2023年度

修　士　論　文

SDNを用いたNDNのアクセス制御方式

指導教員: 上山　憲昭

立命館大学大学院 情報理工学研究科
情報理工学専攻 博士課程前期課程
計算機科学コース

学生証番号: 66112200593

氏名: Yang Zhanbin

概要

Information-Centric Networking (ICN) が，コンテンツを効率的に転送するネットワークとして検討されている．ICN のアーキテクチャの一つに Named Data Networking (NDN) があり，NDN のキャッシュ技術は重要な研究分野である．NDN では要求パケットが到着したルータでコンテンツがキャッシュされている場合，そのルータからコンテンツが配信されるため，コンテンツのオリジナルを提供するパブリッシャに，常に配信要求が届くとは限らず，パブリッシャはコンテンツの要求に対するアクセス制御を行うことができない．

そのため NDN では新たなアクセス制御の仕組みが必要となるが，アクセス制御を NDN の従来の自律分散型の通信方式で実装した場合, 制御情報の伝達効率が低くなる．一方, Software Defined Networking (SDN) は制御情報を交換するレイヤと，ユーザデータが転送されるデータレイヤとを分離し, 制御情報の交換効率を向上させることができる．そこで本稿では, パブリッシャによるデータのアクセス制御を可能にし, 効率的なアクセス効率とキャッシュセキュリティを確保する, SDN ベースの NDN アクセス制御方式を提案する．提案方式では, ユーザ情報をサインとして暗号化し, Interest パケットのパラメタとして追加することで, SDN コントローラの管理下でルータに要求パケットを認証する機能を与える. また CPABE を用いることで, 一旦ファイルがキャッシュされるとデータは安全に保護される．

目 次

第1章　序論

1.1　研究背景

21 世紀に入り, 情報ネットワークは目覚ましい発展を遂げ, ネットワークを介して情報を非常に迅速に伝達することが可能となった. 情報ネットワークが主に提供する二つの重要な機能は, 接続性と共有である. 接続性とは, ユーザが距離に束縛されることなくネットワークを通じて情報を交換できる能力を指し, 共有とは, ユーザがネットワークを通じてサーバー上の必要なリソースを取得できるリソース共有を意味する.

従来のネットワークアーキテクチャは TCP/IP に基づいており, 分層的に協力しながらデータを交換し, ホスト同士は IP アドレスを介して情報を転送する. しかし, 電子商取引, デジタルメディア, ソーシャルネットワーキング, スマートフォンなどのアプリケーションが進化するにつれて, データ転送量が急増し,TCP/IP アーキテクチャではユーザが求めるコンテンツを効率的に配信することが難しくなってきた. そのため, コンテンツ指向性, 移動性, スケーラビリティ, セキュリティに重点を置いた次世代ネットワークが提案されている. その中で, NDN (Named Data Networking）はコンテンツをネットワーク中の主体と見なし, 現在のネットワークの利点と制約を設計に反映させている. 現在,NDN は国内外で注目される研究トピックとなっている.

NDN の大きな特徴の一つは, 対象となるコンテンツがスイッチにキャッシュ可能であり, 次にそのコンテンツにアクセスするユーザが直接キャッシュからデータを取得できるため, パブリッシャから再度リクエストする必要がないことだ. しかし, パブリッシャはコンテンツがどこにキャッシュされるかを制御することができず, 経由する各 NDN ルータは独自のキャッシュポリシーと現在のキャッシュ状態に基づいて, そのコンテンツをキャッシュするか否かを決定する. これは, 暗号化されたコンテンツであっても, リクエストした者が誰であれ, コンテンツが取得可能であることを意味する. これにより, 機密内容の漏えいリスクが高まる可能性がある.

そこで本論文では,NDN のキャッシュ性能を保持しつつ, ルータを利用してアクセス者の身元を識別し, コンテンツアクセスがアクセス許可者のみに限定されることを保証する, ソフトウェア定義ネットワーク（SDN）に基づく NDN アクセス制御方式を提案する. この方式では,SDN のコントローラを通じてユーザの身元管理と鍵管理を行い, ユーザとルータのマッピングを基にしてアクセス制御リスト（ACL）を生成し, 対応するルータに配布する. 同時に, ユーザがコントローラに登録する際には, 対応するユーザ ID と解読鍵を取得する. ユーザがコンテンツをリクエストする際には,ID と Nonce を結合して暗号化し,Interest パケットのパラメータとしてユーザの身元サインを追加する. ルータはこのサインを基にユーザの身元を識別する. パブリッシャが権限情報を更新する必要がある場合,NDN の通信方式を使用して長距離伝送すると, 効率が大幅に低下する. この方式では,SDN の集中制御の特性を利用し, 権限変更などの制御情報をコントローラで伝達することで更新効率を高め, データ層の NDN ネットワークがデータの配布により集中できるようにする.

従来の暗号化方法は一対一の関係であり, 図 1.1 に示されている. つまり, 一方の暗号化には一方の解読が対応している. これにより, 暗号化されたキャッシュは一人のユーザにのみ提供され, そのキャッシュの流通性が大幅に低下し, キャッシュ汚染攻撃（CPA）によってルータのキャッシュサービスの品質（QoS）が低下する可能性がある. このため, 本論文では, 暗号化されたコンテンツ本体を担当する AES と,AES 鍵自体を暗号化する CP-ABE を組み合わせた混合暗号化方式を選択する. コントローラが CPABE の暗号鍵と解読鍵を生成し, パブリッシャとユーザ

に配布する. ユーザは適切な権限の鍵を持っていれば, 明文アクセスを解読でき, パブリッシャも一つの暗号鍵で一度に暗号化するだけである. その結果, 単一の暗号化キャッシュも複数の合法ユーザによってアクセスされ, キャッシュの流通性が保証される. 図 1.2 に提案システムの概要を図時する.

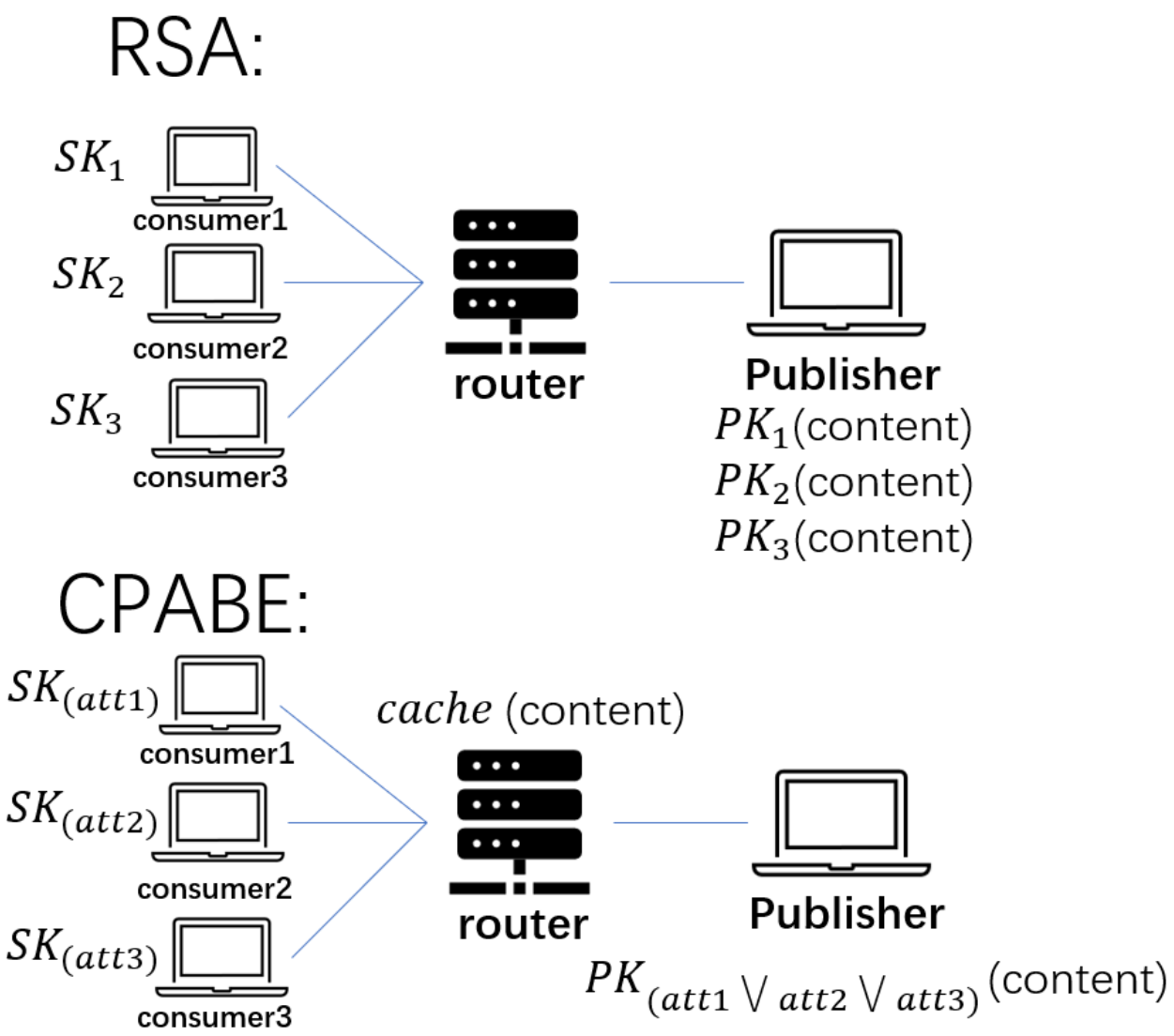

図 1.1: 暗号化アルゴリズムの差異

さらに, この方式では, パブリッシャはアクセス制御を行う必要がなく,CPABE で暗号化された AES キー部分を含むコンテンツ全体がルータにキャッシュされる. これは, パブリッシャがオフラインになったとしても, ネットワーク内にそのコンテンツの完全なキャッシュが存在すれば, そのコンテンツは合法的にアクセス可能であり, 即ちコンテンツの持続性が保証される.

本論文では, 公開されたトポロジ情報を用いて提案方式のパフォーマンスと効率に関する評価を行う. まず, ルータが構成する分散型アクセス制御と, パブリッシャが担当する集中型アクセス制御方案が生成するトラフィックを

比較する. 次に, 伝統的な公開鍵暗号化方式を使用した場合と CPABE を使用した場合のパブリッシャの暗号化コストを比較する. そして,SDN の貢献度を証明するために,SDN を使用する場合と使用しない場合で, ルータが保持する必要がある ACL エントリの数と, パブリッシャが権限を更新するために必要な時間を評価する.

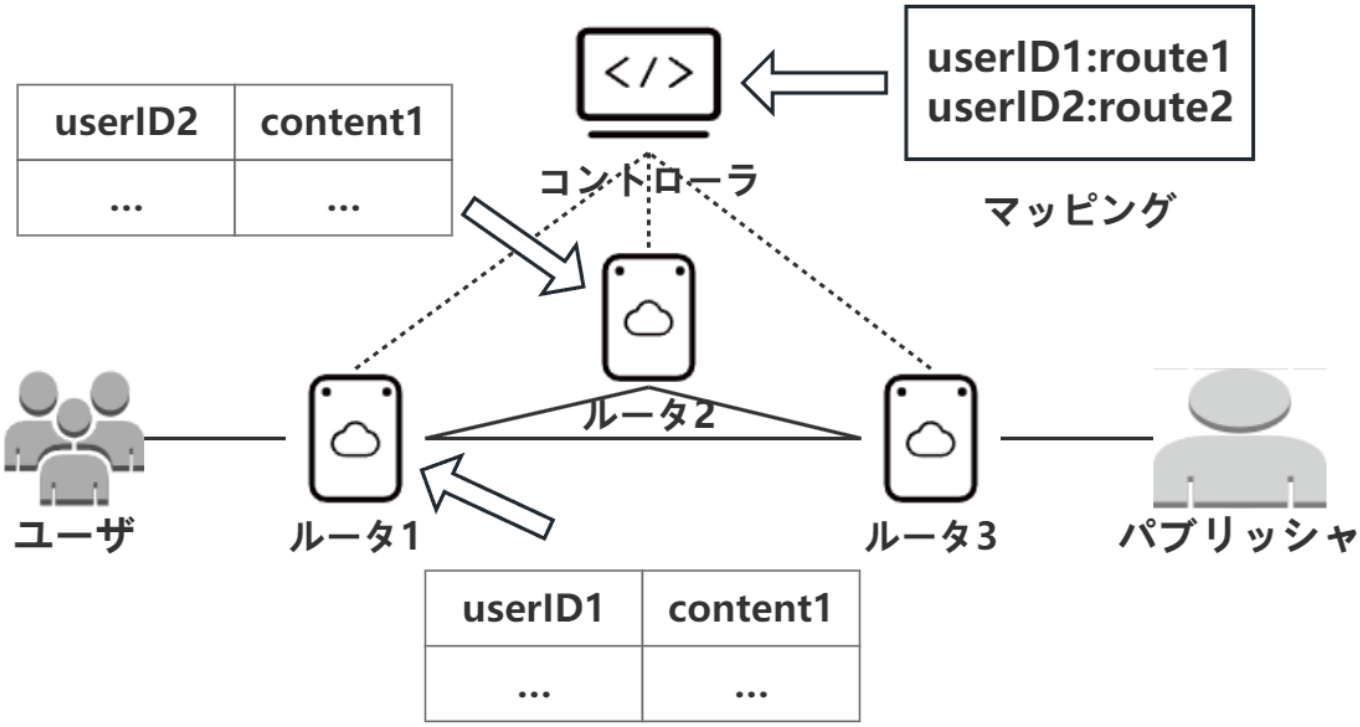

図 1.2: 提案の概念

1.2 研究目的

本論文は,NDN ネットワークにおけるパブリッシャがルータ内のキャッシュに対するアクセス制御を行うことが困難である問題に対して, 新たなアクセス制御方案を提案する. アクセス制御作業を各ルータに分散させ, パブリッシャは SDN コントローラの集中型制御を利用してルータに権限を配布し, パブリッシャのキャッシュアクセス制御機能を実現する.

第 3 章では関連する研究を簡単にまとめた後, 第 4 章で提案方式の各部分の詳細な実装を紹介する. 第 5 章では提案方式の各項目の性能評価結果を示し, 最後に第 6 章で本論文を総括する.

第2章　関連研究

近年の研究では, コンテンツにアクセスするための NDN コンテンツ名検索特性に基づく, コンテンツ名に対するアクセス制御メカニズムがいくつか提案されている [2].[2] の提案は, コンテンツ名を暗号化することで難読化し, 権限のないコンテンツ利用者がその名前にアクセスできないようにする. コンテンツ名を暗号化するための標準鍵を, コンシューマとパブリッシャの間で事前に配布する方法である. パブリッシャ上のデジタル名を使って本人であることを検証することで, 認証を行う. しかし, このメカニズムでは, ユーザ認証を完了するために, パブリッシャが常時オンラインである必要があり, コンテンツの永続性が保証されない. 一方, パブリッシャの認証処理の負荷が大きい.

[1] は NDN における分散アクセス制御のためにルータ＋ CPABE を利用し, パブリッシャのアクセス制御の負荷を低減することを提案している.[1] では, 改良された CPABE が提案されている. ユーザ ID は復号化属性の 1 つとして付加され, ルータによって復号化構造の一部に変換される. この方式は, ルータにアクセス制御機能を与え, アクセス可能なユーザのみがコンテンツにアクセスすることを保証する. しかし, この方式にはいくつかの問題がある. パブリッシャは, ルータにユーザが存在するかどうかを知らないので, 各ルータのユーザ ID ハッシュテーブルは, その領域に属していない多くの ユーザを持つことになり, 不必要なデータオーバヘッドが追加される. また,Interest パケットに含まれるユーザ ID は暗号化されていないため,ID のなりすましにつながる盗聴攻撃を受ける可能性が高く, 結果として不正ユーザが暗号化されたコンテンツキャッシュにアクセスできる. また,NDN の通信特性上, 許可関連の変更情報の送信は非効率である. 本論文では, 署名認証の原理に従ってユーザ ID を暗号化することで, ユーザとルータ間の認証方法を改善し, 認証プロセスのセキュリティを強化する.[6][7] の研究によると,NDN ルータの主要機能は Programming Protocol-independent Packet Processors (P4) を通じて実現可能であり, これは SDN を NDN 上に展開することの実現可能性を証明している.[3][4] におけるコントローラによる地域ユーザー情報の取得方法に基づき, ルータの ACL を簡素化し, ルータの ACL には必要なユーザーのみを保存し, 権限に関連する変更情報をコントローラを通じて転送する. 最後に, 許可を取り消す際のルータへのオーバーヘッド圧力を低減するために, 別の CPABE 改善スキーム [5] を使用する. これにより, ルータによる権限失効に起因する再エンクリプションの回数を減らすことができる.

第3章　提案方式

3.1　初期化

コントローラはユーザの情報, 関連ルータと属性アクセス構造 (Access Structure (AS)) を格納するデータベースを管理する．もし変更がある場合，新しいバージョンの情報を含むハッシュ値を生成し，担当するルータへ通知することで，すべてのルータが所持している情報を常に最新のままに保つ．また提案方式では，データプレーンとコントロルプレーンがそれぞれ独立しており，お互いに干渉できない．すなわちユーザがコントローラにアクセスできず，またコントローラもルータ上のデータにアクセスできない．このような仕組みとすることで，アクセス情報とデータの安全性を保証する．

ここで, コントローラは, 特有のコンテンツ名を占有する. パブリッシャまたはユーザが, このコンテンツ名に基づいて Interest を送信すると, 受信ルータは, このインタレストを openflow packet_in[10] パケットにカプセル化し, コントローラにアップロードする.

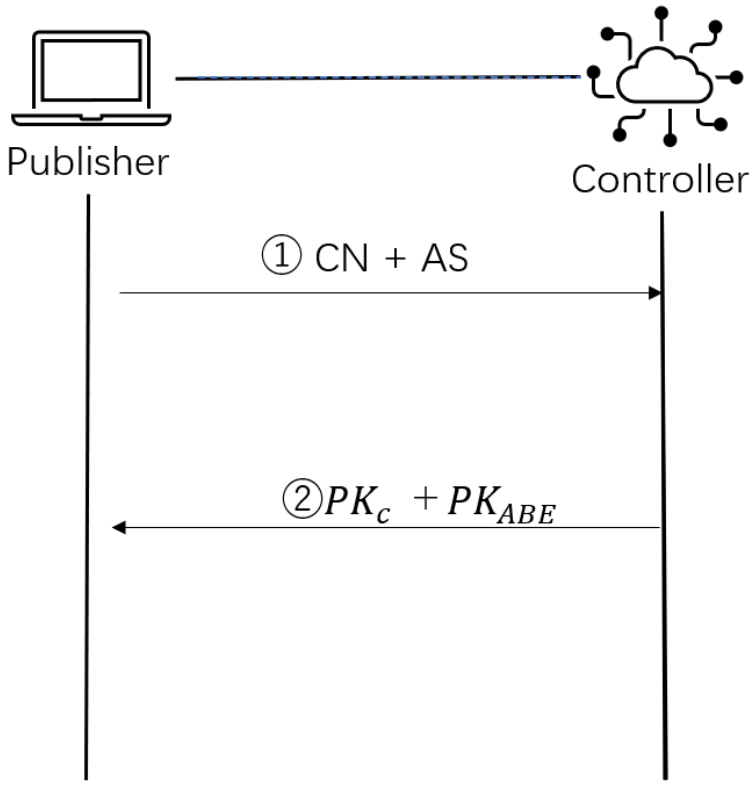

図 3.1: パブリッシャの初期化手順

図 3.1 に，パブリッシャの初期化の各処理ステップを示す.

1. パブリッシャはコンテンツを配信する前に，コンテンツ名 (Content Name(CN)) と AS をコントローラに登録

2. コントローラで生成した CPABE の Public Key (PK_{ABE}) とを受け取る．なお PK_{ABE} の生成には Master Key (MK_{ABE}) が必要である．

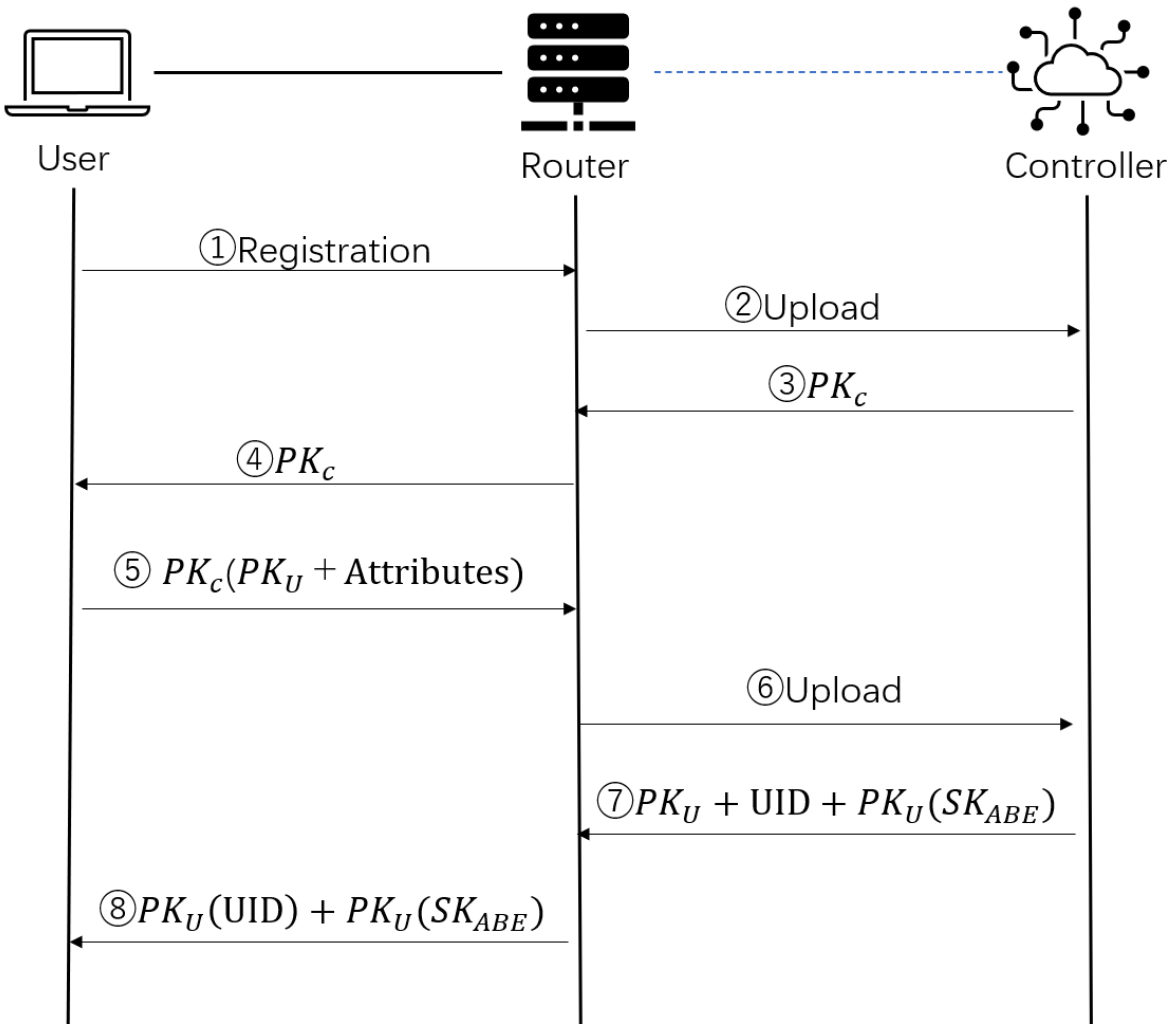

図 3.2: ユーザの初期化手順

図 3.2 に，ユーザの初期化の各処理ステップを示す.

1. 新しいユーザがアプリケーションに参加するとき，登録パケットを最も近いルータに送信
2. ルータはユーザの登録パケットをコントロールに転送
3. コントロールは応答として自身の公開鍵をユーザに返信
4. そのユーザは自身の公開鍵と既存の属性をコントローラに提出するか，属性が何もなければ空のセットを提出
5. コントローラは状況に応じて対応する属性のセットをユーザに割り当てる．同時にルータはユーザの対応ポートを記録し，コントローラがユーザと対応ルータを記録する．そしてコントローラは，ユーザの属性と MK

に基づいて，対応する復号化キー Secret Key (SK_{ABE}) とユーザ ID(UID) を生成する. その後,PK_U,UID と PK_U をルータに返信.

6. ルータは UID とユーザを対応するポートをマッピングして記録する. その後 PK_U を使って UID を暗号化して,PK_U と一緒にユーザに返信.

3.2 暗号化

パブリッシャ側でコンテンツを暗号化して配信する．暗号化ではコンテンツを大小２つの部分に分けて行うが，１つ目 (パート 0) は [5] で暗号化する２つ目の対称鍵暗号秘密鍵である．２つ目は対称鍵暗号方式で，大きい部分を暗号化する．そのためコンテンツの大きい部分はどのようなユーザでも復号可能となり，NDN におけるキャッシング機能を活用することが可能である．[5] では, 時間を AS の属性として追加できるため, 提案中のユーザ権限の有効期間は, ユーザの SK に埋め込まれる．権限の有効期限が切れると, ユーザはコントローラに対して権限の更新する必要があり, コントローラは新しい権限時間を含む鍵を発行する．同時に, ユーザ自身の公開鍵 PKU の長さを権限期間に応じて選択する必要がある.

3.3 アクセス制御

アクセス制御リストの配布は，コントローラ内部のデータベースそのものではなく，ルータの該当ユーザの情報を記載しているハッシュテーブルを送る．コントローラは，ハッシュテーブル (ユーザ ID をもとに，このユーザに対して許可されたアクセスコンテンツを探し出す) の更新のみルータに送ることになる.

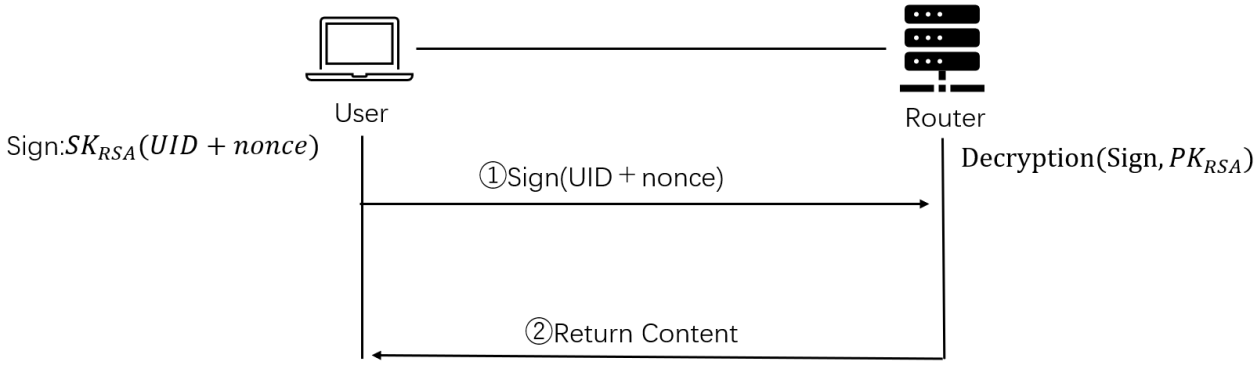

図 3.3: サインの交換手順

実際のアクセス制御は，ルータで行う．図 3.3 に示す通りである. まず, ユーザは ID サインとして, Interest パケットのランダム数 Nonce とユーザ ID をスプライスして暗号化する. そして, これを名前パラメタとして Interest パケットに追加する [8]．ルータは ID サインをハッシュテーブルのユーザ ID と Interest の Nonce に署名することで検証し，一致すれば身元認証はパスする．また，アクセス制御の要求を応答するとき，CPABE で暗号化した部分 (パート 0) のみ返送する．残りのデータは別途要求する必要がある．また認証が通らない場合，直接要求パケットを破棄する. パート 0 は SK で復号化する必要があるので，ルータにキャッシュすることが許可されている.

3.4　権限更新

パブリッシャがアクセス権限を更新したい場合，図 3.4 に示すように，まずコントローラに変更点を PK_C で暗号化して送る．コントローラが変更要求を受け取った後すぐにデータベースの情報を更新する．そして, 確認パケットをパブリッシャに返信し, 新しいアクセス制御情報が対応するルータに送られる．ルータのストレージスペースを節約するため，ルータの管理下に権限があるユーザのみがハッシュテーブルに記録される．

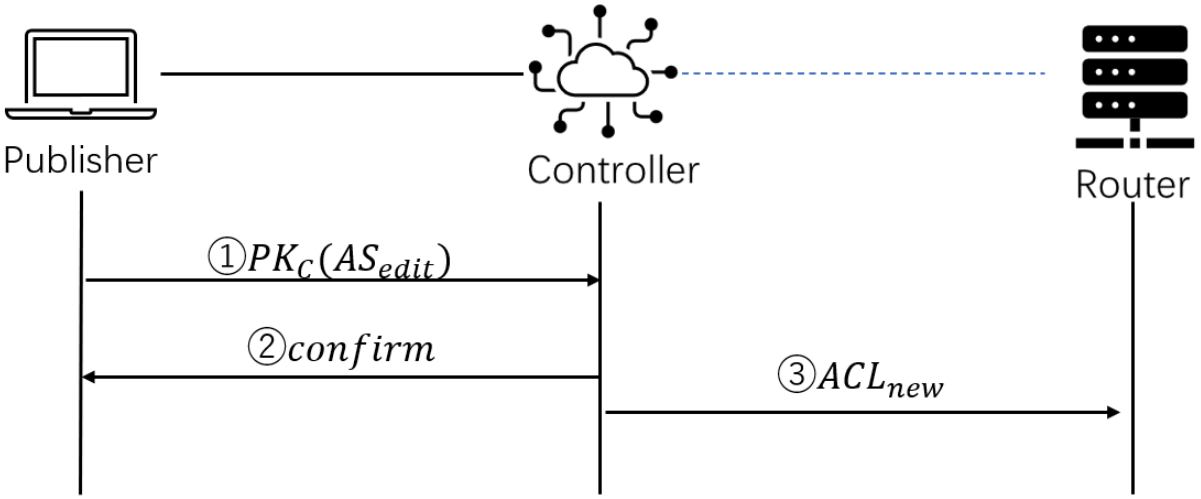

図 3.4: 権限更新の手順

[5] を用いたアクセス制御では，権限の取り消しが二種類に分けられる． 例えば，パブリッシャが認可されたユーザのアクセスを無効にする指示を受け取った場合，そのユーザを管理するルータに対して個別に失効アップデートを送信することができる． この場合，取り消し指示を受け取ったコントローラは，自身のデータベース上に保存しているユーザエントリの情報を更新するだけで済む． 取り消されたユーザの権限がまだ失効していない場合, コンテンツの再暗号化はルータによって実行される． [5] の手法に従うと．このプロセスで取り消されたユーザ ID は, キャンセルリストに追加され, そのユーザがコンテンツを復号できないことを保証する． これにより, パブリッシャの重複暗号化負荷が軽減される．二種類目はグループ単位の取り消しである．つまり，あるコンテンツから，許可が下りた属性のいくつを削除することである．この場合，属性アクセス構造を更新する必要がある．コントローラはこれを受け取り，許可情報を更新して新しいハッシュテーブルを生成し，ハッシュテーブルを対応するルータに送る． ルータは地元の対応キャッシュを削除し, 再度パブリッシャに対応コンテンツをリクエストする必要がある．

3.5　ユーザ遷移

ユーザが別のルータに遷移する場合，ルータへの再バインドのために，コントローラに再登録して送信する必要がある．図 3.5 に，ユーザの再登録の各処理ステップを示す.

1. ユーザは, コントローラの PK_C を使用して, 自分の UID を自分の PK_U で暗号化し, 暗号化されたコンテンツをパラメタとしてカプセル化し,Interest として新しいルータに送信する.

2. 新しいルータは, コントロールのコンテンツ名に基づいて, コンテンツをコントローラにアップロードする.
3. コントローラは, ユーザを新しいルータにバインドし,UID をダウンした PK_U を新しいルータに送信する.
4. 新しいルータは, コントローラからの返事を受信し, 確認の返信をユー ザーに返し, 再登録が終了する.

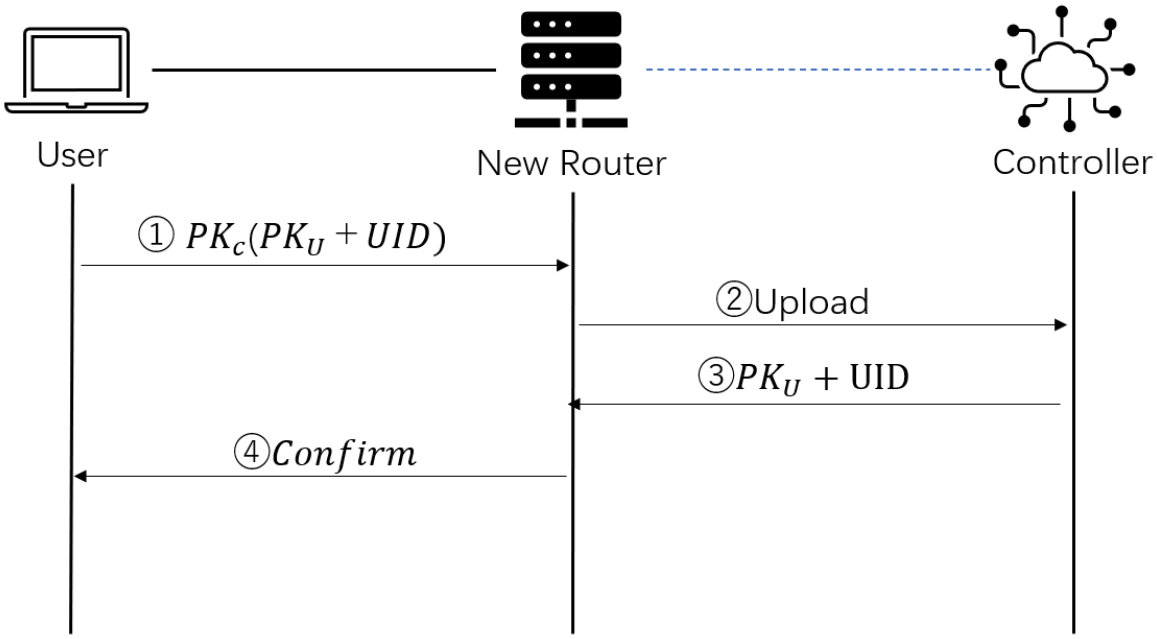

図 3.5: 再登録の手順

第4章　性能評価

本節では，公開されたトポロジ情報を用いて提案方式のパフォーマンスと効率に関することを計算機シミュレーションにより評価する．まず, 提案方式とパブリッシャが担当する集中型アクセス制御方案が生成するトラフィックを比較する. 次に, 伝統的な公開鍵暗号化方式を使用した場合と CPABE を使用した場合のパブリッシャの暗号化コストを比較する. そして,SDN を使用する場合と使用しない場合で, ルータが保持する必要がある ACL エントリの数と, パブリッシャが権限を更新するために必要な時間を評価する.

4.1　安全モデル評価

- 権限管理：提案に鍵の生成と管理は,NDN ルータにとって透明である.CPABE の主鍵 MK はデータ層のルータやユーザには存在せず, コントロール層のコントローラにある. コンテンツの発行者は, コントローラが生成した PK_ABE を使用して暗号化を行う. コントローラはアクセス構造に基づいてユーザに権限を付与し, データ層の任意のユーザはコントローラに直接アクセスすることができないため, 権限管理の安全性が向上する.

- 秘匿性：ユーザはアプリケーションを使用して保護されたデータの暗号化されたアクセスと公開を実現する. 保護されたコンテンツは暗号化された状態でネットワーク全体にキャッシュされ, 転送されるため, データの秘密転送とキャッシュされたデータの安全性が保証される. また, ユーザの SK_ABE は, ユーザの PK_U で暗号化された後に転送され,PK_U も PK_C で暗号化された後に転送される.PK_C はコントローラとの通信以外ではデータ層では無効であり, これにより重要な情報の秘匿性が保証される.

- 前方安全性および後方安全性：[5] は, ユーザの権限が剥奪された後, 鍵が暗号化されたコンテンツを解読できないことを保証する. ユーザが鍵を保持していたとしても,ACL フィルタリングメカニズムにより, 暗号化されたコンテンツにアクセスできず, アクセスしたいコンテンツを取得することができない. したがって, 保護されたコンテンツの前方安全性および後方安全性が保証される.

4.2　トラヒック量の比較評価

シミュレーションの目的は提案したアクセス制御方式の制御パケット量を評価することである. 提案方式と既存方式 [2] で，トポロジ上の全ユーザが目的コンテンツを 1 回要求した場合に，各ノード n に到着する Interest パケット数 B_n を計算機シミュレーションにより比較評価する.

従来方式ではパート 0 を取得する前に，ユーザはパブリッシャでアクセス制御を完了する必要がある. 一方，提案方式では，アクセス制御をルータで行うことができ，パート 0 をキャッシュすることで，完全なコンテンツに対する要求は 1 回のみで，2 回目以降の要求ユーザはルータでコンテンツを取得できる. ユーザ数が m のとき，従来方式はユーザのアクセス制御 Interest パケット (T_{AC}) を m 回，隣接ノードに転送する必要がある. そのため従来方式の各ノードの到着 Interest パケット数は，提案方式の m 倍となる (1).

$$T_{AC} = \begin{cases} 1, & proposed\ method, \\ m, & existing\ method, \end{cases} \tag{4.1}$$

複数のトポロジのシミュレーションでは, 提案方案はすべて異なるフロー削減をもたらす. 図 4.1〜図 4.4 にシミュレーション結果を示す.

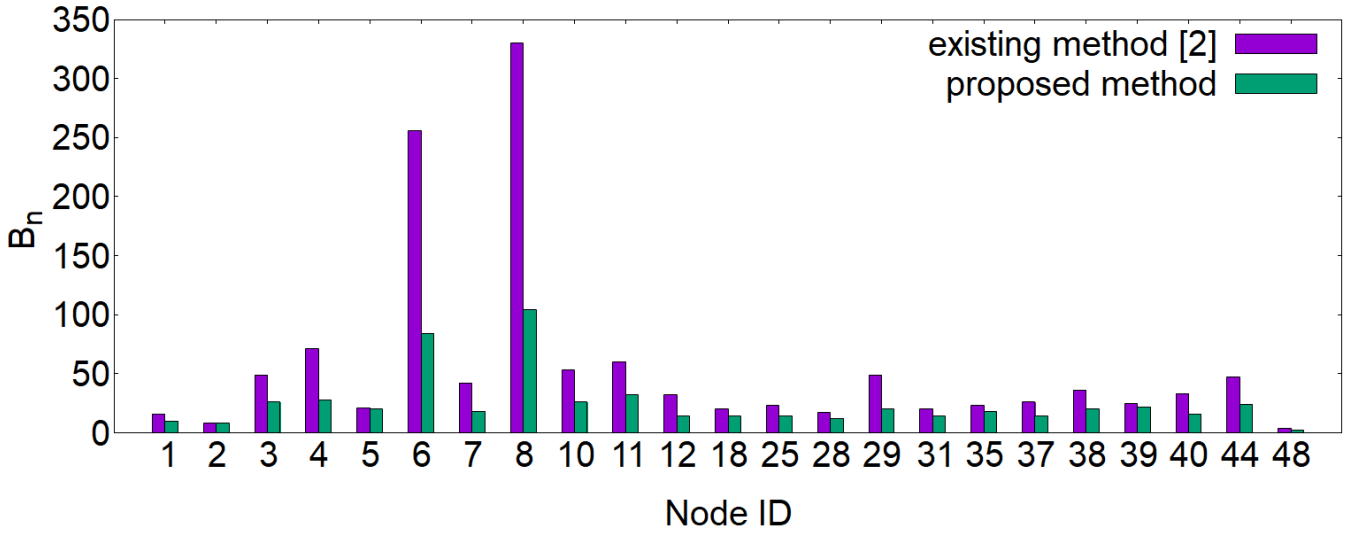

図 4.1: 単一パブリッシャにおける各ノードの到着 Interest パケット数 (Allegiance Telecom)

図 4.1 に単一パブリッシャの場合における 2 つの各方式の各ノードの B_n を，ノード番号に対してプロットする．パブリッシャのトラフィックの削減率は最大で 68 %と高い. リクエストの転送に関与するノードはすべてトラフィック量が削減するが，なかでも主要ハブノードであるノード 6 は，トラフィック量が 66 %減少している.

図 4.2 に複数パブリッシャの場合のシミュレーションで, パブリッシャはノード 24,34,43 に存在し, トラフィックの平均削減率は 63.5%である. 主要ハブノードのトラフィック削減率は 62.8%である. これは提案方式が負荷の高いノードが処理する必要のあるリクエストを効果的に削減できることを意味する.

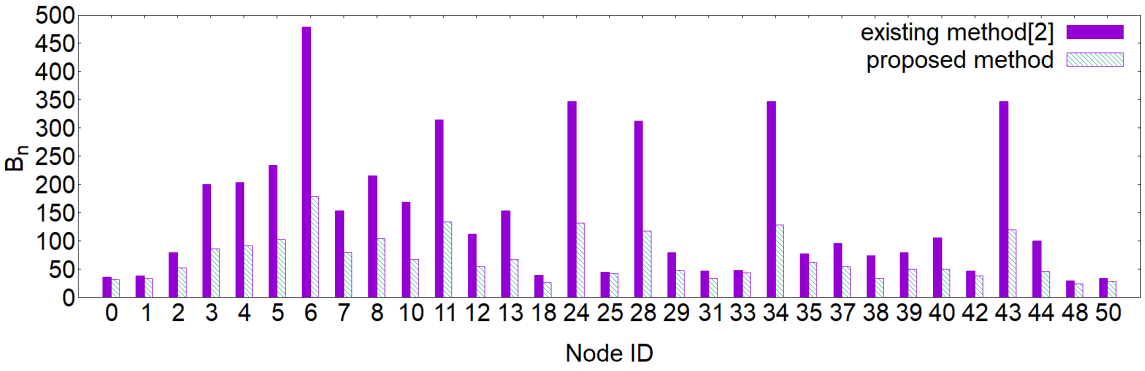

図 4.2: 複数パブリッシャにおける各ノードの到着 Interest パケット数 (Allegiance Telecom)

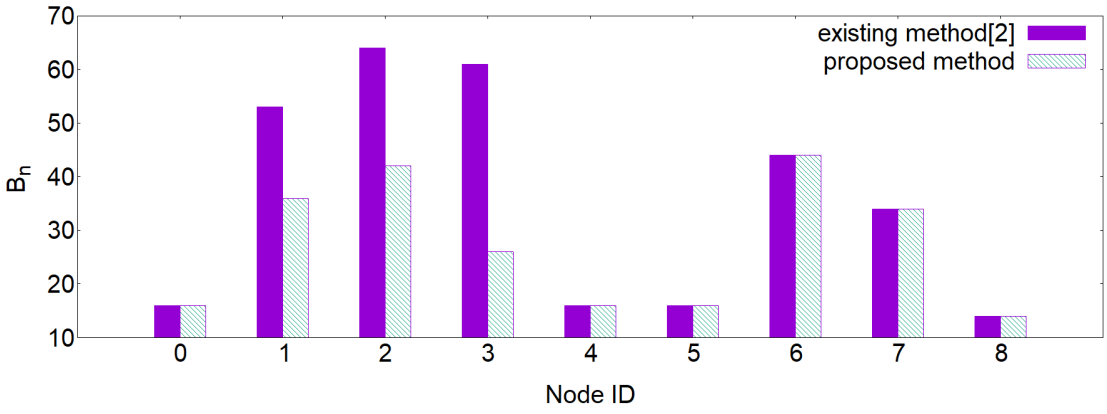

図 4.3: 複数パブリッシャにおける各ノードの到着 Interest パケット数 (GlobalCenter)

図 4.3 のトポロジのノードは, すべて度数 1（各ノードが他のノードに接続されている）であり, パブリッシャのいるノードに直接アクセスできる. そのため，パブリッシャのアクセス制御タスクが軽減され，トラフィックが減少したパブリッシャがいるノードを除いて，ノードのトラフィック量に差異は見られない.

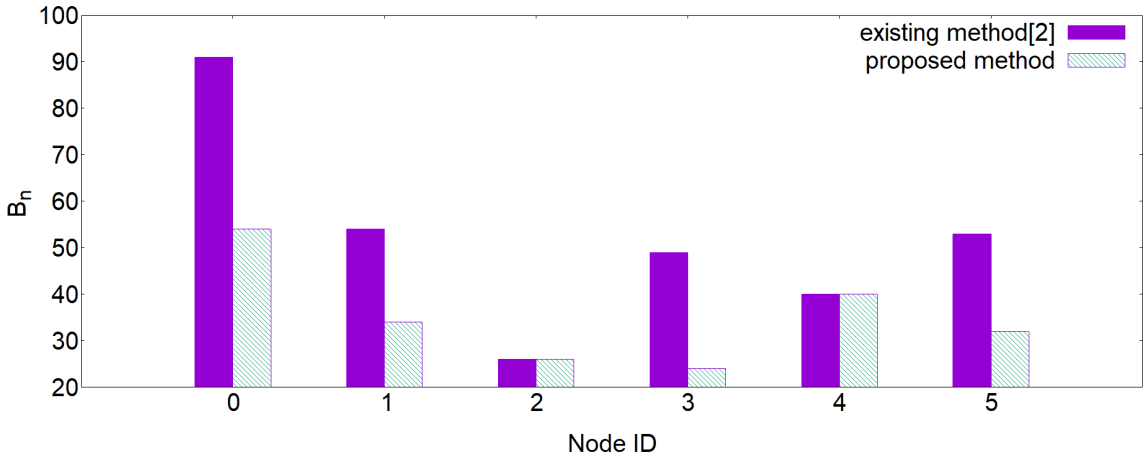

図 4.4: 複数パブリッシャにおける各ノードの到着 Interest パケット数 (Nap_Net_LLC)

図 4.4 のトポロジは簡単過ぎるため, 全ての経路がノード 0 に通過しなければならない. つまり，ハブノードはノード 0 である. パブリッシャにいるノードは 1,3,5, 以上ノードのトラフィック削減をより明確に見ることができる. そして, ノードのトラフィックは依然として 40 %削減されている. しかし, トポロジのノード数が次数分布に対して減少するにつれて, トラフィックの削減率も大幅に減少する. つまり，トポロジが複雑であればあるほど, 提案方式は効果的である.

4.3　暗号化方式のオーバヘッド比較

CP-ABE における暗号化処理では, 複雑なアクセス制御ポリシーとデータを組み合わせることになる. これらのポリシーは通常, 木構造や行列などの属性論理式として表現される. 暗号化処理では, これらのポリシーを評価し, 対応する属性情報を暗号化された暗号文に埋め込む必要がある. パブリッシャがコンテンツを暗号化する際に, 暗号化アルゴリズムがパブリッシャに課す負担は, やはり考慮すべき要素であるため, このセクションでは暗号化アルゴリズムの消費量を評価する.

提案方式は,[5] と共通鍵暗号のハイブリッド暗号方式を用いるため, シミュレーション条件は AES-256(Advanced Encryption Standard) 鍵の暗号化であり, 比較する暗号アルゴリズムは, 現在主流の公開鍵暗号アルゴリズムである Rivest-Shamir-Adleman アルゴリズム (RSA) である. シミュレーション環境は 4 コア,4GB RAM の Ubuntu 20.04 クラウドサーバで,2 つのアルゴリズムの暗号化処理は python 3.7 の Charm ライブラリで実装されている. 2 つの暗号化アルゴリズムが AES-256 鍵を暗号化するのに消費する時間を比較することで, 暗号化の消費がパブリッシャにどれだけ負担をかけるかを評価する.[5] の暗号化で使用される属性の数は 5,10,20,30,40 であり,RSA 暗号化で使用される公開鍵の長さは 1024,2048,4096 である.

以下図 4.5〜図 4.6 にシミュレーション結果を示す.

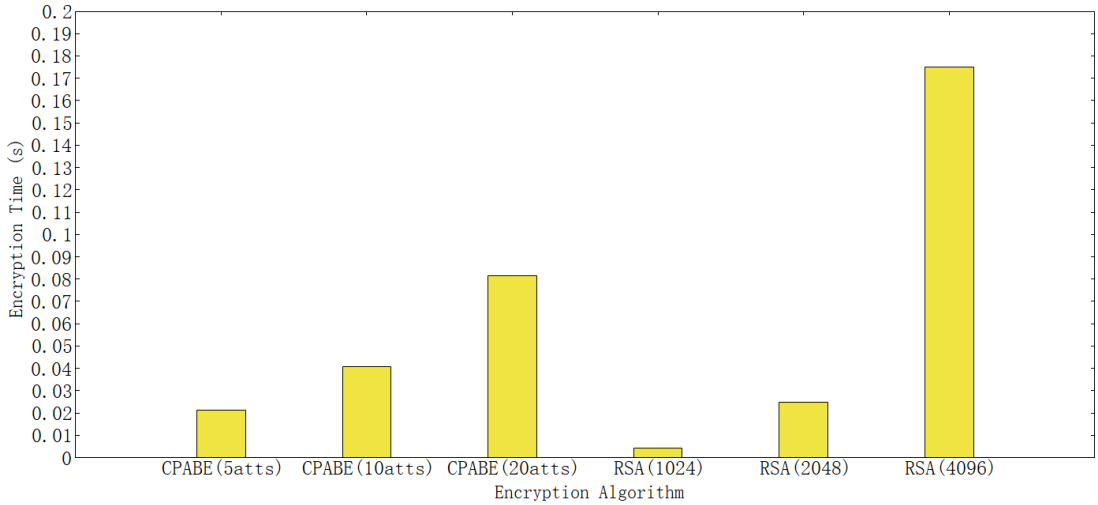

図 4.5: 単一ファイルの暗号化に要する時間

図 4.5 に,2 つの暗号化の異なるケースを, 単一ユーザの場合の暗号化かかる時間に対してプロットする. 公開鍵長 1024bit の RSA が AES-256 鍵の暗号化に要する時間は,[5] のすべてのケー スにおける暗号化経過時間よりもはるかに短いのに対し, 公開鍵長 2048bit の RSA の 暗号化経過時間は,AS 属性数 5 の CPABE の暗号化経過時間とほぼ同じである. これは CPABE が属性ベースの暗号化メカニズムを使用しているためで, 暗号化された暗号文は属性の セットと対応するポリシーに依存する. これらのポリシーは通常, 複雑な論理式として表現され, 暗号化と復号化処理にはペア演算と複雑な数学計算が含まれる. そのため, 公開鍵が短い場合,RSA の暗号化消費量は CPABE よりも少なくなる.

しかし, 公開鍵長 4096bit の RSA の暗号化消費量は,AS 属性数 40 の CPABE の暗号化消費量よりも多い. 一方,CPABE の暗号化消費量は,AS 属性数 5 の 1024bit RSA の暗号化消費量よりも多いにもかかわらず,[5] の消費量の増加は RSA の消費量よりもはるかに少ない. これは,RSA 暗号化の計算消費量が鍵の長さと密接に関係しているためである. RSA は主にモジュロ指数演算を行い, その計算量は鍵長の増加とともにほぼ指数関数的に増加す

る. 一方,CPABE の属性数の増加は主に AS ノード上の多項式の計算を増加させ, その計算消費量は属性数の増加とともにほぼ線形に増加する.

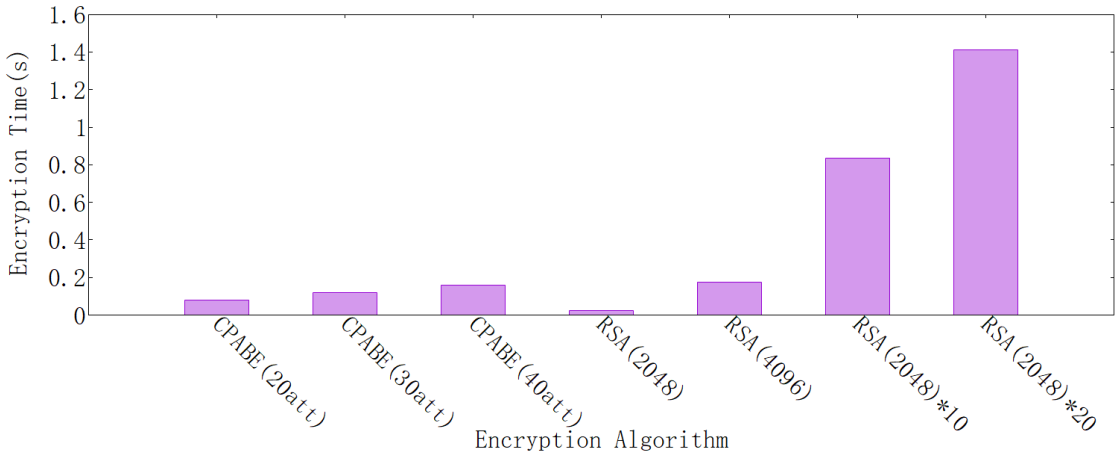

図 4.6: 複数ユーザの場合の暗号化に要する時間

一方, 図 4.6 は, 複数のユーザが存在する場合の 2 つの暗号化の異なるケースを, 暗号化処理時間に対して示したものである. 公開鍵長 2048bit の現在の RSA はまだ安全であると考えられるので, このシミュレー ションではこの条件の RSA を選択した. 複数のユーザが存在する場合, 公開者による暗号化回数対ユーザ数を図 1.1 に示す. RSA は 1 対 1 の暗号化方式に属するため, 発行者はユーザ数に応じてピアツーピアの暗号化を行う必要があり, 図 4.6 のシミュレーションは, ユーザ数が 10 人の場合と 20 人の場合で発行者が同時に暗号化を行った場合である.CPABE は 1 対多の暗号化方式に属し, 消費する暗号化時間は属性数のみに関係する. ただし, 複数ユーザの場合は暗号化する属性数が増える可能性があるため,20,30,40 属性の場合をシミュレートした. マルチユーザの場合に RSA を使用すると, パブリッシャの消費量が急激に増加することがわかるので,CPABE はパブリッシャがコンテンツを暗号化する際のオーバヘッドを効果的に削減できると判断できる.

4.4 ACL の数の削減評価

[1] において, ルータは UID のハッシュテーブルを利用して Interest パケットをフィルタリングする. この ACL に類似したフィルタリングメカニズムは,NDN におけるトラフィックを制限し, 承認された Interest パケットのみが通過できるようにする. これにより, 悪意ある攻撃から機密内容を保護し, 不正アクセスによるリンクの混雑への影響を軽減する.

しかし,NDN の通信方式は接続レスであり, 転送メカニズムはアドレスに基づくものではなく, データ名に基づいている. このため,[1] のルータは多数のフィルタリングエントリ（ACL エントリ）を蓄積することとなる. ルータに過剰な ACL エントリを設定することは, ネットワークのパフォーマンスと管理に多くの影響を及ぼす. 第一に, 多くの ACL エントリの処理には追加の CPU リソースとメモリが必要となり, 処理時間の増加によりネットワークのパフォーマンスが低下する. 次に, ルータのハードウェアリソースがルールの過多により枯渇し, ルータが性能の低いソフトウェアパスを利用して ACL を処理する必要が生じることで, 効率がさらに低下する可能性がある. このため,SDN の集中管理モデルに基づく提案として, コントローラが ACL エントリを集中管理し配布する方法が挙げられる.[3][4] によると, コントローラはユーザとルータのマッピングテーブルを取得でき, 提案方法により各ルータの ACL エントリを訪問制御に必要な最小限のエントリ数（即ち, ルータに直接接続されている承認されたユーザ数）に削減できる.

したがって, このシミュレーションは, 提案方式を使用した後, 単一のルータ内の ACL エントリ数の平均削減率と, トポロジ全体の ACL エントリの削減率を評価するために行われる. コントローラはノード内の登録ユーザに基づいて対応する ACL を作成できるため, 提案における各ルータの ACL エントリ数 n_i はノード内の承認されたユーザ数となる. 一方,[1] におけるルータの ACL エントリ数 n_u は承認されたユーザ全体の数となる.

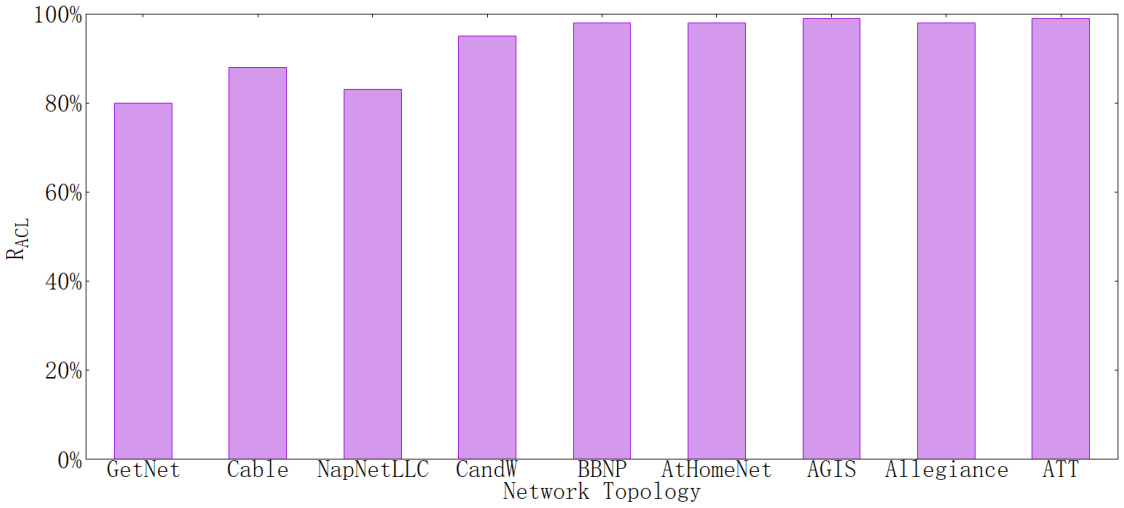

図 4.7: ACL エントリ総削減率

図 4.7 では, 全体の ACL エントリの総削減率 R_{ACL} と各トポロジとの関係が示されている. このトポロジのノード数を N_{node} とすると, 提案された手法と [1] における ACL エントリの総数 n_{ACL} は等式 5.2 のようになる.

$$n_{ACL} = \begin{cases} \sum_{i=1}^{N_{node}} n_i = n_u, & proposed\ method, \\ \sum_{i=1}^{N_{node}} n_u = N_{node} n_u, & existing\ method[1], \end{cases} \tag{4.2}$$

したがって, 総削減率は等式 5.3 のように計算できる:

$$R_{ACL} = \frac{N_{node} - 1}{N_{node}} \tag{4.3}$$

等式 5.3 は, 総削減率がノード内の人数やユーザ数とは無関係であり, ノード数が多いトポロジでの削減効果がより顕著であることを示す. シミュレーションの結果もこの結論を支持し, 図 4.7 ではより複雑なトポロジで常に高い削減率が得られる.

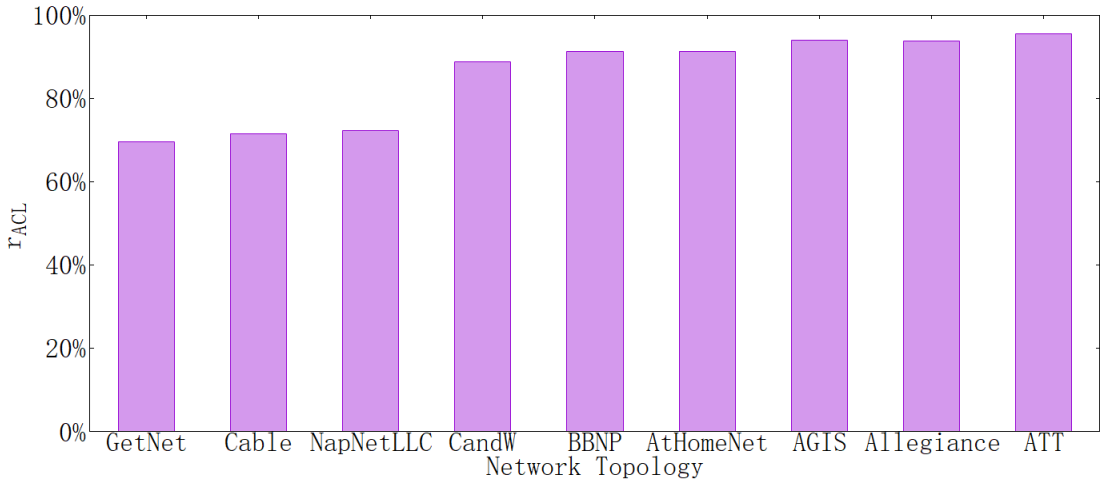

図 4.8: ACL エントリ平均削減率

提案方式が単一ルータの ACL エントリに対してどのような最適化効果を持つかを観察するために, 図 4.8 では各トポロジにおける単一ルータの平均 ACL エントリ削減率 r_{ACL} のシミュレーションを行った.

トポロジ内の各ノードの人口数に大きな差があるため, 単純な平均値では全体のトポロジにおける平均 ACL エントリ数の状況を正確に評価することができない可能性がある. したがって, このシミュレーションでは, ノードの人口を重み w_i とした加重平均数を各ルータの平均 ACL エントリ数として使用し, 人口の多いノードが平均値に与える影響を確保している. 等式 5.4 が平均削減率 r_{ACL} の計算式である.

$$r_{ACL} = 1 - \frac{\sum_{i=1}^{N_{node}} w_i n_i}{n_u \sum_{i=1}^{N_{node}} w_i} \tag{4.4}$$

人口を重みとして加えた後, 各トポロジにおける単一ルータの ACL エントリ数の平均削減率は若干低下したものの, 依然として 70 %の最適化比率を維持し, 最大で 95 %に達することができる. シミュレーションの結果は, 提案方式を使用することでルータの ACL 数を効果的に最適化し, ルータがアクセス制御段階で迅速に検索を行えるようにすることが可能であることを証明している.

4.5 制御情報交換効率の比較

NDN ネットワーク内でルータの ACL を更新する際,NDN のルーティングと転送メカニズムがデータ名に基づいており, 目的地アドレスに基づいていないため, パブリッシャは特定のルータの更新を指定することができない. 全域ルータを一括で更新すると, パブリッシャに一定のアクセス負荷がかかるだけでなく, 更新効率も低下する. したがって, コントローラが権限更新情報を特定のルータに配信することで, 特定のルータ上の ACL をより効率的に

更新できる. しかし, コントローラがパブリッシャからの更新情報を受け取り, 新しい ACL を生成して特定のルータに配信するまでには一定の処理時間がかかり, 転送に一定の遅延が発生する可能性がある. そのため, このシミュレーションでは, 提案方式と NDN 通信モデルの両方を使用して ACL を更新する際に必要な転送時間を評価する.

シミュレーション環境は,Ubuntu 20.04 がインストールされた 4 コア 4GB メモリのクラウドサーバである. 実際のトポロジの分布に基づき,pyNDN を使用して NDN ルータの転送行動をシミュレートし,Ryu+Openflow 1.3+sqlite3 を使用して NDN ルータとコントローラの通信および権限情報の更新をシミュレートする. 更新にかかる時間 T_u は, パブリッシャの次のルータから目的ルータまでを含む.NDN ノードの転送にかかる時間 t_n,openflow の packet_in および packet_out パケットの転送にかかる時間 t_{of}, リンク上の転送にかかる時間の平均値 t_r, ルート上のホップ数 n_h, コントローラの処理にかかる時間 t_c である.NDN 通信ではデータの伝送に Data パケットのみを使用できるため, パブリッシャはまず Interest パケットを送信してルータに通知し, ルータが確認の Data パケットを返信した後, パブリッシャが新しい ACL の Interest パケットを送信し, パブリッシャが Data パケットをルータに送ることができる. そのため, シミュレーションでは NDN ノードの転送にかかる時間 t_n は応答と転送の時間であり,2 回実行する必要がある. 一方, 提案方式のルートではまず起始ルータからコントローラに向かい, 次にコントローラから目的ルータへの転送が必要である.T_u の計算式は等式 5.5 である.

$$T_u = \begin{cases} t_c + n_h(t_r + t_{of}), & proposed\ method, \\ 2n_h(t_r + t_n), & NDN method, \end{cases} \tag{4.5}$$

シミュレーションでは, コントローラをトポロジ内の接近中心性が最も高いノードに配置する. また, パブリッシャをトポロジの中心とエッジに配置し, トポロジの異なる場所での二つの方式の更新時間に差があるかどうかを観察する. シミュレーション結果では, 横軸がホップ数, 縦軸が T_u である. シミュレーション結果の図 4.9-図 4.14 から, ホップ数が一定の範囲以下の場合, 提案方式の配信効率は理想的ではないことが分かる. コントローラの処理時間が長いため,NDN を使用して直接転送する方が効果的である. そのため, 簡単なトポロジやノード数が少ない, またはノードの全体的な度数が高いトポロジである DataXchange_Network_Inc などでは, 提案方式の配信にはより多くの時間がかかる. しかし,CAIS_Internet のような複雑なトポロジで, ノード数が多く全体的な度数が低い（すなわち, トポロジ内のノードの接続方式が多くが直列構造である）場合, 接近中心性が高いノードであってもエッジノードであっても, 長距離ルーティングが存在するため, この時点で提案方式の利点が明らかになる.

しかし,Allegiance_Telecom などの複雑なトポロジでは, 接近中心性が高いノードであってもエッジノードであっても, 長距離ルーティングが存在するため, 提案方式の利点が明らかになる. 権限更新情報は直接目的ルータに送信され, 関連のないルータの処理時間を省くことができる. また,openflow パケットの処理速度は NDN パケットの処理速度よりも高いため, 遠くのルータを更新する必要がある場合, 提案方式は更新効率を向上させることができる.

また, 複数のパブリッシャの状況を評価し, 提案方式による配信効率の最適化を研究するために, 上述の三つのトポロジに対して複数パブリッシャのシミュレーションを行った. シミュレーション条件は次の通りである：パブリッシャは任意のノードにランダムに出現し, 特定のノードのルータ ACL をランダムに更新する. その後, 更新に必要な時間の平均値 T_a を計算する. 各ホップ数内のパブリッシャのサンプルは 100 名である. シミュレーション結果の図 4.15 から, パブリッシャがランダムに生成された場合, トポロジ全体の平均消耗時間の結果は, 提案方式が平均次数が小さいトポロジのを更新する際に優れたパフォーマンスを示すという上述の結論を基本的に支持していることが分かる. 平均次数が小さいトポロジでは, ノード間の最短経路がより多くのノードを通過する必要があるため（図 4.18 の CAIS_Internet のように）,NDN 方式での更新にはより多くの時間がかかる. 平均次数が大きいトポロジ（図 4.17 の ATT のように）を更新する際, 提案方式は NDN の方式に比べて効率が低いが, コントローラの処理ロジックを最適化することで, より優れた更新効率を達成することが可能である.

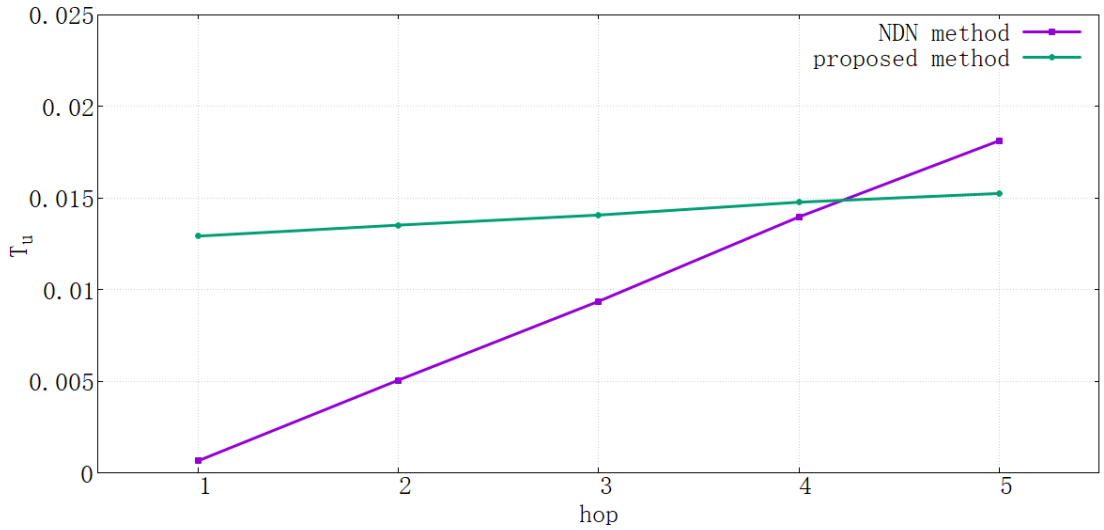

図 4.9: パブリッシャが中心ノードに配置する場合の転送遅延 (トポロジ：Allegiance_Telecom)

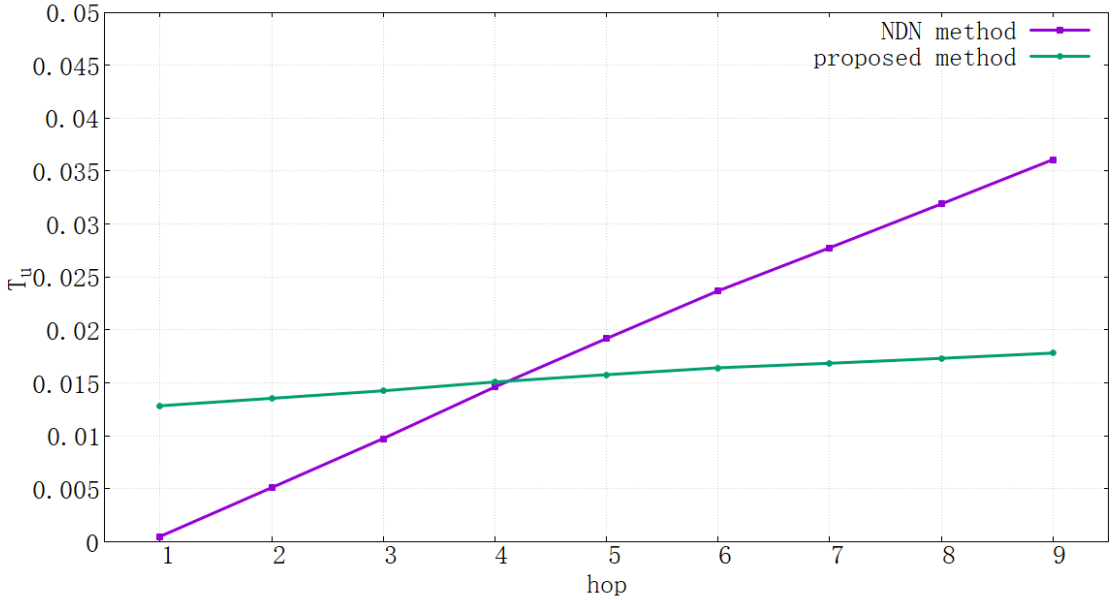

図 4.10: パブリッシャが縁辺ノードに配置する場合の転送遅延 (トポロジ：Allegiance_Telecom)

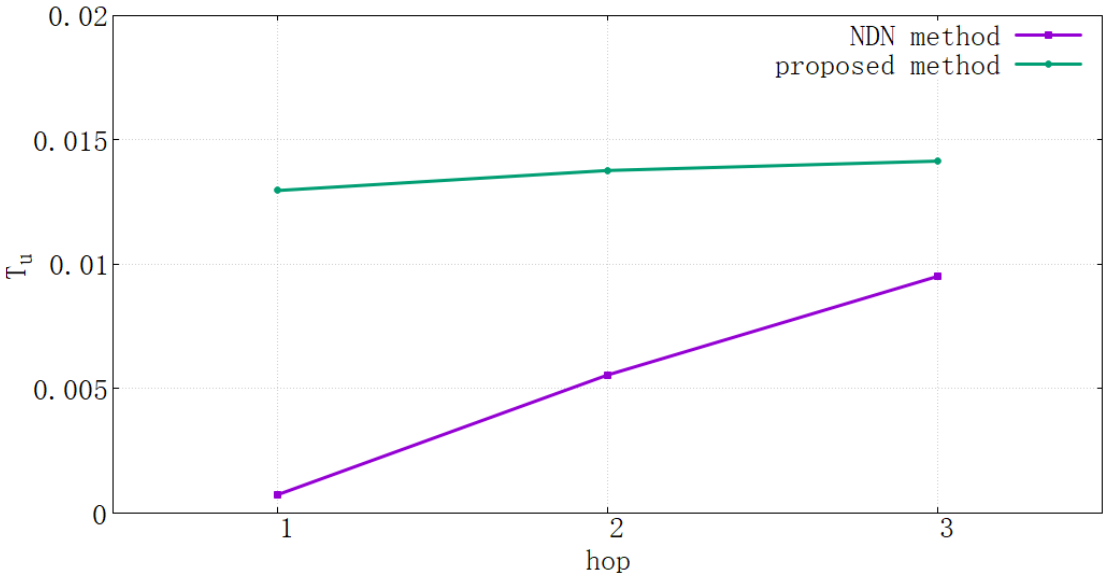

図 4.11: パブリッシャが中心ノードに配置する場合の転送遅延 (トポロジ：ATT)

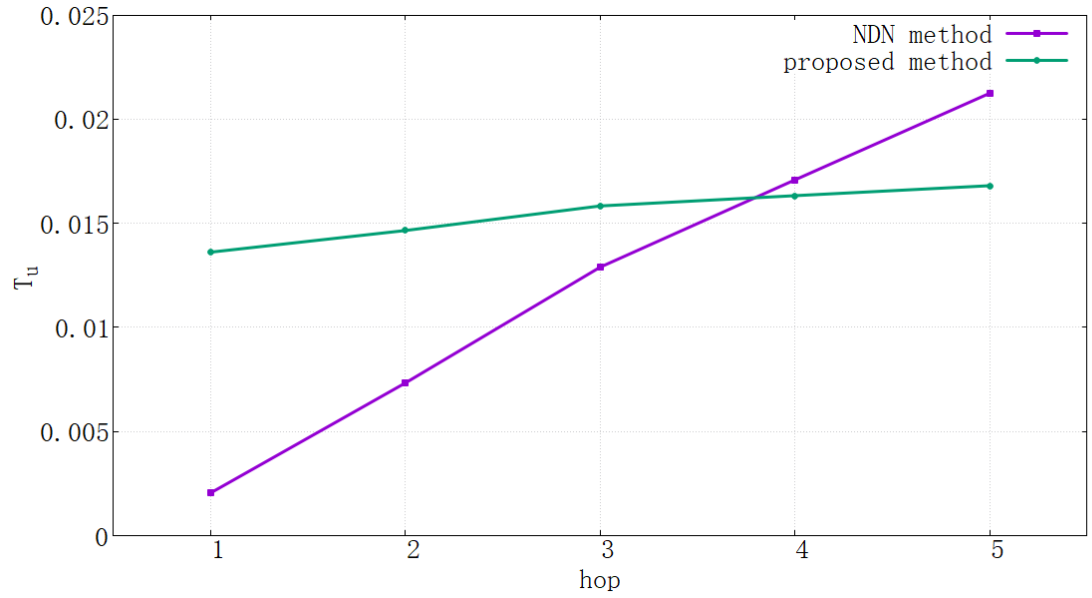

図 4.12: パブリッシャが縁辺ノードに配置する場合の転送遅延 (トポロジ：ATT)

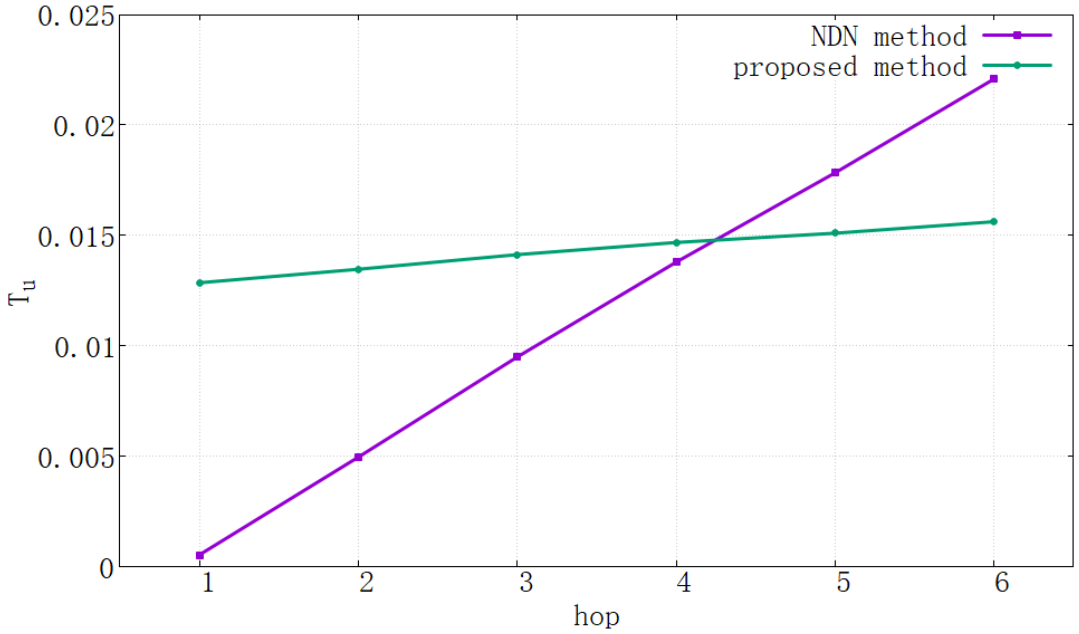

図 4.13: パブリッシャが中心ノードに配置する場合の転送 (トポロジ：CAIS_Internet)

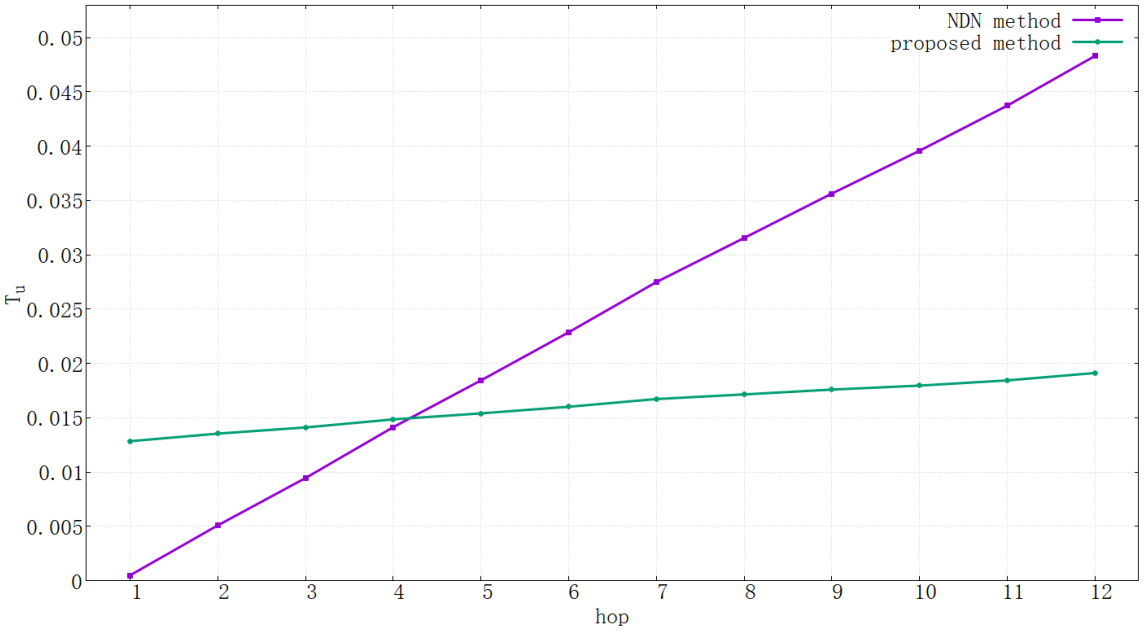

図 4.14: パブリッシャが縁辺ノードに配置する場合の転送遅延 (トポロジ：CAIS_Internet)

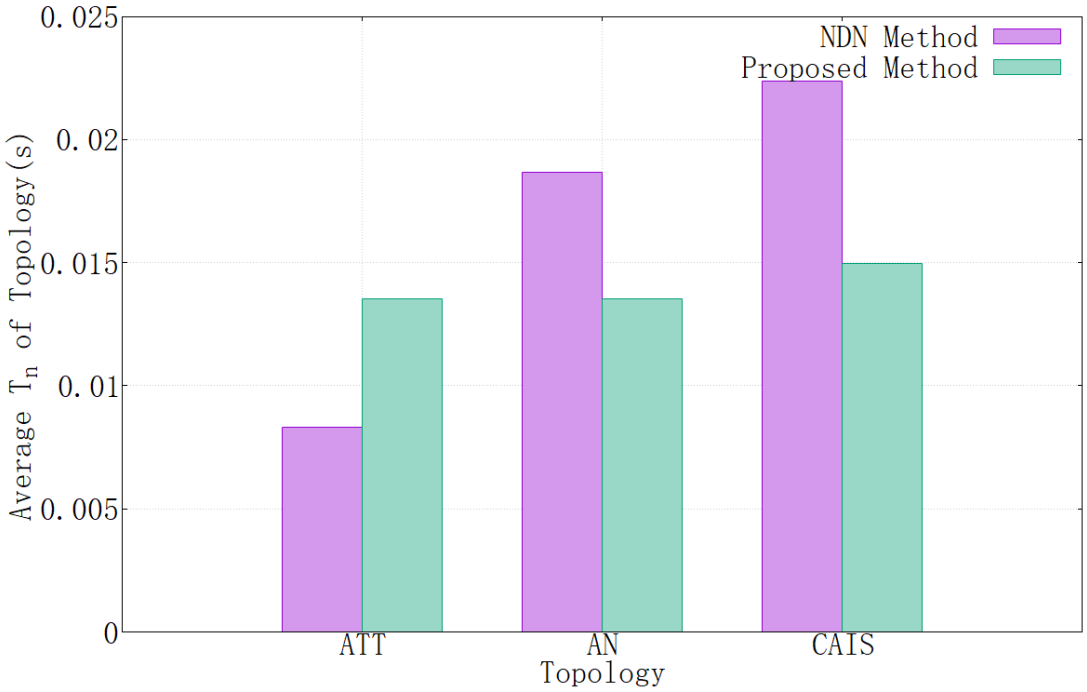

図 4.15: 平均更新時間

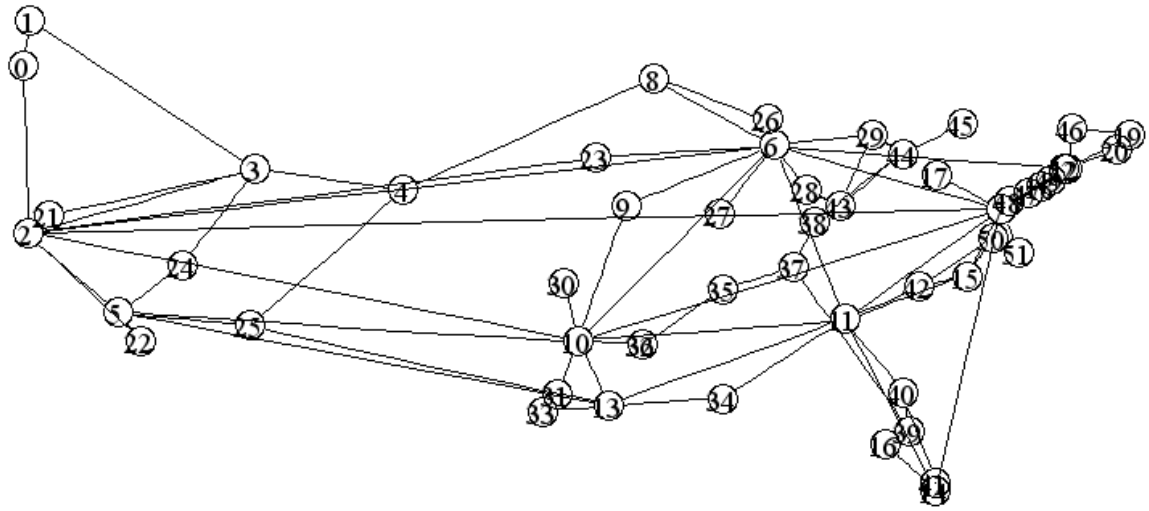

図 4.16: Allegiance_Telecom のトポロジ

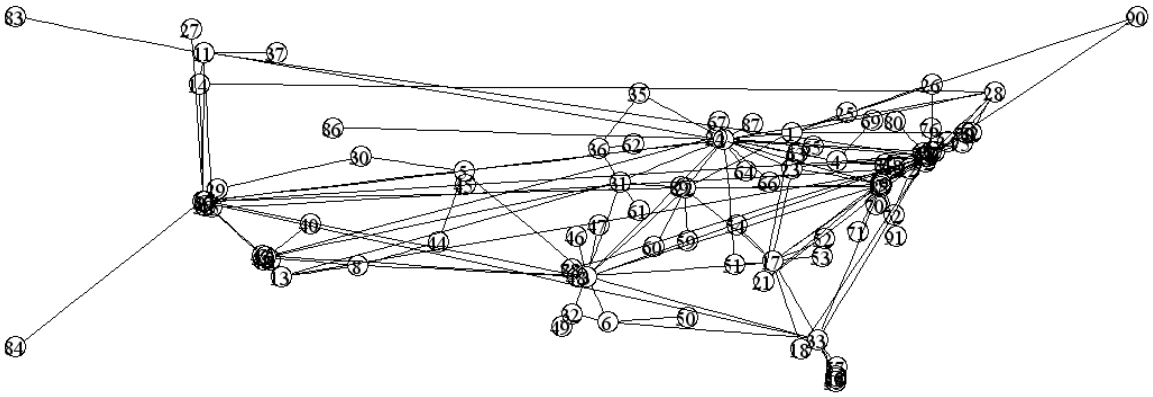

図 4.17: ATT のトポロジ

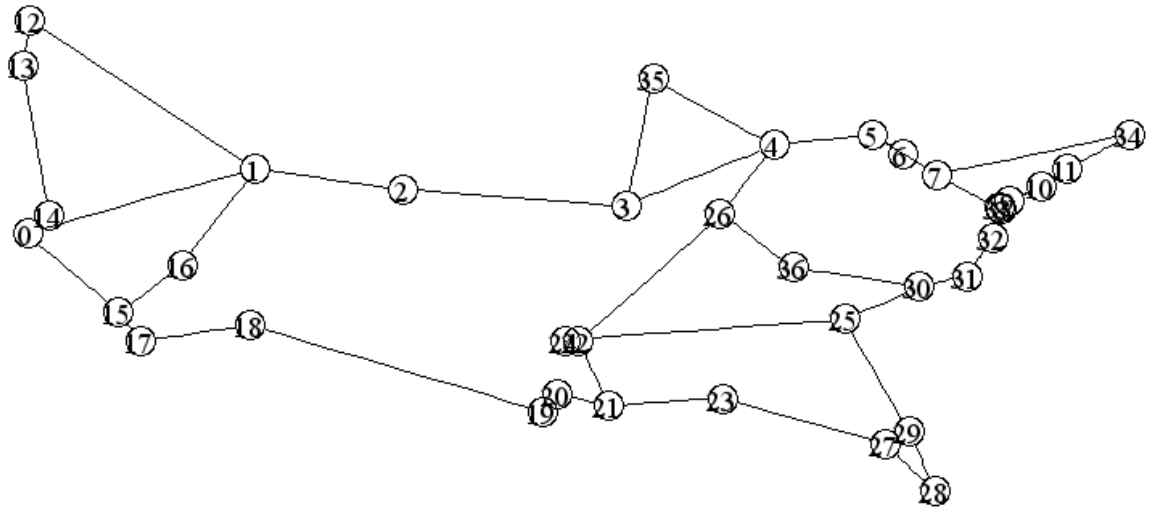

図 4.18: CAIS_Internet のトポロジ

第5章　結論

NDN は最も注目されている次世代の先進ネットワーク技術として, コンテンツの発行者がルータ内のコンテンツキャッシュへのアクセス制御を行うことが難しいため, 効果的なアクセス制御メカニズムは NDN の応用と展開に影響を与える. 本稿では,SDN に基づいた NDN アクセス制御メカニズムを提案し, アクセス制御の作業を各アクセス層ルータに分散させ, ルータがキャッシュアクセスのフィルタリングを担当する. そして, コントローラが暗号化キーとユーザ権限を集中管理し, 重要な秘密情報とデータ層を分離する. 最後に,CPABE の改良アルゴリズム [5] を使用し, 暗号化キャッシュの流通性を保証すると同時に, ユーザの使用期限をユーザのキーに追加し, 暗号化されたコンテンツの安全性を強化する. その後, 提案の安全モデル, トラフィックモデル, 暗号化の消費,ACL の分布状況, およびアクセス権限の更新速度について評価を行った.

安全モデルの分析により, 提案方式は権限管理の安全性, データの安全性, および権限変更後の前方安全性と後方安全性を保証していることが確認された. トラフィックモデル, 暗号化の消費, および ACL の分布状況から，本稿の提案方式が既存の方式に比べてより良い最適化効果を有していることを明らかにした.SDN コントローラを使用して権限を更新する際, 近隣のルータへの更新には,NDN 通信方式よりも長い時間が必要である. 今後は, コントローラの処理ロジックに対するさらなる最適化が求められる.

第6章　謝辞

本研究を行うに当たり，ご指導を頂いた上山教授に感謝します．また日常，有益な議論をして頂いた研究室の皆様に感謝します．

関連図書

[1] WU Zhijun, XU Enzhong. Access control method based on CP-ABE in NDN[J]. Journal of Civil Aviation University of China, 2020, 38(2): 18-24.

[2] Y. Fukagawa and N. Kamiyama, “ Access Control Method with Privacy Preservation in NDN, ” IEEE ICNP 2023 (Poster)

[3] M. Alhowaidi, D. Nadig and B. Ramamurthy, "Cache Management for Large Data Transfers in Named Data Networking using SDN," 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Goa, India, 2019, pp. 1-6

[4] E. Aubry, T. Silverston and I. Chrismen, "Implementation and Evaluation of a Controller-Based Forwarding Scheme for NDN," 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), Taipei, Taiwan, 2017, pp. 144-151

[5] Zhe Liu, Fuqun Wang, Kefei Chen, Fei Tang, and Kaitai Liang. 2020. A New User Revocable Ciphertext-Policy Attribute-Based Encryption with Ciphertext Update. Sec. and Commun. Netw. 2020 (2020)

[6] L. Yu, S. Gao, N. Liu, H. Wang and W. Su, "A Cache-enabled NDN Forwarding Plane based on Programmable Switches," 2022 International Conference on Networking and Network Applications (NaNA), Urumqi, China, 2022, pp. 152-156

[7] S. Signorello, R. State, J. François and O. Festor, "NDN.p4: Programming information-centric data-planes," 2016 IEEE NetSoft Conference and Workshops (NetSoft), Seoul, Korea (South), 2016, pp. 384-389

[8] NDN Packet Format Specification v0.3, 2023/1/30, https://docs.named-data.net/NDN-packet-spec/0.3/.

[9] W. Feng, X. Tan and Y. Jin, "Implementing ICN over P4 in HTTP Scenario," 2019 2nd International Conference on Hot Information-Centric Networking (HotICN), Chongqing, China, 2019, pp. 37-43, doi: 10.1109/HotICN48464.2019.9063219.

[10] OpenFlow Switch Specification Version 1.3.0(Wire Protocol 0x04),https://opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.3.0.pdf

外部発表文献

[1] 楊湛斌, 上山憲昭, “ SDN を用いた NDN のアクセス制御方式 “ , 電子情報通信学会ソサイエティ大会, 2023 年 9 月

[2] 楊湛斌, 上山憲昭, “ SDN を用いた NDN のアクセス制御方式の性能評価 “ , 電子情報通信学会総合大会，2024 年 3 月