Chapter 5
MINTCar:
A Tool Enabling Multiple Source Multiple Destination Network Tomography

Laurent Bobelin
CNRS, France

ABSTRACT

Discovering a network topology and inferring its performances for the client/server case is a well known field of study. However, client/server model is no longer accurate when dealing with Grids, as those platforms involve coordinated transfers from multiple sources to multiple destinations. In this chapter, we first review existing work, introduce a representation of the inferred knowledge from multiple sources and multiple destinations measurements that allows to obtain a well-posed problem, algorithms in order to reconstruct such a representation, a method to probe network, and give some experimental results.

INTRODUCTION

Nowadays institutional grid testbeds are composed of thousands of computing and data storage resources spread worldwide. Connectivity is ensured using either the Internet, or high bandwidth.delay networks such as GEANT (The GEANT website, retrieved 2010) in Europe, TeraGrid (The TeraGrid website, retrieved 2010) or Internet 2 (The Internet2 website, retrieved 2010) in US.

Data-intensive grid applications or even grid middleware itself usually deploy software and resources dedicated to bulk data transfer. For example, EGEE (The EGEE website, retrieved 2010) project uses a hierarchy of tiers. In such a hierarchy, each tier is a data storage center. Tier-0 is located close to the experiment place (for EGEE, at CERN, where data is produced). Tier-1 are national or institutional centers, tier-2 are located close to large computing centers, and tier-3 are located in labs. Tier-0 communicates to tier-1, tier-1s can communicate with every tier-1s and to a subset of tier-2s and tier-2s communicate to a subset of tier-2s and a subset of tier-3s. In such a case, the data transfer paradigm is no more a client/server one: each host is a source, a destination, or both, and each source communicates to a subset of destinations.

DOI: 10.4018/978-1-61350-110-8.ch005

This overlay network is built on top of the real network. It implies that independent logical links can share some physical links. Therefore, it is mandatory to know capacity and topology of the underlying network in order to optimize communications between tiers. If not, some logically independent transfers may compete for the same physical network resource. Unfortunately, most of the time, physical topology is unknown. Moreover, existing monitoring tools like NWS (Wolski, Spring, & Hayes. 1999) or WREN (Lowekamp & al, 2003) only allow modeling basic interactions between transfers.

Internet topology discovery can be done using tools like traceroute (Jacobson, 1989) or traceroute-like measurements. The resulting topology is unlabeled. It is formed by matching IP address of network equipments belonging to the different observed paths. This method has drawbacks: multiple network router interfaces, as well as non-responding equipments, can lead to inaccurate results, and enforce to use dedicated techniques in order to bypass those problems and discover the topology, as in (Viger et al, 2008). Higher-level discovery can also be done using AS-level information provided by ISP using Looking Glass servers, as in (Subramanian et al, 2002). Anyhow those methods suffer the lack of relationships with performances of the links, as they only discover the physical links but not their capacities. In order to retrieve more relevant information, one can use techniques based on monitoring information of links load provided by routers in order to discover the topology to reconstruct the path of flows crossing the network (Zhang et al, 2003).

But all those methods use information that network administrators may not authorize access to. As a grid application runs on hosts owned by organizations applying different security policies, using such tools is most of the time unrealistic. Using application level measurements is then the most efficient way to discover the topology and infer network capacities. This is known in the literature as network tomography (Vardi, 1996).

NETWORK TOMOGRAPHY

Since the last decade, network tomography for the client/server case has been widely studied (see (Castro et al, 2004) for an overview). Resulting topology is a tree where each edge represents a set of physical objects. The root is the server, leaves are clients and inner nodes are disjunction points of paths between the server and its clients. Edges can sometimes be labeled with the capacity of routers and wires belonging to the sub-path represented by it. Packet train based techniques are used to infer disjunction point for a pair of path. Probing is done for each pair of paths. Reconstruction is done most of the time using statistical techniques to estimate likelihood. Roughly speaking, it consists to collapse 2 or more disjunction points into one when capacities of the sub-paths leading to those inferred points are similar. Unfortunately, this method relies on the assertion that the resulting topology is a tree. But a tree cannot characterize the network when multiple sources and multiple destinations are involved, as stated in (Bu, Duffield, & Lo Presti, 2002). New probing techniques, reconstruction algorithms and models must then be developed for this topology discovery and performances inference.

Network tomography is an inverse problem, as stated in (Vardi, 1996). Those problems consist in inferring the structure and characteristic of a system (i.e. the network) based only on its reaction to stimuli (i.e. end-to-end measurements). Canonical inverse problem solving consists in three steps:

- Find an accurate model for solutions, which may enable to pose the problem as a well-defined one (see (Hadamard, 1923) for a definition of it),