ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

BA/SA/MA/Group/Lab:

Efficient Load Balancing

A very simple method to evenly distribute the load among a huge cluster of machines is to allocate the tasks to these machines in a round-robin fashion. An efficient counter object that is shared among several processes can be very useful in this allocation task. In this thesis, we want to improve upon the existing ways of building a shared counter so as to solve the load balancing problem efficiently.

We want the counter to be linearizable, which basically means that if a process increments the counter, then the increment is included in any read operation afterwards. A simple implementation would be to repeatedly try writing the incremented value into a shared register using the compare-and-swap instruction until it succeeds. The problem with this approach is that a process may end up trying forever in the worst case. So, we also want the counter operations to finish in a bounded time even in the worst case. It is possible to solve this problem so that the counter operations take $O(\log n)$ time [1].

Figure 1: An inefficient load balancing

Surprisingly, not everything is solved for this basic problem. For example, one of the problems with the above solution is that the counts only increase and there is no support for decrement. Also, there is no support for re-initialization. The goal of this thesis is to work on some of the many possible extensions of the above solution, and possibly apply it to other problems too. If this sounds interesting to you, please feel free to contact us.

Requirements: Interest in designing and analyzing concurrent algorithms.

Interested? Please contact us for more details!

Contacts

- Pankaj Khanchandani: pankaj.khanchandani@tik.ee.ethz.ch, ETZ G60.1

[1] https://www.tik.ee.ethz.ch/file/68c636f2c14ee9a226e1d5cc91b25c03/counting.pdf