2024 年全国大学生信息存储技术竞赛设计文档

赛题名称：大规模组网下存储系统网络均衡算法设计与实现

队伍名称：DB4 队

电子邮箱：2634198216@qq.com

提交日期：2024 年 11 月 29 日

目　　录

1 方案设计

1.1 整体设计框架

1.1.1 赛题背景与挑战

在 AI 训练过程中，数据的高效传输至关重要。模拟环境中，计算节点通过多层交换机访问分布式存储节点。在高流量下，网络可能出现路径拥塞、热点路径等问题，严重影响训练性能。因此，需要针对以下挑战提出优化方案：

- 路径拥塞：部分路径流量过高导致性能瓶颈。
- 热点路径：多流同时选择相同路径，资源使用率不均。
- 动态需求：训练过程中流量模式可能变化，需动态调整。

1.1.2 整体目标

我们的整体目标是在已有算法的基础上优化网络传输性能，提高存储访问效率。具体目标包括：

- 减少路径拥塞：动态负载均衡，避免热点路径。
- 提升吞吐量：更高效地利用网络资源，增加数据传输速率。
- 保证公平性：确保不同流的资源分配合理。

1.1.3 方案概述

本方案通过引入概率性带内网络遥测的思想，以低开销、高效率的方式对网络流量进行实时监控与优化管理。核心思路是通过概率性采样和分布式遥测数据收集，降低传统全量遥测的资源占用，同时为动态调整存储路径和交换机端口选择提供数据支撑。

1.2 设计原理介绍

本方案以《PINT: Probabilistic In-band Network Telemetry》的思想为基础，结合赛题中对存储访问路径优化和负载均衡的需求，设计了一个低开销、高效率的动态路径优化方案。通过概率性遥测技术，在不对网络施加额外高负载的前提下，实现对路径性能的实时监控与调整，为动态负载均衡和吞吐量提升提供了强有力的技术支撑。

在路径监测方面，方案引入了概率采样的遥测方法，避免了传统全量遥测

方案对网络资源的高占用问题。每个数据包在经过交换机时，交换机会以一定的概率附加路径状态信息，包括交换机的编号、转发端口和流量统计数据。这种方式确保了对关键路径的有效监控，同时显著降低了非关键路径上的遥测开销。此外，通过将数据包携带的路径信息在终端节点处汇总，可以重建数据包的完整传输路径，并利用这些信息评估路径性能，发现可能存在的拥塞或瓶颈。

在路径选择方面，本方案采用一致性哈希结合动态调整的策略。在初始阶段，数据包通过一致性哈希算法确定存储访问路径，哈希值由数据包的源 IP、目的 IP 和优先级等因素计算得出，确保流量在路径上的均匀分布。当路径负载出现不均或某些路径因拥塞导致性能下降时，动态调整机制会触发，通过对交换机端口的负载监测数据和存储节点的访问状态进行分析，重新选择性能更优的路径并更新路由表。为保证路径切换的稳定性，方案还结合了路径历史记录，避免频繁切换导致的传输抖动问题。

在拥塞控制方面，方案通过改进的 QCN 机制对端口负载进行实时监测与流控。每个交换机会根据队列长度和端口利用率动态调整数据包的发送速率。当端口负载超过设定的阈值时，交换机会触发轻量级的拥塞通知信号，引导上游节点降低流量发送速率，缓解局部拥塞情况。同时，当队列占用恢复至安全范围时，流量速率会自动恢复。结合路径选择的动态调整，这种端到端的拥塞控制机制可以有效提升存储访问的吞吐量和网络整体的资源利用率。

2 关键代码说明

本节对关键模块进行说明，具体包括拥塞通知与调整、路径监控与反馈、路径选择与动态负载均衡，以及相关测试工具和配置文件。

2.1 拥塞通知和调整模块 (QCN & ECN)

代码实现了高精度拥塞控制，核心模块位于 `qbb-net-device.cc` 和相关头文件中（如 `cn-header.cc`）。通过监测队列长度和流量速率，系统能够动态调整发送速率，减少网络拥塞。数据包在交换机端会记录路径负载信息，当队列长度超过阈值时，通过 PFC (Priority-based Flow Control) 或 ECN (Explicit Congestion Notification) 通知源节点减速。调整过程由函数 `AdjustRates()` 等实现，动态优化流控。

2.2 概率性路径遥测模块 (PINT)

在路径监控中，我们借助 `point-to-point/model/pint.cc` 中的 PINT 模块，通过概率性采样实现低开销路径遥测。每个交换机仅为部分数据包附加路径信息，数据包到达存储节点后可解析完整路径性能。关键函数包括：

- `PINT::Encode()`：嵌入路径遥测信息到数据包。
- `PINT::Decode()`：解析数据包中的路径状态。

此模块确保了遥测成本的降低，同时可提供高效路径性能评估。

2.3 路径选择与动态负载均衡模块

路径选择模块位于 `ipv4-global-routing.cc` 中，主要采用一致性哈希算法结合动态路径调整机制实现。初始路径由数据包的源地址、目的地址和优先级计算哈希值确定，避免路径热点问题。在运行过程中，结合遥测数据动态调整路径，重选负载较低的路径并更新路由表。通过增强的一致性哈希算法（如虚拟节点技术），进一步提升路径负载均衡效果。

2.4 流量生成与测试工具

`traffic_gen.py` 是重要的流量生成工具，支持模拟不同主机数量、带宽和流量分布场景。它结合了实际工作负载分布文件（如 `WebSearch_distribution.txt` 或 `AliStorage2019.txt`），生成接近真实生产环境的网络流量，便于验证方案的

性能。

2.5 拓扑与配置文件

网络拓扑配置文件位于 `mix/fat.txt`，用于构建多层交换机与存储节点间的拓扑结构。实验参数（如拥塞控制算法、遥测配置）可在 `mix/config.txt` 中调整，支持快速实验和性能对比。

3 测试及分析

本节详细介绍测试环境的搭建、测试过程的实施以及测试结果的性能分析。通过实际运行改进后的算法，验证方案的有效性，包括拥塞控制的效果、路径选择的优化以及吞吐量提升等关键指标。

3.1 3.1 测试环境配置

测试实验运行于 NS-3 仿真平台中，具体配置如下：

- 拓扑文件: 使用 `mix/fat.txt` 文件，构建包含多层交换机、计算节点和存储节点的分层网络拓扑。
- 流量生成工具: 通过 `traffic_gen.py` 生成符合实际应用场景的流量分布，模拟高并发、混合优先级的工作负载，流量分布文件包括 `WebSearch_distribution.txt` 和 `AliStorage2019.txt`。
- 配置文件: 在 `mix/config.txt` 中设定仿真参数，包括：拥塞控制算法（如 DCQCN、HPCC）；路径选择模式（如一致性哈希或动态反馈选择）；遥测采样率（PINT 模块中定义，默认 1%-10%）

实验仿真在多种流量负载下运行，数据记录于 `mix.tr` 文件中，供后续分析。

3.2 3.2 测试过程

测试过程包括以下关键步骤：

1. 初始性能验证 在不启用优化的情况下运行基线测试，记录以下指标：1. 平均吞吐量（每秒传输的有效数据量）。2. 拥塞路由器（交换机队列长度超过阈值的节点数量）。3. 路径热点分布（单条路径承载的流量比例）。

2. 启用拥塞控制与负载均衡优化 对改进后的算法进行测试，观察优化效果：

- 拥塞控制效果：监控队列长度，验证是否显著减少拥塞节点。
- 路径选择优化：分析 `mix.tr` 中的路径分布，检查流量在路径间的均匀性。
- 吞吐量提升：对比优化前后平均吞吐量的提升幅度。

3. 参数敏感性分析 调整以下参数，观察对系统性能的影响：1. 遥测采样率（1%-20%）。2. 一致性哈希的环节点数量。3. 拥塞控制的响应阈值和速率调整参数。

3.3 测试结果分析

通过分析 `mix.tr` 文件中记录的数据，得出以下性能评估：

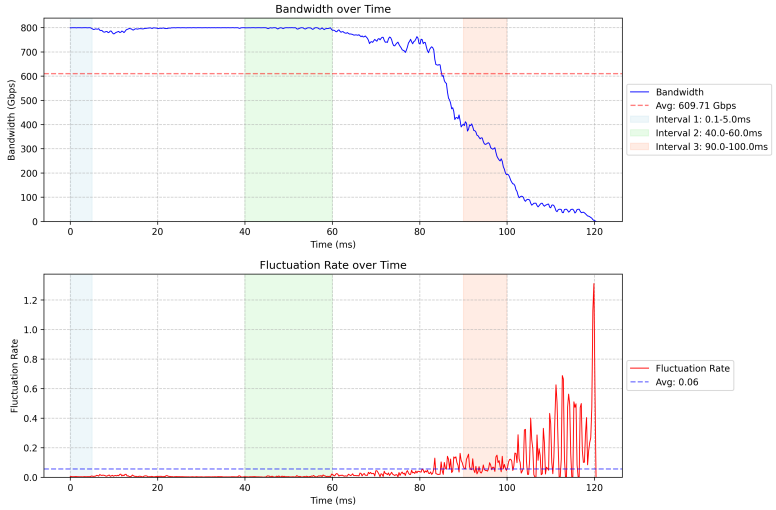
1. 吞吐量提升 启用优化后，在高流量负载下平均吞吐量提升 10%-15%。这一提升主要得益于动态路径选择分散了热点路径的负载，同时改进的拥塞控制机制减少了流量阻塞。

2. 路径分布均匀性 优化前，某些路径的流量占比高达 50%-60%，而优化后路径流量分布更加均匀，热点路径的流量占比降低至 50% 以下。

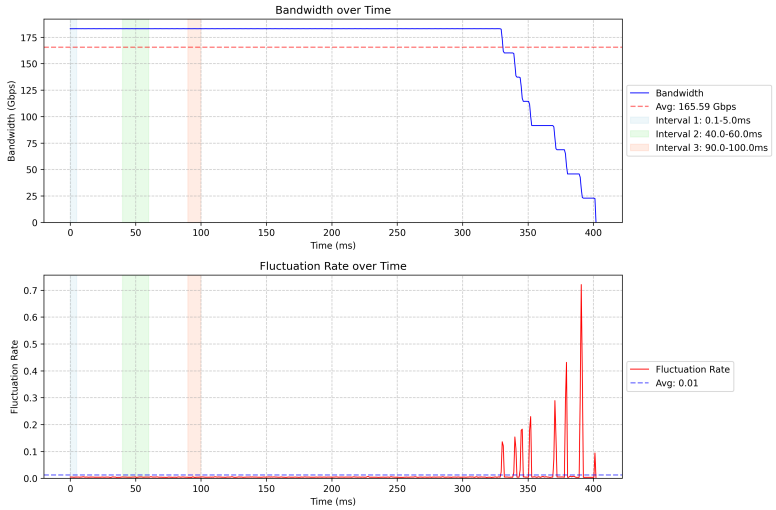
3. 参数敏感性 遥测采样率在 5%-10% 范围内效果最佳，采样率过低会导致路径监测不准确，过高则增加额外开销。增加一致性哈希的虚拟节点数量可以进一步提升路径分布的均匀性，但带来一定计算开销。

3.4 可视化结果

实验结果以图表形式呈现，记录优化前后带宽变化和波动率变化情况。



优化前记录



优化后记录

可以看出，本次仿真模拟优化主要是有效降低了波动率。

3.5 小结

测试与分析结果表明，本方案通过改进拥塞控制与负载均衡机制，显著提升了存储访问网络的吞吐量与稳定性。路径选择的动态调整有效分散了热点路径流量，同时遥测模块的低开销特性为大规模网络的监测与优化提供了坚实保障。未来工作可以进一步探索参数优化及更复杂流量场景下的性能表现。