

# コンテンツ流通網における インターネットワーク誘導情報の配布制御に関する評価

## Evaluation on Distribution Control of In-network Guidance Information in Content Centric Network

柿田 将幸† 谷川 陽祐† 戸出 英樹†  
Masayuki Kakida Yosuke Tanigawa Hideki Tode

### 概要

近年、大容量コンテンツ配信に起因するサーバ負荷の増大が問題となっている。コンテンツキャッシュを用いてサーバの負荷軽減を実現しつつ、コンテンツ指向なネットワークを自律的に形成する手法として、コンテンツ名に基づいてキャッシュへの誘導を行う **Breadcrumbs** と呼ばれる方式が提案されている。また、筆者らは、特定状況下でクエリがルータ間をループするという本方式の問題を解決した **Breadcrumbs+**方式を提案している。従来方式は、ネットワーク上のルータがコンテンツをキャッシュすることを前提としているが、本稿では、キャッシュ機能をユーザに移行させることでさらなる負荷分散を促進する。さらに、誘導情報の積極的な配布により、ユーザキャッシュの利用を促進する方式 **Active Breadcrumbs** を提案し、計算機シミュレーションによりその有効性を示す。

### 1. はじめに

近年、大容量コンテンツ配信に起因するサーバ負荷の増大が問題となっており、解決策としてサーバ以外のノードにキャッシュをもたせるアプローチが存在する [1]。また、新世代ネットワークにおける枠組みとして、コンテンツ指向ネットワークという概念が提案されている [2]。これは、現在のネットワークが IP アドレス等の位置情報を中心に運用されている一方で、ユーザの興味はコンテンツそのものにあり、位置情報にあるのではないという観点から生まれたものである。このような背景から、i3 [3]や DONA [4]に代表される“キャッシュ”と“コンテンツ指向”というアイデアを組み合わせた研究が注目されつつある [5]。Van Jacobson らが提唱する **Content Centric Network** [6]も同様のコンセプトに基づいており、コンテンツの名前空間に基づくルーチング原理が提案されている。このような研究の一つとして、**Breadcrumbs(BC)** [7] [8]と呼ばれるコンテンツキャッシュへの誘導情報を扱う方式が提案されている。本方式では、コンテンツキャッシュを用いてサーバの負荷軽減を実現しつつ、コンテンツ指向なネットワークを自律的に形成する。また、筆者らは、クエリがルータ間をループ上にルーチングされ続けてコンテンツに到達できなくなるという **BC**方式の問題を解決した拡張方式 **Breadcrumbs+(BC+)**を提案している [9]。

一方、**BC**および **BC+**方式は、ネットワーク上のルータのみがコンテンツをキャッシュすることを前提に設計さ

表 1 BC エントリ

| BC エントリ      | 説明               |
|--------------|------------------|
| ContentID    | コンテンツ ID         |
| UpHopS       | 上流ノード ID(複数登録可)  |
| DownHop      | 下流ノード ID         |
| DownloadTime | コンテンツが最後に転送された時刻 |

れている。しかし、この環境ではルータがルーチング以外のタスクで処理能力を消費する問題や十分なキャッシュ容量を確保するのが困難であるという問題が生じる。言い換えれば、**BC** および **BC+**方式は、上記の点で十分な性能が確保された将来のネットワーク基盤を前提としている。

これに対し、本稿では、ユーザがキャッシュを行う、より現実的なネットワーク環境を想定する。キャッシュ機能をルータからユーザに移行して **P2P** ライクなキャッシュ運用を行うことで、広大なキャッシュスペースを確保するとともにルータにかかるキャッシュ機能の負荷をユーザに分散することができる。さらに、本環境において、ユーザキャッシュの利用を促進すると同時に、サーバの負荷とコンテンツ取得遅延を低減させることを目的とした方式(**Active Breadcrumbs; ABC**)を提案する。本方式では、誘導情報を従来より積極的に配布することで、コンテンツ誘導先を適切に制御する。また、シミュレーションを用いた評価により、その有効性を示す。

### 2. Breadcrumbs+の概要

本章では、提案方式のベースとなる **BC+**方式[9]の概要を説明する。なお、**BC+**方式は、**BC**方式 [7]に存在する特定状況下でクエリがルータ間をループ状に転送され続ける問題を解決するために、エントリの一部を拡張し、誘導情報を無効化する方式を改良したものである。以降、キャッシングポリシーは **Least Recently Used(LRU)**規律が採用されているものとする。

#### 2.1 Breadcrumbs による誘導

本方式では、ユーザがサーバからコンテンツをダウンロードする際、その経路上の各ノードがコンテンツキャッシュに加えて“**breadcrumbs**”(BC)という誘導情報を記録する。BCのエントリには、表 1 に示す 4 種類の項目が記録されている。ここで、上流とはコンテンツが転送されてきた方向、下流とはコンテンツが転送されていった方

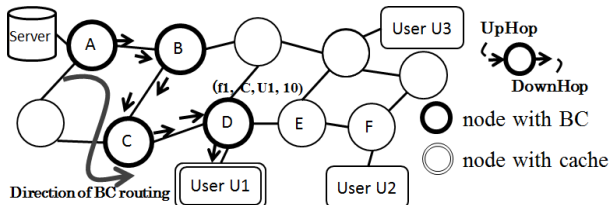


図1 BC trail の例

向を指す。ContentID, DownHop, DownloadTime は 1 エントリにつき 1 情報のみが、UpHopS については、1 エントリにつき 1 つ以上の情報が記録される。あるコンテンツのダウンロード経路上のノードに新たなクエリが到達した場合、対応する BC に記録されている DownHop を順に辿ることによって、過去にダウンロードされたコンテンツの後を追うことができる。この一連の BC の集合を trail と呼ぶ。UpHop または DownHop が null となる場合、そのノードが BC trail の末端であることを表している。クエリがコンテンツの後を追うように誘導されるのは、上流よりも下流のノードにキャッシュが残っている可能性が高いためである。これは各ノードが LRU 規律に基づいてキャッシュを保持していることによる。

クエリがキャッシュを保持するノードに辿りついた場合、そのノードは最短経路に基づく IP ルーチングに従いキャッシュされた当該コンテンツをユーザに返送する。

### 3. ユーザキャッシュを用いた運用

従来方式のBC [7] [8]およびBC+ [9] では、ネットワーク上のルータのみがコンテンツをキャッシュすることを想定している。しかし、これはルータにルーチング以外の機能をもたせることを意味し、結果的にルーチング以外のタスクで処理能力を消費することになる。また、ネットワーク上に存在するコンテンツの種類、大きさは多様であり、各ルータにおいて十分なキャッシュ容量を確保するのは困難である。そのため、本研究では、ルータではなくユーザがキャッシュを行う、より現実的なネットワーク環境を想定する。キャッシュ機能をルータからユーザに移行してP2P ライクなキャッシュ運用を行うことで、広大なキャッシュスペースを確保するとともにルータにかかるキャッシュ機能の負荷をユーザに分散させることができる。しかし、BC, BC+方式において、ユーザキャッシュのみを用いた場合、クエリがBCによってコンテンツから遠ざけられてしまう問題が生じる。以下にその具体例を挙げる。以降では、ルータキャッシュを使用せずに、ユーザキャッシュのみを用いた運用を前提とする。

図1はユーザU1があるコンテンツをサーバからダウンロードした直後の状態を表している。ダウンロード経路上のルータA, B, C, DにBCが記録され、例えばルータDのBCエントリにはコンテンツIDがf1, UpHopSがルータC, DownHopがユーザU1, 最後にコンテンツが転送された時刻が10という情報が記載されている。その後、次の1)~4)が順に生じた後の状態を図2に示す。

- 1) ユーザU2が同コンテンツのクエリqをサーバに送信
- 2) qはルータDにおいてBCにアクセス
- 3) qはユーザU1へ誘導され、キャッシュに到達
- 4) U1からダウンロードされたコンテンツがU2に到着  
この際、ルータD上のBCが更新されると同時に、ユーザU1のキャッシュはLRUポリシーにより消去されるまで

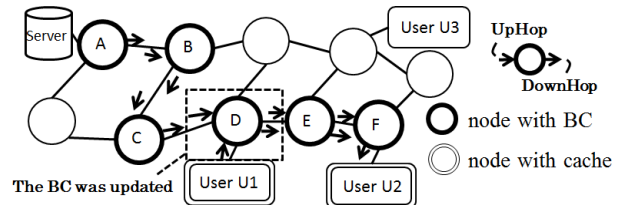


図2 コンテンツから遠ざけられるクエリ誘導

表2 ABCエントリ

| ABCエントリ     | エントリ説明     |
|-------------|------------|
| ContentID   | コンテンツID    |
| CacheNodeID | キャッシュノードID |
| LifeTime    | タイムアウトする時刻 |

の時間が延びる。ここで、ルータDのBC更新の際にDownHopがユーザU1からルータEに書き換えられていることに注意する。これにより、ユーザU1のキャッシュが延命されたにもかかわらず、この後にルータDが受信する新たなクエリは、あと1ホップの場所にあるユーザU1のキャッシュに到達できなくなってしまう。

同様のことがサーバの隣接ルータにおいても発生する。図2の場合、ルータAに到達したクエリは、物理的にあと1ホップで、確実にコンテンツを取得できるサーバに到達できる。しかし、クエリはBCに従いキャッシュノードへ誘導されるので、クエリのホップ数が増加する。このような場合、サーバの処理能力に余裕があればサーバで処理、そうでなければキャッシュへ誘導という適応的な判断を行うことが望ましい。

### 4. 提案方式(Active Breadcrumbs)

本方式は、従来の誘導情報BCに加え、BCに優先する特殊なActive Breadcrumbs(ABC)を用いて、コンテンツ保持ノードが自ら能動的に誘導情報を配置する。なお、ABCには、以下に示す動作原理上、表2に示す3種類の項目が記録されている。

#### 4.1. 誘導情報(ABC)の配布制御

本方式では、キャッシュノードから送信されるメッセージを用いて、ABCの配布や無効化に関する制御が行われる。具体的には、コンテンツの人気度を反映しながらABCを拡散させる。一方、キャッシュノードにかかる負荷が高い場合、ABCを無効化することで当該ノードが処理するクエリ量を抑制する。

キャッシュノードは、自身の負荷状況や当該キャッシュの人気状況に応じて算出されるポテンシャル分布に従い、ABCを配布する周辺のノード集合を決定する。その分布範囲の決定の際に考慮すべき代表的な指標として、メッセージのTTLとABCの総量の2つが挙げられる。決定したノード群に配布メッセージを送信することで、ABCを配置する。なおキャッシュノードは、ABCを配布する対象となる全ノード群へのルーチングテーブルを予め作成しておく。これにより、ABCを配布するノード群に対するメッセージ伝達経路の再現性確保とルート計算コストの低減が図られる。なお、キャッシュノードが各自でABCの配布に関する判断を下すため、異なるキャッシュノードからのABC配布メッセージが同一のルータに届く可能性がある。そこで、ルータは同一コンテンツに

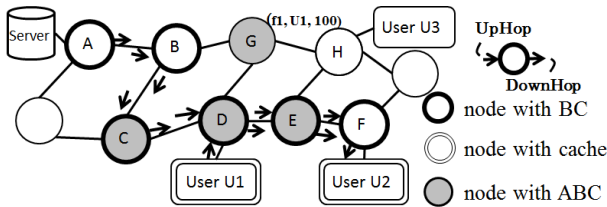


図3 ABC 配布の例

対する複数の ABC を保持することができ、より先に配布された ABC ほど高い優先度で参照されるものとする。

キャッシュノードにおいて、キャッシュがLRUにより削除された場合は、当該ABCに対する無効化メッセージを、配布メッセージを送信したノード群に送信する。なお、キャッシュノードから送信された無効化メッセージが輻輳などにより失われてしまった場合を想定し、ABCにはLifeTimeが設定されている。ABCにアクセスした時刻がLifeTimeよりも大きかった場合、そのABCは無効化メッセージが失われたものと判断され、無効化される。

#### 4.2. ABCによるクエリ誘導

あるコンテンツを要求するクエリが当該コンテンツIDのABCを保持するノードに到達した場合、クエリはABCに記録されているキャッシュノードまでIPルーチングに基づいて転送される。クエリがキャッシュノードに到達した時点でキャッシュが削除されていた場合、再びサーバに向かって転送される。また、本方式では、同一コンテンツに対するBCとABCを1つのノードが同時に保持できる。ただし、当該ノードにクエリが到達した場合、高い優先権をもつABCに基づいて誘導されるものとする。

ABCの拡散の例を図3に示す。ユーザU1がABCを周辺ノード(ルータC, D, E, G)に配布したとする。なお、配布されているABC内の情報はすべて同一であるため、ノードGの保持するABC以外の表記を省略した。ノードGのABCに記録されている情報は、ContentIDが“fl”, CacheNodeIDが“U1”, LifeTimeが“100”であることを示している。例えば、ユーザU3がクエリをサーバに送信した場合、従来ならクエリはノードBまでIPルーチングされてからBCに従ってユーザU2まで誘導されていた。ABCを用いた場合、クエリはノードGにおいてABCにアクセスし、ノードG-D-U1の順にIPルーチングされる。ここで、ABCはBCに優先するため、BCの誘導(図中の矢印の向き)とは異なる方向に誘導されていることに注意する。このようにクエリは、キャッシュノードの近隣に存在するABCの誘導に従うことで、より短いホップ数でキャッシュに到達することができる。

#### 4.3. サーバ用ABC

3章で述べた、BCによってサーバ付近まで到達したクエリがコンテンツから離れる方向に誘導される問題に対応するために、サーバが近隣ルータに到達したクエリの制御に用いるサーバ用のABCを定義する。制御法は通常のものと同様であり、優先度はBCより高く、通常ABCよりも低いものとする。

サーバ用ABCの配布・無効化を制御することで、サーバは自身の負荷状況に応じてクエリ処理量の制御を行うことができる。処理能力に余裕がある場合、サーバ用ABCを配布することで、従来はBCによって遠ざけられていたクエリを当該サーバで処理し、クエリのホップ数が

表3 シミュレーションで使ったパラメータ

| パラメータ               | 値    |
|---------------------|------|
| ルータ数                | 1000 |
| ユーザ数                | 5000 |
| サーバ数                | 50   |
| コンテンツ数              | 1000 |
| 1ユーザがキャッシュ可能なコンテンツ数 | 2    |
| 1ユーザ当たりの平均クエリ発生間隔   | 75   |
| $T_f$               | 500  |

増大するのを防ぐ。一方、処理能力に対して、多くのクエリが到着する場合には、中～高人気度をもつコンテンツに対するサーバ用ABCを無効化し、当該コンテンツに対するクエリはBCに従わせることでキャッシュまで誘導させ、サーバに過負荷がかかることを防ぐ。

## 5. 性能評価

提案方式の有効性を確認するために、C++を用いて作成したシミュレータにより性能評価を行う。

### 5.1. シミュレーション条件

シミュレーションで使った諸パラメータを表3に示す。なお、1パケットが隣接のノード間で転送される遅延を1単位時間とする。また、その他の各条件を以下に示す。

#### ・ネットワークトポロジ

ルータのネットワークトポロジはWaxmanモデル[10]に従う。サーバはルータのいずれかに一様分布に従ってランダムに接続され、ユーザはすべてのルータに5つずつ接続されている。また、作成されたトポロジに対し、任意のルータ間の距離はたかだか16ホップである。

#### ・ABCの配布ルール

簡単化のために、ABCの配布制御は、各ユーザノードにおけるコンテンツキャッシュのダウンロード回数に対して静的に行う。具体的には、各キャッシュノードにおいて、ダウンロード回数が一定回数(1,3,8)を超えたコンテンツに対し、対応するTTL(1,2,3)を設定したABC作成メッセージのフラッディングにより配布する。また、サーバ用ABCを適用する場合は、シミュレーション開始時にサーバと隣接しているルータにのみ配布する。人気度やノードの負荷をより細やかに反映する動的な配布制御は今後の課題とする。

#### ・各コンテンツに対するクエリの発生

各ノードにおいて、独立同一な指数分布に従うランダムな時間間隔でクエリ要求イベントが発生し、人気度を加味した「変則Zipfの法則」に従ってコンテンツを要求する。ここで、変則Zipfの法則とは、同程度の人気度をもつコンテンツがn個ずつ存在すると仮定したZipfの法則である。本評価ではn=10に設定した。

#### ・各種パケットのサイズ

BC方式7)の性能評価モデルに従い、全パケットサイズを一定とし、クエリ、コンテンツ、各種制御用パケットはそれぞれ1個、10個、1個のパケットから構成されているとする。

表4 平均ホップ数

| 方式           | クエリ   | コンテンツ |
|--------------|-------|-------|
| IP           | 9.613 | 9.613 |
| BC+          | 17.55 | 7.877 |
| ABC w/o SABC | 10.62 | 7.126 |
| ABC          | 9.193 | 6.955 |

表5 単位時間当たりのネットワーク中パケット量

| 方式           | パケット総量 | IM    | ABCM  |
|--------------|--------|-------|-------|
| IP           | 7047.3 | 0     | 0     |
| BC+          | 6708.0 | 354.0 | 0     |
| ABC w/o SABC | 5863.7 | 342.4 | 122.8 |
| ABC          | 5636.9 | 340.4 | 105.3 |

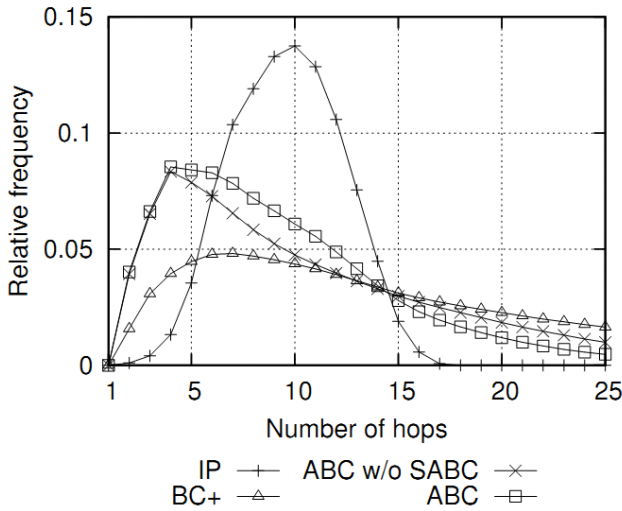


図4 クエリホップ分布

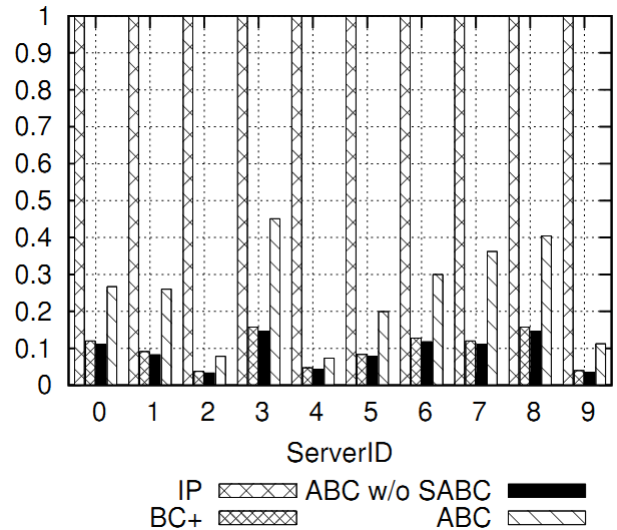


図6 サーバ集中度

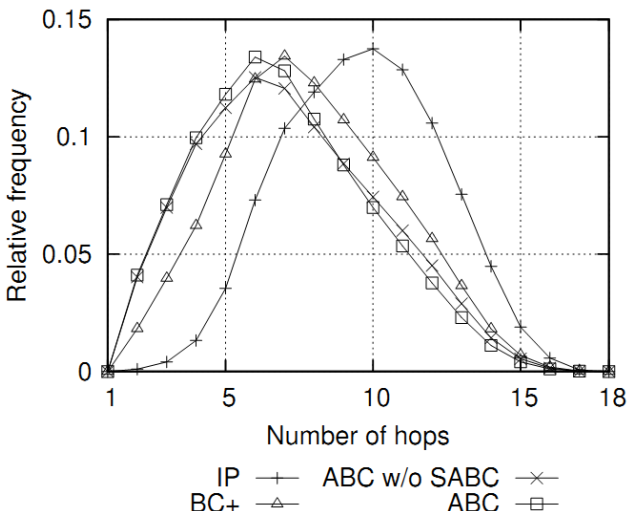


図5 コンテンツホップ分布

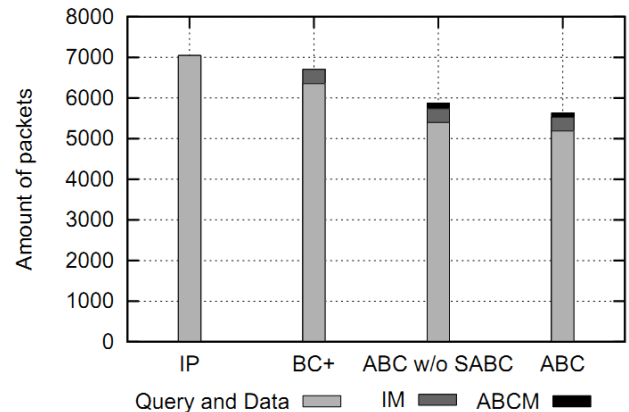


図7 単位時間あたりのネットワーク中のパケット量

配布する方式)

4) ABC w/o SABC: サーバ用ABC を利用しない提案方式

・各種遅延

ベースとしたBC方式[7]に従い、輻輳が発生しないネットワークを近似的に表現し、シミュレーションを行う。具体的には、今回のシミュレーションで扱うルータは各々が十分な処理能力をもつと仮定し、伝送遅延、キューイング遅延といった各種遅延は一定値をとるものとした。1パケットが隣接ノード間で送受信される際の伝送遅延が1であり、シミュレータ内ではこれが単位時間として扱われる。

・評価対象

評価対象は以下の4種である。

- 1) IP: キャッシュを用いず、全クエリがサーバへIPルーティングされる方式
- 2) BC+: BC+方式
- 3) ABC: 提案方式(BC+に加えて、ABC, サーバ用ABCを

5.2. 実験結果

実験結果を表4, 5および図4-図7に示す。ここで、評価指標となっているのは平均クエリホップ数、平均コンテンツホップ数、サーバ集中度およびネットワーク中のパケット量の4つである。平均クエリホップ数とは、ユーザがクエリを送信してからコンテンツに到達するまでに要したホップ数の平均値である。平均コンテンツホップ数とは、コンテンツがユーザに届くまでに要したホップ数の平均値である。サーバ集中度とは、各ユーザノードで発生したクエリのうち、実際にサーバからコンテンツを取得したクエリの割合を表す。この値が低いほど、キャッシュに到達したクエリの割合が大きく、キャッシュを効率的に利用していることになる。なお、シミュレーション内ではサーバの数は50であったが、得られた傾向が



どのサーバでも同じであったことと紙面の都合により、グラフに掲載するサーバは10個(ID 0~9)のみとする。また、図7において、IMとABCMとはそれぞれBC+9)にて使用される制御用パケット、ABCの配布・無効化用の制御用パケットを指す。

実験結果より、各方式を総合的に評価する。

### 5.2.1 BC+方式とABCおよびABC w/o SABC方式の比較

まず、BC+とABCを利用した方式(ABCとABC w/o SABC)を比較すると、ABCを利用した方式のほうが、ユーザから近い位置でキャッシュに到達しているクエリが多いことがわかる。図4において、10以下のホップ数をとるクエリ数に注目すると、ABCを利用した方式がBC+方式を大きく上回っている。同様に、15以上のホップ数をとるクエリ数に注目しても、ABCを利用した方式はBC+方式を下回っている。この裾野分布の形成は、BC trailの長さに制限がないことが原因と考えられる。これらの方式間の差より、BCに優先するABCによる誘導によってクエリがキャッシュに到達しやすくなっていることがわかる。また、図5において、ABCを利用した方式はBC+方式よりもグラフ全体がより小さなホップ数をとる方向に偏っている。この点からも、ABCによってユーザに近い位置からのコンテンツ取得が実現したことがわかる。

さらに、ABCを利用した方式では、BC+方式に比べて制御用パケットの種類が増えているにもかかわらず、ネットワーク中のパケット量を低く抑えることができている。これは、制御用パケットの転送量が増えても、それ以外のパケット転送量がより大きく減少していることを示している。特に、クエリなどに比べてサイズの大きなコンテンツがユーザ近辺のノードから取得できている効果が大きく、パケット転送量を減少させている主な要因と考えられる。

ここで、特にBC+とABC w/o SABCを比較すると平均クエリホップ数、平均コンテンツホップ数、サーバ集中度、ネットワーク中パケット量のすべての指標が改善していることがわかる。

### 5.2.2 サーバ用ABCの有無による差

表4、図4-6より、サーバ用ABCを利用すると、クエリの平均ホップ数は小さくなるがサーバの負荷が大きくなり、キャッシュの利用効率が下がることがわかる。これはサーバまでIPルーティングされる途中でABCやBCにより近隣のキャッシュまで誘導されなかったクエリが存在し、それらがサーバ近くのBCによって遠方まで誘導される可能性を示している。この結果は同時に、サーバ付近のクエリ制御を適切に行うことがキャッシュ利用効率を上げるための一つの要因であること、さらには、サーバ用ABCのON/OFFを切り替えることでサーバ負荷の制御ができることも示している。

## 6. まとめ

本稿では、コンテンツキャッシュを効率的に運用するBreadcrumbs+方式に対し、キャッシュノードが周辺ノードに積極的な誘導情報配布を行うことで、キャッシュ運用のさらなる効率化を行う拡張方式(Active Breadcrumbs)を提案した。さらにシミュレーションによって提案方式の有効性を評価した。今後の課題として、大きく3つが挙げられる。1つ目は、ABC配布の動的な制御のための詳細

設計である。2つ目は、さらなるキャッシュ利用効率の改善と負荷分散を行うために、周辺ノードに限らない、より効果的なABC配置法の確立である。3つ目が、階層的なネットワークポロジを使用した評価である。今回は評価ネットワークとして階層構造をもたないモデルを利用したが、実際のネットワークはバックボーン系とアクセス系に分かれた階層構造になっている。アクセス系からバックボーン系にクエリが送信される前にキャッシュに誘導することができれば、トラフィックの軽減に有効であると考えている。

## 謝辞

本研究の一部は、情報通信研究機構の高度通信・放送研究開発委託研究の成果である。ここに記して謝意を表す。

## 参考文献

- [1] G. Barish and K. Obraczka, "World wide web caching: Trends and techniques," *IEEE Communications Magazine*, May 2000.
- [2] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, "Internet Indirection Infrastructure," *IEEE/ACM Transactions on Networking*, vol. 12, no. 2, pp. 205-218.
- [3] A. Carzaniga and A. L. Wolf, "Content-based networking: A new communication infrastructure," *NSF Workshop on an Infrastructure for Mobile and Wireless Systems*, Oct. 2001.
- [4] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture," *SIGCOMM*, 2007.
- [5] A. Beck and M. Hofmann, "Enabling the internet to deliver content-oriented services," *Proc. 6th International Web Caching and Content Distribution*, June 2001.
- [6] V. Jacobson, D. K. Smetters, J. D. Thornton, M. Plass, N. Briggs, and R. L. Braynard, "Networking named content," *in Proc. 5th ACM CoNEXT*, Dec. 2009.
- [7] E. J. Rosensweig and J. Kurose, "Breadcrumbs: efficient, best-effort content location in cache networks," *in Proc. IEEE INFOCOM 2009*, Apr. 2009.
- [8] E. J. Rosensweig and J. Kurose, "Breadcrumbs: efficient, best-effort content location in cache networks," *UMass Amherst, MA, Tech. Rep. UM-CS-2009-005*, Aug. 2008.
- [9] 柿田 将幸, 谷川 陽祐, 戸出 英樹, "コンテンツ流通網におけるインターネット誘導のための Breadcrumbs 方式の改良," *電子情報通信学会技術研究報告*, vol. 110, no. 372, pp. 75-80, Jan. 2011.
- [10] B. M. Waxman, "Routing of multipoint connections," *IEEE J. Selected Areas in Communications (Special Issue on Broadband Packet Communication)*, vol. 6, no. 9, pp. 1617-1622, Dec. 1998.