

Named Data Networkのためのコンテンツ要求回数に基づいた更新確認アルゴリズム

金子平[†] 木村成伴[‡][†]筑波大学情報学群情報科学類[‡]筑波大学システム情報系情報工学科

1 はじめに

Named Data Network (NDN) は、コンテンツを保持しているサーバ (NDN では Producer と呼ぶ) の場所ではなく、コンテンツ自身の名前のみを指定することで、コンテンツをやりとりするネットワークである。存在する場所を指定しなくて良いことから、コンテンツをルータのキャッシュに保持することで、通信量や配送遅延を削減することなどが実現可能となっている。

Producer はコンテンツがキャッシュされた際の有効期限を設定することができるが、本論文で対象とする Web コンテンツは、不定期に作成・更新されるため有効期限を予測することは難しい。このため、有効期限を長く設定しすぎるとキャッシュに残る時間が長くなり、削減される通信量が多くなるが古いコンテンツを取得する回数も多くなる。逆に、短く設定しすぎると古いコンテンツを取得する回数は少なくなるが削減される通信量は少なくなるという問題があった。

そこで、古いコンテンツを取得する回数を減らすために、著者らはキャッシュを持つルータが更新確認を行う方式を提案している [1]。この方式では、ルータが一定間隔で更新確認を行う定期確認方式と、連続する更新確認の間隔を増加させていく動的確認方式を提案した。特に、後者の方式では、すべてのコンテンツについて、同じアルゴリズムで更新確認間隔を増加させている。しかし、キャッシュに古くなったコンテンツが複数存在する場合、要求回数が多いコンテンツはその分古いコンテンツが取得される回数が多くなる。このことから、要求回数が多いコンテンツは、要求回数が少ないものと比べて、相対的に更新確認の必要性が高いと言える。そこで、本論文ではコンテンツ要求回数に基づいた更新確認アルゴリズムを提案する。

表 1: FIT の例

コンテンツの名前	要求回数	キャッシュされた時刻	現在の更新確認間隔
/content1	4	18:41:00	1800
/content2	8	18:21:00	2400

2 コンテンツ要求回数に基づいた更新確認アルゴリズム

コンテンツ要求回数に基づいた更新確認を行うため、提案アルゴリズムでは、各ルータに Fetch Information Table (FIT) を導入する。表 1 に示すように、FIT はコンテンツの名前、要求回数、キャッシュされた時刻、現在の更新確認間隔のエントリを持つ。このエントリは、当該コンテンツがキャッシュされた時点で作られ、キャッシュから削除された時点で削除される。

2.1 ルータの挙動

提案アルゴリズムにおけるルータの挙動を以下に示す。ここに示す動作の他に、ルータは従来の NDN での中継やキャッシュの動作も行うが、以下では省略する。

なお、ルータから更新確認パケットを受け取った Producer は、コンテンツのバージョンのみ応答し、コンテンツの中身は含めない。

1. ルータがコンテンツを含む Data パケットを受け取り、このコンテンツを新しくキャッシュするときは、FIT のエントリを作る。ここに、コンテンツの名前と要求回数 (1 回)、キャッシュされた時刻を登録するとともに、現在の更新確認間隔を初期値 T [s] とする。また、 T [s] 後に更新確認をスケジューリングする。
2. ルータがキャッシュを持つ Interest パケットを受け取ったときは、その FIT のエントリの要求回数のカウントを 1 増やす。
3. 更新確認パケットの応答を受け取り、当該コンテンツがキャッシュ内にある場合、その更新確認パケットが自分が送信したものでなくても、その

A Cache Updating Algorithm Based on the Request Number of Content for Named Data Networking

Taira KANEKO[†], Shigetomo KIMURA[‡]

[†]College of Information Science, University of Tsukuba

[‡]Faculty of Engineering, Information and Systems, University of Tsukuba

バージョンを確認する。そして、キャッシュ内のコンテンツのバージョンが古ければ、このコンテンツをキャッシュとFITから削除する。

4. 自分が送った更新確認への応答であり、キャッシュ内の当該コンテンツのバージョンが古くなければ、更新確認間隔を以下の手順で変更する。ここで、 k (> 1) は更新確認間隔の倍率である。

- (a) キャッシュ内のコンテンツを人気度 (FIT の情報から計算した単位時間あたりの要求回数) の高い順に並べる。
- (b) すべての人気度が同じでなければ、人気度が最も高いコンテンツと最も低いコンテンツをペアにする。このとき、人気度が同じものが複数あるときは、キャッシュされた時刻が最も古いものを選ぶ。これらのペアを除外し、残りのコンテンツがなくなるか、すべてが同じ人気度になるまで、この手順を繰り返す。
- (c) 更新確認に応答された当該コンテンツがどのコンテンツともペアにならなかった、もしくはペアのうち人気度が高い方だった場合は、FIT の更新確認間隔 d を kd [s] に更新し、その時間だけあとに更新確認をスケジュールする。人気度が低い方だった場合は、高い方の更新確認を kd [s] 後に、低い方の更新確認を $kd + kd^2$ [s] 後にスケジュールし、低い方のFIT の更新確認間隔を kd^2 [s] に更新する。

3 実験

提案方式の有効性を示すために行うシミュレーション実験について説明する。シミュレータは、NDN シミュレータ ndnSIM[2] を用いる。

3.1 実験条件

文献 [1] と同様に、BA モデル [3] を使って 50 台のルータを接続し、その後ランダムなルータに 5 台の Producer と 200 台の Consumer を接続して実験用ネットワークを作成した。リクエストの分布は Zipf の法則に従った。リクエストの頻度は 5 分に 1 回とした。各 Producer は 20 個のコンテンツを持っており、その更新間隔は実際のニュースサイトや SNS の計測結果に基づいて定めた。同一 Producer 内のコンテンツはリンクしているため、一つのコンテンツが更新された場合、その Producer 内のすべてのコンテンツが更新されるとみなした。その他の実験条件を表 2 に示す。

表 2: その他の実験条件

項目名	値
キャッシュサイズ	コンテンツ 10 個分
キャッシュストラテジ	Leave Copy Everywhere
キャッシュリプレースメントポリシー	LRU
ノード間の伝送遅延	1.8 [ms]
ノード間の帯域	1024 [Mbps]
パケットロス	なし
コンテンツの容量	4096 [Byte]
更新確認間隔の初期値	270 [s]
更新確認間隔倍率 (k)	1.5, 2.0, 2.5, 3.0

3.2 評価方法

実験結果の評価指標は、式 (1) の関数 $f(M)$ [1] を採用した。ここで、 M は評価する方式を表し、オリジナルの方式と文献 [1] の方式、提案方式を比較する。また、 $f(M)$ の第 2 項はルータが新しいコンテンツを Producer から取得するときに発生する通信量の期待値を表している。この評価指標を採用することで、更新確認による通信量の増加とキャッシュの有効性を統一的に評価することができる。

$$f(M) = (\text{トポロジ全体の総通信量}) + (\text{古いデータの取得回数}) \times (\text{コンテンツの平均容量}) \times (\text{ホップ削減数の平均}) \quad (1)$$

4 まとめ

本論文では、コンテンツ要求回数に基づいた更新確認アルゴリズムを提案した。今後は、シミュレーション実験を行い、提案方式の評価を行う。

参考文献

- [1] Takatoshi Miwa and Shigetomo Kimura, “Co-operative Update Mechanism of Cache Update Method Based on Content Update Dynamic Queries for Named Data Networking,” Proceedings of 2019 Seventh International Symposium on Computing and Networking Workshops (CAN-DARW), pp. 33–39, 2019.
- [2] “ndnSIM: Simulation of NDN, ICN, Information-Centric Networking,” <http://ndnsim.net/current/>.
- [3] Albert-László Barabási, and Réka Albert, “Emergence of Scaling in Random Networks,” Science, Vol. 286, No. 5439, pp. 509–512, 1999.