Lecture 17 Summary

This lecture talked about KSR-1 and Intel Phi shared memory multi-processors. The first part is about a paper which describes a measurement study of the effects of thread placement on memory access times on the Kendall Square KSR1 multiprocessor. The general KSR architecture is a multiprocessor system composed of a hierarchy of rings of processors. The lowest level is the processor cell, which has a 64-bit superscalar processor and 32MB of local cache memory. Each processor cell is connected to tow neighbors to form the lowest level ring. There were some experiments conducted on KSR1. And results showed that several implications: The placement of the owner thread of a data set affects performance; 2. Suite I shows that additional threads that share a data set can improve performance; 3. The placement of the owner thread of a data set particularly affects the performance of reader threads that are placed on a remote ring; 3. Code changes that allow a shared data set to be distributed among several owners, or stagger the access pattern among readers, can also substantially improve the performance.

The second part was about Intel Xeon Phi Coprocessors. For the environment of test, the authors used Intel Xeon Phi Coprocessors and 16-socket 128-core system from Bull to do the comparison. Due to the foundations in Intel architecture, they used tow different ways for our experiments: 1. Cross-compiled OpenMP programs natively on the coprocessor; 2. The Intel Language Extensions for Offload. To get a first impression of the capabilities of the Intel Xeon phi coprocessors, they used: 1. STREAM to investigate the memory bandwidth of the coprocessor and the coprocessor achieves a better bandwidth compared with the BCS system in this benchmark; 2. EPCC to investigate the overhead of several OpenMP constructs, and results shows that the ability of the coprocessor to handle function calls and other high-level programming constructs allows to offload rather large kernels and helps hide the overhead. To evaluate the performance of a real-world compute kernel, they used a CG solver that runs natively on the Intel Xeon Phi coprocessor. From the results, Intel Xeon Phi coprocessor can reach a very high scalability.