Journal of Network and Computer Applications 34 (2011) 252–261

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Randomized load balancing strategies with churn resilience in peer-to-peer networks

Song Fu [a,*], Cheng-Zhong Xu [b], Haiying Shen [c]

[a] Department of Computer Science and Engineering, University of North Texas, Denton, TX 76207, USA
[b] Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA
[c] Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA

A R T I C L E I N F O

Article history:
Received 27 August 2009
Received in revised form
27 April 2010
Accepted 14 July 2010

Keywords:
Peer-to-peer systems
Randomized probing
Load balancing
Heterogeneous and bounded node capacity
Churn

A B S T R A C T

The objective of load balancing in peer-to-peer (P2P) networks is to balance the workload of peer nodes in proportion to their capacity so as to eliminate performance bottlenecks. It is challenging because of the dynamic nature in overlay networks, the time-varying load characteristics, and the inherent load imbalance caused by consistent hashing functions. It is known that simple randomized load balancing schemes can balance load effectively while incurring only a small overhead in general parallel and distributed computing contexts. Existing theoretical works which analyze properties of randomized load balancing schemes cannot be applied in the highly dynamic and heterogeneous P2P systems. In this paper, we characterize the behaviors of randomized load balancing schemes in a general P2P environment. We extend the supermarket model by investigating the impact of node heterogeneity and churn on load distribution in P2P networks. We prove that by using d-way random choices schemes, the length of the longest queue in a P2P system with heterogeneous nodes and churn for $d \geq 2$ is $c * \log\log n/\log d + O(1)$ with high probability, where c is a constant. Our results have wide applicability and are of interest beyond the specific applications.

1. Introduction

Peer-to-peer (P2P) networks have become, in a short period of time, one of the fastest growing and most popular Internet applications. An important class of P2P overlay networks is the distributed hash tables (DHTs) that map keys to peer nodes based on a consistent hashing function. Representatives of DHTs include Chord (Stoica et al., 2003), Pastry (Rowstron and Druschel, 2001), Tapestry (Zhao et al., 2004), CAN (Ratnasamy et al., 2001), and Cycloid (Shen et al., 2006). In a DHT, each node and key has a unique ID, and a key is mapped to a node according to the DHT definition. The ID space of a DHT is partitioned among nodes and each of them is responsible for the keys whose IDs are located within its ID space. An important goal in designing DHTs is to achieve a balanced partition of the hash space among peer nodes. It is often desirable that each node takes responsibility for a portion of the hash space which is proportional to its processing capacity, measured in terms of its processor speed, available bandwidth, and/or storage capacity. This property should be sustained as nodes join and leave the system. A similar goal is desirable in unstructured P2P networks as well.

However, consistent hashing (Stoica et al., 2003) produces a bound of $O(\log n)$ imbalance of key distribution among peer nodes, where n is the number of nodes in a P2P network. Things are even worse in unstructured P2P systems, where no commonly accepted load distribution mechanism is supported. In addition, users may query geographically adjacent nodes and those that have popular files. These lead to even more imbalanced distribution of workload in the network. When a node becomes overloaded, it cannot store any additional files or respond to user queries any more, which affects the system utilization and user satisfaction. To balance load among peer nodes in a network, lightly loaded peers should be selected to store files or serve queries. Load balancing in P2P networks is an important topic and many related works have been conducted recently (Tewari and Kleinrock, 2006; Shen and Xu, 2006; Zheng et al., 2006; Godfrey and Stoica, 2005; Bienkowski et al., 2005).

It is known that simple randomized load balancing schemes can balance load effectively while incurring only a small overhead in general parallel and distributed computing environments (Xu and Lau, 1997), which makes such schemes appealing for practical systems. The approach of multiple random choices in P2P networks was used in Shen and Xu (2006), Godfrey and Stoica (2005), Kenthapadi and Manku (2005), Zhu and Hu (2005), and Byers et al. (2003). Several peer nodes are probed and the one with the least load is selected to store a file on it or service a user query. Random choice algorithms are scalable and they require a small number of control messages and data structures (Shen and Xu, 2006; Byers et al., 2003). More importantly, they work well in P2P systems with churn, a situation where a large percentage of

* Corresponding author.
E-mail addresses: song.fu@unt.edu (S. Fu), czxu@wayne.edu (C.-Z. Xu), shenh@clemson.edu (H. Shen).

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.07.011

nodes join and leave continuously and rapidly, which leads to an unpredictable network size.

To theoretically analyze the effectiveness of random-choice schemes for balancing workload in distributed systems, researchers have proposed several approaches. Azar et al. (2000) introduced a *layered induction* method, by which the random choice problem was modeled by balls-and-bins. It provides nearly tight results. The *witness tree* approach was proposed by Cole et al. (1998) to tackle the random-choice problem. The probability of a certain event can then be bounded by the probability of occurrence of a witness tree. Generally speaking, the witness tree approach involves the most complexity, and it has been proved to be the most challenging one in obtaining tight results. The *fluid limit* model (Mitzenmacher, 2001; Mitzenmacher et al., 2002) characterizes system dynamics by constructing a family of differential equations, which makes this approach simple yet flexible. In cases where the system dynamics can be modeled by this method, the differential equations generate very accurate numerical results.

However, these theoretical works only analyzed systems with compute nodes having homogeneous and infinite capacities. Moreover, node churn, a defining characteristic of P2P networks, is not modeled by the existing approaches. As a result, we cannot conclude directly that the performance bounds derived in those works will still be valid in P2P networks. In this paper, we analyze the dynamic behavior of randomized load balancing algorithms in a general P2P network, where peer nodes join and leave the network at runtime and they have heterogeneous and bounded capacities.

We model a dynamic P2P system as follows. Load queries arrive as a Poisson stream to a collection of n peer nodes, where n is a random variable reflecting node churn. Nodes are heterogeneous with bounded capacity. For each query, a number of d nodes are chosen independently and uniformly at random, and the query is queued at the node which currently accommodates the fewest number of queries. We refer to such multiple choice processing as d-way probing. Queries arrive to peer nodes at rate λ which is relative to the node population. They are served according to the FIFO protocol, and the service time for a query is exponentially distributed with mean 1.

We extend the supermarket model (Mitzenmacher, 2001) to formulate behaviors of the preceding dynamic system in general cases and to derive system properties. We are interested in characterizing the average response delay and the maximum load on active nodes. However, quantifying these metrics in a general P2P system is nontrivial. It is difficult to find closed-form solutions to the differential equations describing the system dynamics, after we remove certain constraints, such as static system configuration, and homogeneous and infinite node capacities. Instead of solving the equations directly, we study the lower and upper bounds of state variables at equilibrium points with reference to those in special cases. Based on these bounds, we quantify the average response delay and the maximum load, and derive the following properties of d-way randomized probing in P2P networks.

Theorem 1. *For any fixed time interval I, the expected time that a query spends in a dynamic P 2P system with d-way randomized probing* ($d \geq 2$), *denoted by* $T_d(\lambda)$, *over interval* [0, I] *satisfies that* $T_d(\lambda)/\log T_1$ *is close to* $\alpha(1/\log d)$, *for* λ *close to* 1, *where* α *is a constant whose value depends on capacities of peer nodes and the change rate of the node population.*

Theorem 2. *For any fixed time interval I, the length of the longest queue in the dynamic P 2P models with d-way randomized probing* ($d \geq 2$) *over interval* [0, I] *is* $c \log\log n + O(1)$ *with high probability* $(1-O(1/n))$, *where c is a constant depending on capacities of peer nodes, d and the arrival rate of queries, and the form of the* $O(1)$ *term depends on I and some constants.*

The theorems show that two-way randomized probing is asymptotically superior to one-way probing. However, by increasing the number of choices further, efficiency of the load balancing algorithms does not improve significantly. These results are consistent with the findings by the supermarket model (Mitzenmacher, 2001) in special case where the number of servers is fixed, and their capacities are homogeneous and infinite. We also conduct experiments on a P2P network. Experimental results confirm the correctness of our findings. Although the randomized probing algorithms for load balancing are designed and analyzed within the context of P2P networks, the results have wide applicability and are of interest beyond the specific applications.

The remainder of this paper is organized as follows. The basic supermarket model is briefly described in Section 2. Section 3 formulates d-way randomized probing in P2P networks. We investigate the influences of node capacity in Section 3.1 and node churn in Section 3.2. By analyzing the equilibrium points of a P2P system, we quantify the expected time of a query in the system and the length of the longest queue among peer nodes. Experimental results are shown in Section 4. Section 5 presents the related work. Conclusions and remarks on future works are presented in Section 6.

2. Basic supermarket model

A load balancing scheme distributes user requests or storage loads among compute nodes and avoids hot spots. Mitzenmacher (2001) presents a supermarket model based on differential equations to analyze both static and dynamic load balancing strategies. In this section, we briefly describe this model, and in the subsequent sections we will present our extension of the model to formalize randomized probing algorithms for load balancing in general P2P systems.

Supermarket model analyzes properties of randomized load balancing strategies in a special distributed environment. In this environment, user requests arrive as a Poisson stream at a collection of servers. For each request, some constant number of servers are chosen independently and uniformly at random with replacement from the servers, and request waits for service at the server which currently accommodates the fewest requests. Requests are served according to the FIFO protocol, and the service time for a request is exponentially distributed with mean 1. Three underlying assumptions (Mitzenmacher, 2001) are: (a) unbounded server capacities; (b) static server configuration; and (c) homogeneous servers. The author derived the average time of a request staying in the system and the maximum workload of the servers by solving the differential equations of system states.

3. Randomized probing in general P2P systems

In large-scale P2P systems, a great number of peer nodes share resources and send queries to each other. More often than not, they have heterogeneous configurations of storage capacity and processing speed. In addition, dynamics is a defining characteristic of P2P networks, with nodes joining and leaving the network frequently. Load balancing in such large-scale and dynamic distributed environments is challenging. Obtaining the capacity information of all active nodes before dispatching jobs to the most