Hyper-Threading Influence on CPU Performance

João Martins* <martinsj@lip.pt>

Jorge Gomes* <jorge@lip.pt>

Mario David* <david@lip.pt>

Gonçalo Borges* <goncalo@lip.pt>

* LIP – Laboratório de Instrumentação e Física Experimental de Particulas

Outline

- Motivation
- Test Description
- Test Results
- Conclusions
- Work to do
- But..
- References

Motivation

The Intel Hyper-Threading (HT) technology enables one processor core to present two logical cores to the operating system (OS), allowing it to support two software threads at once.

The processor maintains a separate set of registers for each thread, the OS manages the threads as if they were each running on their own core. This will increase the usage efficiency of the CPU execution units, for example, if one thread has a cache miss the other continues processing instructions.

Motivation (cont.)

In this study we evaluate the benefits of HT technology with high energy applications running at NCG site.

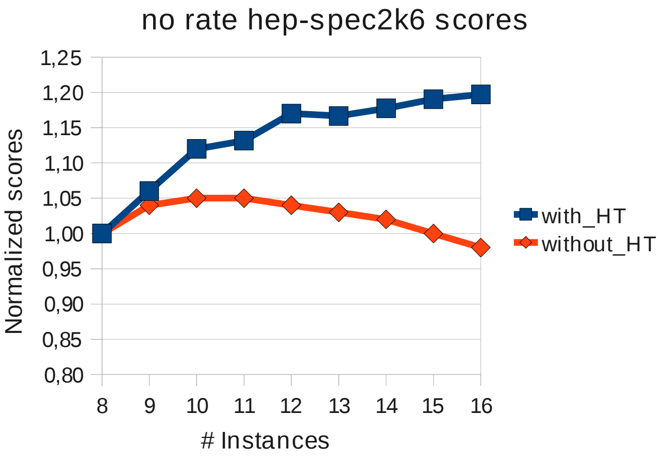
Test Description

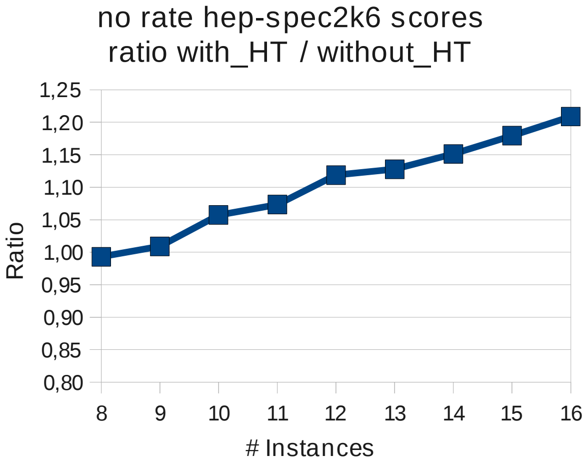
- **Tested Model:** HP Proliant BL460c G6
  2 x Quad Intel Xeon X5550 @2.67GHz, 8MB L3 cache
  24GB RAM
  (Try to test Intel Xeon E5540 but users didn't allow it)
- **Number of servers:** 3
- **Benchmark software:** HEP-SPEC2006
- **Acquisition:** we performed benchmark runs from 8 to 16 instances in the following conditions: rate and no rate with HT, and rate and no rate without HT
- **Number of runs:** at least 3 for each run condition and for each server, we will use the machine scores geometric means for analysis

Results

no rate hep-spec2k6 scores with and without HT

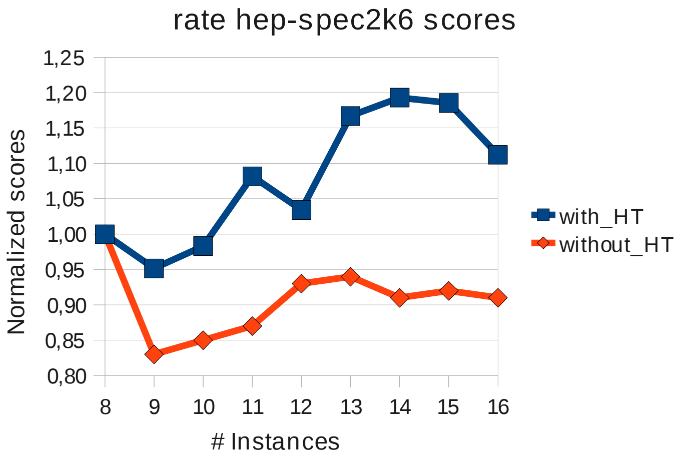
- Modest increase of score values when HT is disable, best scores between 9 and 12 instances;
- Good scale of scores with the number of instances with HT enabled;
- Clear advantage with HT enable, we observe an increase up to 20% for maximum instances pack.

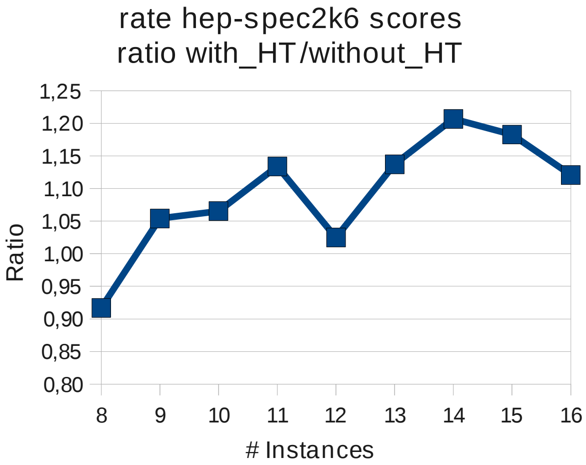




Results (cont.) rate hep-spec2k6 scores with and without HT

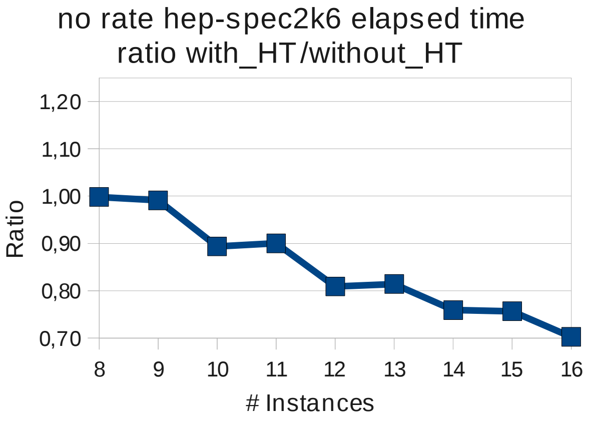
- Clear score degradation with HT disabled;
- Irregular scale of scores with the number of instances with HT enabled;
- We still benefit when HT is enabled but the irregularities suggest there are other elements contributing to the overall profile, probably thread competing over memory access and CPU execution units.



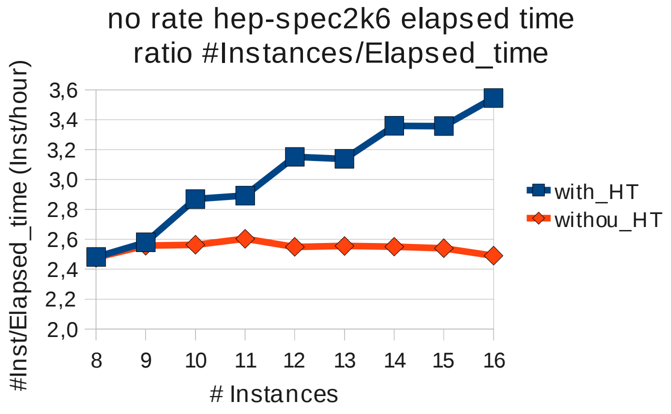


Results (cont.) no rate hep-spec2k6 elapsed time with and without HT

- Light I/O activity because of preparation and benchmarks compilation;
- Clear decrease on elapsed time down to 30% with maximum instances pack when HT is enabled;
- Step profile due to core fill with two threads, one thread per core is the same as no HT enabled;
- Clear increase on the units of work done per unit of time with HT is enabled, it is more or less stable when HT is disabled.



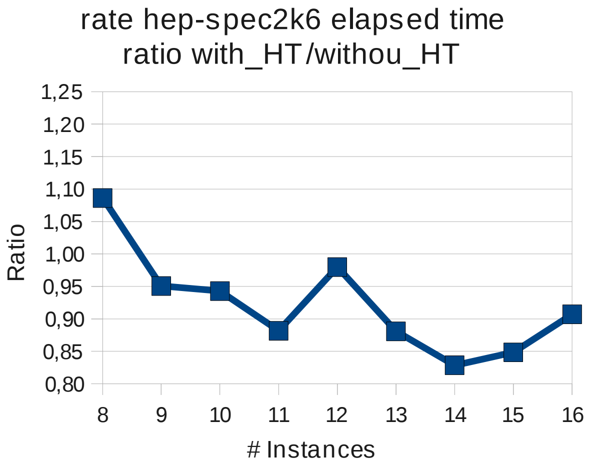
no rate hep-spec2k6 elapsed time ratio with_HT/without_HT

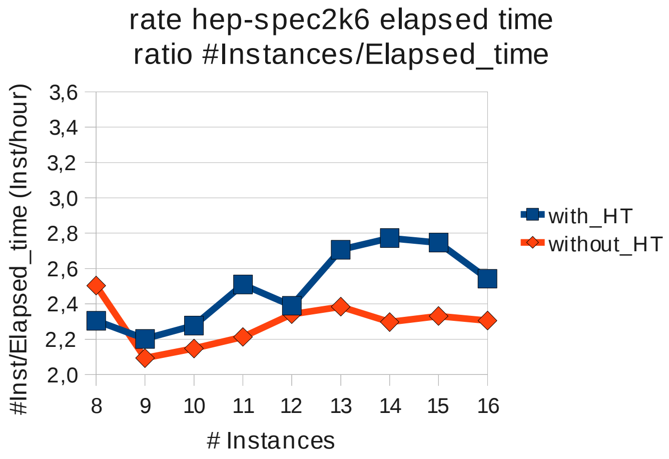


no rate hep-spec2k6 elapsed time ratio #Instances/Elapsed_time

Results (cont.) rate hep-spec2k6 elapsed time with and without HT

- Light I/O activity because of preparation and benchmarks compilation;
- Irregular scale of elapsed time with the number of instances;
- Limited benefits with HT enabled versus HT disabled;
- Step profile not clear, threads should be competing for the same resources, probably memory access and CPU execution units.





Conclusions

- HEP applications with zero I/O activity may benefit up to 20% efficiency increase with HT enabled as long the software threads cope with the number of hardware threads;
- HEP applications with moderate I/O can experience an efficiency increase up to 30% with HT enabled for a fully loaded node;
- HEP-SPEC2k6 is a good benchmark utility to evaluate HEP applications performance but real software threads presents I/O activity, a complementary set of tests is needed to measure HT benefits;
- Parallel applications show an irregular performance profile with moderated increases for a loaded node but in some conditions may show a degradation;
- The actual use of HT technology and the number of allowed threads on a node should depend on the nature of the applications running on it;
- The default OS CPU affinity configuration is not the best strategy for HT technology.

Work to do

- Use of real applications, particularly with medium I/O activity, will give a clear view on the influence of HT technology on performance;
- Extend the number of applications instances over the maximum number of logical cores;
- Find a cleaver way to fill software threads on CPU's hardware threads;
- Repeat all measurements on other server models with HT available at LIP and NCG sites.

But..

- On several occasions we observe crashes on Intel Xeon's L5520 and L5410 servers with heavy I/O activity and HT enabled, it doesn't happen when HT is disabled.

- I believe this fact is enough to exclude the use of Hyper-Threading technology in any situation.

References

- http://www.spec.org
- https://twiki.cern.ch/twiki/bin/view/FIOgroup/TsiBenchHEPSPECWlcg
- http://agner.org/optimize/blog/read.php?i=6&v=t
- http://software.intel.com/en-us/articles/performance-insights-to-int