Reaching the Roof: Automated BLIS Kernel Generator for SVE and RVV

Stepan Nassyr, Kaveh Haghighi Mood, Andreas Herten
Jülich Supercomputing Centre, Forschungszentrum Jülich

About EUPILOT Project

EUPILOT

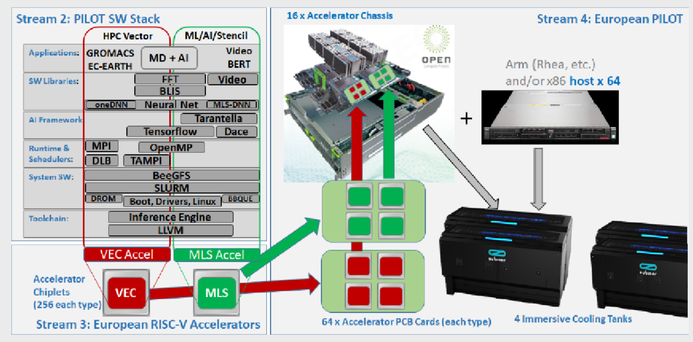
- EuroHPC JU project for All-European HPC system
- 19 European Partners
- 29 999 925 € budget
- Open source/open standards for software, hardware

→ `https://www.eupilot.eu`

VEC Processor

- VEC: RISCV-CPU with wide (256 × FP64) VPU *under development*
- ISA: RVV 0.7.1 (currently testable), 1.0 (in-development)
- 8 lanes per core; 2 FP64 op/cycle/lane → 16 $\text{FLOP}_{\text{FP64}}$/Cycle

Through EUPILOT: Access to FPGA SDV of VEC, at BSC



Introduction

- Accelerated BLAS (Basic Linear Algebra Subroutines): essential for many HPC applications and beyond
- Modern BLAS superset implementation: BLIS (`https://github.com/flame/blis`)
  - Accelerates nearly all BLAS3 methods with optimized GEMM microkernel
  - GEMMTRSM + BLAS1 kernels for full acceleration
- Microkernels usually handwritten and hand-tuned to microarchitecture
- **Our approach:** Automate kernel generation!

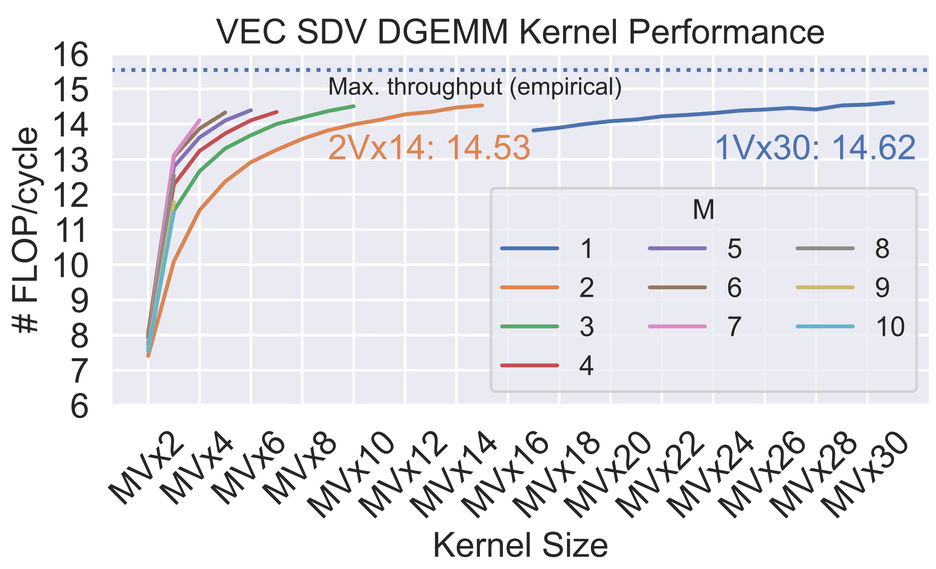
→ Generate ARM SVE and RISCV RVV SGEMM and DGEMM kernels; benchmark on Fujitsu A64FX, Allwinner D1, EUPILOT VEC SDV

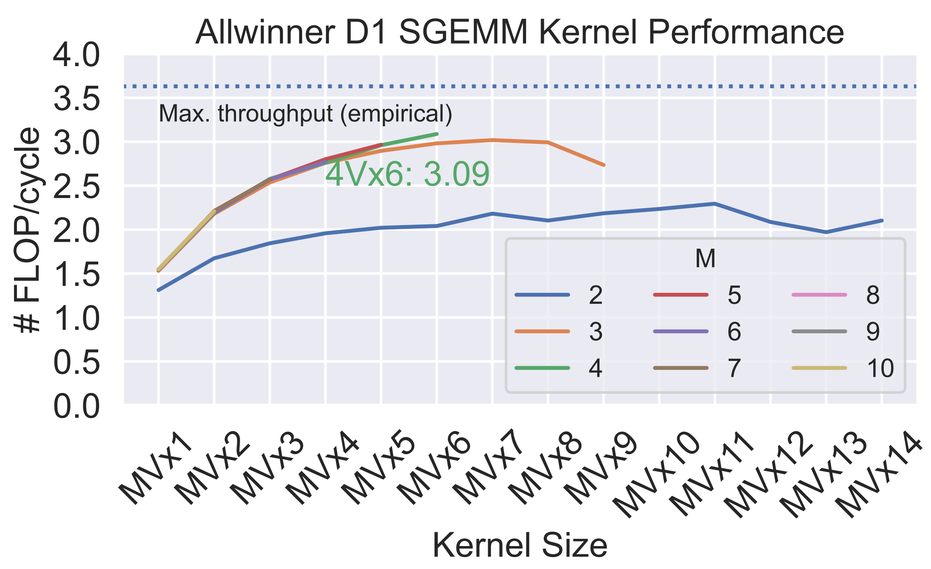
Code generation

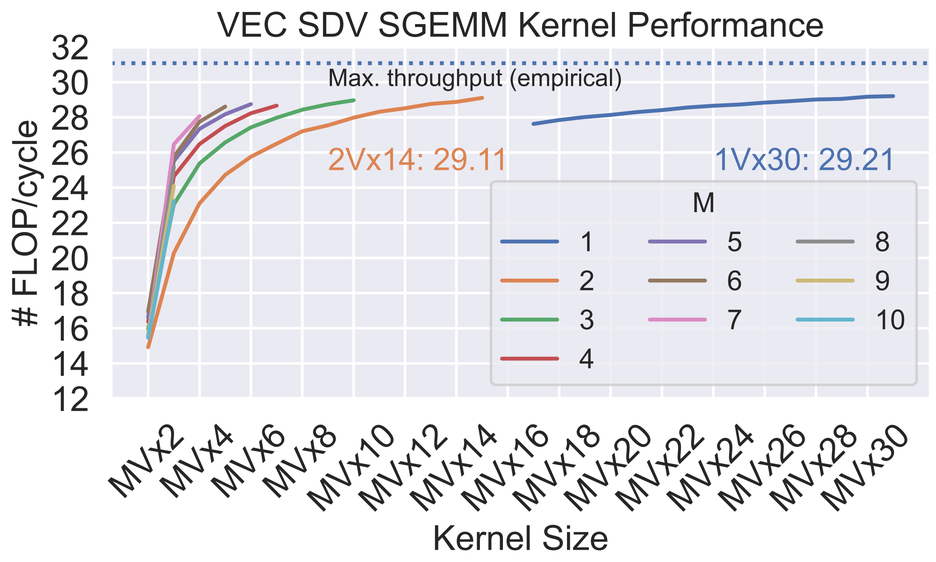
- Kernel code defined by parameters such as
  - Kernel dimensions (size of the C-tile worked on by the μkernel in elements times elements, i.e 8x6, 16x10)
  - VLA kernel size in number of vector registers times elements (i.e 2Vx10, 4Vx6)...
  - Order of operations
- Future optimizations to be controlled by new parameters
- Generates $O(n^3)$ part of the microkernel (Nanokernel) as inline assembly (Supported: x86 AVX2/512, ARM NEON/SVE, RISCV RVV 1.0/0.7.1)
- Benchmarks the µkernel in L1 or with eliminated memory accesses

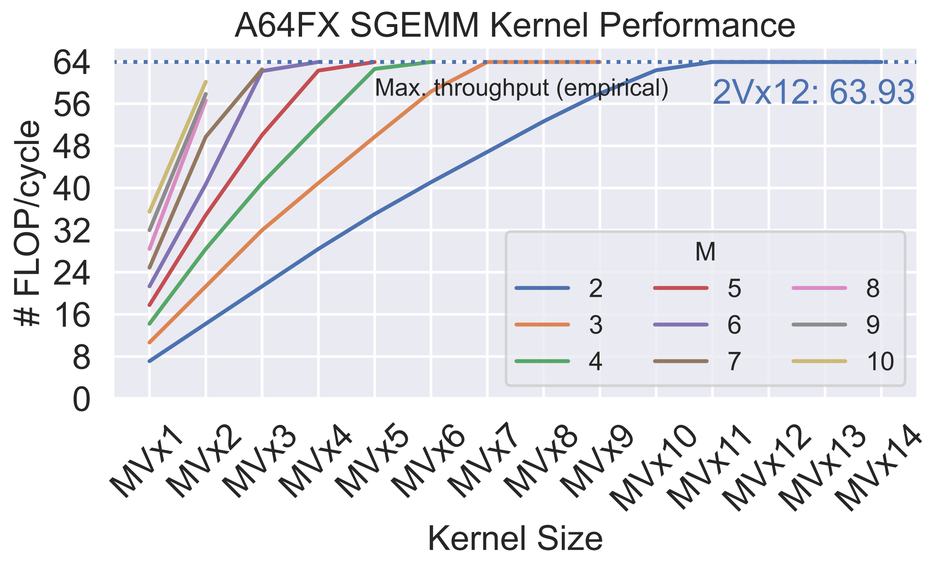
⇒ Quick prototyping and benchmarking of many kernels - focus here: kernel size

- Manually add C-tile update and scaling ($O(n^2)$ part) to the best performer to use as BLIS microkernel









Results

Graphs:

$\mathcal{M}V \times \mathcal{N}$ — Different Lines: #Elements *A* matrix; x Axis: #Elements *B* matrix → #FMAs *(ind.)*

Measured peak (robustness against platform side-effects)

	Emp. Peak FLOP/Cycle		Best Nanokernel Rel. Performance
Processor	*FP32*	*FP64*	
A64FX	63.93	31.95	> 99 %
VEC SDV	31.08	15.54	94 %
D1	3.63	-	85 %

Good utilization on A64FX; VEC architecture details considered in further optimizations; D1 as non-HPC architecture lower optimization priority
Our tool successfully generates highly optimized GEMM kernels for different architectures

Future work

- Co-design with EUPILOT VEC developers
- GEMMTRSM and Level-1 kernels for full BLIS optimization
- Full BLIS microkernel generation
- Use of Matrix-Multiply instructions (ARM SME, Intel AMX, ...) and other relevant current and upcoming technologies
- Optimized BLIS library for the EUPILOT VEC accelerator
- Kernels for other numerical libraries (FFT)
- GPU (generate PTX, CDNA, ...)

Acknowledgements

The European PILOT project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No.101034126. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Spain, Italy, Switzerland, Germany, France, Greece, Sweden, Croatia and Turkey.