How to Design for Scalable Performance—from Multicore to Many-core

By Ekaterina Antakova, Software Engineer, Developer Products Division, Intel

Have you ever threaded an application but seen little performance gain? Have you hit a “scalability ceiling” where performance gains level off as you add more cores? Implementing a parallel algorithm can be a lot of effort. Wouldn't it be great to explore a couple of different implementation schemes and see which is best, before investing in the heavy lifting of full implementation? This is the problem that Intel® Advisor XE 2015 is designed to solve. It creates a framework for software architects to model their design and predict the performance scaling and synchronization issues. Here, we will see how Intel Advisor XE 2015 extends this modeling to support Intel® Xeon Phi™ coprocessors.

Part of the Intel® Parallel Studio XE 2015 family, Intel Advisor XE 2015 helps to assess the opportunities for parallelism in serial code, pinpoint the parts that are ready to benefit from using Intel Xeon Phi coprocessors, and identify the key limiters when parts are not ready to scale. Intel Advisor XE is available at http://intel.ly/advisor-xe.
To enable moving to new CPUs and coprocessors effectively and to ease programmer efforts, Intel provides tools to help identify the best spots to utilize parallel programming to accelerate an application. For example, Intel Xeon Phi coprocessor capabilities can be used effectively on highly parallel workloads. In order to achieve this goal, the programmer has to learn what “highly parallel” actually means in terms of their application. This is the problem which the Intel Advisor XE helps attack with the exciting new features now available.

These features help the programmer to automatically compare Intel Xeon Phi coprocessor performance limits with Intel® Xeon® processor peak performance for a given workload to make informed decisions on which parts of code are ready for porting to the coprocessor. Another new capability is the ability to predict parallel scalability for bigger datasets to help understand how to increase parallel efficiency on modern multicore and many-core hardware.

These modeling capabilities aim to answer three main questions about the analyzed application:

1. **Is the Intel Xeon Phi coprocessor the right fit for this workload?**
   Intel Advisor suitability analysis automatically predicts if Intel Xeon Phi coprocessor performance levels can exceed the Intel Xeon processor performance peaks for a given workload. It also helps determine if the current application structure does not scale well on the coprocessor early in the development cycle.

   Figures 1 and 2 show the predicted performance gain of a potential parallel version of two test loops executing different stages of an image processing algorithm. The green zone shows gain levels which are considered ready for running this workload on the Intel Xeon Phi coprocessor. Examining color-coded zones, it is easy to determine that the first loop shown in Figure 1 scales quite well and looks like a good candidate for porting to the coprocessor. According to this prediction, it should achieve appropriate speedups on 128 coprocessor threads and more, if ported to the coprocessor. Another loop shown in Figure 2 can't be considered “highly parallel,” as its predicted maximum gain becomes flat after 128 coprocessor threads. So this one does not seem to be a good candidate for running on the coprocessor, unless the algorithm can be modified to be more highly parallel.
2. What are the main factors limiting parallel performance and scalability in this application?

For efficient parallelization of serial code, it is essential to understand the main obstacles to desired performance and scalability. A high-level breakdown of parallelism performance losses caused by imbalance in parallel jobs, lock contention, or parallel runtime overheads is provided by Intel Advisor suitability analysis.

Figure 3 shows a loop which has good potential. The three markers show the range of performance possible by changing lock overhead, task chunking, parallel framework, etc. The middle circle moves as the user selects different runtime modeling options. All three markers can move when the user changes the task modeling options. Maximum predicted performance (shown by the upper white rectangles) grows well with the increasing number of CPUs in Figure 3. However, the current performance gain of this loop (shown by the white circles) does not scale well after 64 CPUs according to the Intel Advisor prediction. The reasons for this include too much overhead of parallel runtimes: ~30s in total compared with total parallel execution time predicted for this code of ~165s and significant imbalance between parallel tasks causing waits of around 29s. This is an indication of issues with parallel task granularity and breakdown in this loop. Significant runtime overhead prediction shows that parallel tasks are too fine-grained for this workload, and that creation of every task is done too often, causing too much overhead.
3. **What happens to parallel scalability as the workload size scales up?**

   Suitability Iteration Space modeling aims to predict what happens when workload size increases. It models more iterations in loops, longer execution of iterations, or both at the same time. Run a smaller sample and analyze how the performance will change if dataset size and computation amount increase. Determine a sufficient dataset size to get the most from modern, highly parallel multicore CPUs and coprocessors.

   Figure 4 shows a scalability prediction for a model of a future parallel loop on a smaller test workload and the user interface for modeling bigger data size processing. The current prediction shows quite limited scalability of this workload on 32 and more CPUs.

   But will it scale better with bigger input data? Based on the nature of the analyzed loop and the bigger data to be processed, let’s make an assumption that the iteration number should increase about 125 times, and the duration of each iteration should increase a bit less than this, about 25 times more than the current duration.
Figure 5 shows the updated performance prediction for this loop model with the new data size parameters. In this case, increasing input data—modeled by increased number of iterations and longer iteration durations—leads to better scalability for this loop, making it an appropriate candidate for executing on the Intel Xeon Phi coprocessor.

New modeling capabilities of the Intel Advisor tool are based on measuring CPU-bound work, task granularity (chunking and scheduling), load balancing, lock contention, and overheads of the selected parallel framework. To analyze predicted performance and gains for Intel Xeon Phi coprocessor applicability evaluation, the Intel Advisor model includes CPU frequency parameters, coprocessor-specific runtime overheads, and data transfer overhead for specified data size when modeling offload execution.

Here is what a graphics processing company, an early Beta evaluator, said about the practical benefit of the new workload scaling feature: “Intel Advisor XE 2015 Beta demonstrates a useful ability to estimate dataset size that is essential for choosing policy when chunking big images for multiple renders.”

By modeling the scalability of future parallel code on different numbers of CPUs and on the Intel Xeon Phi coprocessor, Intel Advisor helps identify the most performance-profitable parts of your application. Development efforts can then focus on these code locations. Experimenting with bigger workloads is useful to understand the optimal workload sizes sufficient to saturate many cores on a given target platform. This allows you to explore how an application workload scales under different conditions and discover whether you can get more performance benefits from running on a coprocessor.
Intel Advisor XE Predicts Scalable Performance on Intel Xeon Phi Coprocessors

Intel Advisor XE modeling is a tremendous tool for understanding algorithm scalability in your applications, acquiring a realistic view of the scalability of your program as implemented, and getting feedback on what the limiters are in your code. This is enormously useful. Unfortunately, no tool is available to tell you if there is a radically different approach or algorithm to get the same work done while scaling better. However, with Intel Advisor XE you can determine if you need or want more scalability, and evaluate multiple approaches quickly and easily. This will let you find the best methods available to scale, and identify which applications have enough scaling to use highly parallel systems such as Intel Xeon Phi coprocessors.

Try Intel® Advisor XE

Available in these software tools:
Intel® Parallel Studio XE 2015 Professional and Cluster Editions >