

低遅延・高信頼性通信を実現するネットワーク技術の紹介

加藤 豊行 Toyoyuki Kato

[要 旨]

AI/ML(Artificial Intelligence/Machine Learning)の進化・普及、全産業分野における DX(Digital Transformation)の推進、あるいは IoT(Internet of Things)の普及拡大など、通信トラフィックの増加は続いている。これに対応するべく、テラビットクラスの光通信や B5G/6G・WiFi-6/7 によるギガビットクラスのワイヤレス通信など、通信システムの高速度・大容量化が進められている。その実力を十分に活かすために、伝送制御を担う通信プロトコルについても改良・進化が進められている。本稿では、特に通信ネットワークの低遅延化やリアルタイム対応にフォーカスし、通信プロトコルの進化・改良に関するさまざまな取り組みを紹介する。

1 まえがき

現代社会において、通信インフラストラクチャは日常生活やビジネス活動の基盤となっている。インターネットやモバイル機器の普及とともに、通信技術は急速に進化し、その重要性はますます高まっている^{1)・2)}。通信アプリケーションの多様化によってリアルタイム性が求められる高精度・高品質なデータ通信の需要が高まり、単に高速・大容量だけでなく、時間的確度や決定論的(Deterministic)な特性が強く求められるようになってきた。本稿では、これらの要求に対応するための通信ネットワーク技術、特に低遅延かつ高信頼性の通信を実現するための最新技術トピックについて紹介する。具体的には、次世代通信プロトコルの改良点や新たな通信アーキテクチャ、そしてそれらが実際の通信環境でどのように適用されている

かについて詳述する。これにより、現在および将来の通信ネットワークが直面する課題とその解決策について理解を深めることを目指す。

2 通信ネットワーク利用環境

通信ネットワーク利用環境の概観を図 1 に示す。以下に主要な利用環境を説明する。

社会・生活環境: 光ファイバによる通信ネットワークの高速・大容量化は、スマートシティの実現を支え、リモートワークやオンライン教育を一般化し、柔軟なライフスタイルを実現している。

DX: デジタル技術と通信ネットワークの高度化を背景として、産業全体でデジタル技術を活用したビジネスモデルや業務プロセスの変革が進められ、その基盤となるコンピューティング・ネットワークのさらなる高度化が求められている。

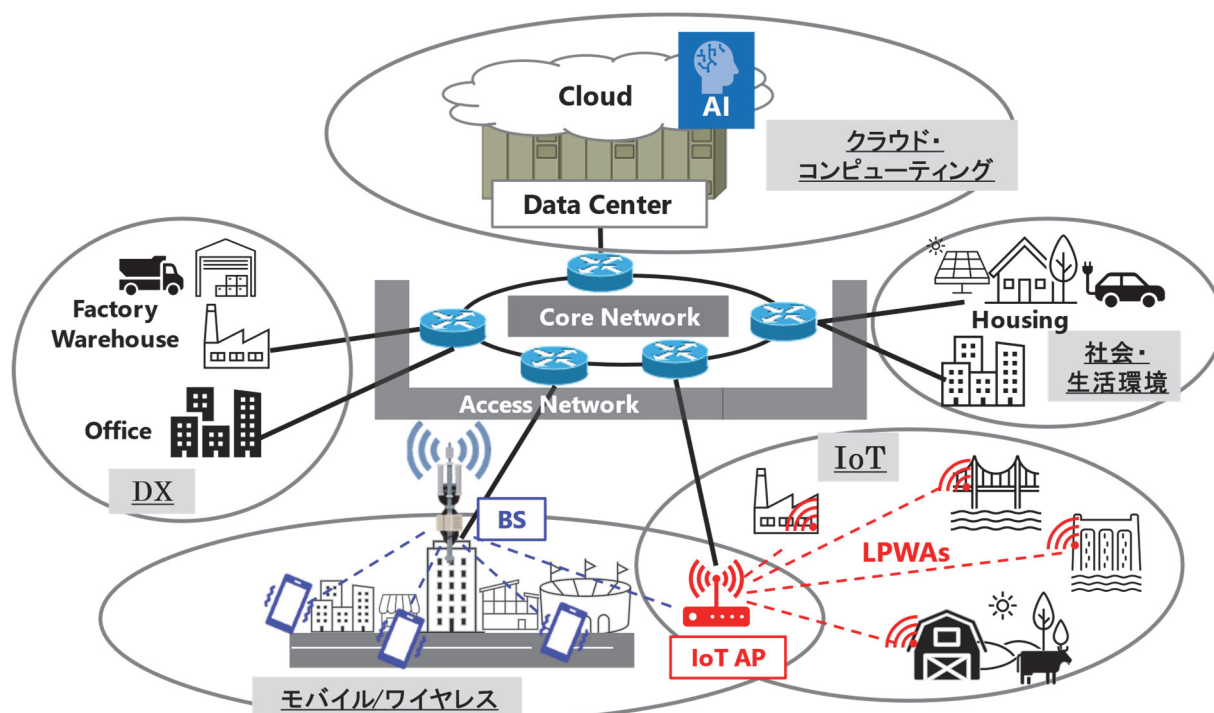


図1 通信ネットワーク利用環境

クラウド・コンピューティング: ICT サービスのクラウド化により、ネットワークの速度や安定性が求められている。また、リモートワークの普及に伴い、さらに高速かつ安定した接続が求められている。

モバイル/ワイヤレス: モバイルデバイスの多機能・高性能化とサービス・アプリケーションの高度化・多様化が進み、通信性能やコスト・パフォーマンスに対するユーザ要求も増大。さらなる高速・大容量・低遅延を目指した 6G の研究が進められている。

IoT: スマートホームや産業用センサー、医療機器など多様な分野で IoT の利用が拡大、アクセス回線の高速化・広帯域化・低遅延化により IoT デバイスの接続性をさらに向上させ、新しいサービスやアプリケーションを創出している。³⁾

このように多様化・高度化する通信ネットワークの利用環境では、共通する要件として低遅延・リアルタイム性が求められる。次章以降、その実現に向けた技術的な取り組みを紹介する。

3 低遅延・高信頼性通信の実現

あらゆる利用分野において、通信ネットワークのさらなる高速化、広帯域化、および低遅延化が共通課題となる。これに応じるように光通信の伝送速度はテラビットに到達し、モバイル・ワイヤレスでは 6G や Wi-Fi6/7 によって数十ギガビットクラスの通信を実現している。その結果、通信プロトコルなどの伝送制御方式では、これまで内在していた伝送遅延が顕在化してきている。

本章では、この伝送遅延の問題と原因・要因を概説し、また主な対応策を紹介する。

3.1 現状課題

通信ネットワーク、有線・無線ともに伝送路の高速化・広帯域化が長年にわたって進められてきた。一方で、伝送制御を担う通信プロトコルにおいては、通信リソースを効率的に利用するためのさまざまな方式・手法が導入され、進化が積み重ねられてきた。しかし、伝送路の高速化・広帯域化が進んだ結果、通信プロトコルの処理や滞留によって生じるオーバーヘッドは、通信全体の伝送時間に占める割合が大きくなっており、特にその主要因とみられる輻輳制御 (Congestion Control) やフロー制御 (Flow Control) の効率化が求められている。

例えば、現在の通信ネットワークを流れるトラフィックの大半は IP (IPv4 および IPv6) である。またその過半を TCP (Transport Control Protocol)⁴⁾ が占めている^{5)~7)}。TCP が設計された 1980 年代初頭は、その当時の低速回線で高効率かつ高信頼性のパッチ通信 (E メールやファイル転送など) を実現することを主目的とし

ていた。その後、通信回線の高速化、多様なネットワークアプリケーションの登場、リアルタイム性や低遅延の要求など、利用環境の変遷を経て、TCP の相対的な老朽化が進行してきている。

TCP は、当初の設計時から利用環境や用途が大きく変化し、現在の要求に適応しきれていないことが指摘されている。具体的には、以下のような課題が存在する。

➤ **伝送遅延の増加:** 高速回線の普及により、TCP の制御メカニズムが伝送遅延を引き起こす要因となっている。

➤ **リアルタイム性の要求:** オンラインゲームやビデオ会議など、リアルタイム性を重視するアプリケーションにおいて、TCP の遅延はパフォーマンスの低下を招いている。

➤ **輻輳制御の非効率性:** 帯域幅遅延積 (Bandwidth-Delay Product: BDP)⁸⁾ が大きな環境では、従来の TCP 輻輳制御アルゴリズムが最適なスループットを達成できない場合がある。

これらの課題に対応するため、TCP の改良や新たな伝送プロトコルの開発が求められている。次節以降、3.2 で TCP/IP プロトコルアーキテクチャについて、3.3 では輻輳制御アルゴリズムの進化と改良、3.4 では伝送プロトコルの改良・改善について詳述する。

3.2 TCP/IP プロトコルアーキテクチャ

TCP/IP プロトコルアーキテクチャの概観を図 2 に示す。最上位層である Application layer は、かつてはテキストベースの電子メールやファイル転送が主体であったが、現在はビデオ配信、オンラインミーティング、リアルタイムゲームから工作機械や自動車にいたるまで多岐にわたっている。また、最下位層である Physical layer の通信媒体も有線の電気通信から光通信や無線通信へと拡大してゆき、また伝送速度も各段に高速となっている。

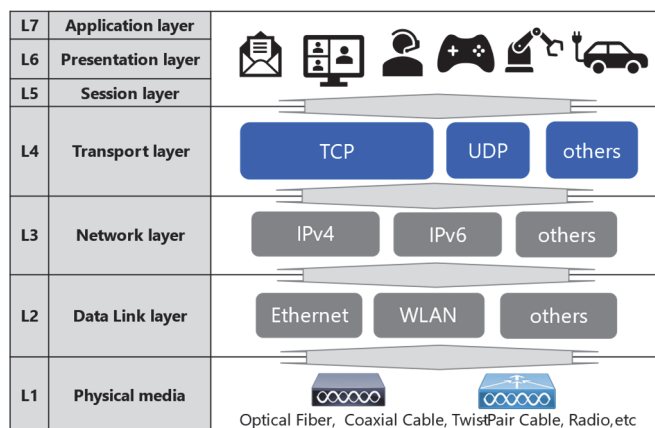


図 2 TCP/IP プロトコルアーキテクチャ

これらの通信制御の中核を担うトランスポート層 (Transport layer) のプロトコルには、主に信頼性を重視した TCP と効率を重視した UDP (User Datagram Protocol)⁹⁾ のいずれかが使用される。

TCP は、データの分割、再構築、エラーチェック、再送機能により信頼性の高いデータ転送を提供するが、処理リソースの消費が大きく、またそのオーバーヘッドによる遅延も発生する。一方で、UDP は、接続の確立やエラーチェックを行わないため、遅延が少なくリソース消費も少ないが、データの損失や順序の保証がないため信頼性は低い。多くのネットワーク環境においては、実際の通信トラフィック量は信頼性が高い TCP の比率が大きい。

TCP では、スライディングウィンドウによる送信データ量の制御と確認応答やシーケンス番号などの情報交換による到達確認を活用し、フロー制御、順序制御、輻輳制御を行うことで通信の信頼性を高めている。さらに AQM(Active Queue Management)の導入やその後の AQM アルゴリズムの改良・進化によって、時代の要求に適応しながら長年にわたって利用されてきた。

3.2.1 AQM(Active Queue Management)

AQM は、ルータやスイッチでキューの長さを動的に調整し、輻輳が発生する前にパケットをドロップまたはマークする。これにより、TCP は輻輳の兆候を早期に検出し、送信レートを適切に調整できる。結果として、ネットワークの効率とパフォーマンスが向上し、安定した通信が可能になる。

最初に AQM の前提となる QM(Queue Management)の構成について凡例を図 3 に示す。短時間の間に複数のデータパケットが到着する場合、十分なバッファ領域が無いとパケットの廃棄が発生してしまう(図 3 a.)。バッファ容量を増やすことでこれを解消できる(図 3 b.)。さらに、データフロー別や優先度別で個別にバッファを設けることで、さらに高品質・高精度な送受信・伝送処理の実現が可能となる(図 3 c.)。

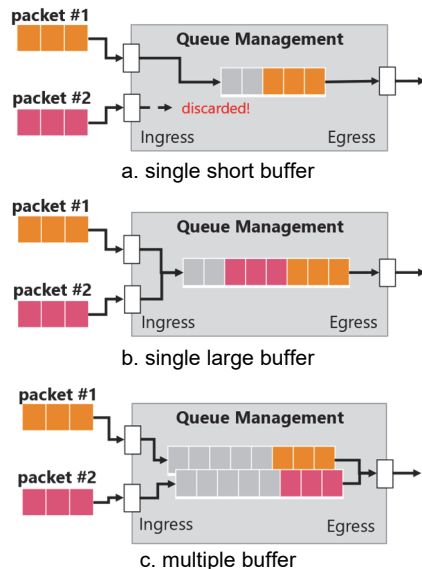


図 3 QM の構成例

初期の AQM アルゴリズムの一つである RED(Random Early Detection)¹⁰⁾では、データ量の増加によりランダムにパケットドロップが発生させ、それに応じて働く TCP のフロー制御によってデータ流量を抑制し輻輳を回避する少々ラフな方式であった。その後、ネットワーク利用の拡大や多様化に応じて、さまざまな AQM アルゴリズムが開発・利用されている。主な AQM アルゴリズムの一覧を表 1 に示す。

表 1 主な AQM アルゴリズム

名称 [発表年]	基本原理	特長
RED [1993]	キューの平均長がしきい値を超えたらパケットをランダムにドロップまたはマーク	特長: キューの長さに応じてパケットドロップの確率が増加 利点: 輻輳の早期検出と防止 欠点: パラメータの調整が難しい
SFQ [1999]	ハッシュ関数を使用してフローをランダムに分割	特長: 計算コストを低減 利点: フロー間の公平性を保つ
HTB [2001]	階層的トークンパケットを使用	特長: 異なる優先度のトラフィックを効率的に制御 利点: 帯域の効率的な管理
CoDel [2012] FQ-CoDel [2018]	キューの遅延がしきい値を超えたらパケットをドロップ	特長: 遅延に基づいてパケットをドロップ 利点: バッファブローットの防止 欠点: 特定条件下でパフォーマンス不安定化の可能性あり
PIE [2013] FQ-PIE [2024]	キュー遅延の目標値維持のため、パケットドロップ率を動的に調整	特長: 遅延を一定に保つ 利点: ネットワークのパフォーマンス向上 欠点: 実装が複雑

3.2.2 Bufferbloat 問題

AQM の導入とその後の改良により、フロー制御・輻輳制御は通信の高速化・大容量化に対応し続けてきた(表 1)。さらに、処理能力向上とバッファ容量増加などのハードウェア性能の向上によって、高度な優先度制御や帯域制御も可能としてきた。

一方で、ネットワーク機器の過剰なバッファリングが遅延の増大を招き、特にリアルタイム通信への悪影響が顕著になってきた。このような現象は Bufferbloat と呼ばれている。ルータやスイッチなど中継装置の過剰なバッファリングによってキューが長くなり、パケットの転送遅延を増加させてしまうことが原因である。特に速いリンクから遅いリンクへのデータ転送のときに顕著となる。

3.2.3 AQM アルゴリズムの進化・改良

Bufferbloat の軽減・解消を目的として、さらなる AQM アルゴリズムの進化・改良が行われている。

FQ-PIE¹¹⁾は、遅延の変化率を監視してドロップ確率を動的に調

整し、輻輳が発生する前にパケットをドロップする。これにより、高いスループットを維持しながら低遅延を実現する。

FQ-CoDel¹²⁾は、パケットがキューに滞留する時間(キュー遅延)を基に輻輳を検出し、一定の遅延(ターゲット遅延)を超えた場合にパケットをドロップする。キューの長さではなく、実際の遅延に基づいて輻輳を管理する。遅延を抑えながらパケット損失を最小限に抑えることができ、特にリアルタイムアプリケーションに適している。

3.3 輻輳制御アルゴリズムの進化・改良

AQM は、ネットワーク内のキューの長さや状態を監視し、適切に管理することで輻輳を防ぎ、ネットワークのパフォーマンスを最適化するために開発・導入されてきた。その後もアプリケーションやサービスの高度化・多様化に対応するため、さらに洗練された輻輳制御アルゴリズムの開発・導入が進んでいる。本節では、そのような先進的な輻輳制御アルゴリズムをいくつか紹介する。

3.3.1 CUBIC

CUBIC¹³⁾は、線形ではなく三次関数に基づくウィンドウサイズの増加によってスケラビリティと安定性を向上させ、いわゆるロングファットパイプ問題^{14), 15)}にも対応する高速長距離ネットワーク向けの輻輳制御アルゴリズムである。主な特長は次のとおりである。

- ・ **ウィンドウサイズの調整**: 従来の TCP の線形ウィンドウ増加関数の代わりに立方関数を用いることで、BDP が大きい環境でも効率的に動作する。

輻輳検出時にはウィンドウサイズを急激に減少することによって、ネットワークの安定性を保ち、輻輳の収束後はウィンドウサイズを急速に増加させて、伝送帯域の回復をよりスムーズに行う。

- ・ **TCP フレンドリー**: 短い RTT(Round Trip Time)や狭帯域幅のネットワークでは、従来の TCP と同様の動作をする。
- ・ **公平性とスケラビリティ**: 異なる RTT を持つフロー間での線形帯域幅共有を実現し、ネットワークの公平性を維持しつつ、長距離ネットワークでも高いスループットを実現する。特に「ロングファットネットワーク」において効果的である。

CUBIC は、Linux や Windows などのオペレーティングシステムの TCP 輻輳制御アルゴリズムとして広く採用されており、さまざまな通信環境でのデータ転送の効率を向上させている。

3.3.2 PCC(Performance-oriented Congestion Control)

PCC¹⁷⁾は、ネットワークのパフォーマンスを最適化するために設計された輻輳制御アルゴリズムである。従来の TCP アルゴリズムとは異なり、パケットレベルのイベントに基づく制御ではなく、実際のパフォーマンス結果に基づいた制御を行う。これにより、ネットワーク

の状態に応じて最適な送信レートを動的に調整し、高いスループットと低遅延を実現する。主な特長は次のとおりである。

- ・ **オンライン学習**: オンライン学習に基づいて送信レートを動的に調整する。これにより、ネットワークの状態に応じて最適なパフォーマンスを維持する。
- ・ **高スループットと低遅延**: パケット損失や遅延を最小限に抑えつつ、高いスループットを実現する。
- ・ **適応性**: ネットワークの変動に迅速に適応し、安定した通信を提供する。

PCC は、送信側が送信レートを調整しながら、実際のパフォーマンス(スループット、遅延、パケット損失など)を観察し、これに基づいて、最適な送信レートを決定し、ネットワークの状態に応じて動的に調整する。特に次のようなシナリオで有効である。

➤ **高帯域遅延積ネットワーク**: 高速なデータ転送が求められる環境で、安定した高スループットを提供する。

➤ **リアルタイムアプリケーション**: 低遅延が求められるビデオストリーミングやオンラインゲームなどのアプリケーションで効果を発揮する。

PCC は、PCC Project¹⁶⁾で開発されており、Linux カーネルモードとユーザモードの実装、ベンチマークツール、あるいは強化学習用モデルなどが GitHub 上で公開されている。

3.3.3 BBR(Bottleneck Bandwidth and Round-trip propagation time)

BBR¹⁸⁾は、従来のパケットロスに基づく輻輳制御とは異なり、ネットワークの帯域幅とラウンドトリップ時間を基にデータ送信速度を調整する輻輳制御アルゴリズムである。これにより、高スループットと低レイテンシを実現し、パケットロスに過剰反応せず安定した通信を提供する。

この BBR は、Google が開発した輻輳制御アルゴリズムで、YouTube や Google Cloud Platform など導入されており、動画の再生やクラウドサービスのパフォーマンス向上に貢献している。また、モバイル環境やパケットロスが発生しやすいネットワークで特に効果を発揮する。IETF でも BBR の標準化が進行中で、次世代通信技術の基盤として重要な役割を果たしている。

3.3.4 L4S(Low Latency, Low Loss, Scalable Throughput)

進化・改良が続けられてきた AQM アルゴリズムに対するさらなる低遅延と高スループットを実現するための新たなアプローチとして、L4S^{19)~22)}が注目されている。L4S は、TCP/IP ベースの通信において、主に遅延を大幅に削減しスループットを向上させることを目的としており、データセンター向けの DCTCP(4.2 参照)をより広い

通信インフラに適合させた規格といえる。ECN(Explicit Congestion Notification)²³⁾を活用して、輻輳状態の検知・通知を最適化し、Bufferbloat を最小限に抑えることでビデオ通話やオンラインゲームなどのリアルタイムアプリケーションのパフォーマンスを向上させる。L4S の基本シーケンスを表 2 に示す。

表 2 L4S の基本シーケンス

ステップ		概要
1	パケット送信	送信側のデバイスがデータパケットを送信する。このとき、L4S 対応の輻輳制御アルゴリズムを適用する。
2	ECN マーキング	ネットワーク内の中継装置（ルーターやスイッチ）は、輻輳発生を検知したときにパケットに ECN マークを付ける。これにより、輻輳の発生を送信側に通知する。
3	パケット受信	受信側デバイスはパケット受信時に ECN マークが付いているかどうかを確認する。
4	フィードバック	受信側が ECN マークの情報を送信側にフィードバックする。これにより、送信側はネットワークの輻輳状態を把握できる。
5	輻輳制御	受信側デバイスからのフィードバックに基づいて、送信側デバイスは送信レートを調整する。これにより、輻輳を回避しつつ、低遅延と高スループットを維持する。

L4S の主な特長は以下のとおりである。

- **低遅延:** 輻輳の兆候を早期に検出し対応することで、パケットの待機時間を最小限に抑える。
- **低損失:** Bufferbloat の軽減により、パケット損失の発生を抑制し、通信の信頼性を向上させる。
- **スケーラブルなスループット:** ネットワークの負荷に応じて柔軟にスループットを調整できるため、大規模なネットワーク環境でも高効率なデータ転送が可能となる。
- **従来の輻輳制御との共存:** L4S の利点を活かしつつ、異なるネットワーク条件やアプリケーションに対して柔軟に対応することができる。

L4S の導入により、特にリアルタイムアプリケーションにおいて、ユーザ体験の向上が期待される。3GPP Release18 では、AR/VR など高速データレートと低遅延通信を必要とする「対話型メディアサービス」の処理効率化を図るため、拡張機能の一つとして L4S の導入が検討されている。²⁴⁾

3.4 伝送プロトコルの改良・改善

本節では、伝送プロトコルの改良・改善について、具体的な技術として QUIC (Quick UDP Internet Connections)²⁵⁾を紹介する。低遅延かつ高速な通信を実現するためには、既存プロトコルの改良や新プロトコルの導入も有効な手段である。その一つとして、この UDP を基盤とした QUIC が提案されまた利用が広がっている。

3.4.1 QUIC

QUIC は、従来の TCP の欠点を補完し、特に低遅延と高速化を目指している。主な特長は次のとおりである。

- **UDP ベース:** TCP のような接続確立のための複雑なハンドシェイクを排除して、接続確立時間を大幅に短縮した。
- **低遅延:** 接続確立時のラウンドトリップ回数を減らすことで、遅延を最小限に抑える。特に、0(ゼロ)-RTT 再接続をサポートしており、一度接続したことのあるサーバへの再接続が非常に高速である。
- **多重化:** 複数のストリームを一つの接続内で並行して処理することができる。これにより、パケットロスが発生しても他のストリームに影響を与えず、効率的なデータ転送が可能である。
- **セキュリティ:** TLS 1.3 をベースにした暗号化を標準でサポートしており、セキュアな通信を実現する。これにより、データの機密性と整合性が保たれる。
- **コネクションマイグレーション:** IP アドレスやポート番号が変わっても接続を維持することができる。これにより、モバイル環境でのネットワーク切り替え時にも接続が途切れない。

QUIC は、HTTP/3 のトランスポートプロトコルとして採用されており、ウェブページの読み込み速度向上とユーザ体験の改善に寄与している。WebRTC などのリアルタイム通信アプリケーションにも利用されており、低遅延での音声やビデオ通話が可能となっている。特にモバイル環境やリアルタイム性が求められるアプリケーションでの利用が期待されている。

4 データセンター向けソリューション

クラウド・コンピュータの普及・利用拡大が続く中、さらに生成 AI の登場によって、データセンター内の高速処理と処理間通信の広帯域・低遅延化の重要度が増している。データセンターでは、ビッグデータの解析、クラウドサービスの提供、リアルタイムアプリケーションの運用など、膨大な量のデータが常にやり取りされている。これらのデータを迅速に処理し、効率的に転送するためには、高速なネットワークインフラが不可欠である。特に、近年増加している AI モデルのトレーニングには、多数のサーバの連携処理と大量のデータ交換が必須であり、大量のデータを迅速に転送する能力が求められる。本節では、L2(Data Link layer)によって実現されるソリューションとして Lossless Ethernet、また L3(Network layer)以上で実現されるソリューションとして DCTCP(Data Center TCP)を紹介する。これらの技術は、データセンター内の通信効率と信頼性を向上させるために不可欠な要素となっている。

4.1 Lossless Ethernet

Lossless Ethernet は、データセンター内の機器間通信に Ethernet 技術を適用し、パケットロスを防止することを目的とした技術である。特にデータセンターや高性能コンピューティング(HPC)環境の機器間通信において InfiniBand²⁶⁾が広く利用されている。複数のプロセッサやストレージデバイス間の通信を効率化し、低遅延と高帯域幅が求められる環境での標準的なインターコネクト技術となっている。

表 3 に InfiniBand の主な仕様を示す。このように、InfiniBand は高速なデータ転送速度を提供し、大規模なデータセンターで求められる高い信頼性を確保する。しかし、施設の規模や使用箇所、用途、SLA(Service Level Agreement)条件によっては、よりコスト対性能の高いソリューションが求められる。このようなニーズに対応するため、相対的に低コストながら高性能の Ethernet 技術²⁷⁾をベースとした Lossless Ethernet が提案され、IEEE802.1Q²⁸⁾で関連規格が策定されている。Lossless 関連規格を表 4、InfiniBand との比較を表 5 にそれぞれ示す。

表 3 InfiniBand の主な仕様

	HDR (High Data Rate)	NDR (Next Data Rate)	XDR (Extreme Data Rate)
リンク当り転送速度	200 Gbps	400 Gbps	800 Gbps
最大リンク数	40 ポート	64 ポート	64 ポート
最大転送速度	8 Tbps	25.6 Tbps	51.2 Tbps
仕様策定	2017 年	2021 年	2023 年

表 4 Lossless Ethernet 関連規格

規格	概要
802.1Qaz : Enhanced Transmission Selection(ETS)	帯域保証付き優先キューイングおよび DCBX プロトコル
802.1Qbb : Priority-based Flow Control(PFC)	優先キューごとのフロー制御
802.1Qau : Congestion Notification(CN)	フローベース輻輳通知

表 5 InfiniBand と Lossless Ethernet の比較

項目	InfiniBand XDR	Lossless Ethernet (RoCE)
最大データ転送速度	800 Gbps (リンクあたり)	400 Gbps (リンクあたり)
最大リンク数	64 ポート	48 ポート
最大帯域幅	51.2 Tbps (64 ポート × 800 Gbps)	19.2 Tbps (48 ポート × 400 Gbps)

遅延	1 マイクロ秒未満	数マイクロ秒
プロトコル	InfiniBand プロトコル	RDMA over Converged Ethernet (RoCE)
フロー制御	Credit Based Flow Control (CBFC)	Priority Flow Control (PFC)
RDMA サポート	あり	あり(RoCE)
コスト	高め	比較的低コスト

Lossless Ethernet の主な特長は次のとおりである。

- ・ **パケットロスの防止**: データ転送中にパケットが失われないように設計されている。
- ・ **低遅延**: 通常のイーサネットに比べて、遅延が少なく、リアルタイムのデータ処理が求められるアプリケーションに適している。
- ・ **RDMA(Remote Direct Memory Access)の活用**: RDMA 技術を使用することで、CPU の介入を最小限に抑え、データを直接メモリに転送することにより処理速度が向上する。
- ・ **高効率なデータ処理**: GPU サーバなどの高性能コンピューティング環境で、データの流れをスムーズにし、計算効率を最大化する。
- ・ **コスト効果**: 高価なハードウェアを最大限に活用することで投資対効果が高まる。
- ・ **スケーラビリティ**: 大規模なデータセンターでも効率的なスケールアップが可能となる。

Lossless Ethernet を実現・構成する主な技術を表 6 に示す。これらの技術の組み合わせによって、Lossless Ethernet は InfiniBand に比べてコスト面で優位でありながら、データ転送の信頼性と低遅延を実現している。これにより、より広範なデータセンター環境での採用が進んでいる。

表 6 Lossless Ethernet を実現・構成する技術

技術	概要
RDMA (Remote Direct Memory Access)	・ ネットワーク越しのメモリ間直接データ転送技術 ・ CPU の介入を最小限に抑え、低遅延で高効率なデータ通信が可能 ・ AI/ML 分野の大規模データ処理において重要
RoCEv2 (RDMA over Converged Ethernet v2)	・ ネット上で RDMA を利用するためのプロトコル ・ イーサネットでロスレスなデータ転送が実現可能 ・ RoCEv2 は UDP/IP ベースでパケットロス減少
フロー制御・輻輳通知	・ Priority Flow Control(PFC)を使用 ・ 特定のトラフィックの優先処理でパケットロス防止

DCQCN (Data Center Quantized Congestion Notification)	・トラフィックの状態をリアルタイムで監視 ・必要に応じてトラフィックを調整し、データセンター内の輻輳を管理する ・パケットロスを防ぎつつ効率的なデータ転送が可能となる
高帯域幅のスイッチと NIC	・400/800GbE など高帯域幅に対応する最新のネットワークインタフェースカード(NIC)やスイッチを使用し、大量のデータを迅速に処理することで Lossless Ethernet の性能要件を満たす

4.2 DCTCP(Data Center TCP)

DCTCP²⁹⁾は、ECN を利用して輻輳の推定精度を高め、低遅延かつ高スループットを実現する輻輳制御アルゴリズムである。DCTCP では、データ転送経路上の中継機器(スイッチやルータ)が輻輳を検知するとデータに ECN マークを付加して転送する。そして、データ受信端ではこの ECN マークをエコーバックして送信元に対して輻輳発生とその程度を通知する。これにより、送信元は的確かつ迅速なウィンドウサイズの調整が可能になる。また、パケット損失が発生した場合には、従来の TCP と同様の輻輳制御アルゴリズムを使用してウィンドウサイズを減少させる。主な特長は次のとおりである。

- ・ **ECN の拡張**: ECN を利用して輻輳の程度を推定する。従来の TCP では、パケット損失を通じて輻輳を検出するが、DCTCP はどの程度のバイトが輻輳に遭遇したかを評価し、これに基づいてウィンドウサイズを調整する。
- ・ **高バースト耐性**: データセンターでは、短いフローと長いフローが混在しており、これを誘因として同一サーバへのデータ送信が集中する「インキャストバースト」³⁰⁾と呼ばれる現象が発生しやすい。DCTCP は、これらのバーストを効果的に処理し、パケット損失を最小限に抑える。
- ・ **浅いバッファのスイッチに最適化**: バッファが浅いスイッチ環境でも高いスループットを維持できるように設計されている。これにより、データセンターのコストを抑えつつ、効率的なデータ転送が可能になる。
- ・ **データセンター内での使用**: データセンター内での使用を前提としており、公共のインターネット上での使用は推奨されていない。

DCTCP は、特に高いコスト・パフォーマンスを求める大規模なデータセンター環境での導入が拡大している。³¹⁾

5 むすび

通信ネットワークの利用拡大が続く中、リアルタイム性、確定性、低遅延といった要求が増大している。本稿では、これらの要求に対

応するための技術について紹介した。長年にわたり広く利用されている TCP/IP アーキテクチャに対して、さらなる進化と改良が進められていると同時に、エッジコンピューティングによる通信経路の短縮や、IoT デバイスにおける CoAP³²⁾など TCP/IP に代わる軽量プロトコルの採用など、直接的・抜本的な対応も行われている。今後も TCP/IP アーキテクチャなどの技術資産の進化・改良が続くとともに、エッジコンピューティングなどの新たな技術パラダイムが融合されることで、通信ネットワークのさらなる進化と発展が期待される。これにより、さまざまな産業分野において、より高性能で信頼性の高い通信基盤が構築され、デジタルトランスフォーメーションの推進に寄与することが予想される。

参考文献

- 1) 独立行政法人 情報処理推進機構:DX 白書 2023, IPA, (2023-3-16)
- 2) 総務省:令和 6 年版情報通信白書, 総務省, (2024-7)
- 3) 岩崎有平, “Ethernet TSN が IoT を変える”, 日本システムクリエイト, <https://www.n-s-c.co.jp/Ethernet-TSN/>
- 4) W. Eddy, “RFC 9293 - Transmission Control Protocol (TCP)”, IETF STD 7, August 2022.
- 5) 株式会社インターネットイニシアティブ, “定期観測レポート「ブロードバンドトラフィックレポート～ブロードバンドトラフィックレポート～2 年目に入ったコロナ禍の影響～」”, Internet Infrastructure Review (IIR) Vol.52, 2021 年 9 月.
- 6) 株式会社インターネットイニシアティブ, “定期観測レポート「ブロードバンドトラフィックレポート～コロナ禍を経てトラフィックは安定増加傾向～」”, Internet Infrastructure Review (IIR) Vol.60, 2023 年 9 月.
- 7) 株式会社インターネットイニシアティブ, “定期観測レポート「IIJ インフラから見たインターネットの傾向～2024 年～」”, Internet Infrastructure Review (IIR) Vol.65, 2024 年 12 月.
- 8) D. Medhi, K. Ramasamy, “7.2.2 Bandwidth Delay Product”, Network Routing (Second edition), A volume in The Morgan Kaufmann Series in Networking, 2018.
- 9) J. Postel, “RFC 768 - User Datagram Protocol”, IETF, August 1980.
- 10) B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, L. Zhang, “RFC 2309 - Recommendations on Queue Management and Congestion Avoidance in the Internet”, IETF4445, April 1998.
- 11) R. Pan, P. Natarajan, F. Baker, G. White, “RFC 8033 - Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem”, IETF, February 2017.

- 12) K. Nichols, V. Jacobson, A. McGregor, J. Iyengar, “RFC 8289 - Controlled Delay Active Queue Management”, IETF, January 2018.
- 13) I. Rhee, L. Xu, S. Ha, A. Zimmermann, L. Eggert, R. Scheffenegger, “RFC 8312 - CUBIC for Fast Long-Distance Networks”, IETF, February 2018.
- 14) L. Kleinrock, “The latency/bandwidth tradeoff in gigabit networks”, IEEE Communications Magazine, April 1992, Volume: 30, Issue: 4, p.36 - 40
- 15) V. Jacobson, R. Braden, D. Borman, “RFC 1323 - TCP Extensions for High Performance”, IETF, May 1992.
- 16) M. Dong, Q. Li, D. Zarchy, B. Godfrey, M. Schapira, “Rethinking congestion control architecture: performance-oriented congestion control”, ACM SIGCOMM Computer Communication Review, Vol.44, No.4, August 2014.
- 17) PCCproject (Performance-oriented Congestion Control project), URL: <https://github.com/PCCproject>
- 18) N. Cardwell, I. Swett, J. Beshay, “BBR Congestion Control”, Internet Draft: draft-ietf-ccwg-bbr-01, October 10, 2024.
- 19) B. Briscoe, K. De Schepper, M. Bagnulo, G. White, “RFC 9330 - Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture”, IETF, January 2023.
- 20) K. De Schepper, B. Briscoe, “RFC 9331 - The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)”, IETF, January 2023.
- 21) K. De Schepper, B. Briscoe, G. White, “RFC 9332 - Dual-Queue Coupled Active Queue Management (AQM) for Low Latency, Low Loss, and Scalable Throughput (L4S)”, IETF, January 2023.
- 22) D. Satish, J. Kua, S. Pokhrel, “Active Queue Management in L4S with Asynchronous Advantage Actor-Critic: A FreeBSD Networking Stack Perspective”, Future Internet, vol. 16, no. 8, July 2024.
- 23) K. Ramakrishnan, S. Floyd, D. Black, “RFC 3168 - The Addition of Explicit Congestion Notification (ECN) to IP”, IETF, September 2001.
- 24) 3GPP TR21.918 Technical Specification Group Services and System Aspects; Release 18 Description; Summary of Rel-18 Work Items (Release 18)
- 25) J. Iyengar, M. Thomson, “RFC 9000 - QUIC: A UDP-Based Multiplexed and Secure Transport”, IETF, May 2021.
- 26) HPC Advisory Council, “Introduction to High-Speed InfiniBand Interconnect”, Technical Workshop on InfiniBand for Trigger/DAQ – indoco (CERN), January 2013.
- 27) IEEE Standard 802.3-2022 “IEEE Standard 802.3 Ethernet”.
- 28) IEEE Standard 802.1Q-2022 “IEEE Standard for Local and metropolitan area networks—Bridges and Bridged Networks”.
- 29) S. Bensley, D. Thaler, P. Balasubramanian, L. Eggert, G. Judd, “RFC 8257 - Data Center TCP (DCTCP): TCP Congestion Control for Data Centers”, IETF, October 2017.
- 30) C. Canel, B. Madhavan, S. Sundaresan, N. Spring, P. Kannan, Y. Zhang, K. Lin, S. Seshan, “Understanding Incast Bursts in Modern Datacenters”, IMC '24: Proceedings of the 2024 ACM on Internet Measurement Conference, p674-680, November 04, 2024.
- 31) A. Dhamija, B. Madhavan, H. Li, J. Meng, S. Khare, M. Rao, L. Brakmo, N. Spring, P. Kannan, S. Ghorbani, “A large-scale deployment of DCTCP”, 21st USENIX Symposium on Networked Systems Design and Implementation (NSDI '24), April 16–18, 2024.
- 32) Z. Shelby, K. Hartke, C. Bormann, “RFC 7252 - The Constrained Application Protocol (CoAP)”, IETF, June 2014.

執筆者



加藤 豊行
先端技術研究所
技術企画部

公知