

A Fuzzy Intra-Clustering Approach for Load Balancing in Peer-to-Peer System

Rupali Bhardwaj¹, V.S.Dixit², Anil Upadhyay³

¹Department of Master of Computer Applications, Ajay Kumar Garg Engineering College, Ghaziabad, UP, India

²Department of Computer Science, Atma Ram Sanatan Dharma College, Delhi University, New Delhi, India

³Department of Applied Mathematics & Science, Bipin Chandra Tripathi Kumaon Engineering College, Dwarahat, Almora, India

(Received October 23, 2011, accepted December 2, 2011)

Abstract. Load balancing algorithms goal is to keep all nodes normally loaded through migration of modules from heavy loaded nodes to lightly loaded nodes. In addition, load balancing must involve low communication overhead and respond quickly to load imbalance in the system. In previous load balancing algorithms, classification of node as heavy, light or normal loaded node is done by using concept of threshold level, which is fixed and predefined. Now, today scenario is to change the status of nodes dynamically according to the state of system. So that, in this paper we proposed an algorithm for load balancing using fuzzy clustering; which improves the performance of the system without pre-defining the threshold values. The proposed algorithm is compared with other existing algorithms and is found to be fast and efficient in reducing load imbalance in Peer to Peer system.

Keywords: heavy weighted, light weighted, fuzzy system, clustering, load balancing

1. Introduction

A Peer to Peer (P2P) system in which every participating node acts both as a client and as a server (servant) and share a part of their own hardware resources such as processing power, storage capacity or network bandwidth. A P2P system will have a number of peers (nodes) working independently with each other. Each node is classified either as heavy or light weighted node using fixed threshold level. The use of single-threshold value may lead to a useless load transfers and make the load balancing algorithm unstable because a node's status may be light weighted when it decides to accept a remote process, but its status may be heavy weighted whenever the remote process arrives. Therefore, a light weighted node becomes heavy weighted node and will again invoke load balancing algorithm to change its status. (Alonso and Cova, 1988) proposed a double-threshold levels algorithm to reduce the instability of the single-threshold level policy; however those two threshold levels are fixed and predefined. One desired feature of load balancing algorithm is change the status of nodes dynamically. Some load balancing algorithms (A Rao & Stoica, 2003; G. Shivaratri & Singhal, 1992; Rajeev Gupta & Prabha Gopinath, 1990; Sonesh Surana & I Stoica, 2004) use fixed threshold levels for load balancing. Load balancing using fuzzy system is a natural extension of double threshold level approach and it improves the performance without pre-defining the threshold values. The remainder of the paper is organized as follows: Section 2 briefly reviews fuzzy logic controller and clustering. The proposed algorithm using fuzzy clustering is presented in section 3. Section 4 & 5 describes existing load balancing algorithms such as Round Robin and BID algorithm and experimental study. Finally, we conclude the paper in section 6.

2. Background

2.1. Fuzzy Logic Controller

Fuzzy logic is a powerful mathematical tool in representing linguistic information and is very useful to solve problems that do not have a precise solution. The architecture of the fuzzy logic controller is shown in figure 1, it includes four components: Fuzzifier, Inference Engine, Fuzzy knowledge rule base and

Defuzzifier (A. Karimi & Saripan, 2009; Ally E. El-Bad, 2002; M.C.Huang & Vairavan, 2003 ; T. J. Ross, 1995)

- Fuzzification

It converts the crisp input value to a linguistic variable using the membership functions stored in the fuzzy knowledge base.

- Fuzzy knowledge rule base

It contains the knowledge on the application domain and the goals of control.

- The inference engine

It applies the inference mechanism to the set of rules in the fuzzy rule base to produce a fuzzy set output.

- Defuzzification

It converts the fuzzy output of the inference engine to crisp value using membership functions similar to the ones used by the fuzzifier.

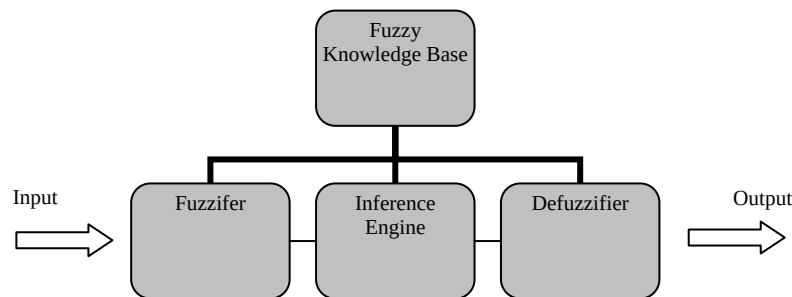


Figure 1: Architecture of fuzzy logic controller

2.2. Clustering

Clustering is a tool for data analysis, which solves classification problems. Its motive is to distribute cases (people, objects, events etc.) into groups, so that the degree of association to be strong between objects of the same cluster and weak between objects of different clusters (Clark F. Olson, 1996). Cluster analysis organizes data by abstracting underlying structure either as a grouping of individuals or as a hierarchy of groups. In brief, cluster analysis groups data objects into clusters such that objects belonging to the same cluster are similar, while those belonging to different clusters are dissimilar. A cluster is therefore a collection of objects which has “similarity” between them and has “dissimilarity” to the objects belonging to other clusters. We can show this with a simple graphical example (Osmar R. Zaiane, 1999).

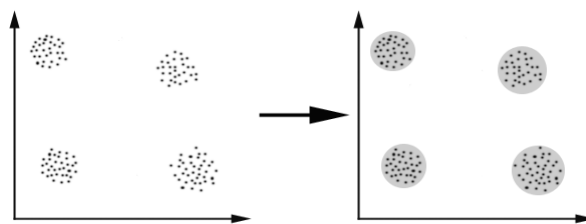


Figure 2: organizing objects into groups whose members are similar in some way

Clusters are formed by two methods-“distance –based clustering”, two or more objects belong to a given distance (in this case geometrical distance). Another kind of method is “conceptual clustering”, two or more objects belong to the same cluster if this one defines a concept common to all that objects. In other words, objects are grouped according to their fit to descriptive concepts.

3. Fuzzy-Clustering Load Balancing Approach

Assuming the poll of dynamic cluster heads in hierarchical method, based on dynamic cluster node; dynamic cluster heads are constructed by using the terms in node and cluster heads will change when different cluster nodes are merged. The degree of similarity between node clusters is calculated based on these dynamic node cluster heads. Dynamic clusters are formed by basically two things-

- (1) Similarity between two data method.