用于加速分布式系统的智能网卡的性能鉴定

严牧田[1)]

[1)](华中科技大学计算机学院 武汉市 中国 430074)

摘 要 智能网卡是一种用于加速分布式系统的设备。由于目前还没有对智能网卡的全面鉴定，现有设计通常只能应用单一的通信路径进行工作的负载卸载。本文是关于离路智能网卡（off-path smartnic）通信路径角度的全面研究，探讨客户端、板载 SoC 和主机通信的关键性能特征，为设计人员提供一些发现和建议。提出了一个同时使用智能网卡多条通信路径的建议，能够为各种分布式系统提供优化的机会，并且在分布式为文件系统和基于 RDMA 的分解键值存储问题上进行了案例研究，证实了优化的有效性。实验结果表明，LineFS 和 DrTM-KV 的性能分别提高了 30%和 25%。

关键词 智能网卡；分布式系统；通信路径；分解键值存储

Characterizing Off-path SmartNIC for Accelerating Distributed Systems

Yan Mutian[1)]

[1)](Huazhong University of Science and Technology, Wuhan, China,430074)

Abstract SmartNIC is a device used to accelerate distributed systems. Since there is no comprehensive characterization of smart NICs, existing designs are usually limited to applying a single communication path for workload offloading. This paper is a comprehensive study on the communication path perspective of off-path smartnic cards (off-path smartnic), exploring the key performance characteristics of client, on-board SoC, and host communications to provide some findings and recommendations for designers. A proposal is made for the simultaneous use of multiple communication paths for smart NICs that can provide optimization opportunities for various distributed systems, and a case study on the distributed-for-file-systems and RDMA-based decomposed key-value storage problems confirms the effectiveness of the optimization. Experimental results show performance improvements of 30% and 25% for LineFS and DrTM-KV, respectively.

Key words smartNIC；Distributed Systems；Communication Path；Disaggregated Key-value Store

1 引言

为了加速分布式系统的传输效率，采用了支持 RDMA，远程直接内存访问的网卡，来提升效率，但是随着通路的传输效率逐渐提高，CPU 资源逐渐显得不足，因此产生了智能网卡，让复杂的计算能够卸载到网卡上面。

在路的智能网卡将网卡内核直接暴露给系统，编程的过程会对系统产生不良的影响，并对开发人员带来负担。为了简化系统的开发，采用离路的智能网卡，在 RNIC 内核旁边附加了一个可编程多核片上系统 SoC（system on chips），开发人员可以将其视为一个独立的服务器。

因为离路智能网卡的计算能力弱于主机，因此无法提高单一的网络路径速度（关键路径理论）之前的工作主要集中在如何利用智能网卡实现分布式系统，但是主要集中在将计算卸载到智能网卡上。相对应的，忽视了 SmartNIC 的基本功能，即联网，对整体性能的影响情况。本文系统地研究了智能网卡的性能特征。不同的路径速度更快的原因和时间。

本文[1]研究发现不同的路径展现的不同性能特征，从网卡到 SoC 的 RDMA 路径比到主机的路径快了 1.48 倍。SoC 的底层硬件细节，包括内存访问路径和 PCIe 和主机不同，涉及 SoC 的 RDMA 请求会出现 48%的性能下降。SoC 和主机之间的路径未能够对 CPU 进行充分的利用，RDMA 两次穿过网卡内部的 PCIe，只能利用一半的带宽，容易受到数据包放大的影响。

2 背景和动机

2.1 RNIC

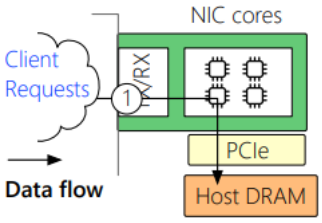

图 1 RNIC 的结构

如图 1 所示，RDMA (Remote Direct Memory Access)是一种低延迟(2μs)、高带宽(200gbps)的网络，广泛应用于现代数据中心。利用 RDMA 的一种直观方法是使用其双边原语(SEND/RECV)加速消息传递，例如基于 RDMA 的 RPC。另外，单侧原语(READ/WRITE2)允许 RNIC 绕过主机 CPU 访问主机内存。具体来说，网卡核心内部使用 PCIe 链路的直接内存访问(DMA)功能来访问主机内存

虽然 RDMA 提高了许多分布式系统的性能[18,76,62,29]，通常提升了几个数量级，但它仍然存在以下两个问题，特别是当 rnic 扩展到更高的性能时。

问题#1:主机 CPU 占用。对于双面基元，分布式系统需要大量的的 cpu 来让一个功能强大的网卡达到饱和。测量表明，24 核服务器在 200 Gbps RNIC (ConnectX-6)上每秒只能饱和 8700 万个数据包(Mpps)，而 NIC 内核可以处理超过 195 Mpps。最近的研究进一步表明，当网络带宽从 25 Gbps 扩展到 100 Gbps 时，分布式文件系统需要 2.27 倍的 CPU 内核来处理网络数据包。虽然部署更强大的 cpu 可以缓解这一问题，但 RNIC 带宽也在迅速增长，目前已达到 400gbps。

问题#2:网络放大。使用单侧 RDMA 基元可以减轻主机 CPU 的压力，因为它允许系统将内存访问卸载到 RNIC。然而，有限的卸载能力限制了系统性能，因为单个请求可能涉及多次 read / write 往返才能完成(通常称为网络放大)。

2.2 从RNIC到SmartNIC

为了解决 RNIC 的局限性，SmartNIC 在网卡上增加了一块 4 - 64gb 的板载内存和各种计算单元(如 SoC)。通过将它们暴露给开发人员，智能网卡可以将定制的计算卸载到它上。具体来说，智能网卡可以分为以下几类。

On-path SmartNIC（在路智能网卡）。在路智能网卡将网卡内核暴露给具有低级可编程接口的系统，允许它们直接操作原始数据包。顾名思义，卸载的代码位于网络处理管道的关键路径上。

但是，在路上 SmartNIC 有两个限制。首先，卸载代码(④)与发送到主机的网络请求(①)竞争网卡内核。如果将过多的计算转移到主机上，发送给主机的正常网络请求将遭受显著的性能下降。其次，由于其低级接口，对路径上网卡进行编程很困难。

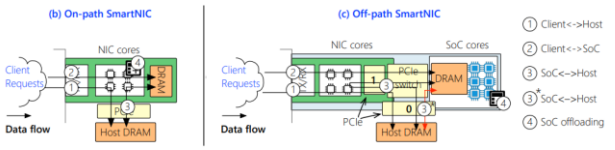

图 2 on-path 和 off-path 的两种 SmartNIC

如图 2 所示，离路智能网卡提供了另一种选择:它将额外的计算核心和内存封装在网卡核心旁边的单独 SoC 中。因此，卸载的代码不在网络处理管道的关键路径上。从 NIC 的角度来看，SoC 可以被视为第二个具有专属网络接口的成熟主机。为了实现网卡核心、SoC 和主机之间的桥接，SmartNIC 内部集成了 PCIe 交换机，实现网络报文的合理调度。

相对于路径上的代码，只要不涉及网络通信(mapping)，卸载代码不会影响主机的网络性能。由于这种明确的分离，SoC 可以运行具有完整网络堆栈(即 RDMA)的成熟内核(例如 Linux)，从而简化系统开发并允许卸载复杂任务。但是，使用离路 smartnic 加速分布式系统通常比使用在路 smartnic 更具挑战性。这是因为 PCIe 交换机延长了所有的通信路径(即①、②、③)，可能会导致性能下降。

图 3 举例说明了在具有单边 RDMA read 的分布式内存中的键值存储上执行 get 请求。客户端首先使用一个(或多个)READ 来查询给定键的索引。根据前面的 READ 返回的索引，发出一个额外的

READ 来检索值。

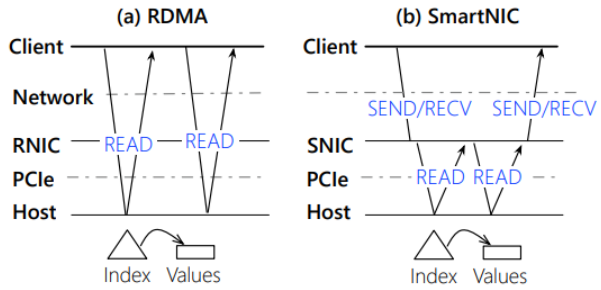

图 3 两种不同的网卡进行数据访问的情况

关于 smartNIC 的其他研究[5]也表明了，在 smartNIC 的使用场景之中，其进行计算的功能在很多情况下，能够带来更多的优势。针对网络流量模式引起昂贵的重传或消耗内存带宽的问题的问题，影响应用程序的性能。而借助其有限的计算资源，产生的轻量级流量冲突避免程序。通过动态线性回归，以及随机森林分类器，能够使用较小的成本，使用带有默认超参数的 RFC 进行二元分类，借助 smartNIC 的计算功能，能够让网络独立于主机应用程序处理资源，使得我们能够优化资源增加系统的弹性。

2.3 研究对象

本文在 Bluefield-2 上进行了研究，Bluefield-2 是一种典型的离路智能网卡，针对卸载通用计算进行了优化。

硬 件 ： Bluefield-2 采 用 成 熟 的 RNIC (ConnectX6)作为网卡核心，实现高速组网。这些核心支持所有 RDMA 操作。它的可编程性来自于集成的板载 SoC，该 SoC 具有 16gb DRAM 和 ARM Cortex-A72(8 核，2.75 GHz)。PCIe 4.0 交换机将网卡核心、SoC 和主机连接在一起，使双向数据传输高达 256gbps。请注意，SoC 通过内部链路连接到 PCIe 交换机，而不是通过 PCIe。具体来说，Bluefield 提供的硬件计数器也意味着它只有两条 PCIe 链路: 一条连接 RNIC 与交换机(PCIe1)，另一条连接交换机与主机(PCIe0)。

软件：SoC 运行完整的 Linux，允许开发人员将其视为普通的 ARM 服务器。内核还承载了一个完整的 RDMA 堆栈，这使得启用基于 RDMA 的通信非常方便。此外，Bluefield 还为 DMA 等高级使用提供了 DOCASDK。

通信基元:RDMA 和 DMA。与 SoC 相关的所有通信路径都使用 RDMA 进行，以简化系统开发。如图 2 所示，客户端可以向 SoC (mapping)发出单侧或双面 RDMA 请求，类似于主机上的双服务器。同时，SoC 也可以通过 RDMA 与主机进行交互，反之亦然。但是，SoC 和主机之间的数据交换必须通过 RDMA 支持的 RNIC (PCIe1 和 NIC 内核)，这给该路径增加了一个隐藏的瓶颈。幸运的是，我们发现 Bluefield 进一步通过 DOCA 提供 DMA 支持(③*)，允许 SoC 使用 DMA 访问主机内存(反之亦然)，绕过 RNIC。

目前 Bluefeild 的研究情况: 以往的研究主要关注 Bluefield 的计算能力，揭示了 SoC 内核在执行卸载任务和发送网络请求方面的相对弱点。这是因为主频和核数都不如主机 CPU。由于 smartnic 的功率限制，网卡与主机 CPU 的相对性能比较不会发生变化。因此，我们将此作为调查的前提。

例如在一些研究中[2]，在给定流量情况下系统分析了 SmartNIC 卸载程序的性能特征，采用以数据包为中心的建模方法，根据数据包在不同硬件实体上的传输方式来研究 SmartNIC 的执行行为，与以执行流为中心的现有模型不同，采用了系统接口、吞吐量建模、延迟建模、模型扩展四个步骤基于现有网卡进行了性能分析。该方法的优化效果相当明显，但是和本文的思路差异较大，更偏向于通过软件模拟的方式对系统的延迟进行分析，探究的是在现有的结构之下如何将其表现结果拟真出来。

在使用该方法对 smartNIC 的性能进行优化的过程中，调整了应用程序的执行并发性，探索了硬件设计的空间，确定了在各种情况下的硬件资源配置，但是需要程序员对底层的硬件架构有比较扎实的了解和研究，估算卸载程序可达到的延迟和吞吐量以及潜在的瓶颈。

另有提出一种软硬件协同设计的异构 SmartNIC 系统[4]，克服了分布式 dlrm 的通信瓶颈，减轻了内存带宽的压力，提高了计算效率,提供了一套 SmartNIC 缓存系统设计(包括本地缓存和远程缓存)和 SmartNIC 计算内核，减少数据移动，减轻内存查找强度，提高 GPU 的计算效率。

在基于 CPU 的高性能计算集群中，实现一个框架[3]，能够支持将任何通信模式卸载到 DPU，并且实现低通信延迟和完美重叠。实现的路径是首先确定局限性和相应的瓶颈，之后提出备选设计并通过备选设计进行方案的探究。是第一个为 DPU 提供高效和通用卸载的框架。

相比之下，很少有研究考虑了 Bluefield 中的各

种通信模式(即①、贴图、投影和交叉)，这是我们工作的主要焦点。本文系统地探讨了 Bluefield 的性能特点，并为未来的系统开发人员总结了有见地的经验教训和建议。

3 智能网卡性能鉴定

如 2.3 节所述，智能网卡的计算能力比 CPU 弱，本文专注于对智能网卡的性能探究，研究智能网卡的各种数据通路传输数据的性能表现，不考虑计算能力的因素。

3.1 单个通路性能测试

本文使用最先进的 RDMA 通信框架进行了实验。对于单侧操作(READ 和 WRITE)，请求者使用 RDMA 的可靠连接(RC)队列对(QP)与一个响应者通信。默认情况下，应答器地址是从 10gb 地址空间中随机选择的。对于双向操作(SEND/RECV)，响应方实现了一个回声服务器，它利用所有可用的内核来处理消息，请求方通过不可靠数据报(UD) qp 与它通信，以获得更好的性能。对于端到端延迟，我们部署一台请求者机器来防止排队效应的干扰。对于峰值吞吐量，我们使用多达 11 台请求机器来饱和响应器。

最后，我们启用了所有众所周知的优化，包括地址对齐[81]，无信号请求和大页面，以防止滥用 RDMA 的副作用。

（1）客户端到主机的通信

对 read、write 和 send/recv，延迟分别高了 15-30%，15-21%，6-9%，延迟来源于 SNIC 和 RNIC 的端到端读取延迟差异，每次通过 PCIe 交换机都需要 300ns 的时间。SNIC 上省略了一次通过 PCIe 交换机的过程，因此 write 的延迟增加略低。SNIC 上 send/recv 延迟增加主要是因为响应者的 CPU 成本增加。

吞吐量上对于小于 512 字节的数据，read、write 和 send/recv 分别降低了 19-26%，15-22%，3-36%，数据包比较大的话延迟相似，因为都会受到网络带宽的制约

总结：网卡、PCIe1 和 PCIe0 的最低贷款将会成为瓶颈，针对小型的请求，“智能”反而会降低效率

（2）客户端到 SOC 的通信

READ 的延迟最多可减少 14%。原因是跳过 PCIe0。然而，它仍然比 RNIC 高 4-15%，因为请求仍必须通过 PCIel 的 PCIe 交换机。对于 WRITE，由于内核的异步完成，SNIC2 提供了与 SNIC1 类似的性能 (见图 4)。对于 SEND/RECV，SNIC 的延迟比 SNI 高 21-30%。

由于 SoC 的计算能力较弱，SNIC 的计算能力较低。

吞吐量上小于 512 字节的有效载荷情况下比客户端到主机的要高 1.08-1.48 倍，甚至 read 效率要高于 RNIC 这是因为 SoC 内存和 PCIe 交换机之间的封装更加紧密。但是 write 要低于 RNIC，因为和主机相比，SoC 的 DRAM 通道较少，限制了写访问的并发性。

send/recv 性能较差，只能够达到主机和客户端之间吞吐量的 64%，因为存在由响应者 CPU 发送回复的瓶颈。

总结：避免倾斜内存访问，SoC 内核可能会影响单边 RDMA 基元的内存访问行为，因为其支持的功能较少。

DRAM 需要一个不太小的范围来使用所有的内存模块，随着地址范围增加，SNIC 的表现会逐渐增加。

避免大的 read 请求，大的有效载荷虽然能够更加充分地利用网络带宽，但是随着请求地有效载荷增大到 9MB，SNIC2 的读取性能会下降，因为 NIC 会受到首行阻塞的影响。

（3）SOC 和主机的通信

从 soc 向主机的延迟非常高，尤其是 read，因为请求者 soc 需要更长的时间才能够完成请求。相反方向的延迟略有减小，但是仍然比 SNIC2 高了 4-17%

吞吐量上对载荷小于 512 字节的请求，主要取决于请求者发布联网请求的能力。SNIC3 的读取吞吐量只能达到 29M 和 51.2M 请求/秒，分别是 S2H 和 H2S

总结：要避免大型读写请求，谨慎启用门铃批处理。

内存映射占用了网卡发送请求的时间，优化内存映射的办法就是门铃批处理，将多次内存映射请求连接在一起，但是它并不是总对主机端有帮助，在读取 SoC 存储器上的请求的时候速度比较慢

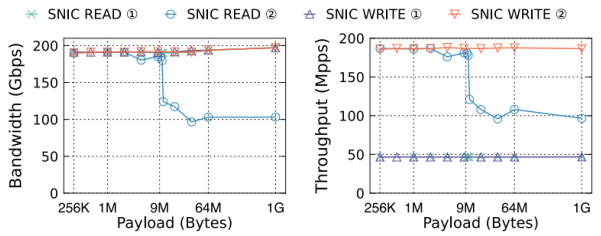

图 4 主机和 SoC 的带宽差别

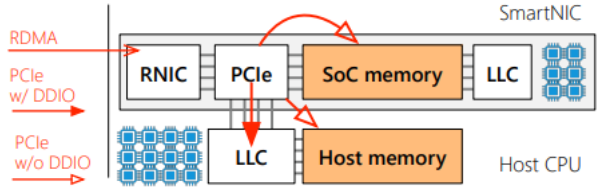

图 5 访问主机和 SoC 内存的不同方法

3.2 采用多通路的性能研究

以前的方法主要是利用 SmartNIC 的单一路径来优化分布式系统的特定功能。

但不能充分发挥 smartnic 的计算和组网能力。此外，仅考虑单一路径可能会忽略不同路径之间对资源(例如 PCIe 和 PCIe 交换机)的干扰。因此，我们首先从整体上研究多路径并发使用的性能特征，然后为设计人员巧妙地使用 smartnic 提供优化指导。

主机和 SoC 的并发通信（①+ ②）：起初的实验方案采用一半客户端向主机发送，另一半向 SoC 发送，性能比单独使用其中的某一个的 read、write、send/recv 分别提高了 1.45、1.5、3.3 倍

该结果和预想不符，因为两个路径应当竞争 NIC 内核，进行假设：SmartNIC 内部为每个端点保留了一些 NIC 内核，因此同时向主机和 SoC 发送的时候能够启用更多的内核。为了探究真实的性能数据，需要调整实验设计思路，排除预留内核对性能的干扰。

重新设计实验对假设进行验证，首先增加请求者机器让 NIC 达到饱和，之后更改响应者，每一个采用 0B 的有效载荷，此时发现两种情况的吞吐量分别比单独使用 1 和 2 高了 4-13%和 5-10%。两条路径的吞吐量总和高于同时使用两条路径。该实验证实了假设，同时对吞吐量的变化进行了量化测试。

同时进行机器间和机器内通信，讨论 1+H2S，发现机器内通信降低了机器间通信的性能对于 read、write、send/recv，性能分别下降 7-15%、4-27%、9-14%，③*的影响略小。

原因是③依赖网卡来支持 RDMA，相比之下，SNIC③*能够用 DMA 避免这方面的干扰。

综上所述，将客户端请求同时发送到主机和 SoC 中，能够更好地利用 NIC 内核处理小型请求。

只有在有多的空闲资源的时候，才要考虑使用机器内通信，因为它经常会阻塞其他的通信，SNIC3 使用的带宽不应该高于 PCIe 带宽极限与网络极限的差值。

3.3 优化思路

本节主要介绍如何巧妙地利用 SmartNIC 的多条通信路径，提升分布式系统的性能。具体来说，考虑到目标分布式系统的功能(例如，分布式文件系统中的文件复制)需要通过 SmartNIC 加速，我们建议设计人员考虑以下步骤:

1.为 SmartNIC 设计潜在的替代方案以支持给定的功能，并根据我们的研究发现的性能特征对其进行优化。

2. 根据系统特定的标准对备选方案进行评估和排序。

3. 依次选择并组合备选方案，直至 SmartNIC 资源饱和。

系统特定标准。这些标准可以是系统设计者希望达到的理想属性，也可以是系统的限制条件。就分布式文件系统中的复制而言，其属性包括低主机 CPU 开销和高网络带宽利用率。对于分解键值存储，其特性包括较少的网络放大、低延迟和高吞吐量。限制主机的 CPU 占用较少或根本没有 CPU 可供我们使用。

图 6。一旦对一个嵌入的权重向量进行了修剪，在训练继续进行时很难评估其对模型准确性的影响。

讨论。目前，我们只考虑以贪婪的方式组合备选方案，这对于现实世界分布式系统中的大多数网络功能来说已经足够。此外，SmartNIC 提供的可用选项通常数量有限。需要注意的是，有效组合备选方案具有挑战性。对于不同的系统，不同的备选方案可能会消耗 SmartNIC 上的不同资源，而它们之间的组合可能会涉及不同程度的资源争夺。我们之前的分析，包括不同通信路径的瓶颈和同时利用 SmartNIC 上的多个路径，将指导设计人员避免大部分性能争用。然而，如何系统地选择和组合不同的路径是我们未来的工作。

4 案例分析

为了提现本文研究的有效性，针对两个常见的分布式系统使用案例进行了研究。

4.1 分布式文件系统的文件复制

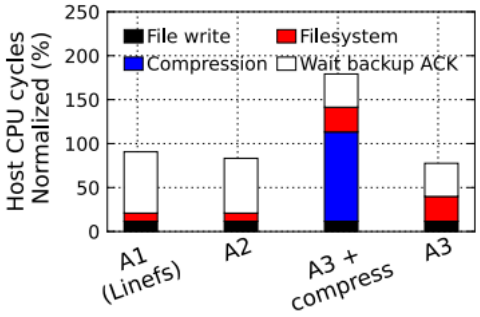

图 6 几种多通路的通信效率

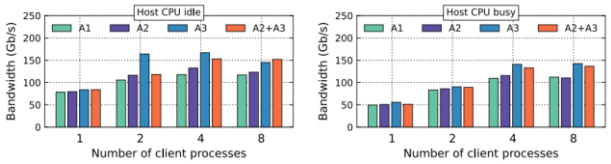

图 7 CPU 带宽情况

文件复制是分布式文件系统容错的关键支柱。随着 RDMA 和非易失性存储器 (NVM) 的出现，一个吸引人的趋势是使用 RDMA 直接在远程 NVM 上复制文件更新以获得更好的性能，即 RDMA 基元可以像 DRAM 一样直接写入 NVM，从而充分利用网络和 NVM 带宽。

设计替代方案。文件复制的理想特性是高性能、高网络利用率和低主机 CPU 开销。在我们的 SmartNIC 上实现文件复制有三种替代方案，如图 8 所示。

1）lineFS 的方法(A1)，将文件复制完全交给 SoC 进行，SoC 负责压缩和复制文件，减少网络传输的数据，同时降低 CPU 使用率，收到复制请求后，主 SoC 读文件、压缩、链式复制，写入远程备份。

2）备选方案 1 中的 3 改成 3*(A2)，减少对带宽的影响

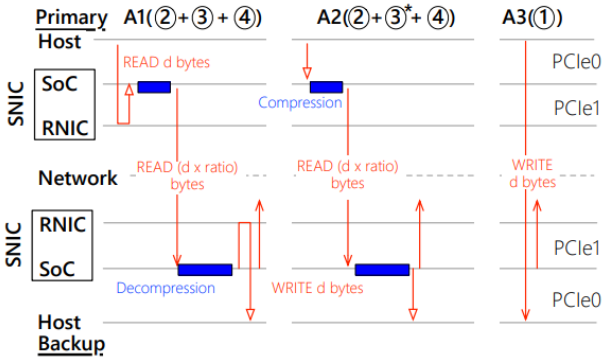

图 8 文件复制备选方案设计

3）主机使用 write 直接把文件写入远程备份，跳过文件压缩(A3)

LineFS 是一种基于 NVM 和 SmartNIC 的先进分布式文件系统。它采用 A1 来复制文件。本文在其开源代码库上进一步实现了 A2 和 A3，并用更高效的 RDMA 实现重写了其后端，以扩展到 200 Gbps 的网络速度，例如异步和分批 RDMA 操作。

优化每种选择。默认情况下，LineFS 的开源代码库中 A1 的块大小为 16 MB。

根据第 3.3 节所述的建议，我们将其缩减为 256 KB，以获得比③更好的性能。这一优化进一步适用于 A2 和 A3。

分析替代方案。A1 是卸载文件复制的最直接方法，可减少通过网络传输的数据（d 对 d × 比率）。因此，理想的峰值带宽为 N/比率，其中 N 为 SmartNIC 的带宽限制。

然而，A1 并未考虑③的 PCIe 占用，它甚至无法使文件传输的网络带宽达到饱和。将主 PCIe 限制（uni）记为 P。

A1 的文件传输带宽 d 受 P1+ 比率限制，因为每个数据包必须通过 PCIe1 输出链路两次。如图 8 所示，一次是从 SoC 到 RNIC（d 字节），另一次是从 SoC 到远程（d×比率字节）。在我们的平台上（p= 256Gbps），只有当压缩率低于 28%时，A1 才会比未压缩文件（其性能瓶颈在于网络 N = 200 Gbps）更好。更糟糕的是，当压缩率较低（≥28%）时，A1 无法使 SmartNIC 的网络带宽达到饱和。例如，在不压缩的情况下（压缩比=1），A1 的峰值仅为 128 Gbps。

图 6所示显示了 A1 在 LineFS 文件写入基准测试中的结果。该基准没有压缩文件。我们可以看到，当主机空闲时，A1 在 8 个客户端的情况下仅达到 117 Gbps。

A2 对 A1 的性能进行了一次全面的提高，如图 6，随着客户端数量的变化，效率为 A1 方案的 1.01 到 1.13 倍，但是 A2 的峰值较低，原因主要是在平台上，write 不能充分利用全部的 PCIe 带宽。其次，SoC 较差的计算能力也成为了文件复制的瓶颈。

设计方案 A3 能够绕过 A1 的 PCIe 占用问题和 A2 的 SoC 性能问题以及慢速 DMA 写入问题，其具有更短的数据路径（如图 8 所示），相比之下等待日志确认所需的时间减少了 40%。缺点是即使不考虑压缩，A3 仍具有更多的 CPU 周期。

选择和组合备选方案。由于 A2 总是优于 A1，所以我们只考虑将 A2 与 A3 结合。正如我们之前分析过的，A3 比 A2 快。因此，在组合路径(A2 + A3)中增加 A3 的比率总是可以提高性能，如图 9 所示。

在启用了高网络利用率的文件压缩时，主机的 CPU 利用率将会提高，因此在增加客户端中路径 A3 的百分比过程中，考虑到固定 50%的压缩比，利用率从 50%降低到 0%。

由于 A2 具有更好的网络利用率，因此采用贪婪方法，首先用 A2 使 SoC 达到饱和，之后使用 A3 进行文件复制，这种方法能够使合并后路径快于 A2 且网络利用率高于 A3。

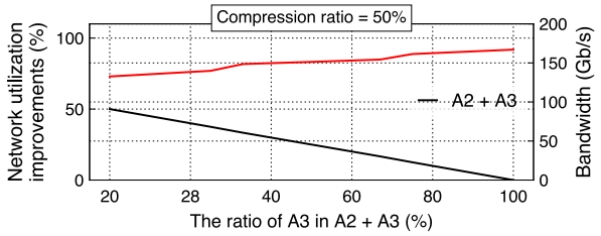

图 9 结合了 A2 和 A3 的网络表现

由图 6 能够看出，主机 CPU 在空闲和繁忙的情况下文件复制的测试结果，在 CPU 空闲时，A2+A3 的运行速度能够提升 7-30%，繁忙时能够提升 4-21%，得益于 SmartNIC 的高效使用和多执行路径的智能利用。

4.2 分解键值存储

基于 rdma 的分解键值存储(RKVS)在现代数据中心中非常普遍。在 R-KVS 中，一个或多个内存服务器存储索引(通常是哈希表)和值。其他机器上的客户机使用 read 遍历索引并检索相应的值来处理请求(即 get)。

设计备选方案：期望的特性是高吞吐量、低延迟和最小的网络放大。限制是我们几乎不能使用主机 CPU(即禁用 SEND/RECV 的路径度量)。SmartNIC 为 R-KVS 提供了五种替代方案，如图 10 所示。

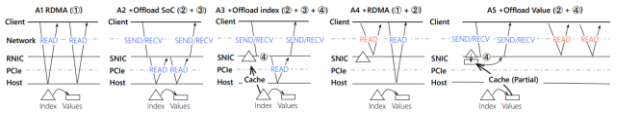

图 10 分解键值存储备选方案设计

1）**客户端使用 read 直接处理请求（A1）**，可预见的容易受到网络放大的影响。

2）**使用 send/recv 向 SoC 发送获取请求(A2)**，SoC 可以遍历索引并通过 RDMA read 读值，能够消除网络放大。

3)解决 2 中从主机本地内存读取数据慢的问题，**卸载索引到 SoC 内存(A3)**，在 smartNIC 上缓存索引。进行实验所用的 Bluefeild2 拥有 16GB 内存，能够完全存储。

4)**使用 read 遍历 SoC 索引(A4)**，用另一个 read 来读主机上面的值，存在网络放大问题

5)**SoC 缓存热键的值(A5)**，避免使用昂贵的通信路径，避免使用之前方法中高昂的通信路径。

鉴于之前的研究建议，在 A2、A3、A5 中，谨慎采用门铃批处理，同时对 A4 和 A5 应用建议，将几个热键复制到多个副本中，避免倾斜访问。对 A2 和 A3 采用 DMA 替代 RDMA 来尽可能减少延迟。通过采用上述建议，能够使当前方案的表现尽可能提升。在采用了这些建议之后，几个备选方案表现如下。

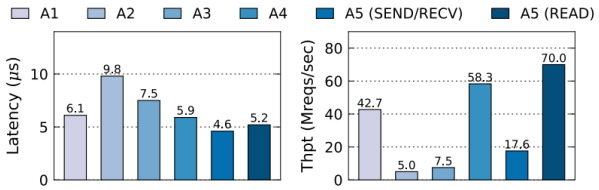

图 11 备选方案表现

如图 11 所示，没有一条路径能够同时实现吞吐量和低延迟。A5 (SEND/RECV)实现了最低的延迟(4.6µs)，因为它完全消除了网络放大问题和昂贵的主机 soc 通信。但是，它的峰值吞吐量(17.6 M 请求/秒)明显低于其他一些替代方案。其中，A5 (READ)和 A4 的吞吐量峰值分别达到 70 M reqs/s 和 58.3 M reqs/s。它们具有更高的吞吐量，因为 RDMA 到 SoC(①)的路径更快(§3.2)。请注意，A5 并不总是可以实现的，这需要在 SoC 内存中缓存所有键值。

因此，如果 SoC 内核成为瓶颈， A4 是一个合适的设计。A1 具有比 A4 更高的延迟和更低的吞吐量，因为 RDMA 到主机(①)相对较慢。A2 和 A3 的瓶颈是缓慢的主机- soc 通信，不适合卸载 KV 存储请求。

综合上述研究考虑，最佳的组合是 A4 和 A5，前几个客户机使用 A5，后续的客户机只用 A4，采用排队理论来估计 SoC 的容量和 RNIC 的能力，能够使效率最大化。

此外，使用 A5 带来了一个挑战，因为客户端不知道哪些值在 SoC 中缓存。尽管 A3 可以用作缓存丢失的后备路径，但它将导致显著的性能下降(参见图 11)。为了解决这个问题，本文提供了一个简单的解决方案:当发生缓存丢失时，SoC 将值的地址返回给客户端，然后客户端发出 READ 以相应地检索值，类似于 A4。在真实的倾斜工作负载中，缓存丢失是罕见的。

如图 12 所示，本文通过增加客户机的数量来绘制该图。A4 + A5 组合的吞吐量峰值为 68 M reqs/s，比 RNIC、A1 和 A4 分别提高 25%、36%和 12%。

之所以省略了 A2 和 A3，因为它们受到 SoC 内核的瓶颈，并且具有极低的峰值吞吐量。A4 + A5 的好处主要来自于利用更快的 SoC RDMA 和 SoC 内核来减少网络放大。

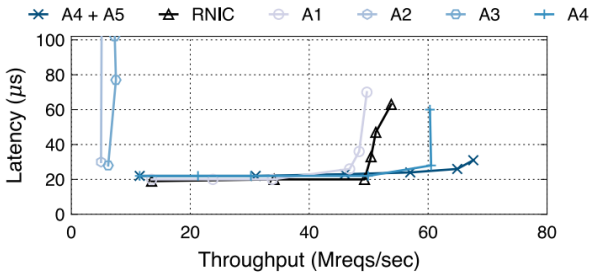

图 12 评估结果

5 总结

现代 smartNIC 发展迅速，已出现的 Bluefeild3 对 Bluefeild2 进行了全面的性能提高[6]。本文针对 bluefeild2 为例，研究其数据通路的同时使用，做到了性能上的提升，该工作具有相当的可扩展性。在 Battle of the BlueFields: An In-Depth Comparison of the BlueField-2 and BlueField-3 SmartNICs 一文中，对 Bluefeild 的两代表现进行了探究，并且分析了在 Bluefeild2 中进行的优化能否在 Bluefeild3 中同样生效，研究表明 Bluefeild3 在性能测试中具备的明显提升，在各种优化之下仍然能够具有较好的表现。本文的研究具备相当的可泛用性。

该文章表示，一方面，带宽本身的提高能够让数据的传输性能得到明显的提升，另一方面，更加强大的计算单元也能够让计算的卸载更加有效，本文的多数据通路并行思路不变的情况下，如何考虑在更加强大的计算单元基础上优化各种方案的配比，在我看来是相当值得探究的方向。

参 考 文 献

[1] Wei X, Cheng R, Yang Y, et al. Characterizing Off-path {SmartNIC} for Accelerating Distributed Systems[C]//17th USENIX Symposium on Operating Systems Design and Implementation (OSDI 23). 2023: 987-1004.

[2] Suresh K K, Michalowicz B, Ramesh B, et al. A Novel Framework for Efficient Offloading of Communication Operations to Bluefield SmartNICs[C]//2023 IEEE International Parallel and Distributed Processing Symposium (IPDPS). IEEE, 2023: 123-133.

[3] Guo A, Hao Y, Wu C, et al. Software-Hardware Co-design of Heterogeneous SmartNIC System for Recommendation Models Inference and Training[C]//Proceedings of the 37th International Conference on Supercomputing. 2023: 336-347.

[4] Guo A, Hao Y, Wu C, et al. Software-Hardware Co-design of Heterogeneous SmartNIC System for Recommendation Models Inference and Training[C]//Proceedings of the 37th International Conference on Supercomputing. 2023: 336-347.

[5] Schonbein W, Matsika T, Grant R E. A Lightweight Network Traffic Prediction Method for SmartNICs[C]//2023 IEEE International Conference on Cluster Computing Workshops (CLUSTER Workshops). IEEE, 2023: 68-69.

[6] Michalowicz B, Suresh K K, Subramoni H, et al. Battle of the BlueFields: An In-Depth Comparison of the BlueField-2 and BlueField-3 SmartNICs[C]//2023 IEEE Symposium on High-Performance Interconnects (HOTI). IEEE, 2023: 41-48.

附录 汇报记录

问题 1：在你所汇报的实验的最后一个案例之中，是如何体现出作者之前进行的研究的？

在文章最后的案例中，作者首先借鉴了之前的优化思路。

首先，通过限制门铃批处理的方式，作者理智地提高了部分实验的效率，其次，作者使用尽可能避免内部通信的方式尽可能降低了时延。但是值得注意到的是，在作者进行的优化过程中，作者只对传输效率进行了探究，但是针对这种实际的应用情况，是能够将计算的负载卸载到 smartNIC 上面的,因此能够在性能上产生明显的优化。因此才只使用方案四和方案五对系统的性能进行尽可能的提高。