Towards clean propulsion with synthetic fuels: A cluster-modularized approach employing hierarchies of simulations and deep learning

M. Bode[a], M. Davidovic[a], J.H. Göbbert[b], B. Dick[c], P. Offenhäuser[c], C. Terboven[d], M.S. Müller[d], T. Lippert[b], M.M. Resch[c], H. Pitsch[a]

[a]Institute for Combustion Technology (ITV), RWTH Aachen University, Aachen, Germany
[b]Jülich Supercomputing Centre (JSC), Forschungszentrum Jülich GmbH, Jülich, Germany
[c]Höchstleistungsrechenzentrum Stuttgart (HLRS), University of Stuttgart, Stuttgart, Germany
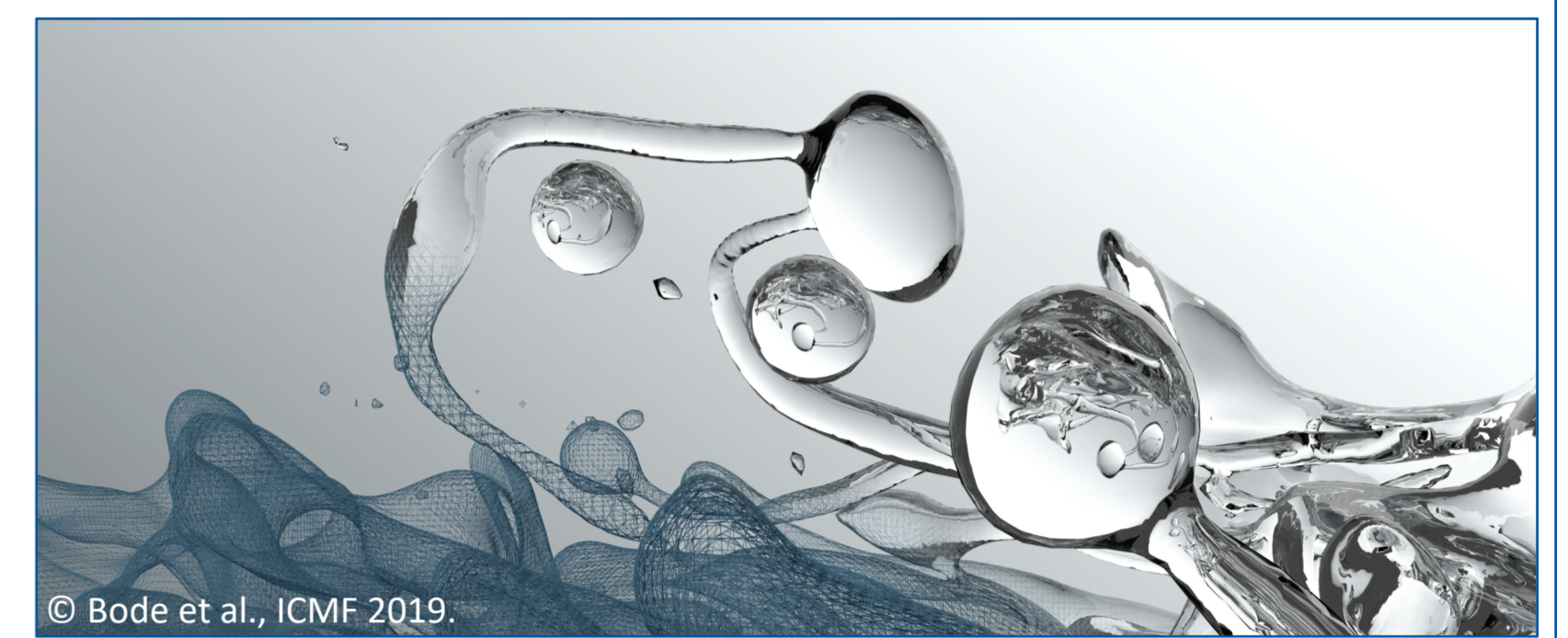[d]IT Center (ITC), RWTH Aachen University, Aachen, Germany

Application Case

- Development of new clean propulsion technology only possible by combining experiments and simulation (finding optimal parameters)
- Fuel injection effects critical but not understood (cavitation, breakup, wall interaction, …)
- Simulation very expensive due to a large range of scales and multi-physics, which also restricts the minimum problem size
- **Minimizing time-to-solution for given accuracy requirement by optimization: node performance** (FLOPS), **scaling** (efficiency of large runs), **coupling of high-order models** (DNS) **and reduced-order models** (LES), **data transfer speed/size, cluster selection**
- ➡ Our hierarchical approach using optimized LES/DNS and optimal computing resources enables an iterative development cycle (edge2cloud)
- ➡ CNNs (DL) are trained on-the-fly with DNS data and used as BC generator for LESs, optimizing data transfer from TBs to GBs

Cavitation
Wall interact.
Breakup
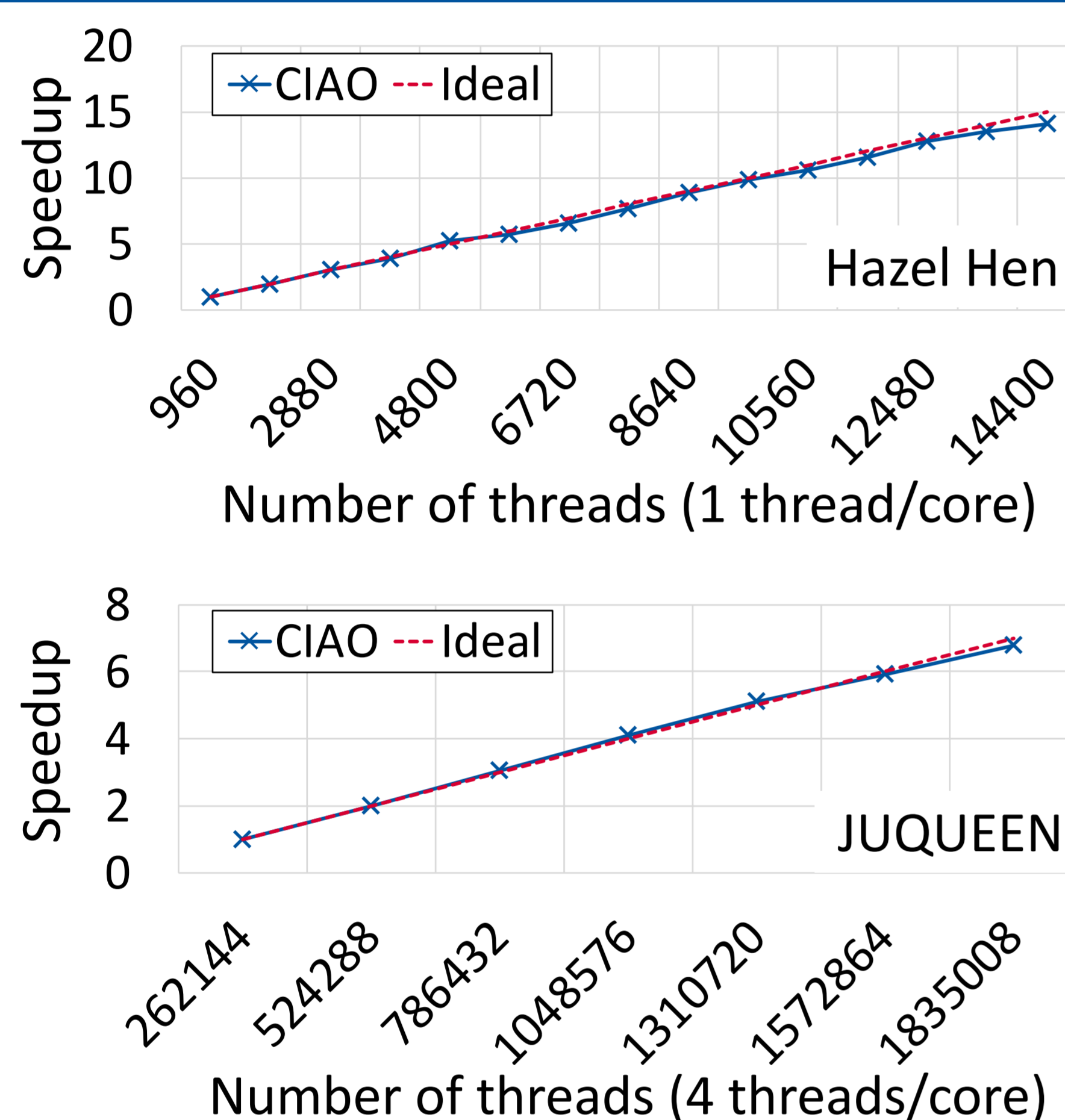© Bode et al., JSAE 2017.
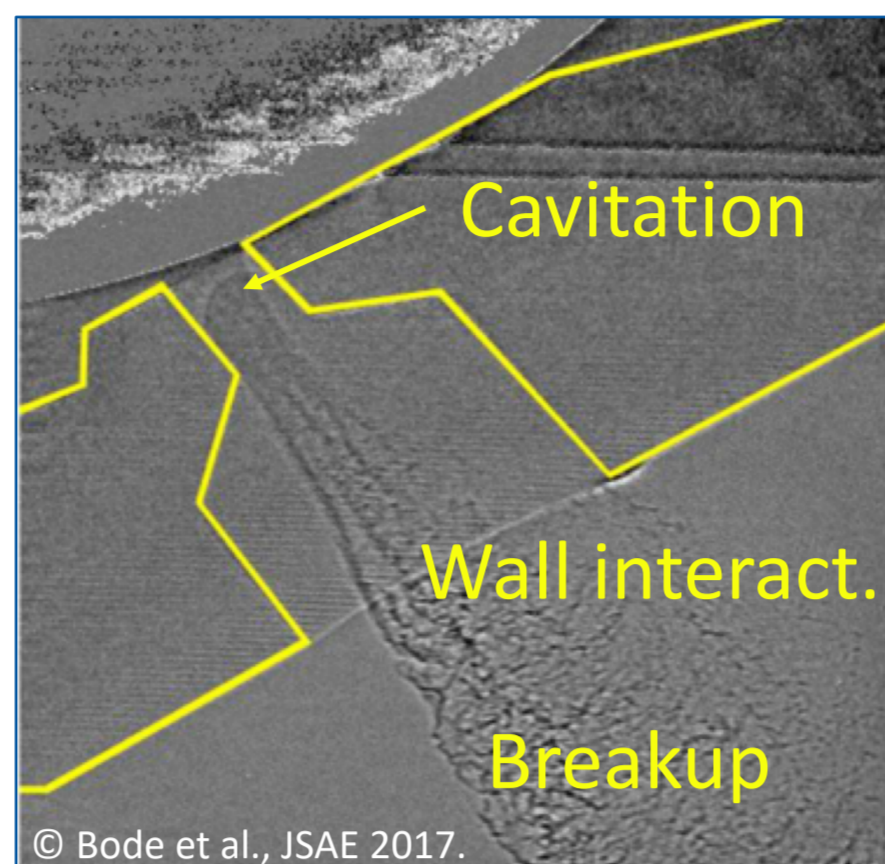
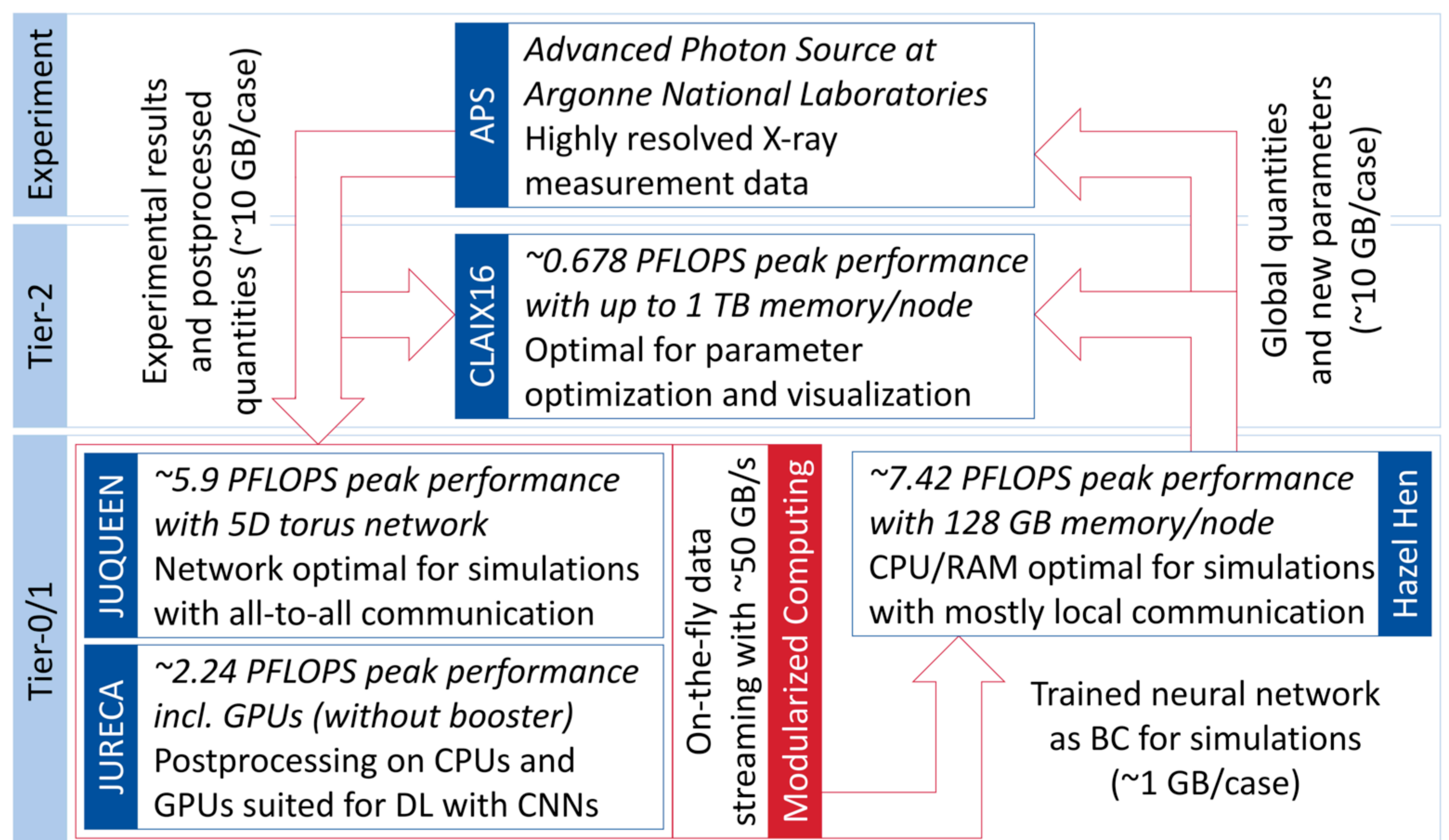## Modularized DNS&DL @ JUQUEEN&JURECA

- Liquid breakup simulations (DNS) are large and expensive but can be run in low Mach limit
- ➡ All-to-all communication which benefits decisively from JUQUEEN's 5D torus network
- Simulations were performed using up to 1.84 Mio. threads and **378.4 TFLOPS** (full JUQUEEN)
- JURECA's GPUs were used to train a CNN using on-the-fly data streaming, modularized computing, and statistically similar data
- A CNN architecture with 3D sub-boxes as input and five hidden layers was employed as BC generator
- ➡ Implicit consideration of spatial gradients, which are important for proper BCs for the LES
- **400.2 TFLOPS** were achieved during training

© Bode et al., ICMF 2019.

Hierarchical Computing Approach

Speedup — Number of threads (1 thread/core) — CIAO / Ideal — Hazel Hen
960, 2880, 4800, 6720, 8640, 10560, 12480, 14400

Speedup — Number of threads (4 threads/core) — CIAO / Ideal — JUQUEEN
262144, 524288, 786432, 1048576, 1310720, 1572864, 1835008

LES scaling on Hazel Hen (top) and **DNS scaling** on JUQUEEN (bottom); minimum number of threads chosen based on minimum RAM requirements

Experiment — APS: *Advanced Photon Source at Argonne National Laboratories* Highly resolved X-ray measurement data

Tier-2 — CLAIX16: *~0.678 PFLOPS peak performance with up to 1 TB memory/node* Optimal for parameter optimization and visualization

Tier-0/1 — JUQUEEN: *~5.9 PFLOPS peak performance with 5D torus network* Network optimal for simulations with all-to-all communication

JURECA: *~2.24 PFLOPS peak performance incl. GPUs (without booster)* Postprocessing on CPUs and GPUs suited for DL with CNNs

Hazel Hen: *~7.42 PFLOPS peak performance with 128 GB memory/node* CPU/RAM optimal for simulations with mostly local communication

Experimental results and postprocessed quantities (~10 GB/case)

Global quantities and new parameters (~10 GB/case)

On-the-fly data streaming with ~50 GB/s — Modularized Computing

Trained neural network as BC for simulations (~1 GB/case)
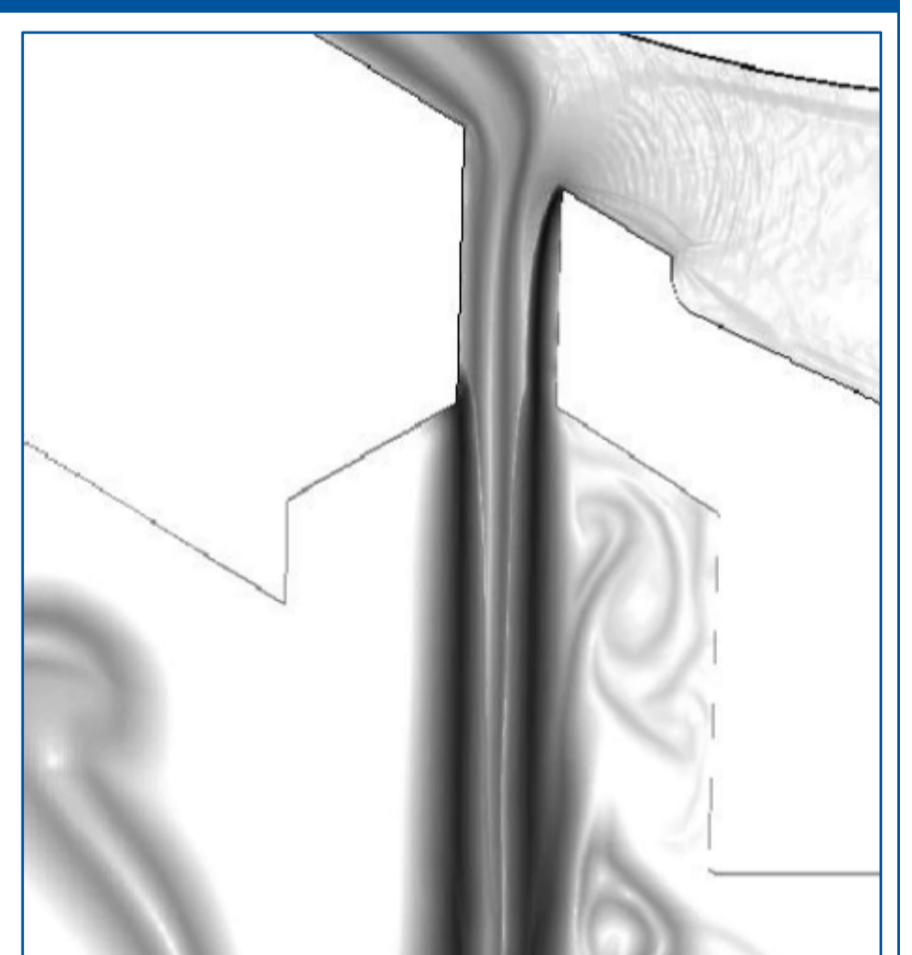
LES @ Hazel Hen

- Compressible injection simulations (LES) are less expensive and require mainly local communication
- ➡ Network less important for scaling, and LES performed on Hazel Hen due to good ratio of CPU power and memory per node
- Specific vectorization with OpenMP instructions accelerated time-to-solution of LES by 17.2%
- LES able to run with up to **130.5 TFLOPS** on 85% of Hazel Hen after cache usage optimization
- Trained CNN used as BC and allowed studying broad parameter space
- ➡ Predictive simulations/designed experiments possible

Golden Spike Award 2018

Achievements

- CIAO used for optimizing propulsion technology
- Optimal cluster choices and optimizations enable very good scaling (overall) and node performance (compared to other CFD codes)
- Modularized computing reasonably reduced critical intra-cluster data transfer
- CNN trained on DNS data reduced inter-cluster data transfer and provided accurate BCs for LES parameter study at the same time
- **Hierarchical approach reduced time-to-solution (overhead for inter-cluster data transfer included) by 52.0% compared to a single cluster with similar peak performance**
- ➡ Iterative development cycles as feedback to the experiment possible
- ➡ Optimizations with respect to nozzle geometry, injection conditions and fuel properties were realized (cf. Figure)

Acknowledgements & Reproducibility

DNS – Direct Numerical Simulation
LES – Large Eddy Simulation
CNN – Convolutional Neural Network

DL – Deep Learning
BC – Boundary Condition

CIAO – Inhouse code framework
CFD – Computational Fluid Dynamics

Frankfurt, June 16th – 20th, 2019
ISC High Performance 2019