1. The Linux Scheduler: aDecade of Wasted Cores, EuroSys 2016

   This paper focuses on discovering and solving bugs arising from load balancing of the existing solid Linux scheduler as modern hardware get more complicated. It also presents two tools for discovering these bugs.

   This paper has its strength in that it dealt with various bugs that may occur during the load balancing process with easy-to-understand examples and reasonable performance improvements. However, it is regrettable that the scheduler was considered only as a work conserving aspect, and that it did not consider power consumption.

   It would be better to add a graph that shows the correlation between power consumption and performance improvement in bug fixes.

2. Arachne: Core-Aware Thread Management, OSDI 2018

   This paper has the theme that by allowing the application to know information about the core and request the core directly, not the thread, it can expect latency by application, increase the efficiency of the throughput, and efficient implementation of thread.

   The strength of this approach is that the application can use system resources very efficiently in that it can control the core directly. But they did not address the blocking I/O problem which schedule activation does do.

   It would be better to add a reference to whether it is safe to give user applications permission to control the core. I wish there was a reference to how to deal with the user application that malicious use the core.