Load Balancing Using Load Threshold Adjustment and Incentive Mechanism in Structured P2P Systems

Kyoungsoo BOK[†], **Jonghyeon YOON**[†], **Jongtae LIM**[†], *Nonmembers*, *and* **Jaesoo YOO**[†a)], *Member*

SUMMARY   In this paper, we propose a new dynamic load balancing scheme according to load threshold adjustment and incentives mechanism. The proposed scheme adjusts the load threshold of a node by comparing it with a mean threshold of adjacent nodes, thereby increasing the threshold evenly. We also assign the incentives and penalties to each node through a comparison of the mean threshold of all the nodes in order to increase autonomous load balancing participation.
key words: *structured P2P, load balancing, load threshold, incentive*

1. Introduction

A structured P2P system is organized according to some specific topology [1], [2]. In the structured P2P system, when popular information occur, a load may be concentrated on a specific node. Recently, various studies have been conducted for load balancing in P2P systems. In [3], a diffusive load balancing proposed is proposed for parallel computing system. The each node collects the load state of its neighbors and decides load transfers between its neighbors. In [4], an incentive mechanism was proposed for assigning an incentive or penalty to increase nodes' participation in load balancing. However, this incentive mechanism only encouraged nodes to participate in load balancing that was not mandatory. Thus, many nodes were still reluctant to participate in load balancing, thereby reducing the load balancing performance.

   We proposed a load balancing scheme considering only load threshold adjustment in the structured P2P in our previous work [5]. This paper is an extended version of the existing scheme [5] to improve load balancing performance by using the automatic adjustment and incentive mechanism. The load threshold of each node is adjusted through the mean value of the thresholds of adjacent nodes. Furthermore, by giving incentives to nodes that actively participate in load balancing or penalties to nodes that participate reluctantly based on a comparison with the mean load threshold of all the nodes, nodes that are reluctant to participate in load balancing are induced to participate in load balancing more actively.

2. Proposed Load Balancing

2.1   Overall Procedure

In this paper, we propose a new load balancing scheme to evenly increase nodes' thresholds and encourage nodes to participate in load balancing actively. Since the P2P system is used for data sharing purposes, the load is determined by the write size for storing the shared data and the size of the read for providing the shared data. In this paper, the load of a node is defined as the sum of the size of the data requested to be read and the size of the data to be written in the time interval. Each node calculates the mean threshold of its own adjacent nodes. By comparing the mean threshold with its own threshold, a node increases its own threshold if its own threshold is lower than the mean threshold. Therefore, each node can reduce its threshold deviation with adjacent nodes, since the thresholds between nodes are uniformly adjusted. Once the thresholds are adjusted, each node compares the mean threshold of all the nodes to its own threshold. If its own threshold is higher than the mean threshold, an incentive is given. Otherwise, a penalty is given. Incentives or penalties affect each node's intention of load balancing, thereby encouraging nodes to have high thresholds.

   A load state is divided into three phases through two thresholds, as shown in Fig. 1. The higher threshold out of the two thresholds is called the over-loaded threshold T_o and the lower threshold is called the under-loaded threshold T_u. P2P systems are organized autonomously by user participation. Therefore, the thresholds T_o and T_u are voluntarily provided by the user according to their capabilities. Each node can update the threshold at any time as needed. The over-loaded is defined as a load that is higher than T_o, while middle-loaded is defined as a load that is higher than Tu and lower than T_o. The under-loaded is defined as a load that is lower than T_u. As such, a load state is divided into
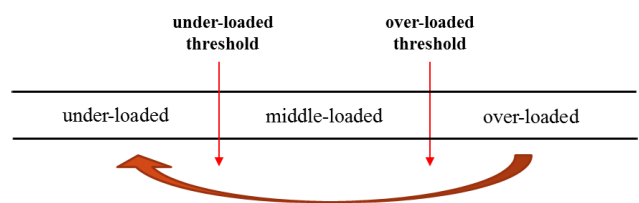

Fig. 1   Three-phase load states according to load ratio based on two thresholds

three phases, thereby alleviating load balancing complexity by excluding middle-loaded nodes through performing load balancing only from over-loaded to under-loaded nodes.

Since load balancing is conducted based on load state according to the threshold during load balancing and the circumstances of the changing thresholds of nodes. Thus, information about each node threshold is needed to adjust the threshold. Moreover, information about the adjacent node list is also needed to know adjacent nodes.

If each node manages load information about the node locally, it takes a lot of time and communication costs to check the load status of the nodes. Load information about each node is managed by multiple nodes, called load information management nodes similar to [4]. Each load information management node stores the load information for nodes within a specific load range. Each node stores the read and write request information for the data locally and then calculates its load status. When the load status of a node is changed, its load information about its load status, threshold, and incentive information are sent to the corresponding load information management nodes via the TCP protocol for stable information collection.

2.2 Threshold Adjustment

We increase the threshold of each node automatically, thereby improving load balancing performance as a result of increased load balancing participation. A large number of nodes in P2P systems mainly aim to improve their own performance. As a result, nodes tend to set a low threshold to receive fewer loads from other nodes during the load balancing process. In load balancing schemes using a threshold, the higher the threshold is set, the better the load balancing efficiency. This is because a node can maintain a middle-loaded state longer when the same load is applied if the over-loaded threshold is set higher so that the node can process its own load without sending the load to other nodes.

Figure 2 shows an example of the load balancing process according to a threshold. In the Fig. 2 (a), N1 has an 83 load. Assuming that N1 sets its over-loaded threshold as 92, as indicated by the dotted line, N1 processes its own load, because its load state is middle-loaded. On the other hand, assuming that N1 sets its over-loaded threshold as 69, as indicated by the dotted line, N1 sends its own load to other under-loaded nodes, such as N6 and N11, because its load state is lower than the over-loaded threshold value. Similarly, the lower the under-loaded threshold is set, the longer the node can maintain its under-loaded state so that it can receive loads from over-loaded nodes to facilitate better load balancing. The Fig. 2 (b) shows an example of the load balancing process according to the under-loaded threshold. Assuming that the under-loaded threshold is set to 44, as indicated by the dotted line in Fig. 2 (b), the node sets a higher under-loaded threshold value than it has currently. This means it is in an under-loaded state and available to receive loads from other over-loaded nodes, thereby facilitating load balancing. On the other hand, assuming that
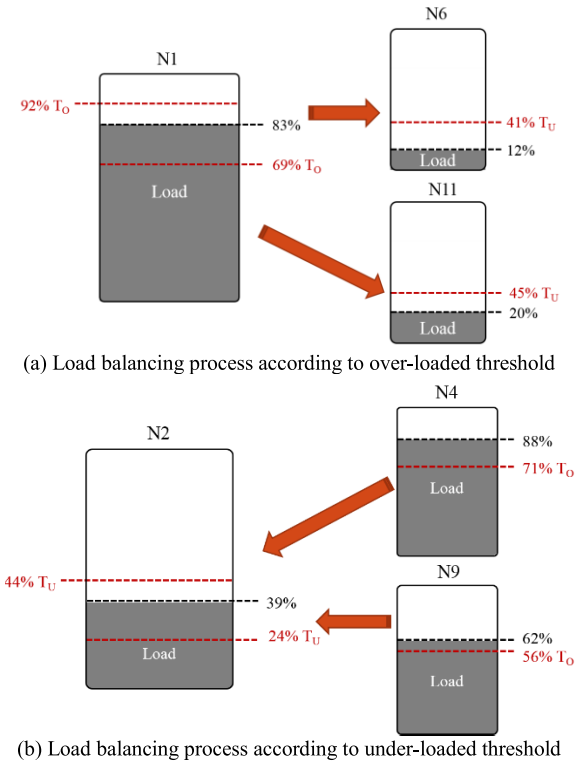

(a) Load balancing process according to over-loaded threshold

(b) Load balancing process according to under-loaded threshold

Fig. 2 Load balancing process according to thresholds

the node set its under-loaded threshold as 24, as indicated by the dotted line, the node cannot receive loads from other over-loaded nodes, because its load state is middle-loaded. As shown above, if the threshold of each node is set high to make nodes participate in load balancing actively, it will have a positive effect on load balancing.

Since load balancing is conducted based on load state according to the load threshold, the load threshold adjustment is performed when the load state of a node becomes over-loaded or under-loaded. Each node adjusts its own threshold through the mean threshold of adjacent nodes. Each node calculates the mean thresholds of adjacent n predecessor nodes and n successor nodes including itself. T_{un} and T_{on} are the mean thresholds of T_u and T_o. If a node's T_u or T_o is smaller than the calculated T_{un} or T_{on}, then the node increases its threshold using Eqs. (1) and (2), where α is a constant value defined by the P2P system. On the other hand, if a node's T_u or T_o is equal to or larger than the calculated T_{un} or T_{on}, the node's threshold is maintained without change, because if the node's threshold is lowered due to the lower mean threshold of adjacent nodes, it can lower the overall network threshold.

$$T_u = \left(1 + \frac{\alpha}{100}\right) T_u \qquad (1)$$

$$T_o = \left(1 + \frac{\alpha}{100}\right) T_o \qquad (2)$$

2.3 Incentive and Penalty

After the automatic threshold adjustment, an increase in the

threshold of each node is encouraged through an incentive or penalty. Both automatic threshold adjustment and active threshold adjustment are encouraged to increase the load balancing participation of nodes. To assign incentives and penalties based on the load threshold of each node, first, T_u and T_o of all nodes are calculated, and the mean thresholds T_{ua} and T_{oa} are obtained through the mean of T_u and T_o of all nodes.

All nodes in the P2P network compare their T_u and T_o with T_{ua} and T_{oa}. If their T_u and T_o are larger than T_{ua} and T_{oa}, the incentive β is assigned by Eq. (3). On the other hand, if their own T_u and T_o are smaller than T_{ua} and T_{oa}, the penalty $-\beta$ is assigned. The incentive and penalty information is also managed in the load management node as the load information of each node, which may affect the load balancing process.

$$\beta = Min(|T_u - T_{ua}|, |T_o - T_{oa}|) \qquad (3)$$

Each node receives an incentive or a penalty during the load balancing. A node actively participating in load balancing increases its load by the incentive β. That is, the load of the node that receives the incentive is increased, and its request can be distributed to other nodes. On the other hand, a node that passively participates in load balancing receives a penalty and reduces its load by $-\beta$. Therefore, the request of another node is performed instead.

3. Performance Evaluation

To evaluate the performance of the proposed load balancing scheme, we conducted simulations through Java programs running with the Windows 7 operating system. To show the superiority of the proposed scheme, we compared the proposed scheme with the incentive mechanism [4]. In the experiment, 1,000 nodes were used, and the number of additional added nodes was set to 50~200. We performed the evaluation by setting α to 10% and β to the same value as the incentive information. If load is concentrated on a particular node, system response time and network performance may be degraded. However, the actual response time and network performance may vary depending on how the P2P system is configured and the type of service. Therefore, we evaluate performances in terms of load distribution, threshold distribution, and node load state.

Figure 3 shows the threshold distribution according to the number of nodes. The threshold distribution shows the load thresholds of the nodes after the load balancing process. The threshold of each node was increased through the proposed scheme. Numerically, the mean thresholds of all nodes were increased in the proposed scheme compared to the existing scheme. In addition, the deviation of the threshold between nodes was also decreased because the threshold increase adjustment of thresholds below the mean threshold of adjacent nodes.

Figure 4 shows the comparison of threshold changes before and after additional node insertion when additional
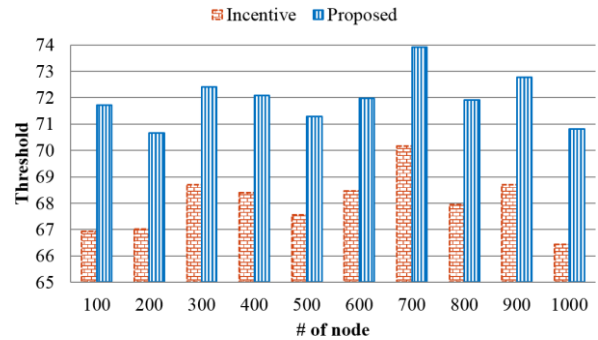

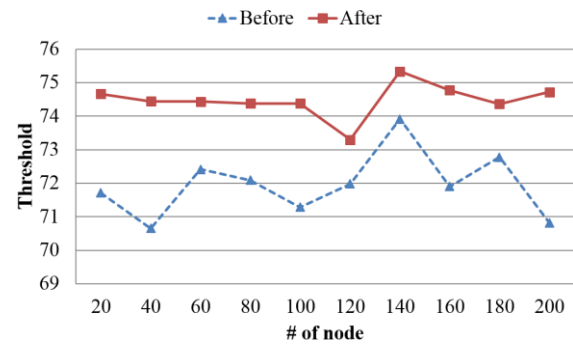**Fig. 3** Threshold distribution according to the number of nodes

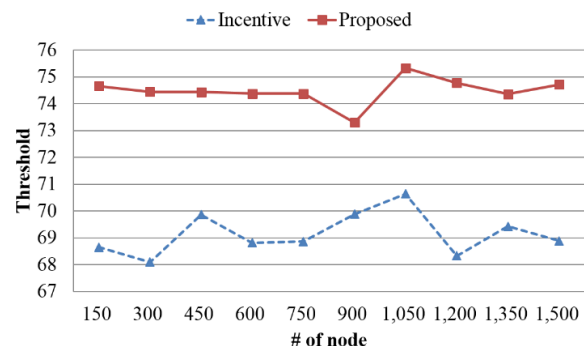**Fig. 4** Threshold distribution before and after added nodes

Fig. 5 Threshold distribution after additional node insertion

nodes were added to the network. "After" shows the threshold of each node through the proposed scheme when 200 nodes were gradually added to the network after performing the proposed scheme in the initially constructed network. The mean threshold of all nodes was gradually increased. The deviation of thresholds between nodes was decreased.

Figure 5 shows the comparison result of the threshold distribution of nodes after additional node insertion between the incentive mechanism and the proposed scheme. The initial number of nodes is 1,000 and the number of added nodes is 500. Compared to the incentive mechanism, thresholds of nodes were increased, and the deviation of the threshold between nodes was decreased in the proposed scheme. The mean threshold of all nodes was increased by about 4% compared to the existing scheme, while the standard deviation of thresholds between nodes was decreased by about
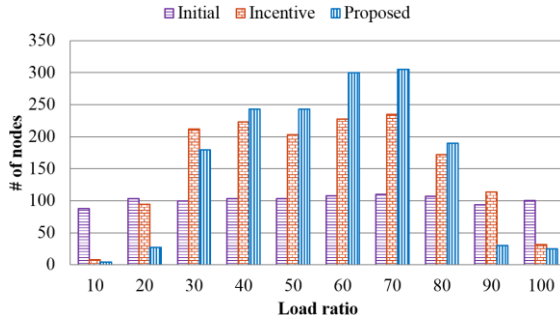
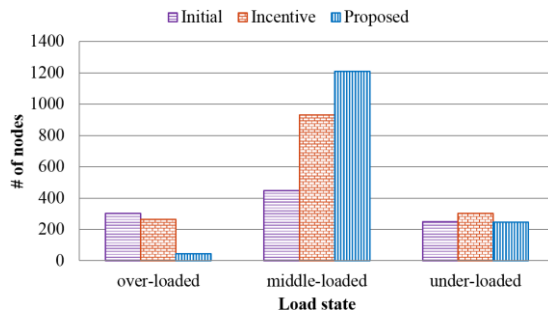**Fig. 6**   Node distribution according to load ratio

Fig. 7   Node load state distribution

23% compared to the existing scheme.

Figure 6 shows the distribution of load ratios after load balancing among initially set nodes in the existing load balancing scheme and the proposed scheme. "Initial" refers to the distribution of initially set nodes. The load distribution after load balancing through the proposed scheme showed that loads that were near the median value were increased more than those in the incentive mechanism and that loads that were too high or too low were decreased more than those in the incentive mechanism.

Figure 7 shows the load state distribution according to load balancing. The load state distribution after load balancing via the proposed scheme showed that the number of over-loaded nodes was decreased by about 83%, the number of middle-loaded nodes was increased by about 30%, and the number of under-loaded nodes was decreased by about 19% compared to those balanced via the incentive mechanism.

4.   Conclusion

In this paper, we have proposed a new load balancing

scheme in structured P2P. The proposed scheme not only encouraged threshold adjustment of nodes through incentives but also improved the participation of passive nodes in load balancing by adjusting nodes' thresholds through the mean threshold of adjacent nodes. Furthermore, the proposed scheme helped nodes that did not know their appropriate thresholds to adjust their thresholds, thereby increasing the load balancing participation of the nodes to improve the load balancing performance. The performance evaluation result showed that the proposed scheme increased load thresholds overall and decreased the deviation of thresholds between nodes, thereby adjusting threshold uniformly. Future research will evaluate the response time and network performance of the proposed scheme by applying it to real systems.

References

[1] X. Chen and J. Zhang, "Building decentralized resource space on a structured P2P network," Concurrency and Computation: Practice and Experience, vol.23, no.9, pp.866–879, 2011.

[2] S. Ghamri-Doudane and N. Agoulmine, "Enhanced DHT-based P2P Architecture for Effective Resource Discovery and Management," Journal of Network and Systems Management, vol.15, no.3, pp.335–354, 2007.

[3] Y. Qiao and G. Bochmann, "Load balancing in peer-to-peer systems using a diffusive approach," Computing, vol.94, no.8-10, pp.649–678, 2012.

[4] X. Meng and T. Li, "A dynamic load balancing scheme with incentive mechanism in heterogeneous structured P2P networks," Computer and Electrical Engineering, vol.39, no.7, pp.2124–2134, 2013.

[5] K. Bok, J. Yoon, J. Lim, and J. Yoo, "Load Balancing with Load Threshold Adjustment in Structured P2P," Proc. International Conference on Big Data and Smart Computing, pp.213–218, 2018.