

Set 1 - SIMD

Issued: February 16, 2015

Hand in: February 23, 2015, 12:00 noon

Question 1: SIMD compatibility

- a) Find out which SSE/AVX version your machine and compiler support.¹ Which speedup do you expect from it?
- b) Find out which SSE/AVX version does the Euler cluster support.

Question 2: Parallel computation of squares of binomials

The provided skeleton code computes many squares of binomials.

```
1  for( int i = 0; i < N; ++i ) {  
2      z[i] = x[i]*x[i] + y[i]*y[i] + 2.*x[i]*y[i];  
3  }
```

- a) Vectorize the code using the manual SSE or AVX intrinsics and the report the speedup obtained compared to the serial version. Does it meet your expectations?
- b) Now try to do the same using automatic vectorization and study the compiler's vectorization report to check that the loop is indeed vectorized. Comment on the performance relative to your manual version. **Hint:** With gcc the reduction is only vectorized with the `-funsafe-math-optimizations` option.²

Summary

Summarize your answers, results and plots into a PDF document. Furthermore, elucidate the main structure of the code and report possible code details that are relevant in terms of accuracy or performance. Send the PDF document and source code to your assigned teaching assistant.

¹<http://software.intel.com/sites/landingpage/IntrinsicsGuide/> for Intel processors

²`-funsafe-math-optimizations` enables more aggressive optimizations by allowing the compiler to violate some details of IEEE and ISO standards. This can cause problems with codes relying on, e.g., correct treatment of special values like infinities and NaN. Reference: <http://gcc.gnu.org/onlinedocs/gcc-4.8.2/gcc/Optimize-Options.html#index-funsafe-math-optimizations-906>