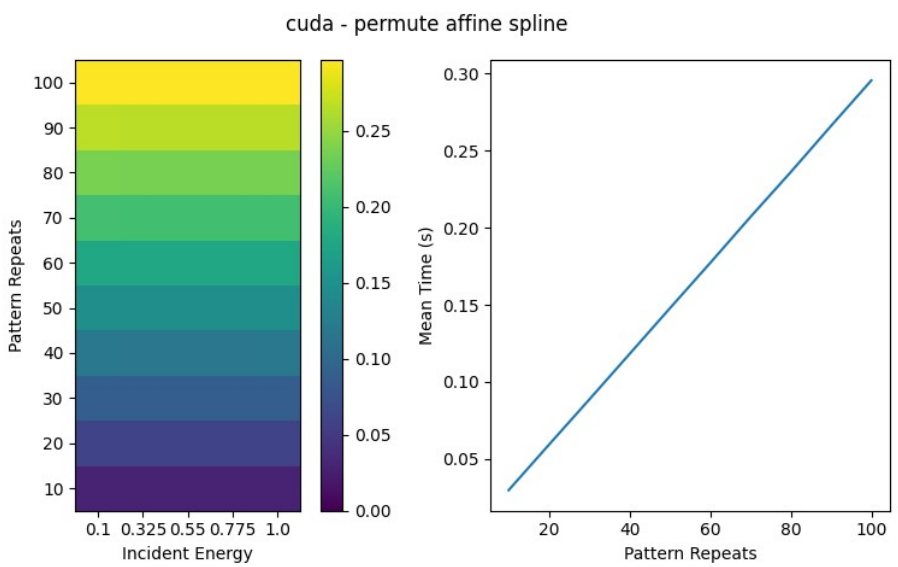
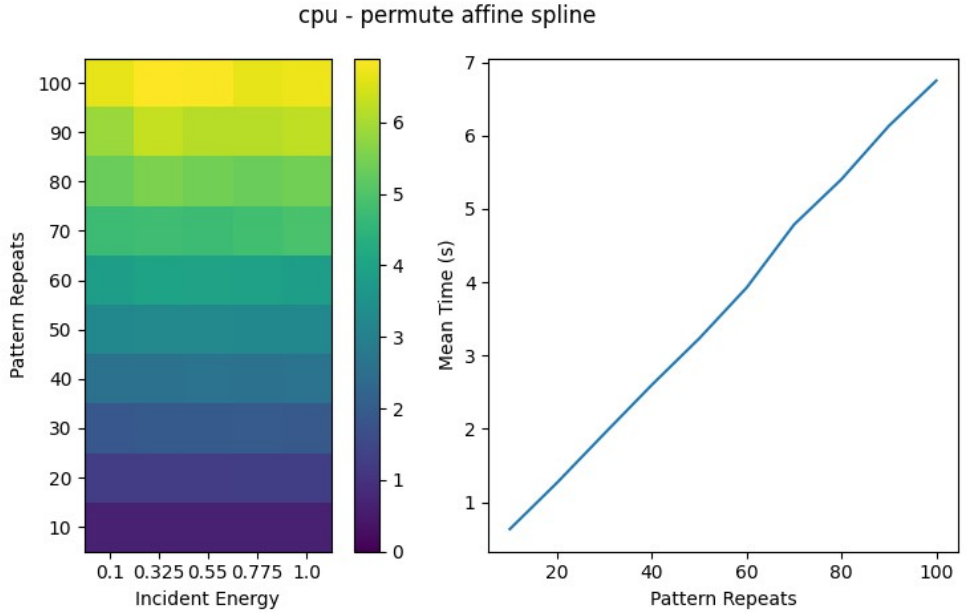
Showerflow

Some notes and timings

- Just checking it’s worth trying to reduce the number of operations in showerflow. Process is;
  - Chose a sequence of operations (affine coupling, spline, permutation)
  - Make flow models that repeat this sequence n times, where n = 10, 20, … 100
  - Without training, generate 10k points from the flow model. Try this 10 times, but only average the last 9 rounds (avoid possible setup costs).
  - Try for 5 different conditioning values. Shouldn’t really make a difference?
- Assumption is that training won’t make a difference either.

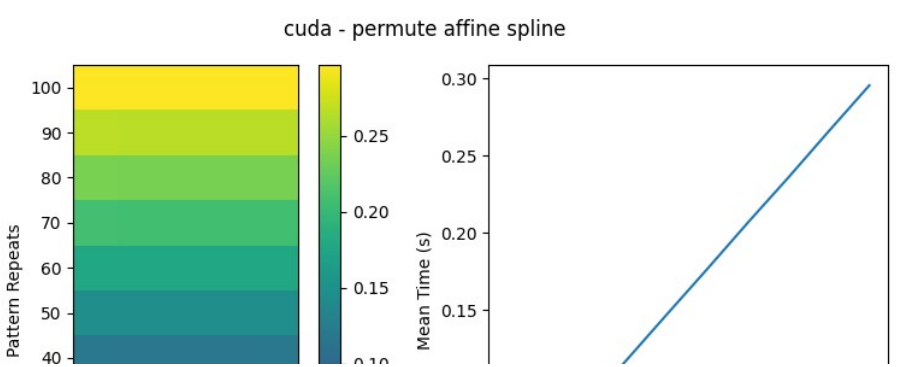
Do fewer operations run faster?

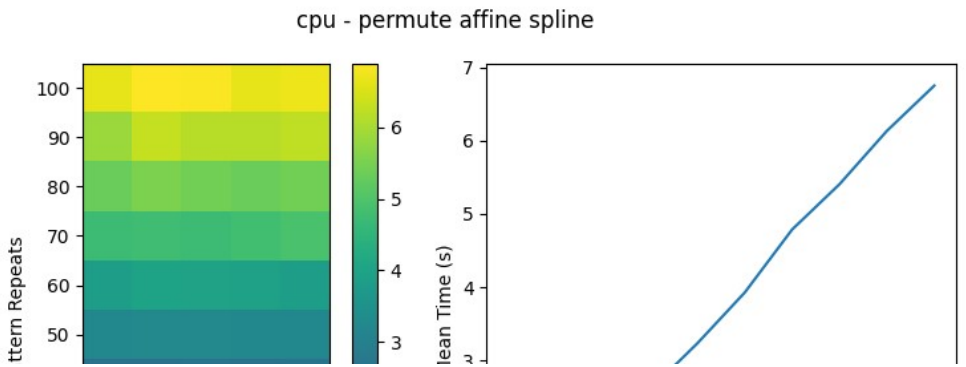


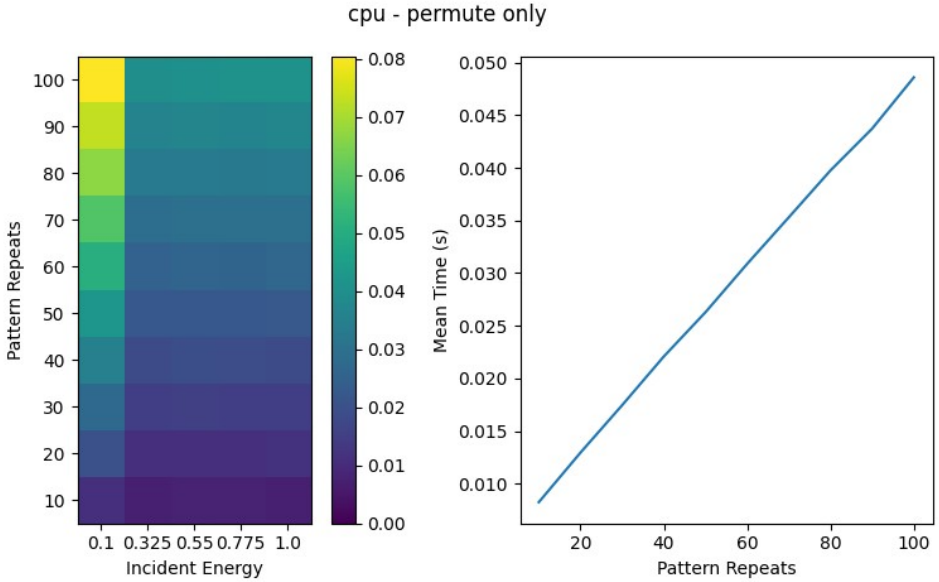


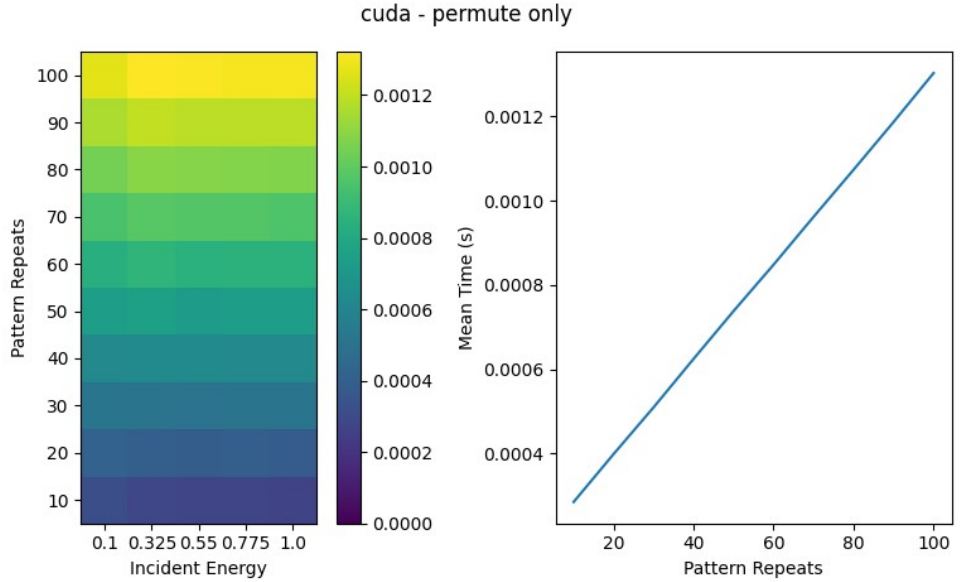
Run time vs number of operations appears to be linear on both CPU and GPU

Do fewer operations run faster?



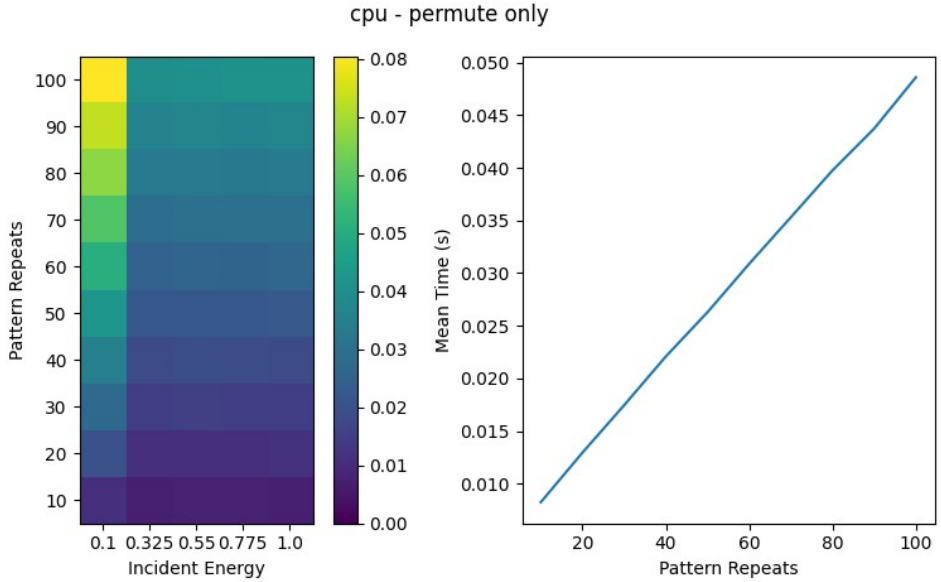


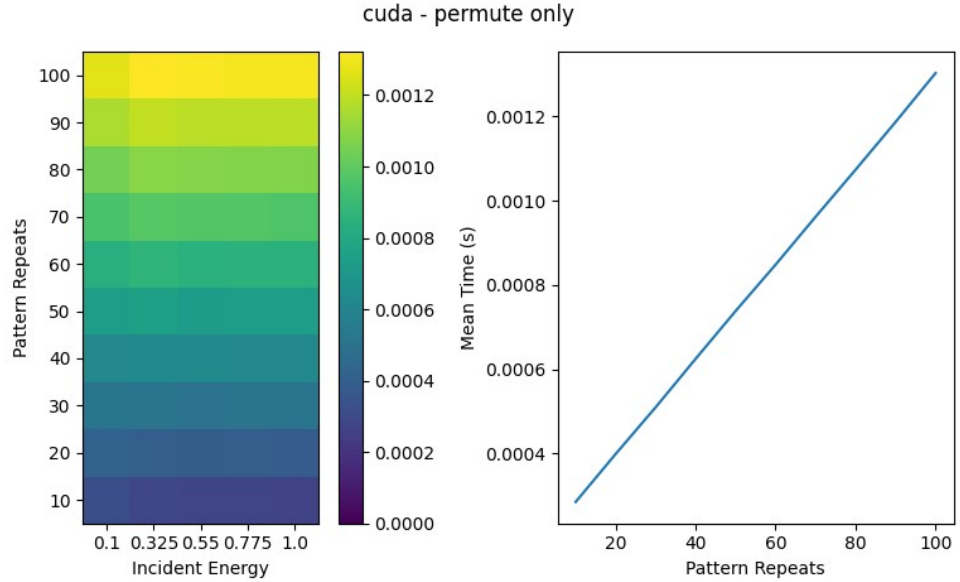




Do fewer operations run faster?

Something odd happens when the conditioning value is close to 0. Maybe need trained models to get definitive results.





Do fewer operations run faster?

Showerflow is a strange probability distribution

- Values it predicts;
  - 0) Total clusters in event (normalised for dataset)
  - 1) Total visible energy in event (normalised for dataset)
  - 2) Center of gravity in x
  - 3) Center of gravity in y
  - 4) Center of gravity in z
  - 5:35) Clusters per layer (normalised for dataset)
  - 35:65) Energy per layer (normalised for dataset)

0 has rigid constraints from 5:35

1 has a rigid constraints from 35:65

3 has rigid constraints 5:65

Anywhere these constraints don't get satisfied, formally the log_prob should be -inf.
That would be impossible to train with.