1. **Power provisioning for a warehouse-sized computer, ISCA 2007**

   This paper says it is becoming important to make the best use of a given power capacity in the data center system. But there are three obstacles to predicting, controlling and managing power consumtion in big data center system. First, there is large difference of up to 40% between the actual peak power consumption of the system and the theoretical peak usage. Second, the actual power consumption of the server depends greatly on the amount of activity. Finally, the tendency of power consumption in large scale system varies because of different applications. Then, this paper shows the characteristic of 3 large scale workloads as well as a workload mix from an actual datacenter. And characterizing power consuming tendency of those workloads on three level (Rack, PDU and Cluster). And suggest that it is better to build power provisioning strategy on cluster level, not on rack level. Furthermore, this paper suggest that system need to think the power efficiency on activity range not only on peak performance level to exploit utilization of power budget.

   It has an advantage that this paper is the first power usage study at the real data center level and workloads and that precisely measuring the power consumption of large scale cluster on real workloads and shows those measure on multi-level. but they didn't show actual result of applying the new power provisioning strategy.

   It would be better show the actual difference of power efficiency between conventional one and their own strategy.

2. **Where is the energy spent inside my app? Fine Grained Energy Accounting on Smartphones with Eprof, Eurosys 2012**

   In this paper, the problems of energy profiling and accounting of smartphone apps are studied, making three specific contributions that enable the development of smartphone's energy-aware apps. First, fine-grained energy profiler for modern smartphones called Eprof. It focuses on energy accounting policies: how to map the power draw and energy consumption back to program entities. Second, it reports for the first time on its experience using Eprof to analyze energy consumption in several apps. It shows that, in most popular free apps, performing the task related to the purpose of the app consumes only a small fraction (10-30%) of the total app energy. Finally, this proposes a bundle, a new accounting presentation of app I/O energy that helps developers quickly understand and optimize the energy drain of app.

   It has an advantage in that it deals with difficulties in measuring energy consumption in mobile environments and analyzes them through the infrastructure based on the FSM power model and presents alternatives. But Eprof requires modifications in the Android framework to trace the API calls; the application code, if using the Android NDK, should also be modified. This aspect can be seen as a weakness.

It would be better to add an explanation of applications with energy-intensive application level code. I think it may not work.