StencilFlow: Mapping Large Stencil Programs to Distributed Spatial Computing Systems

Johannes de Fine Licht*, Andreas Kuster*, Tiziano De Matteis*, Tal Ben-Nun*, Dominic Hofer*†, Torsten Hoefler*

*Department of Computer Science, ETH Zurich, Switzerland †MeteoSwiss, Switzerland

{definelj, kustera, tdematt, tbennun, dohofer, htor}@ethz.ch

Problem

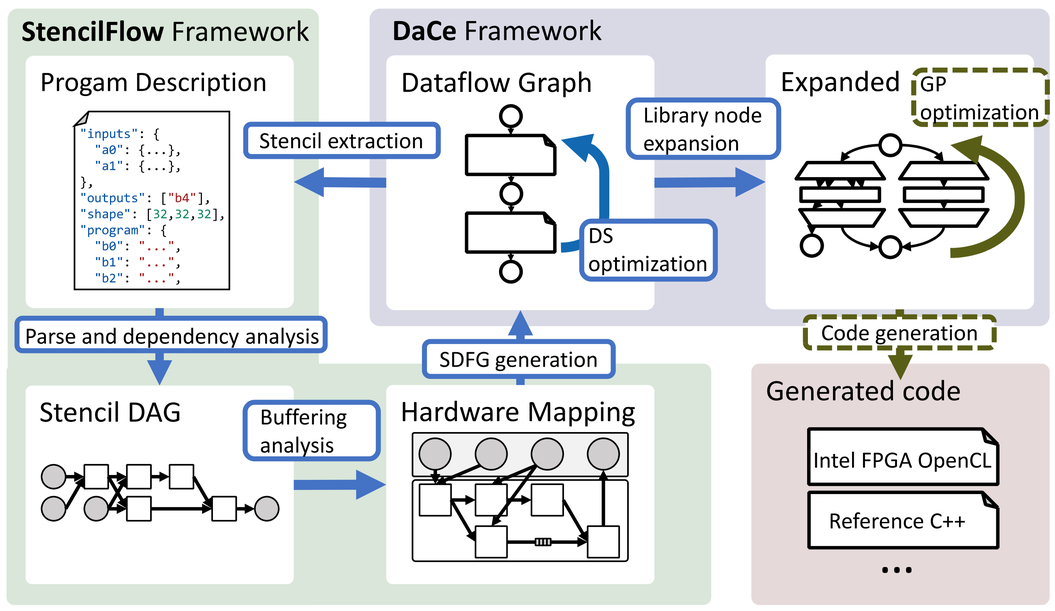
Reconfigurable hardware offers *massive spatial parallelism*. However, *mapping* of heterogeneous stencil computation by hand while maximizing temporal locality is difficult.

Proposed solution

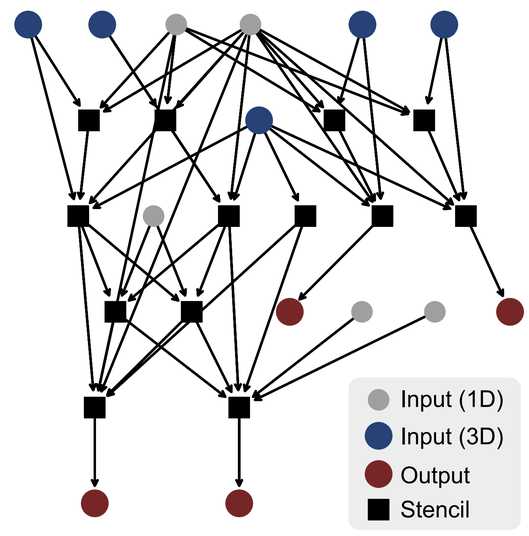
We introduce *StencilFlow*, an end-to-end analysis, optimization and code-generation stack built on the *DaCe* framework:

- Enables generation of complex high-performance stencil programs from high-level DSL input
- Reaching highest recorded single and multi device performance on Stratix 10

End-to-end Workflow



Case Study: Horizontal Diffusion



Complex dataflow graph of a horizontal diffusion stencil program used in numerical weather prediction models. *StencilFlow* achieves perfect temporal reuse by mapping it to spatial hardware.

	Runtime	Performance	Peak BW.	%Roof.
Stratix 10	1,178 µs	145 GOp/s	77 GB/s	52%
Stratix 10*	332 µs	513 GOp/s	∞ GB/s	–
Xeon 12C	5,270 µs	32 GOp/s	68 GB/s	13%
P100	810 µs	210 GOp/s	732 GB/s	8%
V100	201 µs	849 GOp/s	900 GB/s	26%

Horizontal diffusion benchmark. *without memory bandwidth constraints

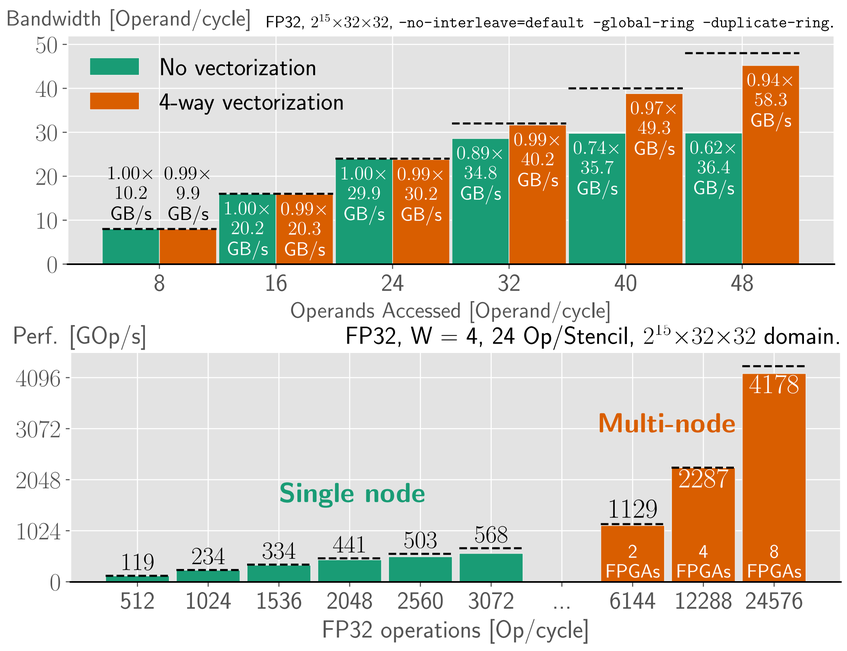
Open Source

StencilFlow and DaCe are both available free and open source on github.com/spcl/{stencilflow, dace}

Results

	Performance	ALM	FF	M20K	DSP
Total		103 M	3.7 M	11.7 K	5760
Avail.		692 K	2.8 M	8.9 K	4468
Diffusion 2D W=8 (Ours)	1,313 GOp/s	449 K 64.8%	1329 K 48.0%	2565 28.6%	2304 51.6%
Diffusion 3D W=8 (Ours)	1,152 GOp/s	567 K 81.9%	1606 K 57.9%	5357 59.8%	3072 68.8%
Diffusion 2D (Zohouri et. al. [8])	913 GOp/s	471.4 K 68.0%	1173.6 K 42.3%	2204 24.6%	3844 86.0%
Diffusion 3D (Zohouri et. al. [8])	934 GOp/s	450.5 K 65.0%	1078.2 K 38.9%	8684 97.0%	3592 80.4%
Waidyasooriya and Hariyama [21]	630 GOp/s	Arria 10 GX 1150			
SODA [9]	135 GOp/s	ADM-PCIE-KU3			
Niu et al. [22]	119 GOp/s	Virtex-6 SX475T			
Ben-Nun et al. [14]	139 GOp/s	Virtex UltraScale+ VCU1525			

Comparison to hand-tuned code and other frameworks.



Deep pipeline performance (scalar/vectorized) and bandwidth.

Acknowledgements: The authors wish to thank Tobias Kenter, Christian Plessl, Jakub Beránek, the Swiss National Supercomputing Centre (CSCS), and the Paderborn Center for Parallel Computing (PC²) for their valuable help, and access to compute infrastructure.