security and Privacy in Computing and Communications (RATSP2015)*, Vol.1, pp.710–717 (2015).
- [9] 森 研太, 神本崇史, 重野 寛: コンテンツ指向型ネットワークにおける帯域を考慮した分散キャッシング手法, *情報処理学会論文誌*, Vol.57, No.2, pp.611–619 (2016).
- [10] Kamimoto, T., Mori, K., Umeda, S., Ohata, Y. and Shigeno, H.: Cache protection method based on prefix hierarchy for content-oriented network, *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pp.417–422 (2016).
- [11] Brodersen, A., Scellato, S. and Wattenhofer, M.: YouTube Around the World: Geographic Popularity of Videos, pp.241–250 (2012).
- [12] YouTube: YouTube Trends Map, available from (<https://www.youtube.com/trendsmap>) (2016).
- [13] NS-3 Consortium: ns3, available from (<http://www.nsnam.org/>) (2016).
- [14] Afanasyev, A., Moiseenko, I. and Zhang, L.: ndnSIM: NDN simulator for NS-3, Technical report, NDN (2012).
- [15] Afanasyev, A., Mahadevan, P., Moiseenko, I., Uzun, E. and Zhang, L.: Interest flooding attack and countermeasures in Named Data Networking, *IFIP Networking Conference, 2013*, pp.1–9 (2013).



神本 崇史 (学生会員)

2015年慶應義塾大学理工学部卒業。  
現在、同大学大学院理工学研究科前期  
博士課程在学中。



佐藤 和也 (学生会員)

2016年慶應義塾大学理工学部卒業。  
現在、同大学大学院理工学研究科前期  
博士課程在学中。



重野 寛 (正会員)

1990年慶應義塾大学理工学部計測工  
学科卒業。1997年同大学大学院理工  
学研究科博士課程修了。現在、同大学  
理工学部教授。博士(工学)。情報処  
理学会論文誌編集委員、同高度交通シ  
ステム研究会幹事等を歴任。現在、情  
報処理学会マルチメディア通信と分散処理研究会主査、電  
子情報通信学会英文論文誌B編集委員、Secretary of IEEE  
ComSoc APB。ネットワーク・プロトコル、ITS等の研究  
に従事。著書『ユビキタスコンピューティング』(オーム  
社)、『情報学基礎第2版』(共立出版)等。電子情報通信学  
会、IEEE、ACM各会員。