

情報指向ネットワークにおける キャッシュ利用効率向上のための 一検討

木村一統¹, 新井イスマイル², 藤川和利²

¹奈良先端科学技術大学院大学 情報科学研究科

²奈良先端科学技術大学院大学 総合情報基盤センター

研究背景

- 2018年にはインターネットトラフィックの約75%を動画等の大容量コンテンツが占めると予測されている[1]
 - トラフィック量の増大
- ICN(Information Centric Networking)では、コンテンツをネットワーク上でキャッシュすることによりトラフィック量の削減可能

[1] Cisco VNI Service Adoption Forecast, 2013-2018, <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html>

本研究の目的

- ICNにおけるキャッシュヒット率の向上
 - トラフィック量の削減
 - レスポンス時間の短縮
- ユーザリクエスト(Interest Packet)をキャッシュの方向へと転送することでキャッシュヒット率を向上

既存研究

- Breadcrumb[2]
- Caching Location and Searching(CLS)[3]

[2] E. J. Rosensweig, and J. Kurose, "Breadcrumbs: Efficient, BestEffort Content Location in Cache Networks," in INFOCOM, 2009.

[3] Li, Yang, et al. A chunk caching location and searching scheme in content centric networking. In: Communications (ICC), 2012 IEEE International Conference on. IEEE, 2012. p. 2655-2659.

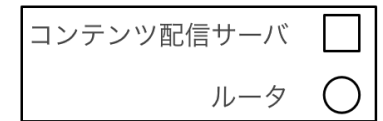
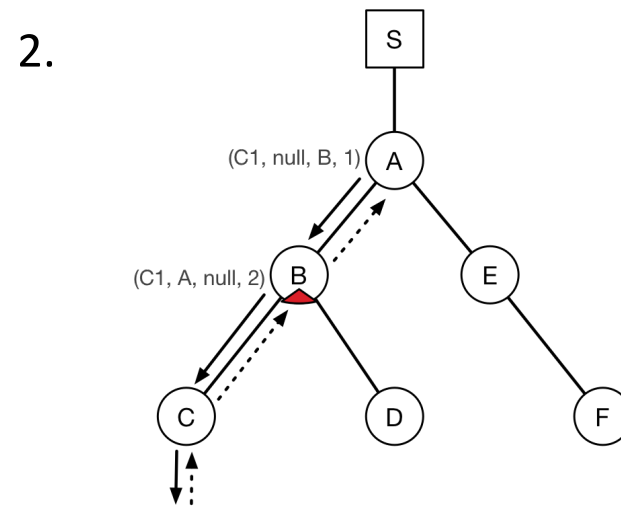
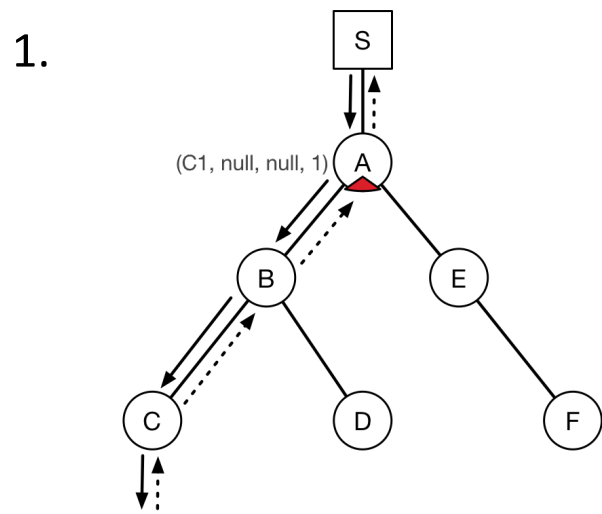
Caching Location and Searching (CLS)

- ネットワーク上の各ノードが、キャッシュの存在する方向を示すテーブルを持つ
- テーブルの内容に従いユーザリクエストをキャッシュに向けて転送することで、キャッシュヒット率を向上させる

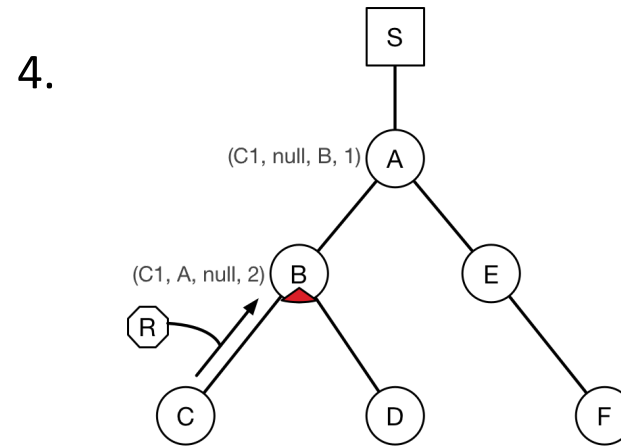
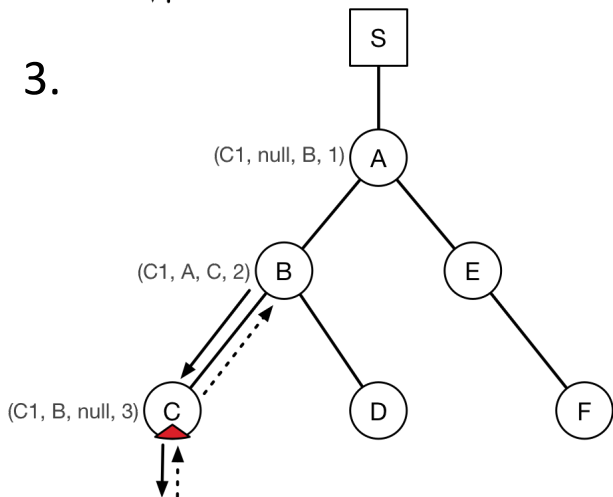
CLSテーブル

- CLSの各ルータは以下の4要素から構成されるテーブルを持つ
 - コンテンツID
 - データを受信したインターフェース
 - データを送信したインターフェース
 - コンテンツ配信サーバからのホップ数

CLSの動作例1



(コンテンツID,
データ受信インターフェース,
データ送信インターフェース,
サーバからのホップ数)

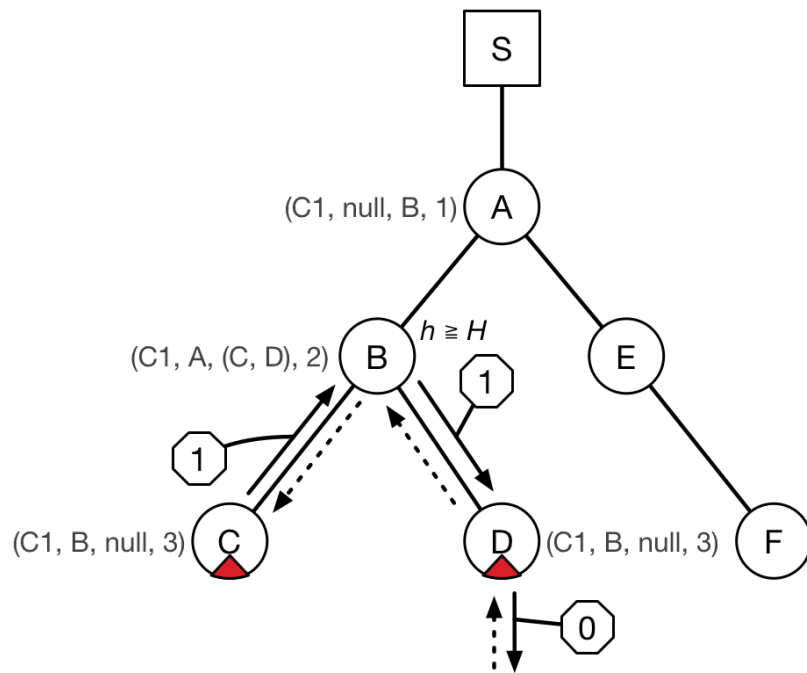


CLSのルール

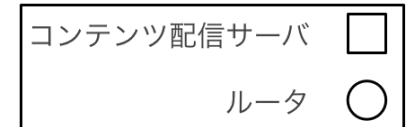
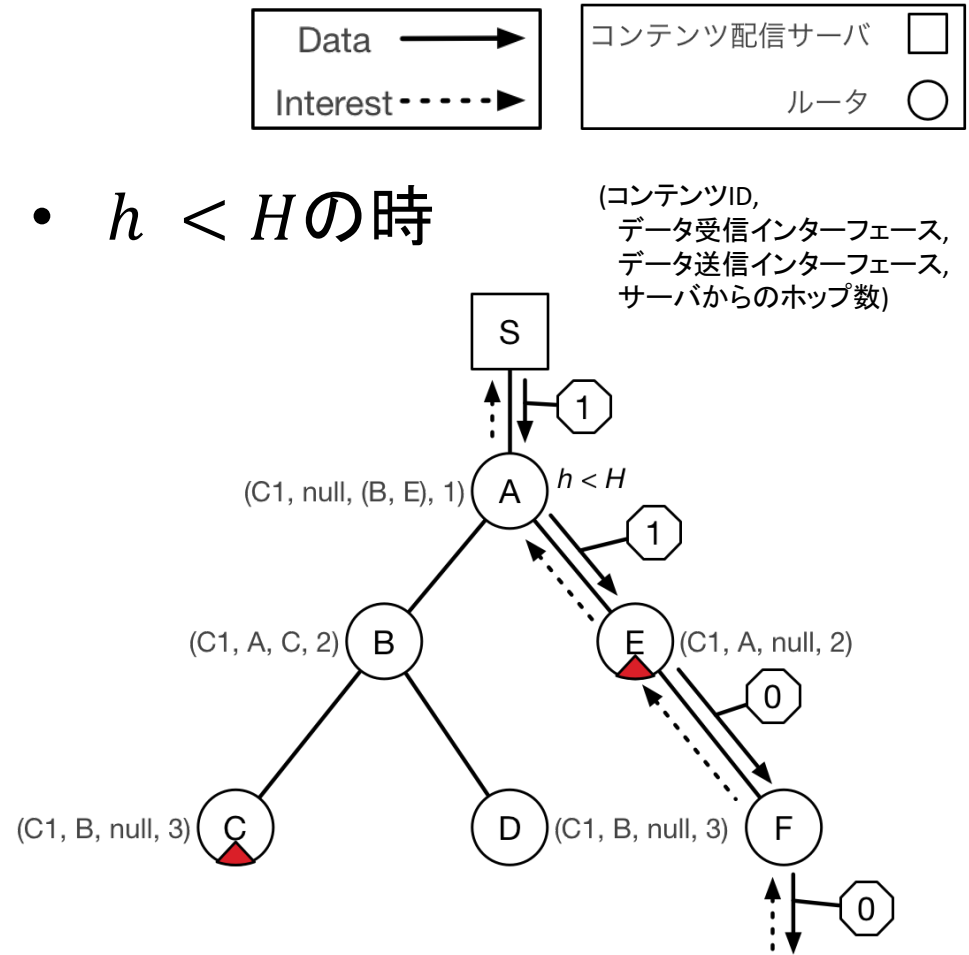
- CLSではサーバとリーフルータ間でキャッシュが重複しないことを保証
- h : コンテンツ配信サーバからのホップ数
 H : サーバとリーフルータ間の高さの $\frac{1}{2}$
と定義
 - $h \geq H$ の時、リクエストをキャッシュ方向へ転送
 - $h < H$ の時、リクエストをサーバ方向へ転送

CLSの動作例2

- $H = 1.5$ の場合
- $h \geq H$ の時



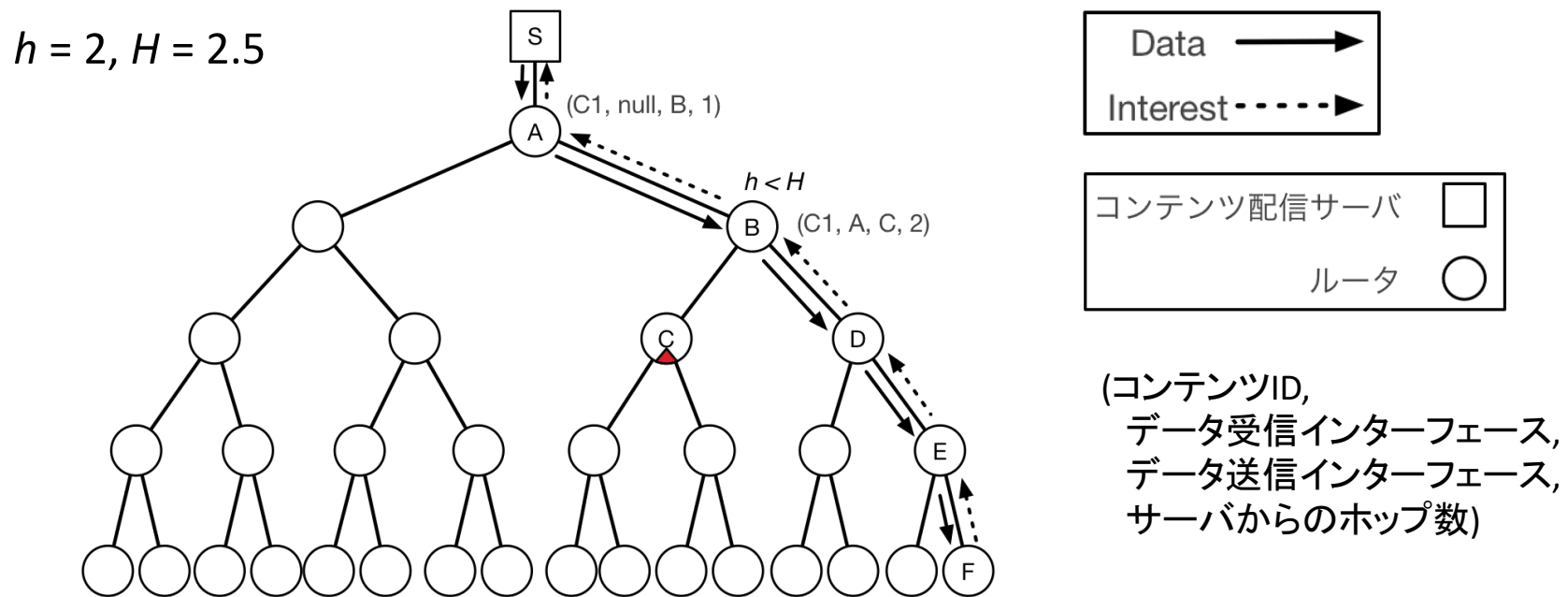
- $h < H$ の時



(コンテンツID,
 データ受信インターフェース,
 データ送信インターフェース,
 サーバからのホップ数)

CLSの問題点1

- $h < H$ の時、サーバに近いキャッシュが活用されない
 - 必ずサーバ方向へ転送される



CLSの問題点2

- 各ルータに対してサーバとリーフルルータ間の高さの $\frac{1}{2}$ を入力する必要がある
 - ルータ管理コストの増大

提案手法

- キャッシュからのホップ数を保持するためにCLSテーブルを拡張
- キャッシュからのホップ数を伝達するための新しいパケットHopCountを定義

CLSテーブルの拡張

- 拡張されたCLSテーブルは以下の5要素から構成される
 - コンテンツID
 - データを受信したインターフェース
 - データを送信したインターフェース
 - コンテンツ配信サーバからのホップ数
 - キャッシュからのホップ数

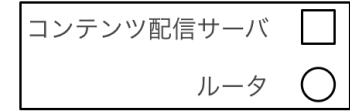
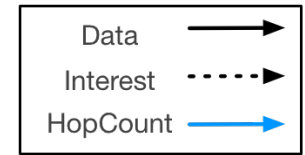
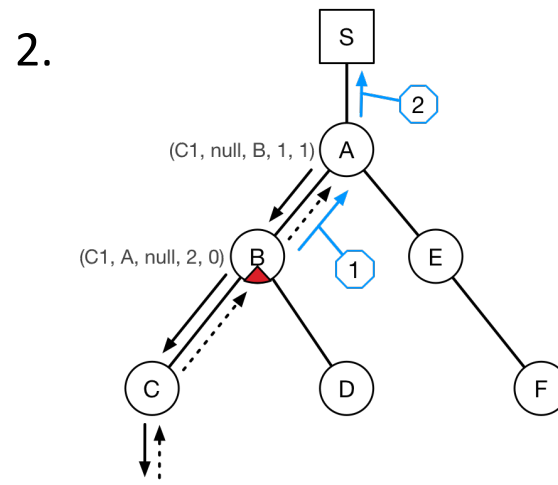
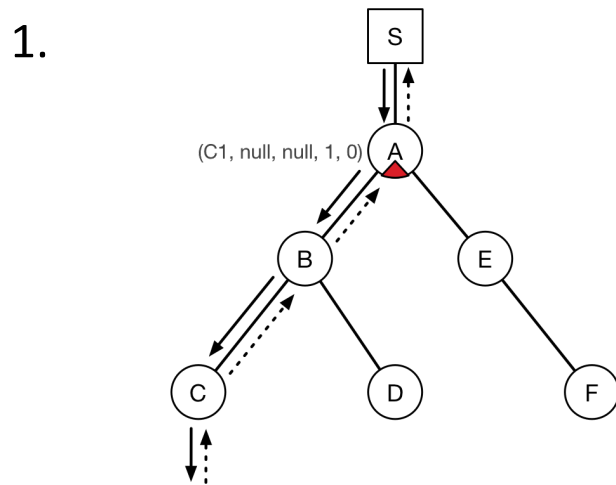
HopCount/パケット

- フィールドとしてキャッシュからのホップ数 hc を持つ
- キャッシュを持つルータから $hc=1$ で送信され、転送される度に hc の値をインクリメント
- パケットを受け取ったルータは hc の値をキャッシュからのホップ数として保持

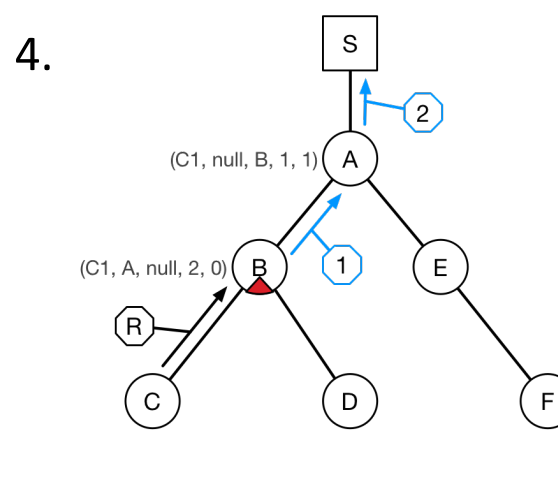
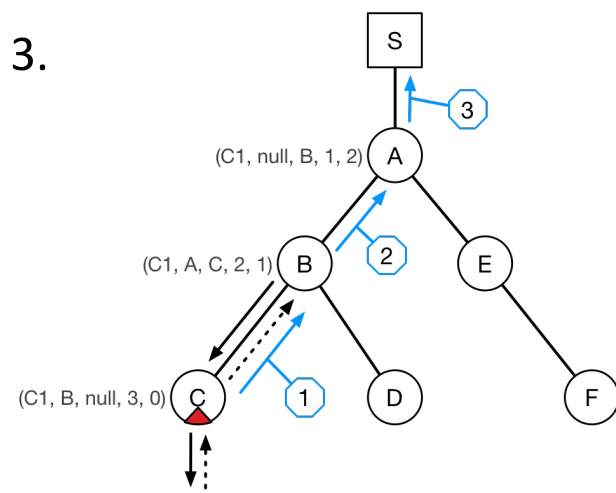
HopCountパケット送信のタイミング

- HopCountパケットを送信するタイミングは以下の2つ
 - ルータにキャッシュが追加された時
 - ルータからキャッシュが削除された時

提案手法の動作例

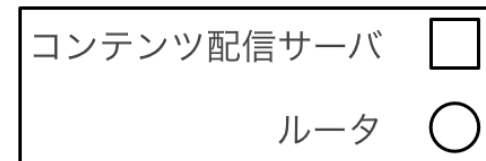
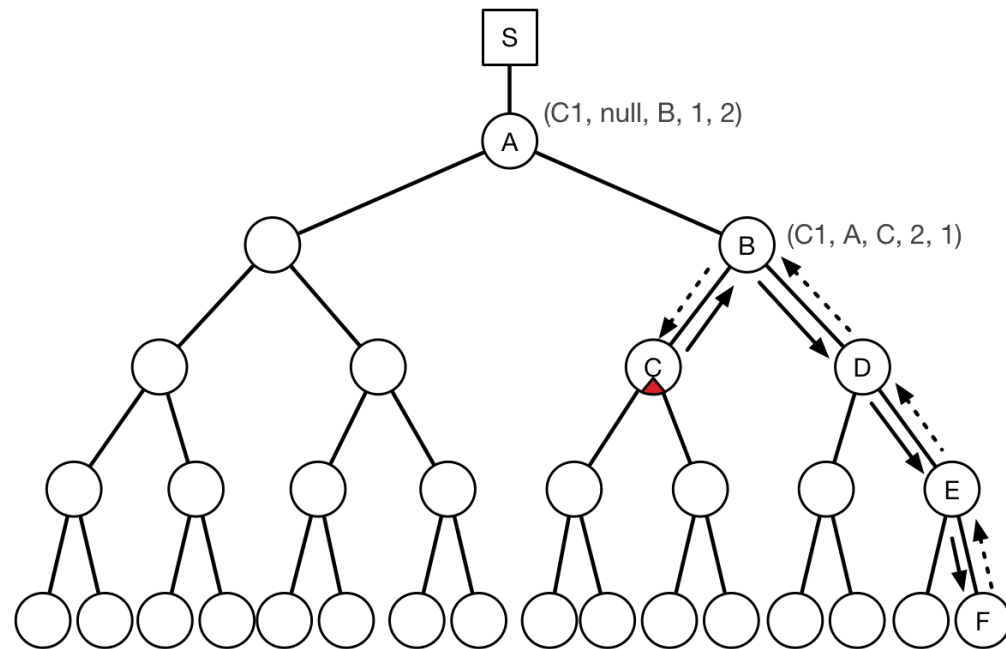


(コンテンツID,
データ受信インターフェース,
データ送信インターフェース,
サーバからのホップ数,
キャッシュからのホップ数)



提案手法の利点1

- 各ルータがコンテンツ配信サーバとの距離、キャッシュとの距離の両方を保持するため、適切に転送可能



(コンテンツID,
データ受信インターフェース,
データ送信インターフェース,
サーバからのホップ数,
キャッシュからのホップ数)

提案手法の利点2

- 各ルータに対してサーバとリーフルルータ間の高さの $\frac{1}{2}$ を入力する必要がない
 - ルータ管理コストが削減

今後の課題

- 実装

- CLSのテーブルを拡張
- 新しいパッケージを定義

- 評価

- キャッシュヒット率
- レスポンス時間

- 木構造以外のトポロジーで提案手法が有効か

- 有効でない場合テーブルが複雑化

まとめ

- ICNでネットワーク上のキャッシュ利用効率を向上させる手法を提案
 - CLSでサーバに近いキャッシュが活用されない問題を解決
 - ルータ管理コストの削減
- より多くのトポロジーで有効か検証する必要がある