Practical Exercise #14

Affinity: A Truce with the Computing Core

Exercises

1. Create a visual representation of a pair of different hardware architectures. Disclose the hardware characteristics of these devices.

2. Run the benchmark suite on your hardware using the script in Listing 14.1. What did you learn about how to best utilize your system?

3. Modify the program used in the vector addition example (vecadd_opt3.c) in Section 14.3 to include more floating-point operations. Take the compute kernel and change the operations in the loop to the Pythagorean formula:

c[i] = sqrt(a[i]*a[i] + b[i]*b[i]);

How have your results and conclusions about the best placement and bindings changed? Do you now see any benefit from hyperthreading (if you have it)?

4. In the MPI example from Section 14.4, include the vector addition compute kernel and generate a scaling plot for the kernel. Then modify the kernel with the Pythagorean formula used in Exercise 3.

5. Combine vector addition and the Pythagorean formula in the following procedure (either in a single loop or in two separate loops) to allow for greater data reuse:

c[i] = a[i] + b[i];

d[i] = sqrt(a[i]*a[i] + b[i]*b[i]);

How does this change the results of the placement and affinity exploration?

6. Add source code to assign placement and affinity within the application from one of the previous exercises.