A pathway for performing long-term climate simulations on Summit

Gunther Huebler, UW-Milwaukee
Collaborators: Vincent Larson, John Dennis

Supported by NGD Atmosphere Physics

Our goal is to perform long-term climate simulations on Summit

In order to perform long-term simulations, we need to achieve large throughput (e.g., >~ 5 SYPD).

In order to get an allocation on Summit, we need to use GPUs efficiently. This requires providing the GPUs with a lot of parallel work.

Doing cloud-resolving simulations provides a lot of work for the GPUs, but has a throughput of only ~1 SYPD

High resolution (e.g., 3 km) means a short time step (e.g., 15 s), which means poor throughput.

The small time step prevents us from performing long-term (e.g., 100-year) climate simulations.

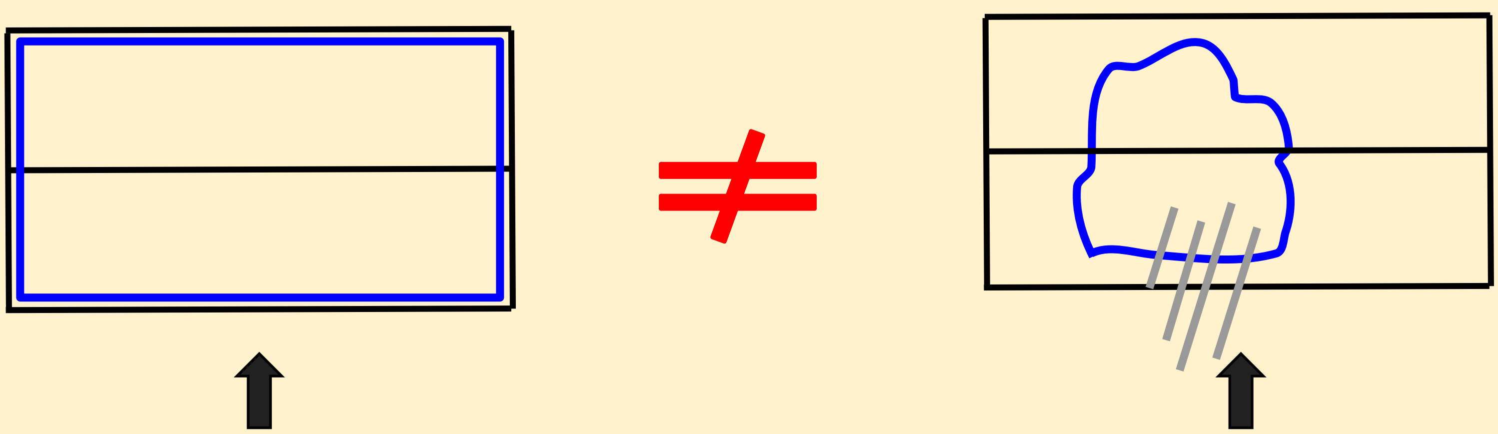
An alternative way to get an allocation on Summit is to use coarser resolution and subcolumns

We propose to use coarse horizontal resolution (e.g., 25 km). This allows us to use a long time step, which in turn allows us to perform long-term climate simulations.

In addition, generating multiple subcolumns within each grid column and performing physics calculations on them will provide parallelizable work for the GPUs.

Subcolumns are designed to drive physical processes, e.g. microphysics, using subgrid-scale variability:

For instance, we'd like to account for the effects of partial cloudiness on drizzle rate. We'd also like to account for within-cloud variability.

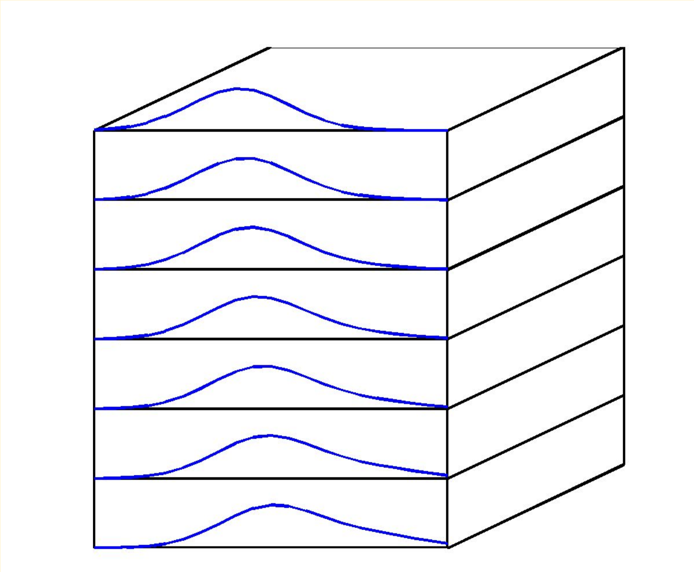


How a model with no subgrid information handles microphysics

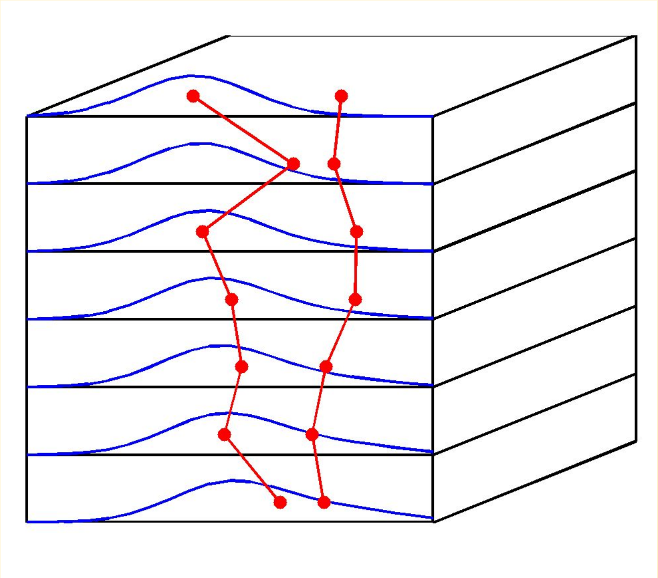
What we are attempting to do

Our subcolumn generator is named SILHS - Subgrid Importance Latin Hypercube Sampler and consists of 4 steps

1. Predict the PDF of subgrid variability at each grid level.



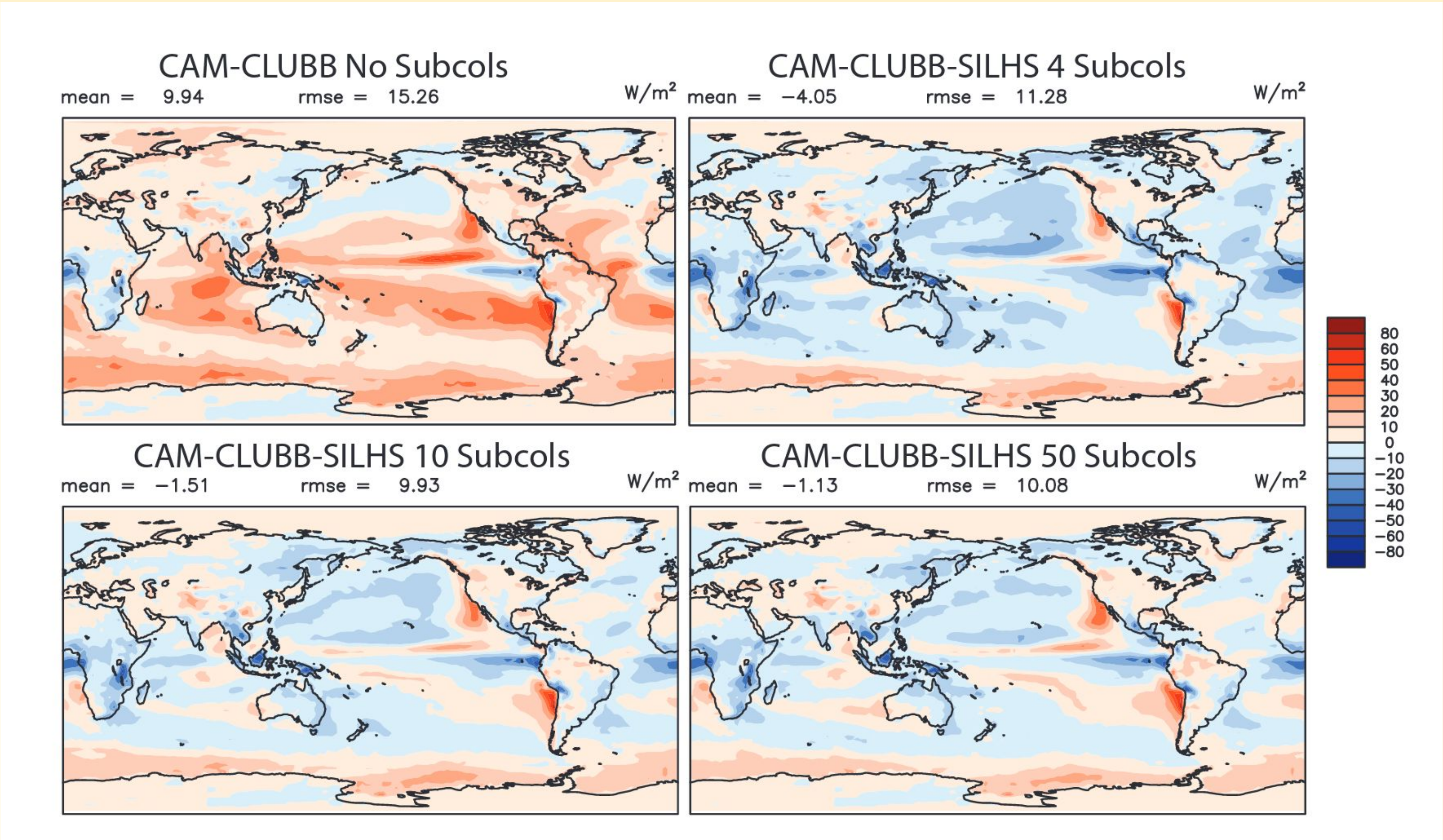
2. Generate subcolumns consistent with the subgrid PDF at each level.



3. Feed each subcolumn into the microphysics parameterization (e.g., MG3).

4. Average the microphysical tendencies from the subcolumns and feed them back into the large-scale (host) model.

Increasing the number of subcolumns to 10 improves 5-year average climatology

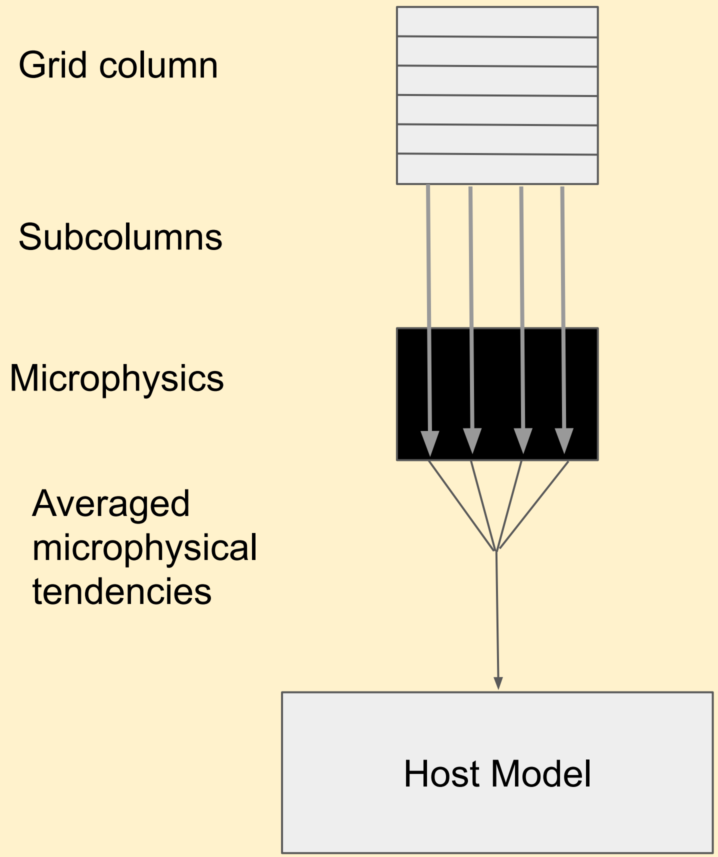


5-year average of cloud brightness. This run was tuned using 10 subcolumns. Thayer-Calder et al. (2015)

SILHS adds a layer of parallelism to climate models

On each subcolumn we perform a microphysics calculation. Then we assign the averaged tendencies back to the grid column.

Calculations on subcolumns are completely independent of each other.



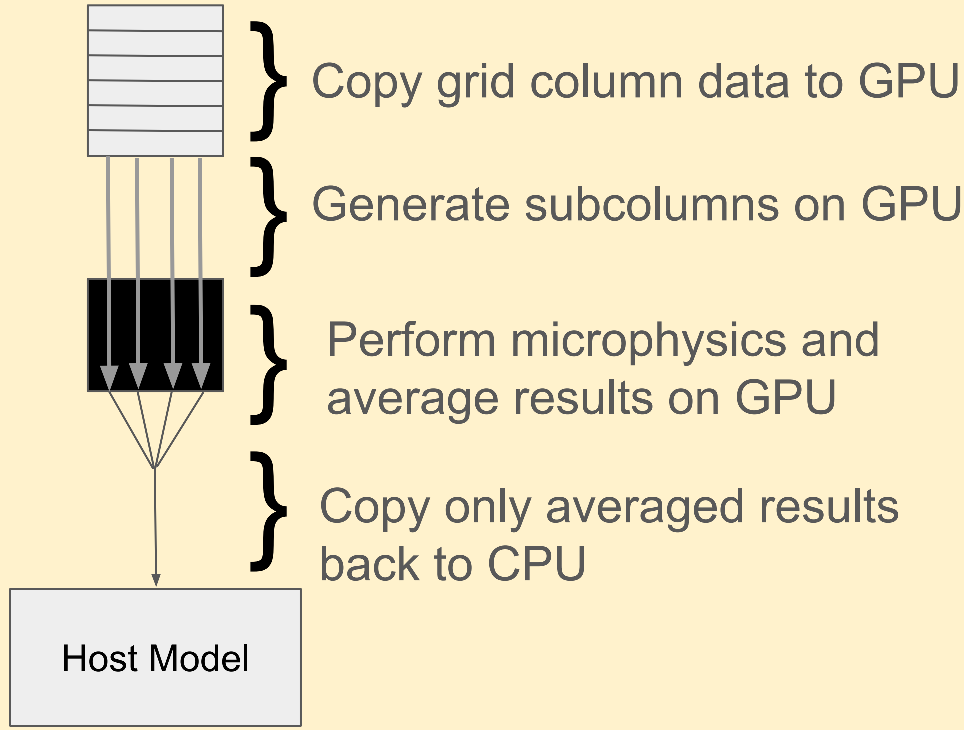
In principle, subcolumns can be computed efficiently on GPUs

Subcolumns are highly parallel:

- The only atomic step is the averaging. This is done only at the end of each physics time step (e.g., 5 min).

The subcolumn data never needs to be copied to and from the GPU.

- The subcolumns can be generated on a GPU.
- Only the grid averages need to be copied back to the CPU. The communication is minimized.



We have refactored SILHS and MG3 to run on GPUs

The code that generates subcolumns (SILHS) on the GPU is complete.

MG3 is also running on GPUs, thanks in part to help from John Dennis. (We can port P3 when it is available.)

The GPUized code has been tested for efficiency, and we’ve been able to make some performance estimates.

Estimating Relative Efficiency on Summit

We've timed the cost of generating subcolumns and calculating microphysics for a single-column simulation on a workstation with a P4000 GPU.

We've extrapolated those timing numbers to Summit, with a ne120 simulation in mind.

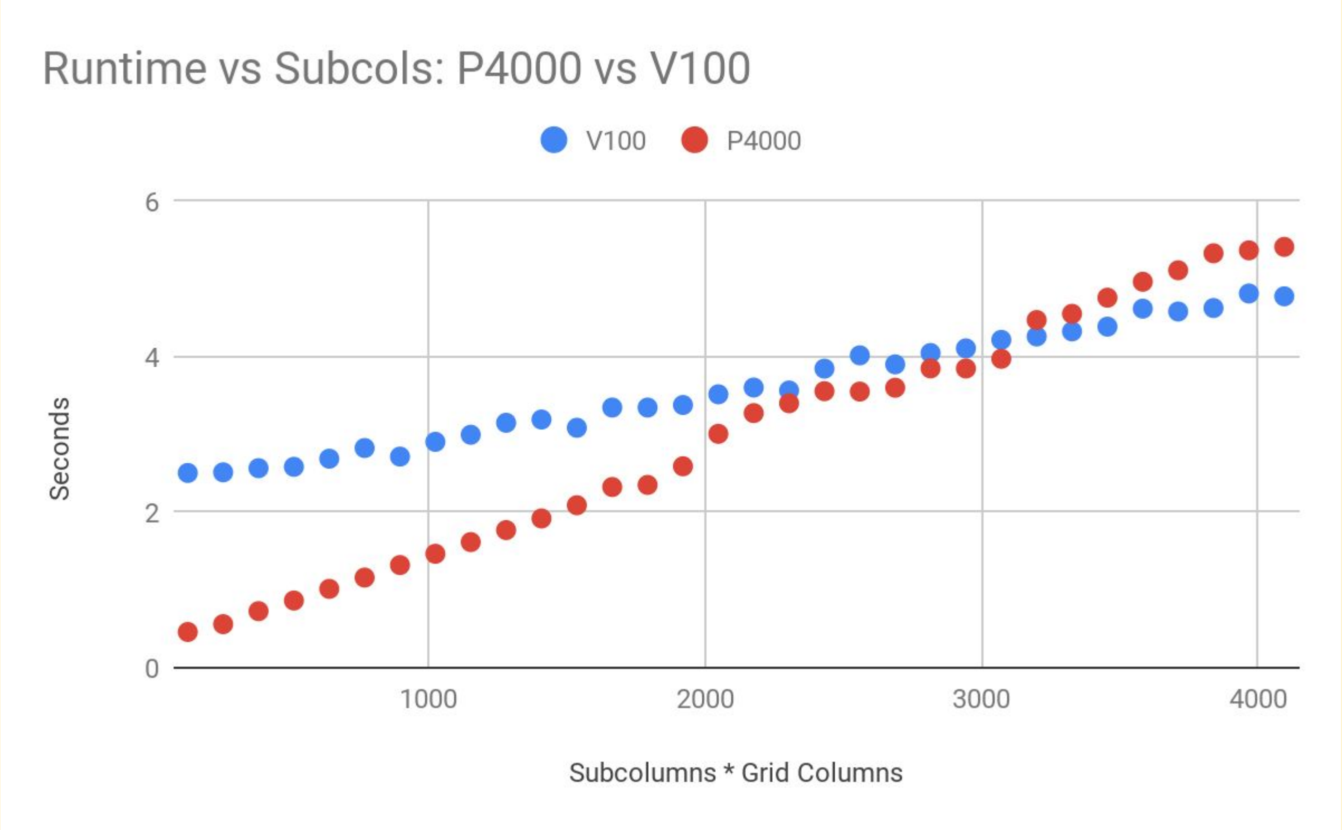
A ne120 simulation on Summit has

- 777602 grid columns / 4608 nodes = 169 grid columns / node

Each node on Summit has

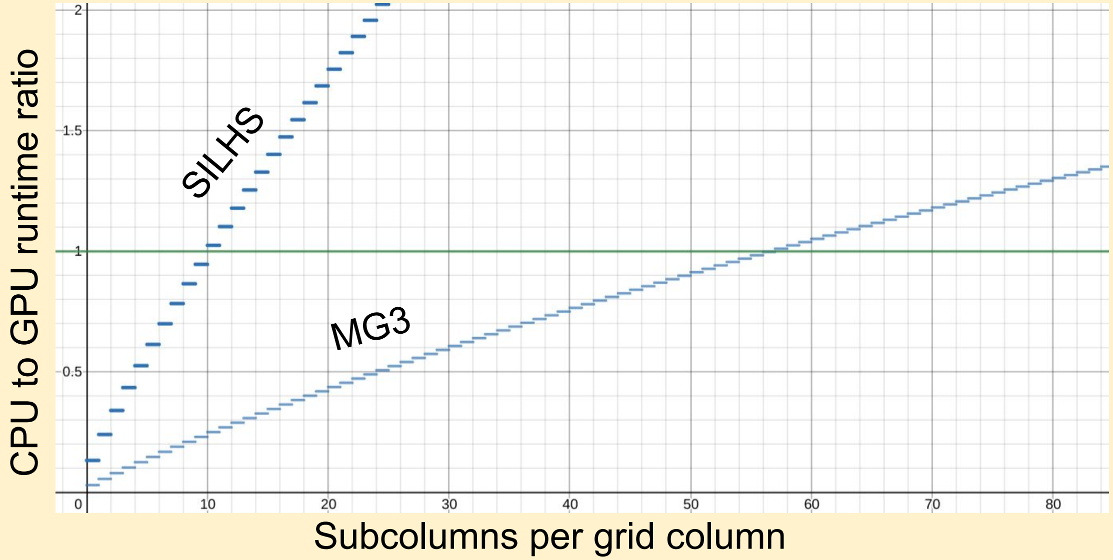
- 44 CPU cores
- 6 Nvidia V100 GPUs
- => ~1 GPU per 7 CPU cores

Comparing our Workstation GPU (P4000) with the V100 GPU that is used on Summit



V100s are less efficient than the P4000s (testing hardware) at low subcolumn numbers, but scales significantly better. This suggests that we may be able to beat estimates by a greater margin if subcolumns per grid column exceeds ~70. Again referring to a ne120 simulation using all of Summit.

Number of subcolumns beyond which Summit's GPUs are faster than its CPUs



Using trendline estimates of GPU performance, we're able to compare runtimes and estimate that, on Summit with a ne120 simulation

- SILHS becomes more efficient on GPUs than CPUs @ 10 subcolumns per grid column
- MG3 becomes more efficient on GPUs than CPUs @ 57 subcolumns per grid column

Efficiency analysis: Interpretation

The additional cost of using a fully GPUized version of SILHS or MG3 cannot be evaluated using these results.

These tests demonstrate that we can use GPUs to beat CPU performance while using a reasonable number of subcolumns (10-70).

As GPU architectures improve and become more powerful, we will be able to either:

- Maintain subcolumn counts and see increases in throughput
- Increase subcolumn counts while maintaining throughput

Recap

Use of subcolumns can improve the accuracy of existing climate models while still maintaining throughput. This provides a way to use GPU-based supercomputers to perform long-term climate simulations.

As a prototype, we're calculating microphysics on subcolumns, but in the future we could calculate radiation and aerosols on subcolumns as well.

There's still work to be done, but initial tests are promising. Contact us if you'd like to collaborate with us on this project!