High-Performance Matrix Computations
Homework #1

Prof. **Paolo Bientinesi**

pauldj@aices.rwth-aachen.de

Due: Friday, May 20th, 23:59pm

Matrix-matrix product (GEMM)

Operation: $C := C + A * B$

where $A \in \mathcal{R}^{m \times k}$, $B \in \mathcal{R}^{k \times n}$, and $C \in \mathcal{R}^{m \times n}$

Implementations

- **“BLAS-0”**: Explicit triple loop; 6 variants.
- **BLAS-1**: Inner products; 2 variants.
- **BLAS-2**: Matrix vector products; 2 variants. Outer product; 1 variant.
- **BLAS-3**: Direct call.

Assignment

Goal: Study the performance of different implementations of GEMM

1) **Implement** at least 4 variants, covering all four BLAS levels.

2) **Validate** your routines. Make sure you are computing the right quantity. Explain how the correctness of your code is assessed.

3) **Time** your routines.

4) **Report** the *Performance* and the *Efficiency* of the variants.

   a. *Repetitions*. Repeat the experiments K times and include statistics (min, max, med, . . .).

   b. *Caching*. Repeat the experiments, making sure that at each iteration, the matrices are evicted from cache.

5) **Present** your results (visually).
   Annotate the *in cache* vs. *out of cache* boundaries.

6) **Explain** the experiments. Comment. And provide your data.

Rules and Submission

- Format: pdf, html, org.
- **Use an optimized BLAS**: OpenBLAS, MKL, BLIS, Accelerate. Do not use the reference BLAS! (`http://www.netlib.org/lapack/`)
- Code in C or through ELAPS.
- Sequential execution. No multi-threading.
- No need to (attempt to) optimize your code manually.
- Turn in results as well as code.
- Include a description of the architecture used.
- Submission by email to `pauldj@aices.rwth-aachen.de`
- Email's subject: “`HPMC-16 HW1 your_last_name`”
- **Deadline: Friday, May 20th, 23:59pm**