24.77 Pflops on a Gravitational Tree-Code

- Another N-Body problem, like molecular dynamics on Anton
- But a bit different scale – astronomical
  - Simulate the evolution of our Milky Way galaxy, with fine structure dynamics, and include halo effects (dark matter)
  - Over billions of years (with a time step of 75K years)
- Since want to simulate hundreds of billions of objects (stars) for many time steps (> 100,000), $O(n^2)$ not possible
  - Use Barnes-Hut tree algorithm, to aggregate objects into spatial groups (like an oct-tree), for forces for distant objects
  - Becomes an $O(n \log n)$ algorithm, but more complex to compute forces

Computation Issues

- Run on modern supercomputers, with CPUs and GPUs
  - Swiss Piz Daint (#6 on Top500) – 5272 nodes, each with Intel CPUs and NVIDIA Kepler GPUs, Cray Aries interconnect
  - ORNL Titan (#2 on Top500) – 18688 nodes, each with AMD CPUs and NVIDIA Kepler GPUs, Cray Gemini interconnect
- Biggest optimization is doing all computations (tree-walk, force computations, time integration) on GPUs, and overlapping communication between nodes (done by CPU) with GPU computations
  - Need to use GPU local memory on Kepler GPU
  - Use Local Essential Tree (LET) and Space Filling Curve (SFC) methods to assign sub-domains to nodes (to minimize communication)
Results

- Scientific results show that improved resolution enables the simulation to provide insight into origins of the observed structure of the galaxy and how it has evolved.
- Performance results show that the code scales well (both strong and weak scaling) and gets 24.77 Pflops on Titan (86% efficiency) and 6.18 Pflops on Piz Daint (95% efficiency).
- More impressive, for a single time step that does $\sim10^{16}$ floating point operations, takes only about 5.5 seconds.
  - So could do 8 billion year simulation with 242 billion particles in about a week on Titan (18600 GPUs).