ECE5658: Operating Systems Design

Paper Reading Assignment

2019712410 Jeon Hyung-Joon

The Linux Scheduler: A Decade of Wasted Cores

Summary:

The objective of this paper is to address the issues of the Linux task scheduling algorithm on a multi-core/multi-processor architecture, and introduce tools related to diagnosing such issues. As stated in the paper, the OS thread scheduler must ensure that ready threads are scheduled on available cores. Unfortunately, due to logically erroneous code (up until version 4.1), Linux-based operating systems are prone to having tasks pending within run-queues upon just a single core, while having the other cores in idle state. To investigate deeply on this issue, the authors developed various diagnostic tools. Then, to solve the issue, they came up with several code modifications, which are proven to both decrease latency and enhance throughput of task scheduling in Linux.

Strengths and Weaknesses:

It can be noticed that the paper nicely describes the potential bugs in detail, specifying from which version a bug began to exist to the impact that it has. As demonstrated in the paper, the patch for such bugs were proven to have prominent enhancements in task scheduling performance. With diagnostic applications, undesired behavior in scheduling can be detected, and this could enable further improvements, if any exists, of the Linux scheduling scheme in future releases.

However, the reliability of one bug fix seems questionable. As a solution for the "Overload-on-Wakeup" issue, the authors claimed that newly woken up threads should always select the core that stayed idle for the longest time period. This could cause overhead of mandating the cores to be almost always ready as long as the number of tasks to be run is greater than the number of existing cores, leading to higher power consumption of hardware. Though the paper assumes a system with only non-idle cores throughout the whole session lifetime, this is not the case in reality. Such adoption of new policy can be detrimental to Linux-based machines that require energy efficiency such as multi-core mobile phones. To make things worse, this code modification may cause excessive heat distribution across cores, causing serious thermal performance regression.

Suggestions for Improvement:

To alleviate the side-effects of the patch for "Overload-on-Wakeup" issue, I suggest including additional algorithm or adjusting the modification itself that can prevent excess activation of cores in the Linux system. After that, I recommend that the authors test out their modified fix and monitor if there are any side-effects related to power consumption. From these two pieces of opinion, I believe the reliability of the main idea of the paper can be strengthened further.