

6-2 情報指向型ネットワーク技術

6-2 Information-/ Content-Centric Networking

朝枝 仁 松園和久 大岡 睦

Hitoshi ASAEDA, Kazuhisa MATSUZONO, and Atsushi OOKA

より高度化する通信サービスを効率的に実現するため、NICT では「情報指向ネットワーク技術 (Information-Centric Networking (ICN) あるいは Content-Centric Networking (CCN))」と呼ばれる新しいネットワークアーキテクチャの研究開発を行っている。具体的には、ICN/CCN 実現のために必要となる「ネットワーク内キャッシュ」「経路制御」「トランスポート」「セキュリティ」などの基礎研究に加え、Cefore と呼ぶ通信オープンソース・ソフトウェア・プラットフォームの開発を行っている。本稿では、特に「リアルタイムストリーミング用トランスポート技術」及び「Cefore 実装」に関して説明する。

We have been working on a new network architecture called “Information-Centric Networking (ICN)” or “Content-Centric Networking (CCN)”. ICN/CCN effectively facilitates future advanced communication services. As the fundamental research, we have been studying in-network caching, routing, transport, and security for ICN/CCN. We also have been developing an open software platform named “Cefore”. In this paper, we describe a newly developed transport mechanism for real-time streaming over ICN/CCN and the Cefore implementation.

1 まえがき

これまでインターネットはメールやファイル転送のような一対一の情報伝達を実現するための通信基盤として発展してきた。しかし近年、SNSをはじめとするコミュニティー通信や、多数のユーザーが同時に視聴する超高精細なビデオ配信のための通信、IoT と呼ばれるモノとモノ (あるいはモノとヒト) がつながる通信が通信量の大半を占めている。また近い将来に向け、ネットワーク上に流通する大量のデータから必要なデータを抜き出し、人工知能 (AI) や機械学習機能を用いて情報解析し、最適な情報を人間やロボット、センサーなどへ提供可能にする通信技術の研究なども進められており、インターネットサービスは更なる進化が予想される。このように、人々が求めるアプリケーションやサービス、環境は常に変化し、データ量やネットワーク接続されるデバイス数も今後益々爆発的に増加するであろう通信環境において、クラウドや Content Distribution Network (CDN) などの既存技術を拡張していくだけでは多様かつ新しい要求に対応できない。このため、従来の Internet Protocol (IP) 通信よりも高速で効果的、効率的、スケーラブルで省エネルギーな通信、そしてセキュアでプライバシーも考慮した安全な通信を実現する、新しい通信プロトコル、

新しいネットワークアーキテクチャの提案が欠かせない。

2 情報指向ネットワーク技術

将来的なインターネット通信サービスが直面する諸問題を解決するための新たなネットワークアーキテクチャとして「情報指向ネットワーク技術 (Information-Centric Networking (ICN) あるいは Content-Centric Networking (CCN))」が注目されている。この通信技術の原点になったものは、「本来、通信サービスが求めるものはサーバーへのアクセスではなく、『情報』の取得にほかならない」という考えである。つまり、「もし近くに自分が欲しいコンテンツを持っている人 (実際はネットワーク機器や PC など) がいるならば、そこからコンテンツを取得すればよい」という発想である ICN/CCN では、情報を取得するために必ずサーバーの位置 (IP アドレス) を調べてそのサーバーにアクセスしなければならないというこれまでの通信プロトコルの制約を排除し、「情報 (若しくはコンテンツ)」そのものを示す識別子 (コンテンツ名など) を利用し、それを指定してネットワークから情報 (コンテンツ) を取得することを可能とする。

3 情報指向ネットワーク技術の概要と特徴

ICN/CCN は従来のホスト中心の Internet Protocol (IP) からコンテンツ中心の通信モデルに移行する新しいネットワークアーキテクチャである。ICN/CCN では、従来の IP アドレスではなく、「コンテンツ名」を情報識別子として通信を行う。また、ルーターがデータ転送時に自律分散的にキャッシュする「ネットワーク内キャッシュ」を利用する。すなわち、ユーザーやアプリケーションからのデータ要求パケット (Interest と呼ぶ) を受け取ったルーターは、そこに示されたコンテンツ名を基に自身のキャッシュを調査し、キャッシュデータを保持している場合は、ユーザーに直接に (サーバーなどを介さずに) 要求されたデータを転送する。このためユーザーは遠方にあるかもしれないサーバーやクラウドへの接続を課せられることなく、近隣のネットワークの中から所望のデータを取得することが可能となる。さらに、マルチパスやマルチホーミング、マルチキャストといった効率性を高める通信技術をサポートすることで、ネットワーク資源の有効活用も行える。結果として、ユーザーにとっては遅延が少なく高品質な通信が、サービス提供者にとってはコンピュータやネットワーク資源を効率的に利用した省エネルギーな通信が可能となる。概念を図 1 に示す。

このようにいくつもの優位性がある ICN/CCN であるが、ICN/CCN の実現には、解決すべき研究テーマが数多く存在する。例えば「ネットワーク内キャッシュ」という技術に関して、ルーターが保持するキャッシュの容量が多ければ多いほど、キャッシュヒット率は向上し、ネットワーク全体で見たレスポンスは早くなる。しかし、キャッシュ容量を大きくすることはコ

ストの上昇も招くため、相対的な検討が必要になる。また、たとえキャッシュ容量を大きくしたとしても、全てのルーターが同じコンテンツ (例えば人気のあるコンテンツだけ) をキャッシュしていたのではネットワーク全体としてのパフォーマンス向上の効果は薄れてしまう。こういったコンテンツをどこに配置されたルーターでどれくらいの期間キャッシュすべきかなどは重要な研究課題である。

また、ICN/CCN 通信の前提になる「情報識別子」若しくは「コンテンツ名」に関し、グローバル・ユニークな (世界でただひとつの) 識別子をどう定義するかという議論も欠かせない。しかし現在の ICN/CCN においては、「情報識別子」や「コンテンツ名」というものは抽象的なものとしてとらえることができ、ウェブで用いられる URL のような階層的な名前 (例えば、/example.com/today/video.mpg) も、「○×倉庫の温度」というのも ICN/CCN で扱える情報識別子になる。ここで重要なことは、その情報識別子を、どこで、どのようにして用いるかということであり、つまりネットワークやサービスの適用範囲 (スコープあるいは名前空間) を決めれば、識別子自体が必ずしもグローバル・ユニークな識別子である必要はなくなる。とは言え、グローバル・ユニークな情報識別子に対する取決めも必要であり、これに関しては、ルール作りや標準化と共に議論されていく必要があり、時間をかけて多くの組織と一緒に決めていくものと考えている。

ICN/CCN 実現のために必要となる様々な研究テーマにおいて、NICT では「ネットワーク内キャッシュ」「経路制御」「トランスポート」「セキュリティ」などの基礎研究に加え、Cefore と呼ぶ通信ソフトウェア・プラットフォームの開発 [1] を行っており、それをオープンソースとして公開している。以下では、特に

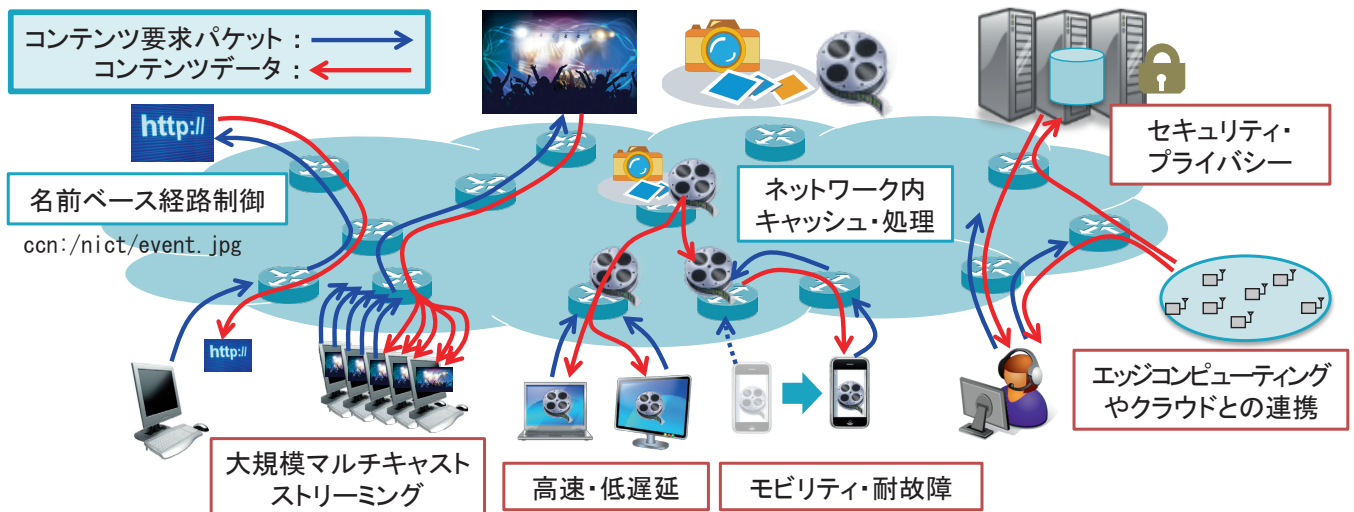


図 1 情報指向ネットワーク技術 (ICN/CCN) の概念図

「リアルタイムストリーミング用トランスポート技術」と「Cefore 実装」に関して説明する。

ところで、言葉の定義として、厳密には ICN と CCN は異なる意味をもつ。ICN はより抽象的なネットワーク技術を示し、CCN は ICN の代表的な一提案にすぎない。つまり、CCN 以外の ICN アーキテクチャもいくつか提案されているが、これまでの ICN 研究の潮流と CCN の特徴や利点を鑑み、NICT では CCN をベースとした研究を行っているため、本稿では ICN と CCN は同義のものとし、これ以降、CCN という言葉で統一して説明する。

4 ネットワーク内キャッシュとネットワーク内符号化を活用したリアルタイムストリーミング

近年 4K/8K などの高品位な映像(ストリーミング)サービスに対する需要が増しており [2]、今後も更なるストリーミング転送量の増加が予想される [3]。特にライブ放送のように低遅延が要求されるリアルタイムストリーミングサービスにおいて、ユーザーに対する通信品質 (Quality of Experience (QoE)) を高めることは最重要課題のひとつと言える。高品位ストリーミングにおける QoE 向上を目的とした場合、通信量の増加に応じてネットワーク容量を対応させる技術、あるいはデータ欠損を防ぐために再送される重複データ転送量を抑える技術が必要になる。しかし高品位なストリーミングであるがゆえに、ネットワーク帯域を占有してしまう可能性が高く、結果として輻輳をより増幅させてしまい、輻輳発生時に適切にデータ受信が行えず QoE の低下を招く恐れがある。このような背景において、我々は、CCN の概念に基づく、ネットワーク内キャッシュとネットワーク内符号化を活用したリアルタイムストリーミング用トランスポート技術を開発した [4]。当技術は、従来のホスト中心型の制御では不可能であった、ネットワークによる通信帯域や通信状態の把握及びそれに基づくデータ転送量の制御を可能とし、さらに従来の CCN よりも制御メッセージ転送量を減少させることで、ユーザーに対する QoE を向上させる。

提案手法の基本的な考え方は、CCN の特徴である「ホップバイホップ転送」を利用した通信制御技術である。簡単にいうならば、ストリーム受信者が送出するデータ要求パケット (Interest) と、それに対して送られてくるデータパケットがノード間 (例えば、ルーター間) で交換されるタイミングにおいて、各ノードが「リンク許容遅延」及び「パケット損失率」を把握しネットワーク内キャッシュとネットワーク内符号化に

よる冗長データを利用して損失データを許容遅延内に復元する [4]。さらに、ネットワーク資源を有効活用するために、「Symbolic Interest (SMI) [5]」と呼ばれる特殊な Interest メッセージを用い、効率的なマルチキャスト転送及びマルチパス転送を行い、コンテンツフロー同士が公正にネットワーク帯域共有を行えるように映像品質の調整を行う。

ストリーム受信者及びルーターが行う手順は以下のとおりである。受信者はデータ要求として Interest メッセージを送出する際、パケット中のオプションルヘッダにて SMI であることを明示し、ルーターが管理する「Pending Interest Table (PIT) (コンテンツ名と下流のデータ転送先インターフェースの対応表)」に、エントリーが維持される最大生存時間 (SMI_LT) (例えば、2 秒) を指定する。ルーターは SMI を受信して PIT に当該コンテンツ名のエントリーを挿入する際、データのセグメント ID の代わりに、「++segment」という予約語を付与し (例えば、/example.com/video1/++segment)、各データが持つ単一のセグメント ID を除く部分の名前が一致したストリームデータは全て送信することを記録する。SMI_LT が期限切れになるまで該当エントリーは消去されず、受信者は継続的にストリームデータを受信できるため、受信者は数秒間隔で SMI を転送し、上流のルーターが保持する PIT の SMI_LT を更新する。この手法は、受信者に複雑な通信制御を課すことなく、またコンテンツの再生ビットレートにも依存することなく、制御メッセージ (Interest メッセージ) 量を抑制しながら、常にマルチキャスト通信を行い、無駄な重複データ転送を回避させることができる。

図 2 に提案手法のシステムアーキテクチャを示す。SMI を利用することで、従来の受信したいパケット単位に要求しなければならない高頻度な Interest 送出を抑制することが可能となり、パケット単位ではなく時間単位でリアルタイムストリーミングデータを受信することができる。Regular Interest (RGI) は従来の CCN が行う受信パケット単位の Interest であり、これは損失データの再送を要求するために用いる。Control Interest (CNI) は、主に上流ノードに付与すべき符号化データの量を指定するために用いる。ビデオデータは Scalable Video Coding (SVC) による階層符号化データを用いる。

コントロールプレーンでは、受信者はビデオ品質 (例えば、コーデックの階層) を指定して SMI を発行する。各ルーターは上流リンクのネットワーク状態と推定リンク許容遅延を基に、(1) 使用するリンクとビデオ品質を決定し SMI を送信し、(2) 目標データ転送成功率を達成すべく、必要に応じて冗長符号化データの使用

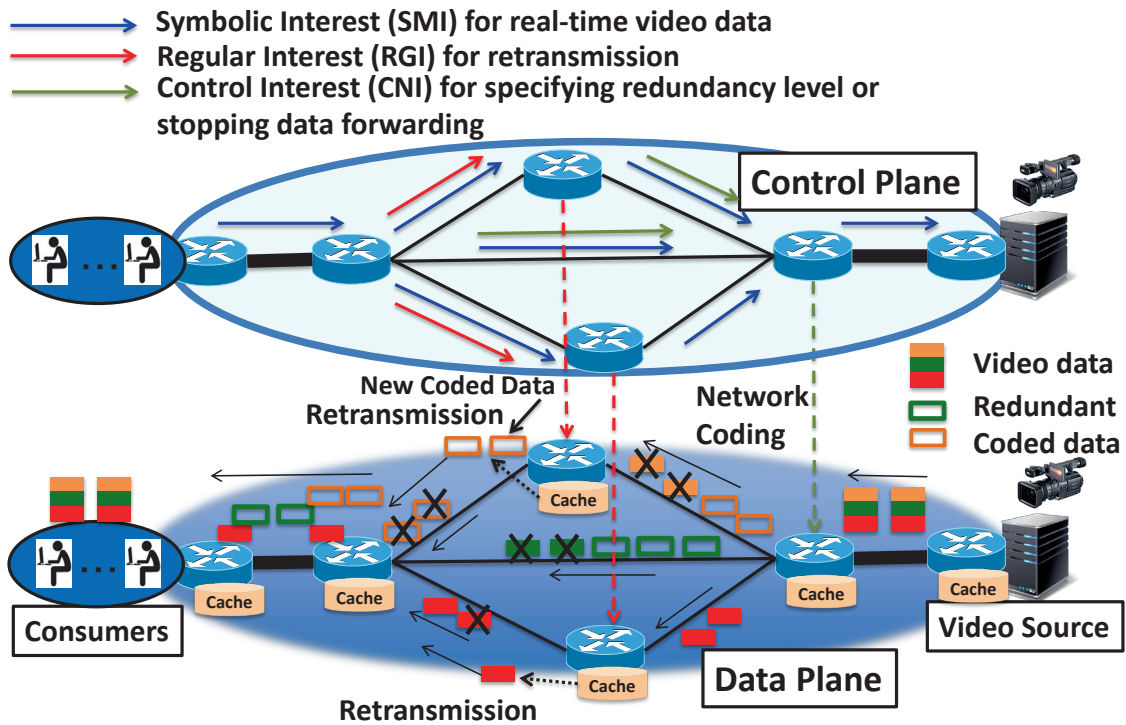


図2 提案手法のシステムアーキテクチャ

とその転送量を決定し CNI にて通知し、(3) RGI を用いて上流にキャッシュされている損失データまたは符号化データを要求する。任意のフローに関してパスの切替えや転送中止を行う場合にも CNI を用いて上流ルーターに通知を行う。リンク許容遅延推定を各ルーターが行えるようにするため、受信者は SMI にアプリケーションに依存する許容遅延をオプションヘッダに記述する。ルーターは定期的に CNI を上流及び下流ルーターに転送し、その応答パケットを受信することでアップリンク/ダウンリンクの往復伝送遅延の最大値と最小値を把握する。SMI 及びデータを転送する際、上流/下流の推定片道伝送遅延の最大値/2 と最小値/2 を SMI 及びデータのオプションヘッダに加算する。ルーターは、SMI とデータのオプションヘッダを参照することで、データを下流のノードに転送する際のリンクの最大及び最小許容遅延の推測を行う。

データプレーンでは、PIT に基づきビデオデータの中継が行われる。パケット欠損時にデータ修復を行うための冗長符号化データは、符号化グループの総データ転送数がソースブロックサイズ (k) 未満の場合のみ転送される。RGI を受信した際には、ビデオデータがキャッシュされている場合はそのデータを送信し、キャッシュに存在しない場合は同じ符号化グループの異なる k 個以上のデータがキャッシュされているか否かを調べ、存在する場合にはそれらから新しく生成した符号化データを送信する。存在しない場合は、上

流へ Interest (RGI) を送信する。

ところで、各ルーターが推定するリンク許容遅延は、各受信者とその転送経路に依存するため一意ではない。したがって、最大/最小リンク許容遅延及びパケット損失率を考慮し、データ転送成功確率の平均値を利用して冗長符号化パケットの転送量を決定する。具体的には、各ルーターはデータ転送成功率の閾値を満たす再送処理を考慮したうえで、最小冗長符号化パケットの転送量を決定する。

各ルーターは、各リンクの帯域幅と帯域割当てポリシーに応じて、受信すべき通信経路とビデオ品質を決定する。本手法では、最も空き容量のある経路を優先的に利用し、コンテンツフローごとに帯域が公平に共有されるように割り当てる。コンテンツフローの重要度が高いデータ (例えば、SVC のベースレイヤ) を優先的に要求し、もし任意のフローが利用している経路の利用可能帯域を超える場合は、貪欲法 (greedy algorithm) にて最も空き容量のある経路を利用し、可能な限り高いレイヤのデータを受信して QoE の向上を図る。

以下、SMI と許容遅延を考慮したストリームデータ転送のシミュレーション結果について述べる。シミュレーションでは、図3に示すような米国学術機関である Abilene のネットワークトポロジを用い、既存研究のひとつであるマルチパス輻輳制御技術提案 [6] と提案手法の比較を行い、QoE モデル [7] の評価を行った。各リンクの帯域幅/キュー長/伝送遅延は、

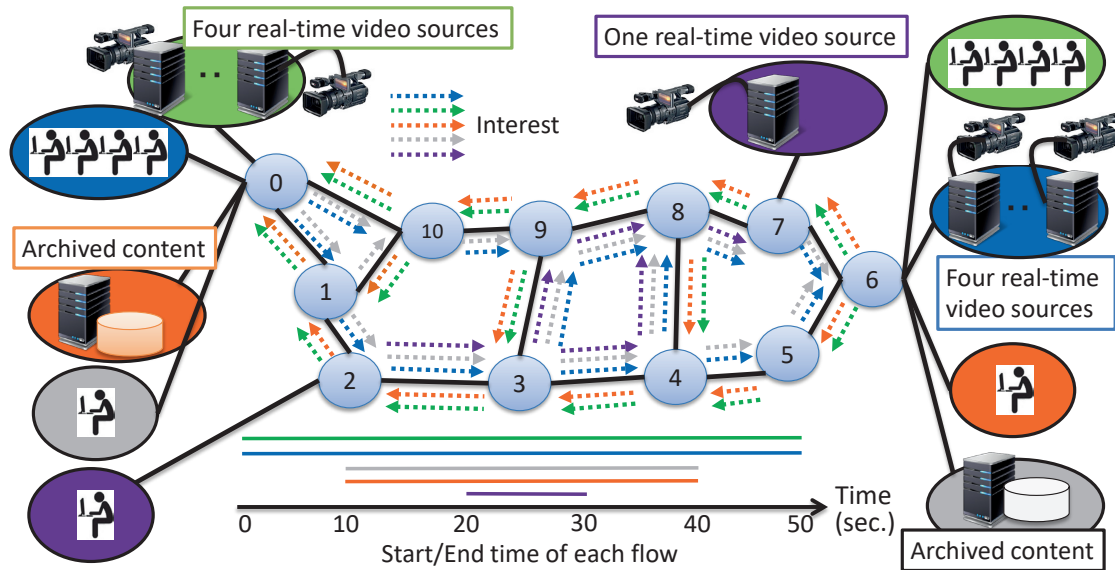


図3 評価ネットワークトポロジ (Abilene トポロジ)

それぞれ 100 Mbps / 100 パケット / 10 ms である。データ転送成功率の閾値を 0.995、ソースブロックサイズ (k) を 5 に設定した。Interest とデータサイズは、それぞれ 120、1024 バイトに、ビデオアプリケーションの許容遅延は 150 msec に設定した。各ビデオの最大転送レートは 35 Mbps であり、SVC のベースレイヤ / レイヤ 2 / レイヤ 3 のそれぞれの転送レートは 20 Mbps / 10 Mbps / 5 Mbps である。

図4は、提案方式と比較方式の正規化された QoE 値を示している。提案手法は、比較手法と比較して高い QoE 値を達成している。これは、マルチパス活用によりネットワークリソースを有効活用しつつ、ビデオ品質適応とデータ損失リカバリが機能しているためである。一方、比較手法でもマルチパスを活用しているが、データ損失発生に反応して受信映像品質を下げるため、高い QoE を達成することが困難になっている。また、図5で示すように、提案手法は SMI を活用することにより高レート Interest を抑制するため、Interest トラフィックを比較手法と比較して劇的に抑えられることが証明された。

以上、我々が提案したネットワーク内キャッシュや柔軟なマルチパス転送、ホップバイホップ転送といった CCN 通信の利点及びネットワーク内符号化機能を活用したリアルタイムストリーミング用トランスポート技術を紹介した。提案手法は、(1) SMI により高レート Interest と重複データ転送を防ぎ、(2) リンクの許容遅延を考慮し、必要に応じて最小限の冗長符号化データを付与することで高い転送効率達成を図るとともに、(3) 各リンクのネットワーク帯域を有効活用し、各フローに公正に帯域を割り振ることで効率的に高い映像品質データを受信者側へ転送することで QoE 向

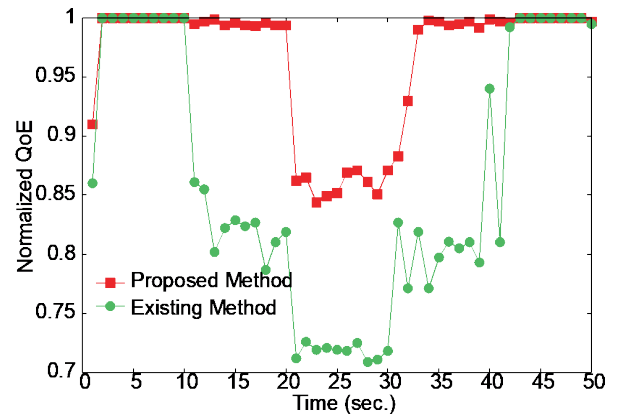


図4 QoE 評価結果

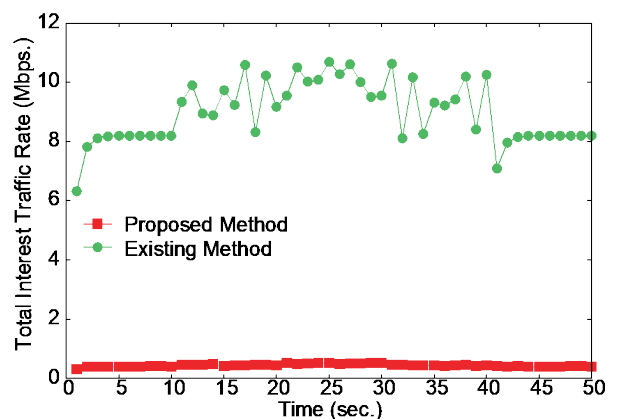


図5 リンク (9, 8) における Interest トラフィックのレート

上を図る。評価では、既存手法と比較して、SMI による無駄なトラフィックを抑制すると同時に、高い QoE (最大 25% 向上) を達成できることを確認した。今後の課題として、文献 [8] のアイデアを応用し、受信者モビリティに対応すべくプロトコルの拡張を行う。

5 Cefore

CCN は IP 通信が持つ問題点を根本から修正するため、ネットワークを白紙から設計しようという「クリーンシート」な試みから生まれた技術であり、それゆえに基礎研究としてのアルゴリズム設計やプロトコル設計だけでなく、それを評価するための実装・実証も 1 から行わなければならなかった。このため、CCN 研究に追従する形でいくつかの CCN の参照実装が開発・公開された。ここでは、代表的な参照実装として CCNx [9]、NFD (Named-Data Networking Forwarding Daemon) [10]、CICN (Community ICN) [11] を説明する (詳細は各ホームページ及び文献 [12] を参照のこと)。

世界で最初に公開された CCN の参照実装は、米国 PARC Inc. が 2009 年に公開した CCNx [9] である。当時はバージョン 0 系列として概念実証を目的として公開されたため、パフォーマンスや通信の効率性などは全く考慮されていなかったが、CCN が実際にインターネット上でも動くということを実証したという点で高く評価された。2014 年には標準化団体の IETF の姉妹組織である IRTF にて、CCNx-1.0 のパケットフォーマット [13] が発表され、同時に、これに沿った実装として CCNx-1.0 が開発された。しかし、CCNx バージョン 0 系列と 1 系列にはメッセージの互換性がなく、また、CCNx-1.0 が開発当初はオープンソースでなかったため、ユーザーにとっては利便性に乏しく、最終的には、CCNx の開発は中断されることになる (しかしその後、コードとライセンスは以下で説明する CICN に移管される)。

次に公開された参照実装は、2014 年に米国 Named-Data Networking (NDN) プロジェクトが開発した NDN Platform [10] である。NDN Platform は NDN Forwarding Daemon (NFD) と呼ばれるソフトウェアルーターと、通信用の各種アプリケーションやライブラリを含むソフトウェア基盤である。NDN は CCN 同様、ICN の一実装であり、CCN と概念も通信方式もほぼ同等であるが、NDN と CCN ではパケットフォーマットが異なるため、これらに相互互換性はない。

2017 年には、Cisco Systems, Inc. が PARC の保有していた CCN 関連の知的所有権を買収し、上記の CCNx-1.0 実装は FD.IO というプロジェクト内の Community ICN (CICN) [11] と呼ばれる実装に併合された。CICN は 2 つの実装形態で開発が継続されており、1 つは FD.IO コミュニティーが公開する Vector Packet Processing (VPP) フレームワークを利用したプラグインとして、もう 1 つは VPP に非依存のソケッ

トベースのパケットフォワーダーとして公開されている。

CCN は新しいネットワークアーキテクチャとして様々な可能性を秘めており、技術の適用先や導入先も多岐にわたる。このため、IoT などのセンサーネットワーク環境への適用においては、センサーノード上で稼働するための軽量かつ省電力な実装が求められ、バックボーンなどの基幹ネットワークへの適用においては、非常に高速なデータ転送を行うルーター若しくは高速キャッシュ処理が可能なルーター実装が求められる。しかし既存の参照実装は、高速化を目指すことでハードウェア要件が高くなる、あるいは動作環境に制約を与えてしまうなどの問題や、逆に軽量化を求め過ぎて通信性能を向上させられないなどの問題があった。また、特定の用途に特化させないために汎用性を求めた結果、実装が複雑になり、機能拡張が困難になり、秒進分歩の技術を開発・評価する目的に適さないなどの問題も見られた。

急速な発展を続ける CCN 研究に対し、我々はその研究促進と社会展開を促す参照実装の重要性を認識し、さらに上記で述べたような適用範囲に対する依存性を軽減するため、軽量かつ拡張性の高い CCN 通信用ソフトウェア・プラットフォーム、「Cefore」[1]を開発し、2017 年にオープンソースとして一般公開した。ソースコードは改変も自由に行え、商用においても利用可能なライセンスとなっている。Cefore の特徴は基本機能と拡張機能のソフトウェア分離にある。基本機能である軽量なデータ転送(フォワーディング)デーモン cefnetd と、ネットワーク内キャッシュをつかさどる Content Store (CS) デーモン csmgrd、そして各種ツール群で構成されている。

cefnetd は CCNx-1.0 メッセージを転送するために必要最低限の機能のみを有する。具体的には、コンテンツ所有者 (publisher と呼ぶ) 若しくはキャッシュルーターへの通信経路表に相当する「Forwarding Information Base (FIB)」と、コンテンツ名と下流のデータ転送先インターフェースの対応表である「Pending Interest Table (PIT)」の 2 種類を持ち、CCNx-1.0 メッセージパケットを転送する。cefnetd は Linux 系 OS や macOS に加え、Android の上で稼働する。また、センサーノードとして用いられる Raspberry Pi のように計算資源の乏しい環境でも動作する軽量性も併せ持つ。cefnetd は汎用性が高く、機能拡張性にも優れており、通常は、ユーザーは cefnetd や csmgrd 本体のプログラムを改変することなく、独自に開発したい機能やアルゴリズムは cefnetd あるいは csmgrd のプラグインライブラリーとして、若しくはこれらと連携する外部(デーモン)

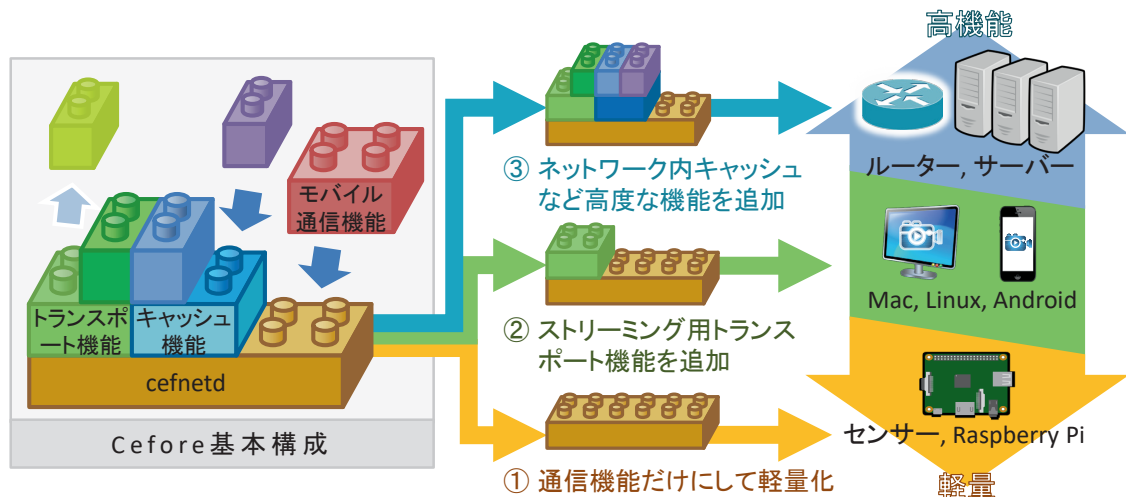


図6 Cefore ソフトウェアコンポーネントのイメージ図及び機能拡張

プロセスとして実装する。つまり、アプリケーション独自のトランスポート機能やモビリティ（ハンドオーバー）機能などをプラグインとして実装し、cefnetd 本体に組み込むことが可能となっている（図6 参照）。外部デーモンプロセスとの連携も容易であり、例えば、処理負荷が大きいCSによるキャッシュ機能は、外部デーモンプロセスである csmgrd と連携して稼働する。csmgrd も cefnetd 同様、キャッシュアルゴリズムをプラグインとして開発可能である。Least-Recently Used (LRU) や Least-Frequently Used (LFU)、CUSH [14] などのキャッシュ置換アルゴリズムは Cefore の標準パッケージに含まれており、誰でも利用可能である。ほかにも、cefnetd は CICN と同じ CCNx-1.0 メッセージフォーマットを用いるため、相互に通信可能であるが、NFD とも通信できるようにフォーマットを変換する NDN プラグインが Cefore 標準パッケージに含まれている。

Cefore にはいくつかの便利なネットワークツールも含まれる。例えば、コンテンツ名に基づいてデータのアップロードやダウンロードが可能なツールとして cefputfile と cefgetfile が提供されている。また、cefnetd 本体が持つ FIB やキャッシュ状態を調べる cefstatus や、キャッシュノードへの疎通性を確認する cefping などのツール、コンテンツ取得の通信遅延や取得経路などのネットワーク全体の情報を観測可能なユーティリティである cefinfo (IRTF では ccninfo として提案) [15][16] などが提供されている。さらに、アプリケーション開発を容易にするために、Python での開発を可能とする cefpyco パッケージも公開している。

CCN 通信の研究には参照実装が重要であるとともに、その実装を用いたプロトコルやアプリケーションの実験・検証を行うための評価環境も必要となる。新

しいネットワークアーキテクチャをインターネットへ適用した場合の効果・影響などを検証する最も代表的な評価環境として、広域テストベッドが挙げられる。CCN 実装などの検証を行うテストベッドとしては、NDN testbed [10] や CUTEi [17] などが開発され利用されている。テストベッドでの実験検証は、最も現実に近い形で評価が得られるといったメリットがあるが、構築・設定のコストが高い、ノード数やトポロジーの柔軟性に乏しい、多様な要因から挙動が不安定になりやすい、実験結果の再現性を保証するのが難しいなどの問題がある。

テストベッドのような実ネットワークを用いた検証環境のほかに、シミュレータやエミュレータを使用する方法がある。シミュレータは数学的なモデルによって実際のネットワークの振る舞いを計算する手法であり、再現性が高く、大規模なネットワーク検証にも用いられる。NDN では ndnSIM [10] というシミュレータが開発されている。しかし再現性が高い一方、シミュレータは通信モデルの設計において現実的な挙動を捨象しており、特にモバイル環境における無線通信検証などは、その評価結果に関し、現実性が低くなるなどの欠点がある。

エミュレータは仮想化技術によって単一の計算機上にノードを生成する手法であり、大規模なネットワーク構築が容易でありながら、実機検証と同等の現実性を持つ。各仮想ノードは論理的に分離されており、それぞれが固有のプロセスグループ、計算資源、そしてネットワーク空間を有する。仮想ノードを稼働させる計算機の資源に余裕がある限り、任意の数の仮想ノードを作成できるため、大規模なネットワークを柔軟に構築可能である。CCN エミュレータとしては Mini-CCNx [18]、Mini-NDN [10]、vICN [11] が開発されており、それぞれ CCNx、NDN、CICN の各ソフトウェ

6 ネットワークの効率的な資源配分を目指す研究開発

アルーター用の仮想ネットワークを構築する機能を提供している。しかし仮想ノードが構成するネットワークトポロジは計算されたリンク遅延などを含んだ仮想的なものであり、シミュレータ同様、モバイル通信の検証などに精度を欠く。

我々は Cefore を用いた現実的かつ大規模な CCN 通信環境評価のため、CCN エミュレータ「Cefore-Emu」を開発し公開した [1][19]。Cefore-Emu は、(1) Linux コンテナベースのネットワークエミュレータ方式により軽量に動作、(2) ネットワーク設定を容易にする設定ファイル及びその自動生成ツールの提供、(3) 現実の物理ネットワークとエミュレータ内のネットワークを接続・融合したハイブリッド検証環境を実現などの特徴を持つ。以下では、それぞれの特徴について詳説する。

Cefore-Emu は Mininet-HiFi [20][21] というコンテナベースのネットワークエミュレータに基づいて開発されており、その軽量性、設定容易性、現実性を引き継いでいる。エミュレータ方式には主に仮想マシン方式とコンテナ方式が存在する。仮想マシン方式はそれぞれの仮想マシンで独立した OS を管理し OS レベルで挙動を再現する方式であり、コンテナ方式はハードウェアに導入された OS (ホスト OS と呼ぶ) と同一の OS 上にコンテナと呼ばれる仮想ノードを生成する。コンテナ間ではプロセスグループやネットワーク空間が論理的に分断されているため、各コンテナを独立した計算機として扱える。コンテナはホスト OS の計算資源やソフトウェアを直接使用できるため、OS ごとに仮想化する仮想マシン方式より軽量に稼働させることが可能である。しかし、コンテナがホスト OS の計算資源を共有するため、コンテナ間での計算資源の干

渉が起り得る。そこで、Mininet-HiFi はコンテナごとの計算資源 (CPU・メモリ) も独立させる機能を有しており、これによってコンテナ間の干渉を防ぎ、実験結果の現実への忠実性を保証している [21]。

次に、Cefore-Emu の設定方法及び動作に関して説明する。図 7 のように、ユーザーは最初に設定ファイル cefemu.conf を作成する。これはノードとリンク及びノードの種類 (nodetype) から構成されている。本例では 3 ノード (h1、h2、h3) がキャッシュルーター s1 を介して接続している (図 7 右トポロジ参照)。nodetype は Cefore の設定ファイルを指定するための項目であり、キャッシュの有無や自作プラグインのパラメータを設定することができる。本例では s1 に cachemode タイプを指定し、ルーターでのみキャッシュを行うように設定している。設定ファイルを作成した後は、cefemu コマンドを実行することで仮想ネットワークを起動できる。Cefore-Emu はノードへの接続用インターフェースとして Command Line Interface (CLI) を提供しており、任意の仮想ノード上でのコマンド実行や、Cefore-Emu の提供するネットワーク操作機能の利用を対話的に行うことができる。

最後に、ハイブリッド実験環境について説明する。既述のとおり、シミュレータやエミュレータでは、ネットワーク状態、特に無線通信環境などにおける通信遅延やデータ欠損などの通信品質を評価する実験において、評価の精度に対する信頼性が問題になることが多い。そこで Cefore-Emu では、無線インターフェースを含む実際の物理インターフェースをエミュレータに接続し、エミュレータが構成する仮想的なネットワークトポロジと物理ネットワークを融合して実験評価できるハイブリッド検証環境を実現した。図 8 の例では、

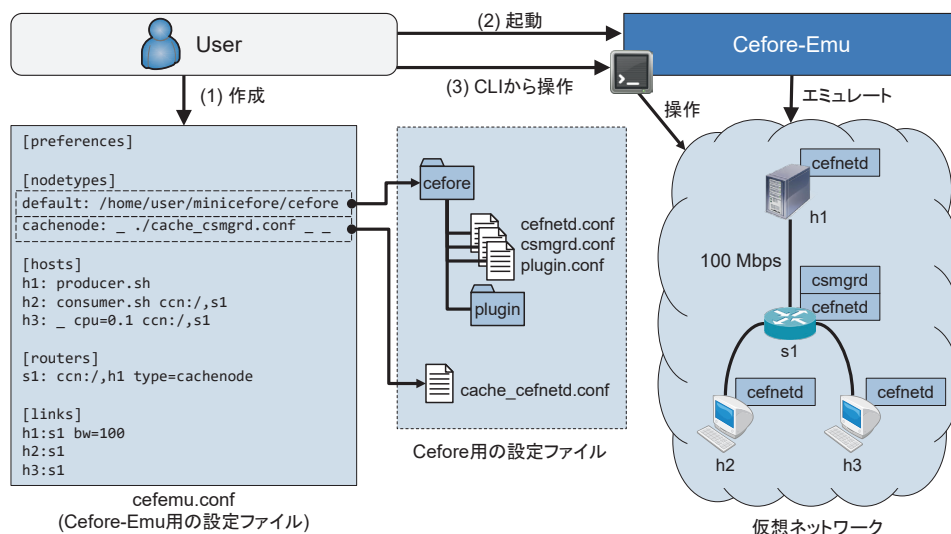


図 7 Cefore-Emu の簡単な設定例とエミュレートされた仮想ネットワーク

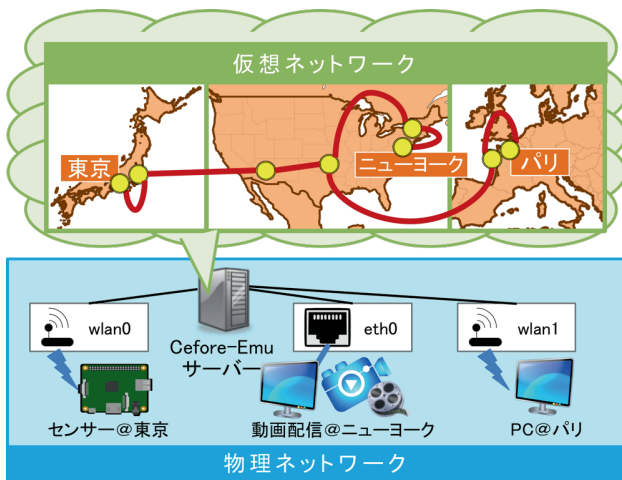


図8 Cefore-Emuによるハイブリッド実験環境例

Cefore-Emu を稼働させるサーバーが、Ethernet 経由で PC と接続し、また2つの WiFi アクセスポイント (USB デバイス) とも接続している。この環境を用いることで、ニューヨークにある PC から動画配信を行い、仮想的なインターネットトポロジを経由し、東京とパリにある WiFi アクセスポイント経由で動画を受信している Cefore 通信のようなシナリオを再現することが可能となる。このようなハイブリッド環境は、cefemu.conf のノード設定に接続したい物理インターフェース名を記述するだけで構築できるため、大規模な模倣ネットワークと現実の無線通信を融合した実験が容易に行える。

6 今後の展望

ここまで、ICN/CCN の概要や特徴、NICT が取り組む ICN/CCN 研究や開発、参照実装などに関して説明してきた。ICN/CCN は次世代のネットワークアーキテクチャとして、IP 通信が抱える様々な課題を克服するために、世界各国で研究開発が進められている。近年では、その基礎研究に留まらず、ベンダーやキャリアなどによる応用研究や実装も進められており、益々その実現が期待されている。

今後の展望として、特に第5世代移動通信システム (5G) に関しては、5G のコア技術のひとつとして ICN/CCN 技術が含まれると考えられており、海外の大手ベンダーはそこにビジネスチャンスを見いだそうとしている。5G では物理ネットワークを「スライス」という論理的なネットワークに分割して運用することが可能となり、つまり IP とは独立した ICN/CCN スライスを定義し、IP サービスと並行して、特定の ICN/CCN サービスやアプリケーションを ICN/CCN スライス上で実現できる。5G の中心的なサービスとなる IoT や高速・大容量・超低遅延通信などは、正

に ICN/CCN が得意とするコア技術 (マルチキャスト、マルチパス、サーバーレス通信、移動体通信 (モビリティ)) を必要とするサービスであるため、より品質の高いサービス提供が実現できると考えられる。

キャリアインフラの視点では、既存の IP ネットワークを全て ICN/CCN に置き換えるという「ハードランディング」な考え方だけに固執するのではなく、部分的に、若しくはこれから作られていく新しいエッジコンピューティングやフォグコンピューティングに ICN/CCN 機能を融合させる方法も有効であろう。キャリアインフラ以外の技術展開に関しては、特定のネットワーク、例えば小規模なネットワークやエンタープライズネットワークなどで ICN/CCN が展開されていく方向も考えられる。また、インターネット上の特定のコンテンツ配信、例えばマルチキャストやマルチパスを利用した 4K/8K レベルの高品位かつ低遅延リアルタイム・ストリーミング配信や、時差通信機能を用いた耐災害ネットワークなど、アプリケーションレベル (オーバーレイ) で ICN/CCN を利用していくのも効果的である。

NICT では、新しいネットワークアーキテクチャとして非常に大きな可能性を持つ ICN/CCN に対し、今後も様々な研究課題を克服し、研究のレベルアップとオープンソースの成熟によりその優位性を実証していく。また ICN/CCN を更に発展させ、ネットワークを単なる「(データを流すだけの) 土管」から、より知性と機能を持つインフラストラクチャとして進化させ、未来の通信サービスを創造し、豊かな社会への貢献を目指す。

【参考文献】

- 1 "Cefore," Available at: <https://cefore.net/>.
- 2 P. Seeling and M. Reisslein, "I. want. pixels. (Entering the age of 4K)," IEEE Potentials, vol.33, no.6, pp.27-30, 2014.
- 3 "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update," 2014-2019 White Paper.
- 4 K. Matsuzono, H. Asaeda, and T. Tuletto, "Low Latency Low Loss Streaming using In-Network Coding and Caching," Proc. IEEE Infocom'17, Atlanta, United States, May 2017.
- 5 K. Matsuzono and H. Asaeda, "NRTS: content name-based real-time streaming," Proc. IEEE CCNC, Las Vegas, United States, Jan. 2016.
- 6 G. Carofiglio, M. Gallo, L. Muscarillo, M. Papalini, and S. Wang, "Optimal multipath congestion control and request forwarding in information-centric networks," Proc. IEEE ICNP, Oct. 2013.
- 7 X. Yin, V. Sekar, and B. Sinopoli, "Toward a principled framework to design dynamic adaptive streaming algorithms over HTTP," Proc. ACM Hotnets, Oct. 2014.
- 8 K. Matsuzono and H. Asaeda, "NMRTS: content name-based mobile real-time streaming," IEEE Communications Magazine, vol.54, no.8, Aug. 2016.
- 9 "CCNx," Available at: <https://github.com/ProjectCCNx/ccnx>.
- 10 "Named Data Networking," Available at: <http://named-data.net/>.
- 11 "CICN," Available at: <https://wiki.fd.io/view/Cicn>.
- 12 朝枝仁, 松園和久, "情報指向ネットワーク技術におけるプロトタイプ実装と評価手法," コンピュータソフトウェア, vol.33, no.3, pp.3-15, 日本ソフトウェア科学会, 2016年8月.

6 ネットワークの効率的な資源配分を目指す研究開発

- 13 M. Mosko, I. Solis, and C. A. Wood, "CCNx messages in TLV format," IRTF ICNrg Internet-Draft (work in progress), July 2018, Available at: <https://tools.ietf.org/html/draft-irtf-icnrg-ccnxmessages-08>.
- 14 A. Ooka, E. Suyong, S. Ata, and M. Murata, "Scalable cache component in ICN adaptable to various network traffic access patterns," IEICE Trans. Commun., vol.E101-B, no.01, Jan. 2018.
- 15 H. Asaeda, K. Matsuzono, and T. Turletti, "Contrace: a tool for measuring and tracing content-centric networks," IEEE Communications Magazine, vol.53, no.3, pp.182-188, March 2015.
- 16 H. Asaeda and X. Shao, "CCNinfo: Discovering Content and Network Information in Content-Centric Networks," IRTF ICNrg Internet-Draft (work in progress), June 2018, Available at: <https://tools.ietf.org/html/draft-asaeda-icnrg-ccninfo-01>.
- 17 H. Asaeda, R. Li, and N. Choi, "Container-based Unified Testbed for Information-Centric Networking," IEEE Network, vol.28, no.6, pp.60-66, Nov. 2014.
- 18 C. Cabral, C. E. Rothenberg, and M. Magalhaes, "Mini-CCNx: Fast Prototyping for Named Data Networking," Proc. ACM SIGCOMM ICN Workshop, Aug. 2013.
- 19 大岡睦, 朝枝仁, "Mini-Cefore: Container-Based Large-Scale Cefore Emulator," 電子情報通信学会情報指向ネットワーク技術特別研究専門委員会 (ICN研究会), 2017年12月, 広島.
- 20 "Mininet," Available at: <http://mininet.org/>.
- 21 N. Handigol, B. Heller, V. Jeyakumar, B. Lantz, and N. McKeown, "Reproducible network experiments using container-based emulation," Proc. ACM CoNEXT '12, Dec. 2012.



朝枝 仁 (あさえだ ひとし)

ネットワークシステム研究所
ネットワーク基盤研究室
上席研究員
博士(政策・メディア)
情報指向ネットワーク、経路制御、通信プロ
トコル



松園和久 (まつその かずひさ)

ネットワークシステム研究所
ネットワーク基盤研究室
研究員
博士(政策・メディア)
情報指向ネットワーク、ネットワーク符号化



大岡 睦 (おおおか あつし)

ネットワークシステム研究所
ネットワーク基盤研究室
博士(工学)
研究員
情報指向ネットワーク、ルータ、キャッシュ
技術