IFERC Workshop (online) on GPUs Programming (16th December 2020)

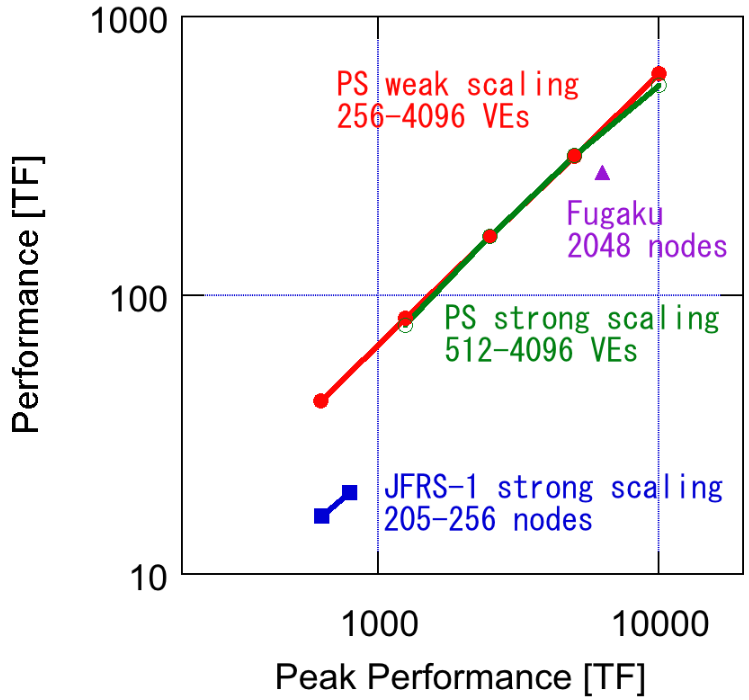
Conclusion and close

Masatoshi YAGI

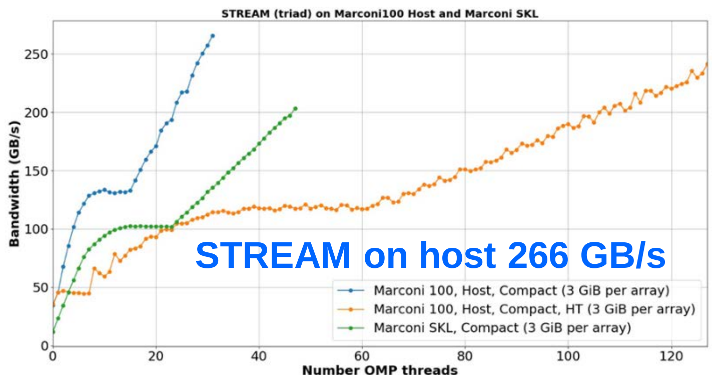
National Institutes for Quantum and Radiological Science and Technology(QST), Rokkasho Fusion Institute, Japan

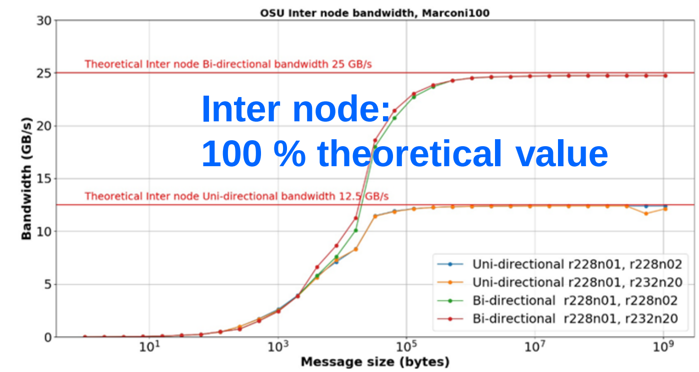
Summary

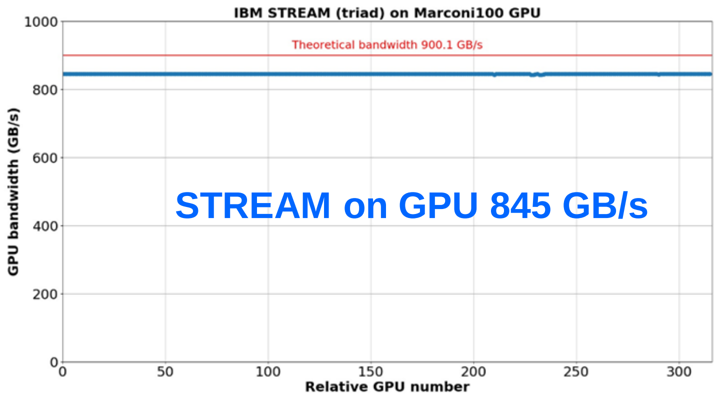
- Performance of MEGA is compared among JFRS-1, Plasma Simulator, and Fugaku.
- Good performance of MEGA is demonstrated for both strong and weak scaling on Plasma Simulator.
- MEGA achieved 6.3 – 6.7% of the peak performance for 256 – 4096 VEs on Plasma Simulator.
- Performance of MEGA on JFRS-1 and Fugaku is 2.5% and 4.4%, respectively.

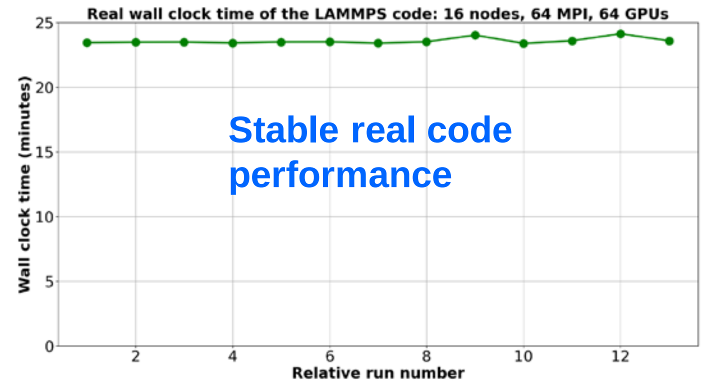


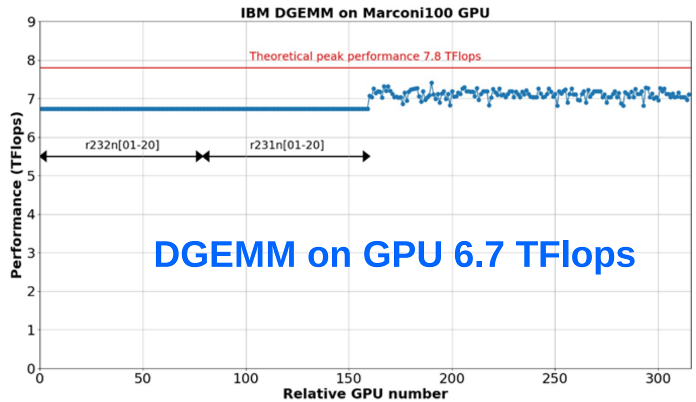
Benchmarks and validation of the Marconi100 HPC system

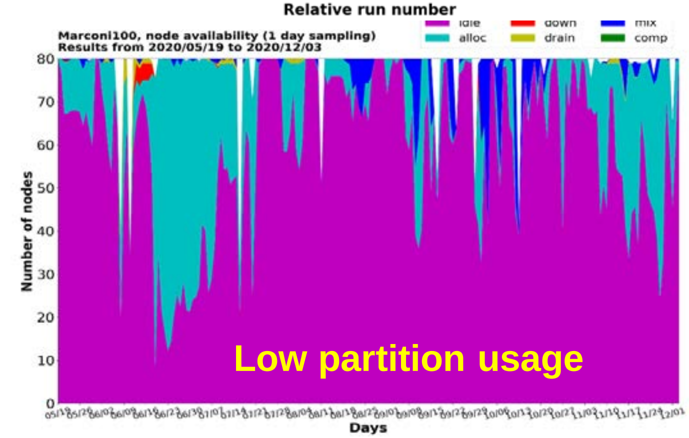








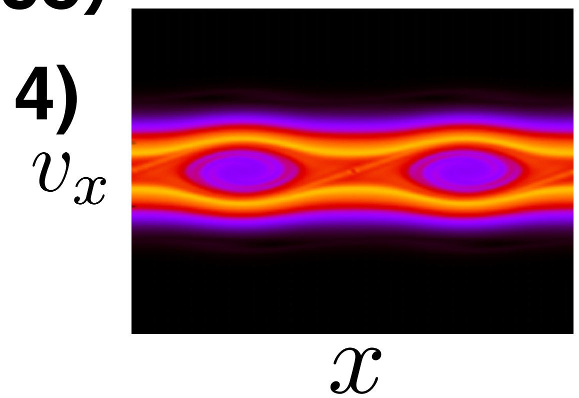




Performance portable implementation with Kokkos/Directives

4D Vlasov-Poisson equation （2D space、2D velocity space)

- Vlasov solver: Semi-Lagrangian, Strang splitting (1D x 4)
- Poisson solver: 2D Fourier transform



Kokkos version of Poisson solver

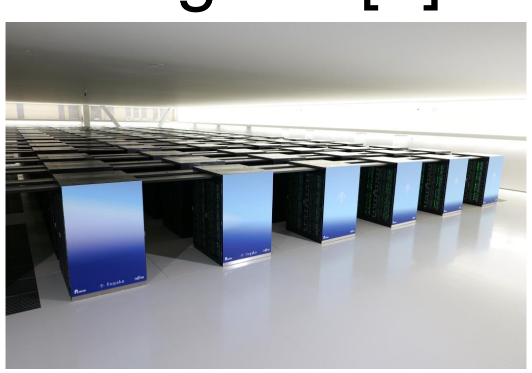
```
// Forward 2D FFT (Real to Complex)
fft_->rfft2(rho_.data(), rho_hat_.data());
Kokkos::parallel_for(nx1h, KOKKOS_LAMBDA (const int ix1) {
  for(int ix2=1; ix2<nx2h; ix2++) {
    double ky = ix2 * ky0;
    double k2 = kx * kx + ky * ky;

    ex_hat(ix1, ix2) = -(kx/k2) * I * rho_hat(ix1, ix2) * normcoeff;
    ey_hat(ix1, ix2) = -(ky/k2) * I * rho_hat(ix1, ix2) * normcoeff;
    rho_hat(ix1, ix2) = rho_hat(ix1, ix2) / k2 * normcoeff;
  }
  …
});
// Backward 2D FFT (Complex to Real)
fft_->irfft2(rho_hat.data(), rho_.data());
fft_->irfft2(ex_hat.data(),  ex_.data());
fft_->irfft2(ey_hat.data(),  ey_.data());
```

Performance against SKL (OpenMP)

	Time [s]	Speedup
Skylake (OpenMP)	278	x 1.00
Skylake (Kokkos)	192	x 1.45
TX2 (OpenMP)	589	x 0.47
TX2 (Kokkos)	335	x 0.83
P100 (OpenACC)	21.5	x 12.9
P100 (Kokkos)	15.6	x 17.8
V100 (OpenMP4.5)	16.9	x 16.4
V100 (OpenACC)	10.0	x 27.8
V100 (Kokkos)	6.79	x 40.9

Single code works on CPUs/GPUs

Fugaku [1]



Summit [2]

Achievements

Good performance portability keeping readability and productivity with Kokkos (Abstraction of memory and parallel operation)

[1] https://www.r-ccs.riken.jp/en/
[2] https://www.olcf.ornl.gov/summit/
[3] Y. Asahi et al., OpenACC meeting, September, Japan
[4] Y. Asahi et al., waccpd (SC19), November, US

ORB5 on GPUs

- ORB5 ported to GPUs with OpenACC
  - Particles live on GPUs, rest of code on CPU
  - Details in Ohana et al., CPC 2020
- Memory capacity limiting factor on m100
  - Sets minimum parallelization / maximum problem
- Strong scalability now (often) limited by field solver
  - Mixing OpenACC+OpenMP path forward