jh191003-NWJ

高速大容量トラフィックキャプチャ/ジェネレータの開発

中村 遼（東京大学）

概要

本研究では NVMe ストレージと Ethernet NIC が P2P DMA を用いて直接パケットデータをやり取りすることで、ハードウェアの性能を限界まで活かした高速、大容量なトラフィックキャプチャ/ジェネレータの開発を行う。今年度は、P2P DMA のためにデバイス上のメモリをユーザランドから扱うライブラリの設計と実装を行い、Ethernet NIC と NVMe SSD で P2P DMA を行うためのソフトウェア基盤を開発し、トラフィックキャプチャ/ジェネレータアプリケーションを実装した。実験では、P2P DMA を行うために必要な、デバイス上にメモリを持つ NIC および SSD が 2019 年度時点で存在しないため、FPGA ベースのデバイスを PCIe リンク上のメモリバッファとして用いる形で性能を計測を行った。実験の結果、このメモリデバイスを用いた構成が原因となって DMA と同等の性能を P2P DMA で出すことは出来なかったが、その原因調査を通じて P2P DMA を活用するために必要な技術的要素の確認と今後の検討を行った。

1　共同研究に関する情報

1.1　共同研究を実施した拠点名

東京大学

1.2　共同研究分野

■超大容量ネットワーク技術分野

1.3　参加研究者の役割分担

中村　遼・東京大学 情報基盤センター
役割: トラフィックジェネレータ開発

明石　邦夫・情報通信研究機構
役割: トラフィックキャプチャ開発

2　研究の目的と意義

本研究では、超高速、大容量のネットワークを前提とした汎用マシンによるトラフィックキャプチャおよびジェネレータに関する研究開発を行う。Ethernet の高速化に伴って、x86 の汎用マシンに搭載可能な PCIe による Network Interface Card (NIC) も、100Gbps、200Gbps と高速なものが登場している。しかし現在の General Purpose Operating System (汎用 OS) では、100Gbps という高速リンクをアプリケーションも含めて効率的に使い切ることは難しい。これは、現在の OS における、ネットワーク越しにデータをやりとりするためのアーキテクチャが、10Base-T などの低速なネットワークの時代から大きく変化していないことに起因する。

そこで本研究では、PCIe デバイス同士が直接データをやりとりしてネットワークと通信する手法を用いたアプリケーションの開発を行う。提案手法では、NVMe のもつ Controller Memory Buffer (CMB) を用いることを想定し、NVMe ストレージと NIC 間で、CPU や

メインメモリを経由することなく、直接 PCIe バス越しにデータをやりとりする。これによって、汎用マシンの PCIe バスというハードの性能を限界まで使い切った、高速、大容量なデータ通信の実現を目指す。本研究ではこの第一歩として、提案手法を用いた高速、大容量のトラフィックキャプチャ、およびジェネレータの研究開発を行う。

本研究ではまず Peer-to-Peer DMA (P2P DMA) と呼ばれる、CPU とメインメモリをバイパスするデバイス間通信を用いた NIC と NVMe 間のデータ転送を実現する。その上で、P2P DMA を用いたトラフィックキャプチャとトラフィックジェネレータを開発する。これら 2 つのアプリケーションは、直接パケットを NVMe と NIC 間でやりとりすることで実現可能であり、ファイルシステムやネットワークプロトコルスタックといった実装を必要としないため、提案手法の有効性を確認するのに適したアプリケーションである。

3 当拠点公募型研究として実施した意義

本研究で開発するものは、高速、大容量のネットワークアプリケーションである。そのため、実装物を SINET5 を用いて複数拠点に跨って高速ネットワーク上で実験すること想定し、拠点公募型共同研究として実施した。

4 今年度の研究成果の詳細

P2P DMA では、DMA の宛先としてメインメモリではなく PCIe デバイス上のメモリを用いる。そのためまず、この PCIe デバイス上のメモリを Memory-Mapped I/O (MMIO) を通じてアプリケーションから柔軟に利用するためのライブラリの開発を行った。このライブラリを用いることで、ユーザランドアプリケーションは通常のメモリ領域と同じようにデバイス上のメモリを扱うことができる。そして、Ethernet NIC で P2P DMA を利用するために本ライブラリを高速パケット I/O フレームワークである netmap [1] に統合した。さらに、NVMe SSD でも P2P DMA を利用するため、NVMe SSD のユーザランドドライバ実装である UNVMe [2] にも本ライブラリを統合した。

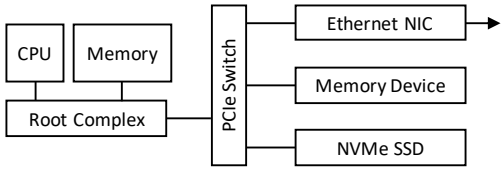

図 1　トラフィックジェネレータ用 PC の構成

PCIe デバイス上のメモリを柔軟に利用するためのライブラリと Ethernet NIC および NVMe SSD ドライバへの統合によって、Ethernet NIC と NVMe SSD 間での P2P DMA が可能となった。このソフトウェア基盤を用いてさらに、トラフィックジェネレータ/キャプチャアプリケーションの実装を行った。本アプリケーションは、CPU とメインメモリを経由することなく、パケットを SSD から読み出して送信ないしは保存することができる、新しいトラフィックジェネレータアプリケーションである。

実験として、本アプリケーションの性能を計測した。2019 年度時点では、残念ながら外部から PIO および DMA でアクセスできるメモリ領域 (CMB) を搭載した NVMe SSD は販売されなかったため、実験では MMIO 経由でアクセスできるメモリを持つ FPGA ベースのデバイスを用いた。図 1 に、実験に用いた PC の構成を示す。マザーボード上の PCIe スイッチ配下に、Ethernet NIC、上述のメモリ用 FPGA デバイス、そして NVMe SSD を搭載

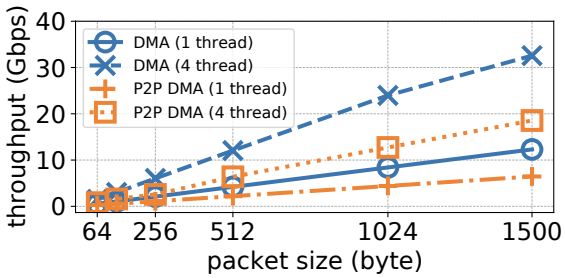

図 2　トラフィックジェネレータの送信性能

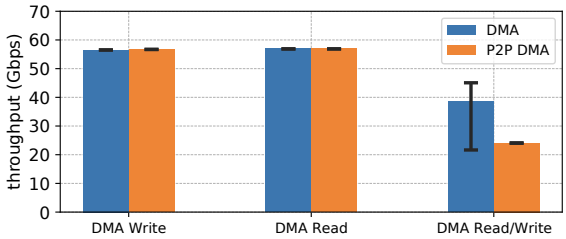

図 3　メインメモリとメモリデバイスに対する DMA のスループット

し、アプリケーションはまず NVMe SSD からパケットデータをメモリデバイスに読み出し、Ethernet NIC がメモリデバイス上のパケットをネットワークに送信する。Ethernet NIC には Intel XL710 40Gbps NIC を、NVMe SSD には Samsung PM1725a NVMe SSD を用いた。CPU には Intel Core i9-9820X 3.3GHz 10 core CPU を、メモリには Crucial の 16GB DDR4-2666 メモリを 2 枚用いた。

図 2 に、トラフィクジェネレータで送信したパケットを netmap pkt-gen を使って受信し、スループットを計測した結果を示す。実験では、送信するパケットのサイズを変えながら、パケットバッファとしてメモリデバイスを使った場合 (図中 P2P DMA) とメインメモリを使った通常の DMA の場合 (図中 DMA) の性能をそれぞれ計測した。

図 2 が示すように、結果として P2P DMA ではメインメモリを経由するほどの性能を出すことはできなかった。パケットサイズが 1500 バイトの際、メインメモリを経由する DMA の場合 4 スレッドで約 32Gbps に対して、P2P DMA では約 19Gbps となった。この原因は、メモリデバイスの特性だと考えられる。図 3 に、pcie-bench [3] というベンチマークデバイスを用いて、PCIe デバイスからメインメモリに対する DMA のスループットと、計測に用いたメモリデバイスに対する P2P DMA のスループットを計測した結果を示す。なお pcie-bench から P2P DMA を行うために、開発したライブラリを pcie-bench に統合した。DMA Read または DMA Write のみの場合、メモリデバイスへの P2P DMA でもメインメモリへの DMA と同等のスループットが出ている。一方 DMA Read と DMA Write を同時に行った場合、メモリデバイスへの P2P DMA は DMA の半分ほどのスループットとなった。

トラフィックジェネレータの性能計測実験では Ethernet NIC と NVMe SSD の間にメモリデバイスを挟んだため、メモリデバイスは NVMe SSD からの DMA Write (NVMe SSD からの読み出し) と Ethernet NIC からの DMA Read (Ethernet NIC へのパケット送信) の両方を処理しなければならない。これは、図 3 が示すようにメインメモリへの DMA Read/Write よりも低いスループットとなる。結果として、トラフィックジェネレータの性能も P2P DMA を用いた方が通常の DMA よりも低くなった。

P2P DMA で性能が出ないという本実験の結果は、中間バッファとしてメモリデバイスを用いたことに起因している。メモリデバイスが必要だったのは、P2P DMA の宛先となれるメモリを持つ Ethernet NIC または NVMe SSD がまだ存在しないためである。本来 P2P DMA では 2 つのデバイス同士、本研究では

Ethernet NIC と NVMe SSD が、直接データをやり取りすることが想定される。また 2 つのデバイス間でデータをやりとりする場合には、必要な DMA は DMA Read か DMA Write のどちらかのみである。そのため、メモリデバイスを含む 2 つのデバイス間の P2P DMA は図 3 からメインメモリ対する DMA と同じだけのスループットを得られることが期待できる。本実験を通じて、今後 P2P DMA による性能向上や CPU およびメインメモリの負荷の低減を目指すためには、やはり CMB 対応の NVMe SSD や、同等の機能を持つ Ethernet NIC が必要であることが改めて確認された。

5　今年度の進捗状況と今後の展望

今年度は、P2P DMA のためにデバイス上のメモリをユーザランドから扱うライブラリの設計実装を行い、Ethernet NIC と NVMe SSD のドライバフレームワークに本ライブラリを統合することで Ethernet NIC と NVMe SSD で P2P DMA を行うためのソフトウェア基盤を実現した。その上で、トラフィックキャプチャ/ジェネレータアプリケーションを実装した。しかし、残念ながら 2019 年度には P2P DMA を行うために必要なデバイス上にメモリを持つ NIC および SSD が存在しないため、FPGA ベースのデバイスを PCIe リンク上のバッファとして用いる形で性能を計測した。結果的にこのメモリデバイスを用いた構成が原因となって DMA と同等の性能を P2P DMA で出すことは出来なかったが、一方で P2P DMA を効率的に実行するためにはやはり PCIe デバイス側の対応が必要であることが改めて確認できた。今後、CMB 対応の NVMe SSD の登場を待ちつつ、Ethernet NIC に CPU とメモリを搭載した Smart NIC 等を用いて P2P DMA を行う方法を検討する。

6　研究業績一覧（発表予定も含む）

参考文献

[1] Luigi Rizzo., ”netmap: A Novel Framework for Fast Packet I/O”. In Proceedings of the 2012 USENIX Annual Technical Conference (USENIX ATC 12), pages 101-112, Boston, MA, USA, 2012.

[2] MicronSSD/unvme: User Space NVMe Driver, https://github.com/MicronSSD/unvme

[3] Rolf Neugebauer, Gianni Antichi, José Fernando Zazo, Yury Audzevich, Sergio López-Buedo, and Andrew W. Moore. 2018. Understanding PCIe performance for end host networking. In Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (SIGCOMM ’18). ACM, New York, NY, USA, 327-341.