Datacenter Networks
Project Progress Report

Krishangee Bora
Swapnil Pandey
Wasfi Momen

Introduction

Datacenter Networks (DCN) is one of the fastest-growing research topics due to the arrival of cloud computing and the future progress of edge computing. Previously, DCNs were restricted by hardware, geographical location, and strict function domains. With these aspects now unlocked, research for DCNs deal with managing the queue of several, multipurpose flows across wide geographical distances.

From the IEEE Infocom journal, we consider two conference papers on the topic of DCN: “Scheduling Jobs across Geo-Distributed Datacenters with Max-Min Fairness” by Chen et al. and “Multi-Tenant Multi-Objective Bandwidth Allocation in Datacenters Using Stacked Congestion Control” by Tian et al.. The former takes a theoretical approach in solving a minimization problem in order to “optimize all the job completion times” and an example network scheduler to handle the queuing of all tasks in a geo-distributed network. The latter tries to solve the bandwidth allocation problem by improving congestion control for use cases within an Application Driven Network (ADN). In the following sections, we explain the problem that exists with DCNs and the way these papers attempt to solve their respective problems.

Detailed Problem Statement

DCNs wish to provide services for multiple organizations and users with both hard constraints on bandwidth and soft constraints on service guarantees. Research of the general problem on how to compact large data sets, store them in hardware, and virtualize delivery is already integrated to solutions like Apache Spark, Google’s MapReduce [google paper], and structures like Hadoop Distributed File System which make up cloud computing. The problem is that the demand for large-scale applications now both have the means to get near real-time computability, but DCNs need to optimize for application-specific goals. Applications such as dense-data sensor networks, media content delivery, and social media “oversubscribe” to a DCN model and make it troubling to allocate bandwidth on a wide scale with Quality of Service (QoS).

Part of the solution is the actual selection of tasks in the queue to complete across a distributed network. In a datacenter network, there is much more intra-network communication than inter-communication. Compiled with the fact that datacenters now are globally distributed, this intra-communication cost needs to optimized. By taking the approach of a linear programming problem, a max-min fairness can allocate a finite share of resources, isolated from other network tasks, and reach an optimal completion time for all network tasks.

Another part of the solution relies in the unique congestion control of each application within a DCN in order to provide some form of service guarantee. TCP only considers a two party fairness for congestion control but does not consider the congestion occurring in other parts of the network. A data-link layer protocol can be used to limit congestion within and between DCN clusters for latency or deadline goals. A solution can also consider the costs of the edge network between a DCN and a user to optimize congestion even further for multiple flows both interior and exterior parts of a DCN.

Motivation

For example, Facebook's [Facebook DCN] datacenters ran into the issue of oversubscription in a Three-Tier DCN where clusters had huge communication costs compared to the cost of sending the information to the user. While this problem was alleviated with a different physical network structure, a software solution should slide.

Related Work

Plan of Attack

Our plan is to exploit vulnerabilities of the algorithms provided in the papers. The problems are dependent on each other since calculations must include communication costs at each step of the process.

One aspect to exploit could be the scalability aspect of the DCN. Both papers used available resources to model their respective problems and mention with additional hardware resources their solutions would run faster with either a better CPU or with additional servers. Utilizing GSU's HPC cluster could be used as an experimental setup to see if either paper's algorithm truly does stand when considering large-scale DCNs.

A better aspect is to compromise the solution within a particular use case and produce a non-optimal solution to the problem. For a multi-tenant and objective purpose DCN, injecting congestion at

Research Progress

Two papers mentioned constitute the two aspects of a general DCN that requires a service guarantee. Our current aim is to introduce a case which results in non-optimal solutions which degrade network performance. DCNs are very susceptible to huge loss if even one cluster fails to handle congestion or allocate resources accordingly.

Bibliography

https://ieeexplore.ieee.org/document/8056949 [Scheduling]
https://ieeexplore.ieee.org/document/8056944 [Multi-Tenant]
https://static.googleusercontent.com/media/research.google.com/en//archive/mapreduce-osdi04.pdf
http://delivery.acm.org/10.1145/2960000/2959075/p617-wang.pdf?ip=73.137.123.74&id=2959075&acc=PUBLIC&key=AA86BE8B6928DDC7%2E44E017FFD298019C%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&__acm__=1552248891_a30d899d1c70dfb2dcd8a31d5f3dc944 [ADN]

https://code.fb.com/production-engineering/introducing-data-center-fabric-the-next-generation-facebook-data-center-network/ [Facebook DCN]