Dynamic I/O Model Recommendation System With Machine Learning

*Note: Sub-titles are not captured in Xplore and should not be used

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—In a typical database and file system, using asynchronous I/O is generally a good way to optimize processing efficiency. However, asynchronous I/O may not a more efficient way in all situations. In this work, we use Machine Learning(ML) techniques to learn I/O model's performance, and set up a client/server system to recommend the more efficient I/O model under different system loads. The experimental result shows that our system has a 15% performance improvement compared to using asynchronous I/O alone.**

***Keywords*—asynchronous I/O, synchronous I/O, Machine Learning, performance prediction**

I. Introduction

With revolution of "Big Data", data has grown exponentially in data centers. Therefore, data center has to process hundreds of millions of pictures and hundreds of billions of messages each day. How to improve processing efficiency is a hot issue of a data center. Due to the huge speed gap between cpu and I/O device, I/O is one of the bottlenecks of the issue. Using asynchronous I/O is a common way to boost I/O speed, however, it isn't in all situations of system loads.

Synchronous and asynchronous I/O are two types of I/O synchronizations as ***figure 1*** shows. In a synchronous I/O job, it starts a thread for I/O operation, and it would hang immediately until the operation is finished. While in a asynchronous I/O job, it would start a thread to send a I/O request to Kernel by calling a function, if the request is accepted successfully, it continues to process other jobs. The kernel signals the calling thread when the operation is finished, then the thread interrupts its current job and processes the data from the I/O operation as soon as possible. Even if base on ***io-uring***.

To help improving system processing efficiency, we purpose to using a recommendation system based on Machine Learning(ML) This system is running as a daemon like a server and the process requesting the I/O is the client, it collects system I/O data for Machine Learning's training data, and decide what I/O synchronizations the process uses.

The main challenge of our system is performance, so ...

In summary:

Identify applicable funding agency here. If none, delete this.

1) lightweight
2) respond
3) Self-optimization

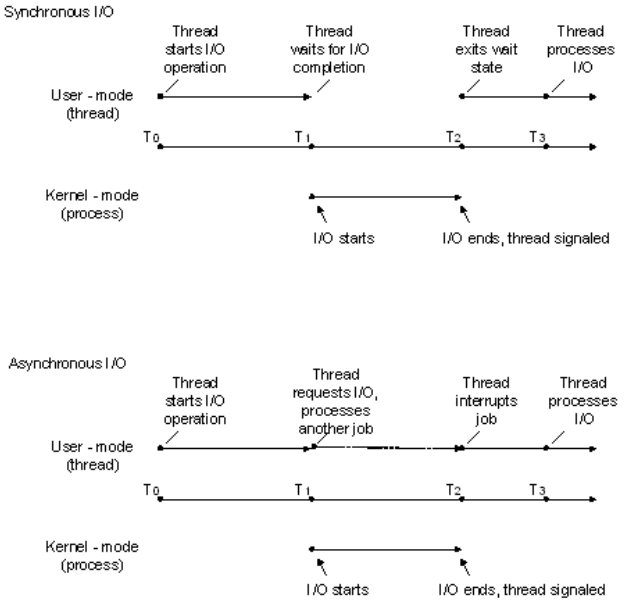

Fig. 1. Asynchronous and Synchronous I/O

II. Design

- evaluate io-uring performance and compare to other io-engins and synchronous
- why choosing machine learning and decision-tree
- how to connect client and server

III. Implementation

- collect data
- train data
- build the system
- test the system

IV. Evaluation

- io-uring performance
- compare single I/O work performance between used and none-used our system
- compare multi I/O works performance between used and none-used our system

V. Related Work

- hot issue in I/O

VI. Conclusion

improvement of our system and future usage scenario

Acknowledgment

References

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.