Scalability Problems In Shared Memory

José Alves, Rui Brito

Universidade do Minho

Braga, March 2013

Index

1. Memory Bandwidth and Computational Intensity
2. Measure Task Granularity
3. Measure Programs Without Synchronisation
4. Measure Loads Per Task
5. Conclusion

Memory Bandwith

$Mem\_bandwidth = ((PAPI\_L2\_TCM) \times 64 \times 1 \times 10-9)/exec\_time$
$measured\_mem\_bandwidth = (8427602 \times 64 \times 1 \times 10^{-9})/0.88 = 0.6129 GB/s$
$System\_memory\_avail\_bandwidth = 14.8473 GB/s$

As seen in the December presentation, conv-diff has some locality problems (has it can also be seen by the high number of L1 misses (14937146)).

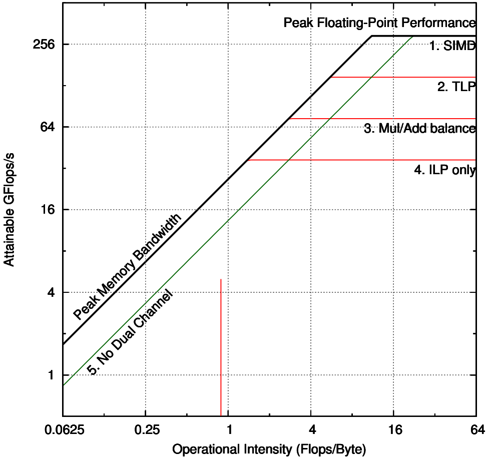

Figure : Roofline for rMBP and conv_diff

Task Granularity

In conv_diff:

- Only two parallel pieces of code run in parallel
- Large chunks of code
- Thread creation overhead is thus minimized

conv_diff is unsuitable for this sort of optimization:

Excessive task synchronization

- Reduction can't be used because values are updated in an array of pointers
- Synchronization must be forced on attributions

Loads Per Task

- Slight improvement using dynamic and guided scheduling
- Workload distribution also isn't the biggest problem

Conclusion

- Bad data locality hinders the entire application performance
- Implementation either AoS or SoA is expected to improve performance drammaticaly
- High chance that locality problem hides other problems
- Maybe what is not a problem now will prove to be so in the near future