

Reducing Time of Application Migration Across Datacenters Using Load Balanced Parallel TCP

Wongsatorn Thongthaworn, Prapaporn Rattanathamrong
Thammasat University

Motivation

- Application migration across data centers is needed by organizations in many scenarios, such as when deploying an updated application from a test site to a production site, or when a data center is damaged and cannot maintain current services due to disasters or other reasons.
- The problem of application migration is downtime for validating data to destination data center.
- We are particularly interested in existing work (Kang, 2014) that improving performance of application migration when a number of applications to be migrated across data centers is large. By performs parallel application migration on wide-area network environment to maximize number of successful application migration operation in disaster scenario.
- We also interested in load balancing end-to-end migration requests that can assist in maximizing the achieved total bandwidth, and resulting in lower total migration time.

Problem Statement



- The problem statements are defined into 2 parts.
 1. What is the **appropriate number of parallel migration requests** that can minimize total migration time ?
 2. What is the **migration request pairs between hosts** in both data centers that can balance connection load of all hosts ?

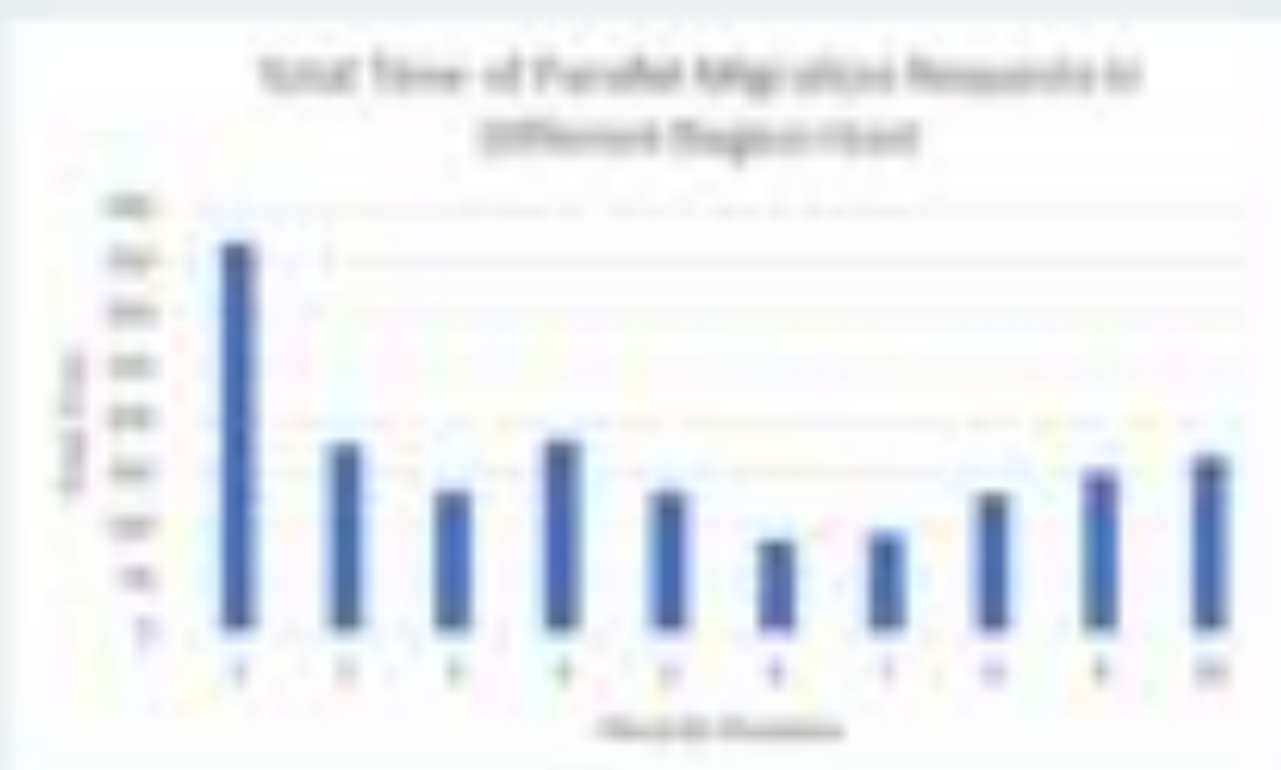
Proposed Method



Our ongoing research proposes a middleware for managing application migration requests consisting of 3 parts:

- 1) a **parallel window controller** inspired by TCP congestion control algorithm (previously proposed by Kang). This controller manages the number of parallel migration using network bandwidth as feedback.
- 2) a **load balance scheduler** that manages the application migration requests from multiple hosts. The scheduler uses a load balance algorithm based on optimal semi matching for parallel migration.
- 3) a **migrator coordinator** that fulfills parallel application migration requests by coordinate between the source and destination datacenters according to the controller and scheduler.

Preliminary Result



This preliminary experimental use iperf command to find a number of appropriate migration requests. A source data center locates at Singapore, and a destination data center locates at Sydney. This experiment uses iperf 1 to 10 parallel streams to represent as migration requests, and send 2GB files to represent as virtualization unit.

The result show that an increasing in number parallel streams, the total migration time is decrease until 6 parallel streams. After that, if parallel streams are more than 6 streams, total migration time is increase because of network congestion. Therefore, 6 parallel streams is appropriate migration requests.

Future Work

From the congestion problem in parallel application migration. We will implement parallel window controller to manages the number of parallel migration request properly. We also implements application migration on multiple hosts by using and load balance scheduler.

The further implementation are planned to improve application migration time in real enviornment.