chunking

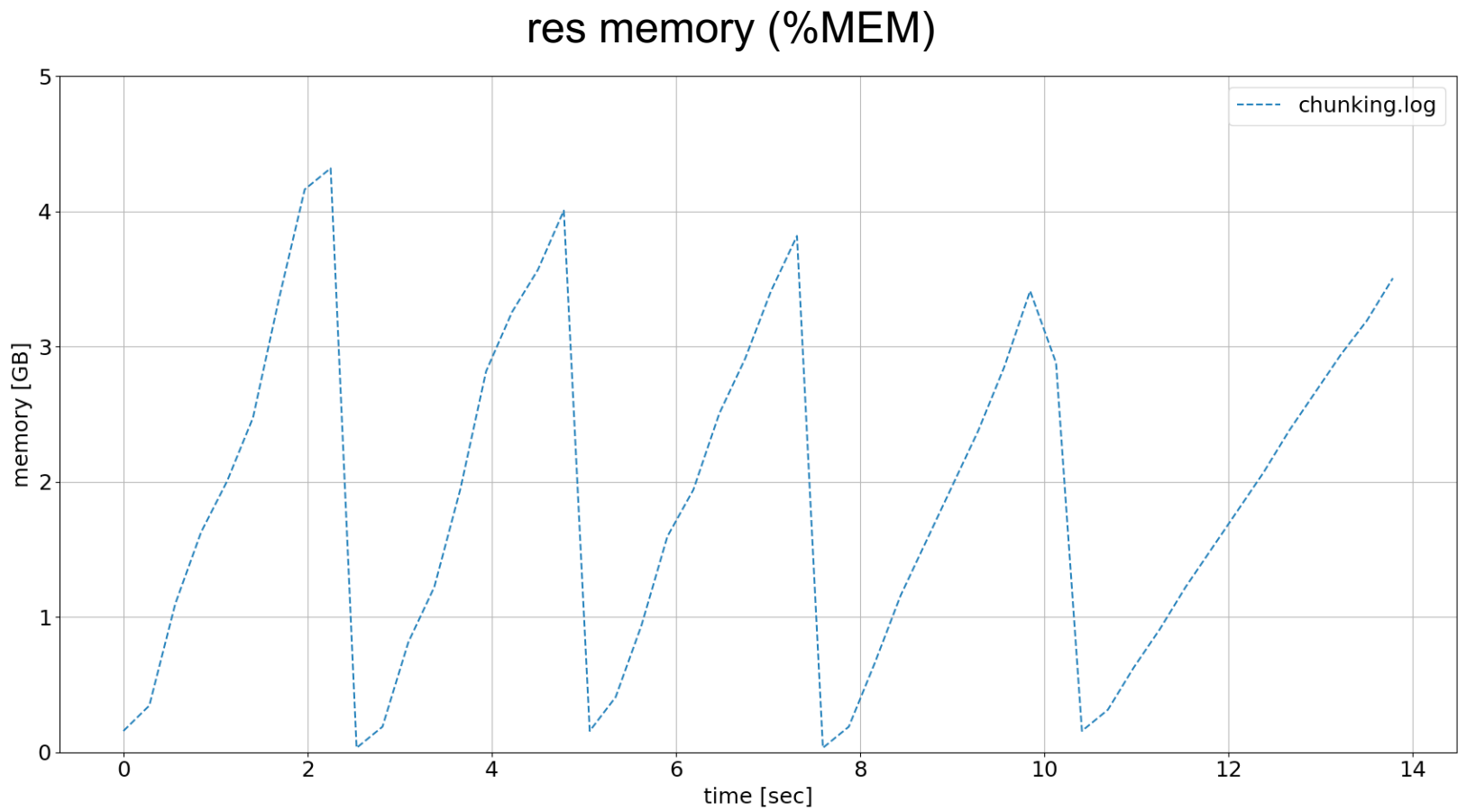
pytorch/src/DNNROIFinding.cxx

```
iframe (tags) -> Eigen -> Eigen (downsample) -> Tensor ->

torch.chunk(nchunk, dim)
loop chunks {
  torch::jit::IValue -> ITensorSet -> (inferencing) ITensorSet -> Tensor
}
torch.cat{{}, dim}

-> Eigen -> Eigen (upsample)
```

Memory for 1, 2, 4, 8,100 chunking

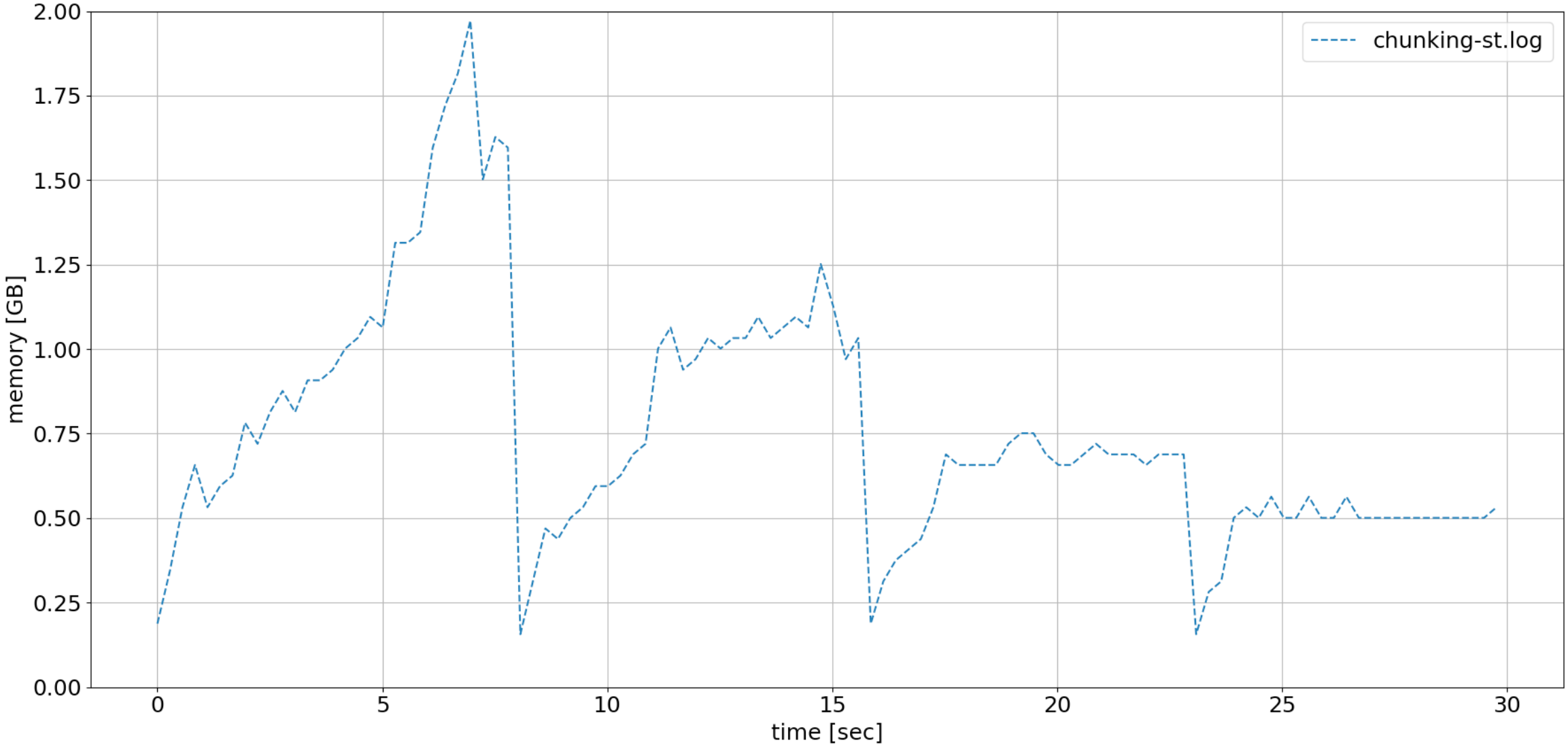
ExeMon:

```
 16 Memory: MEM: total: size=1.10656e+07K, res=5.06988e+06K increment: size=8.70751e+06K, res=4.87017e+06K inference
 35 Memory: MEM: total: size=9.09029e+06K, res=4.45606e+06K increment: size=6.73219e+06K, res=4.25552e+06K inference
 58 Memory: MEM: total: size=8.05338e+06K, res=4.10013e+06K increment: size=5.69553e+06K, res=3.89966e+06K inference
 89 Memory: MEM: total: size=7.57813e+06K, res=3.96266e+06K increment: size=5.22004e+06K, res=3.763e+06K inference
304 Memory: MEM: total: size=7.01704e+06K, res=3.71398e+06K increment: size=4.65894e+06K, res=3.51409e+06K inference
```



Brookhaven National Laboratory

chunking 1, 2, 4, 8, `torch::NoGradGuard no_grad;`



Brookhaven National Laboratory

Gray's fix + no_grad

- sbndgpvm04
- ref: WCT 0.27.1
- new: Gray's fix + ~~no_grad~~ (BV already fixed it!)
- chunking
- found a bug

```
35  void Pytorch::TorchService::configure(const WireCell::Configuration& cfg)
36  {
37      auto dev = get<std::string>(cfg, "device", "cpu");
38      m_ctx.connect(dev);
39
40      auto model_path = Persist::resolve(cfg["model"].asString());
41      if (model_path.empty()) {
42          log->critical("no TorchScript model file provided");
43          THROW(ValueError() << errmsg{"no TorchScript model file provided"});
44      }
45
46      // Use almost 1/2 the memory and 3/4 the time.
47      torch::NoGradGuard no_grad;
```

	Total time, sec	OSP time, sec	dnn time, sec	VmHWM, GB	
ref, nodnn	28			1.4	
ref, dnn	113		19*2=38	7.7	113-28-38=47
new, dnn	108		19*2=38	7.8	
new, dnn, 3500	70		10*2=20	5.2	
new, dnn, 3500, 4chunks	80		~21	2.4	
new, dnn, 3500, 8chunks				1.6	

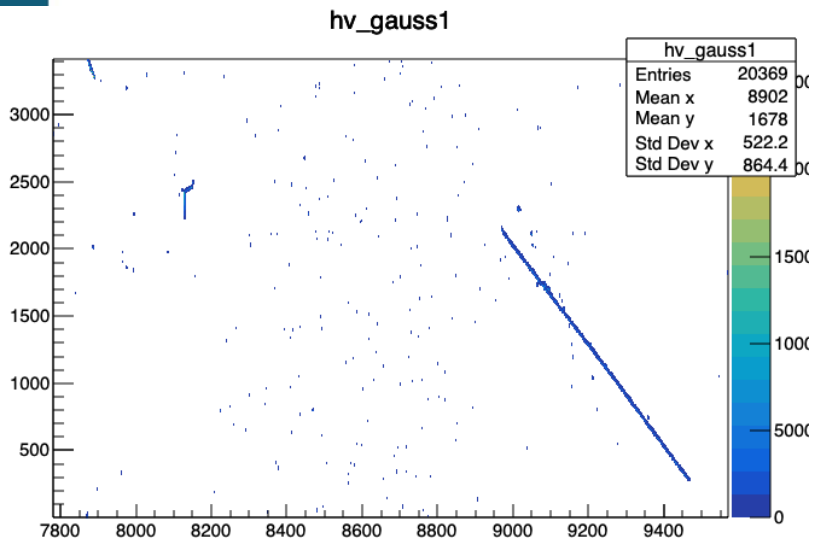
~47sec in preparing additional output in OSP?

- reduce OSP resolution

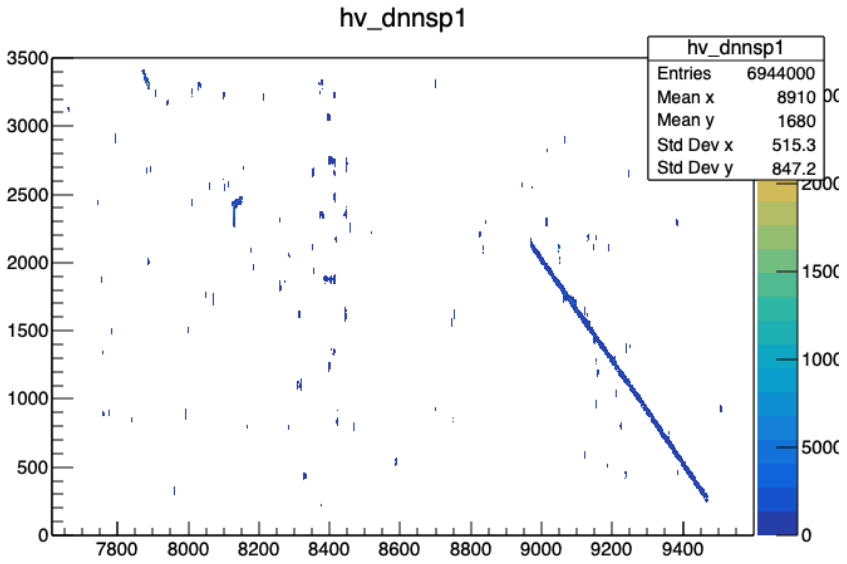
Brookhaven National Laboratory

Still bugs to fix

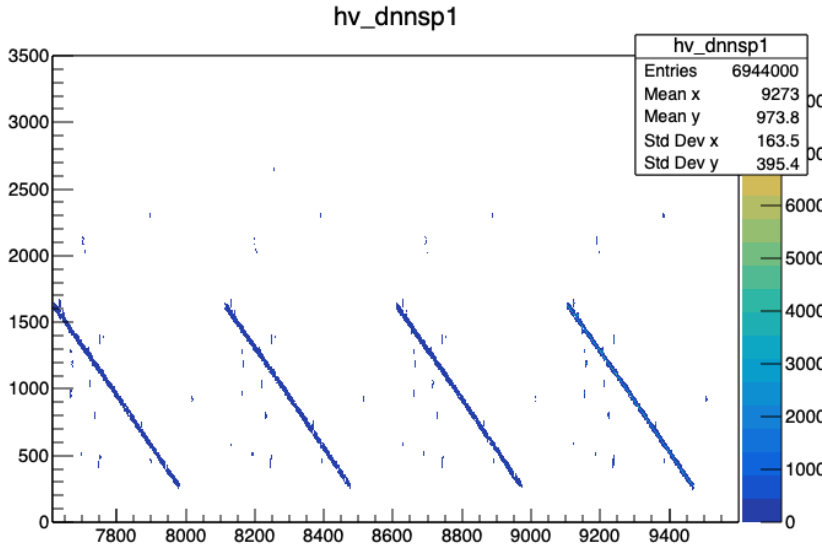
ref-nodnn



new-3500



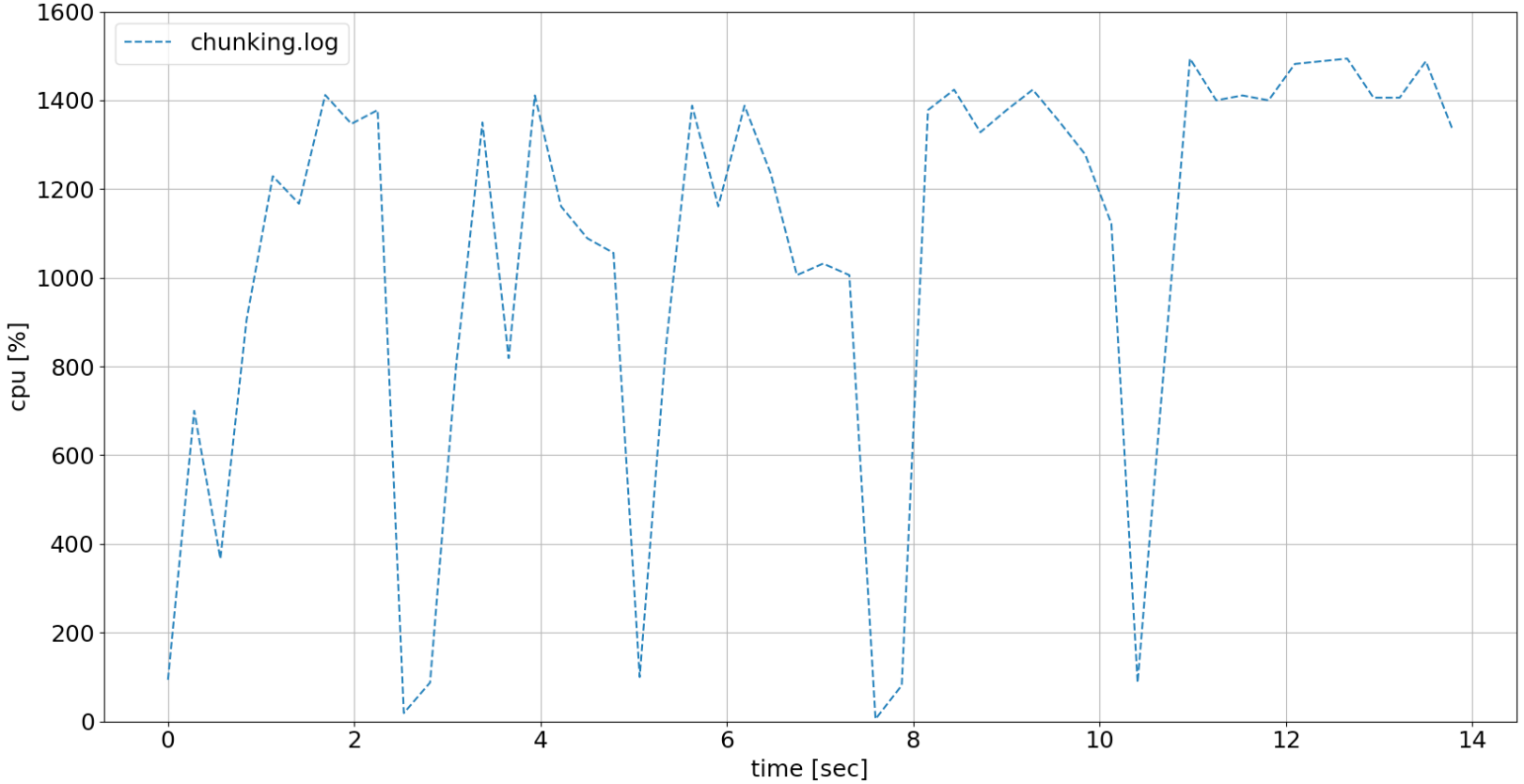
new-3500-4chunks



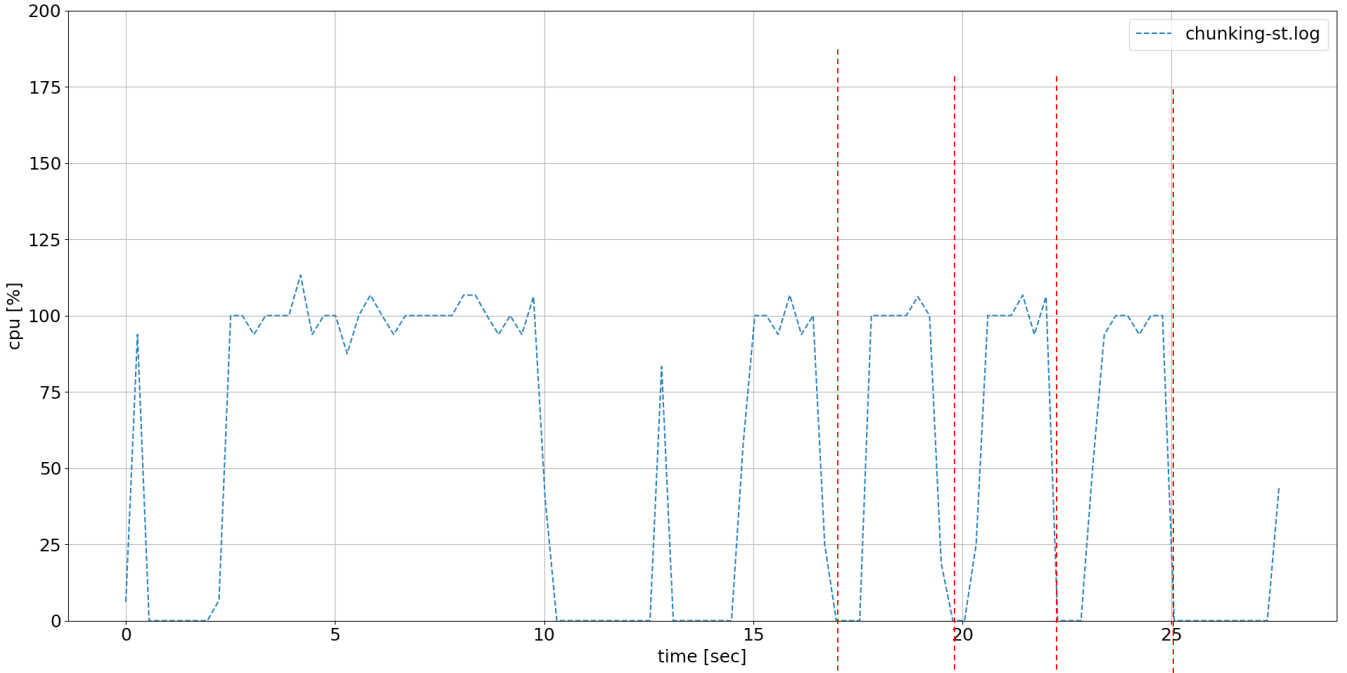
Brookhaven National Laboratory

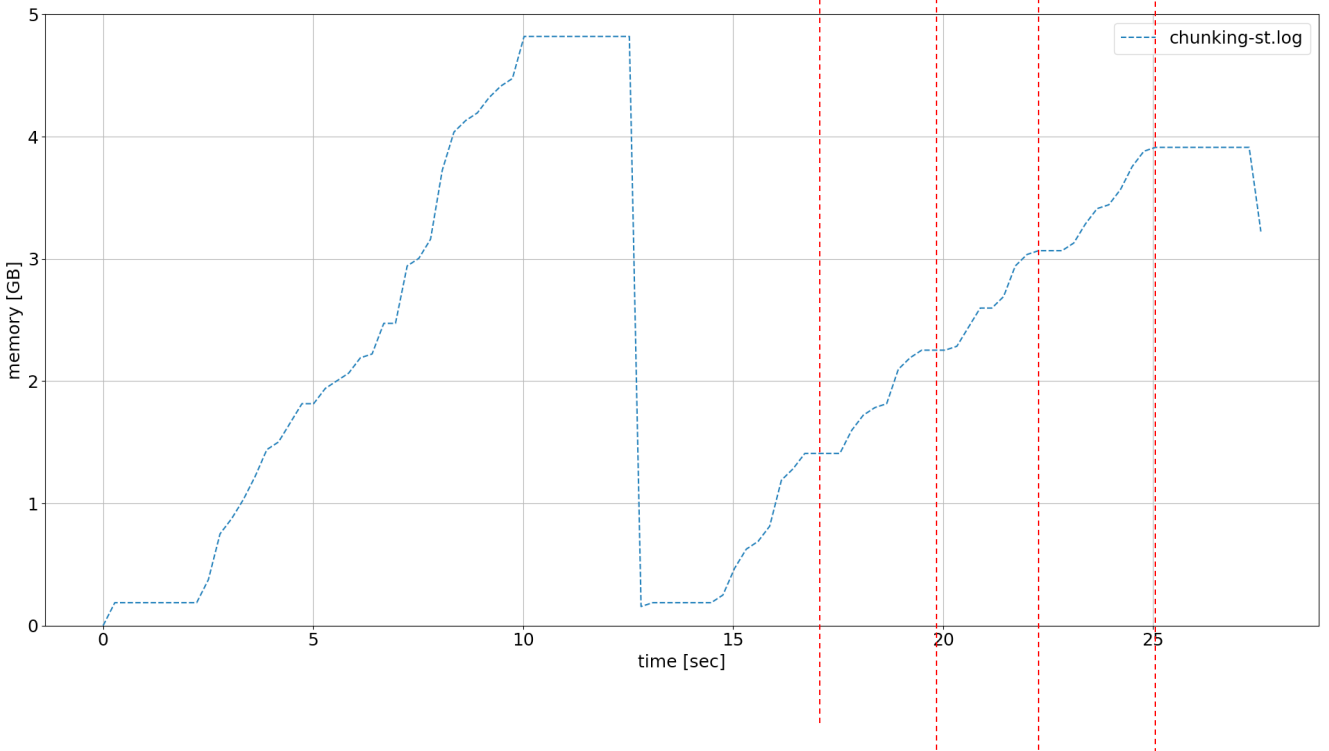
backups

Memory for 1, 2, 4, 8,100 chunking



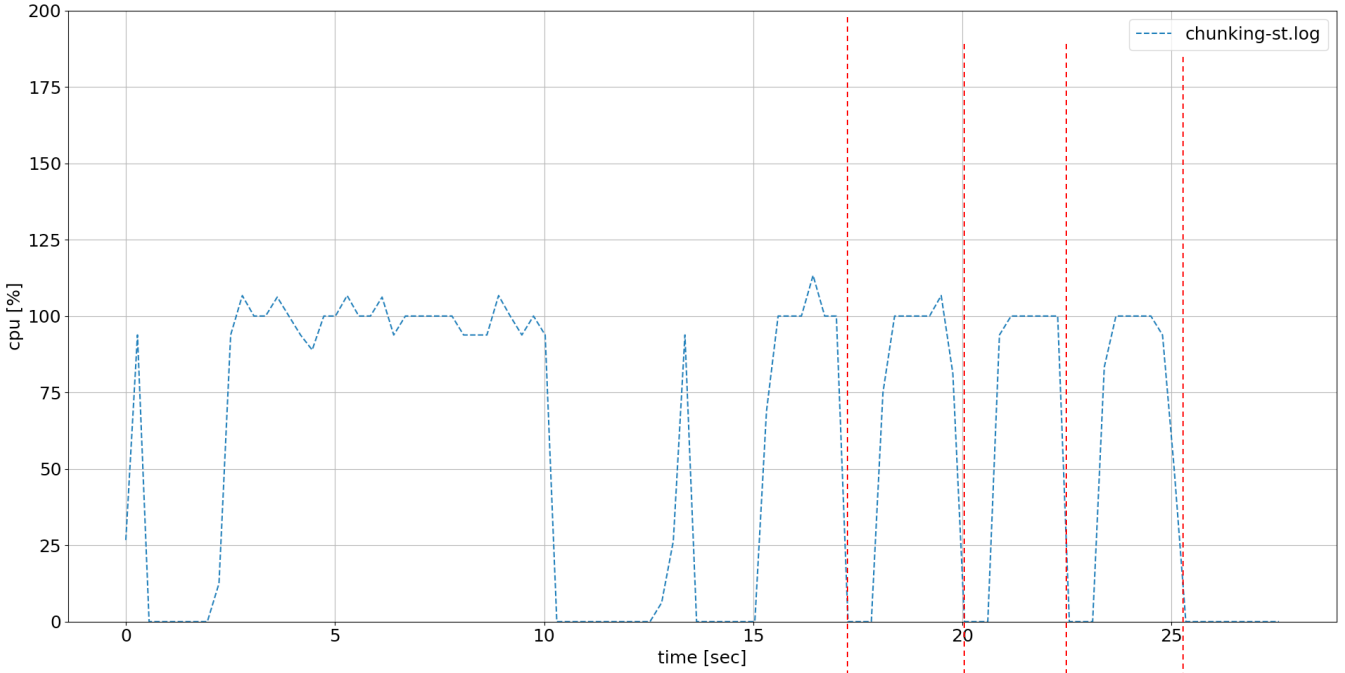
orig

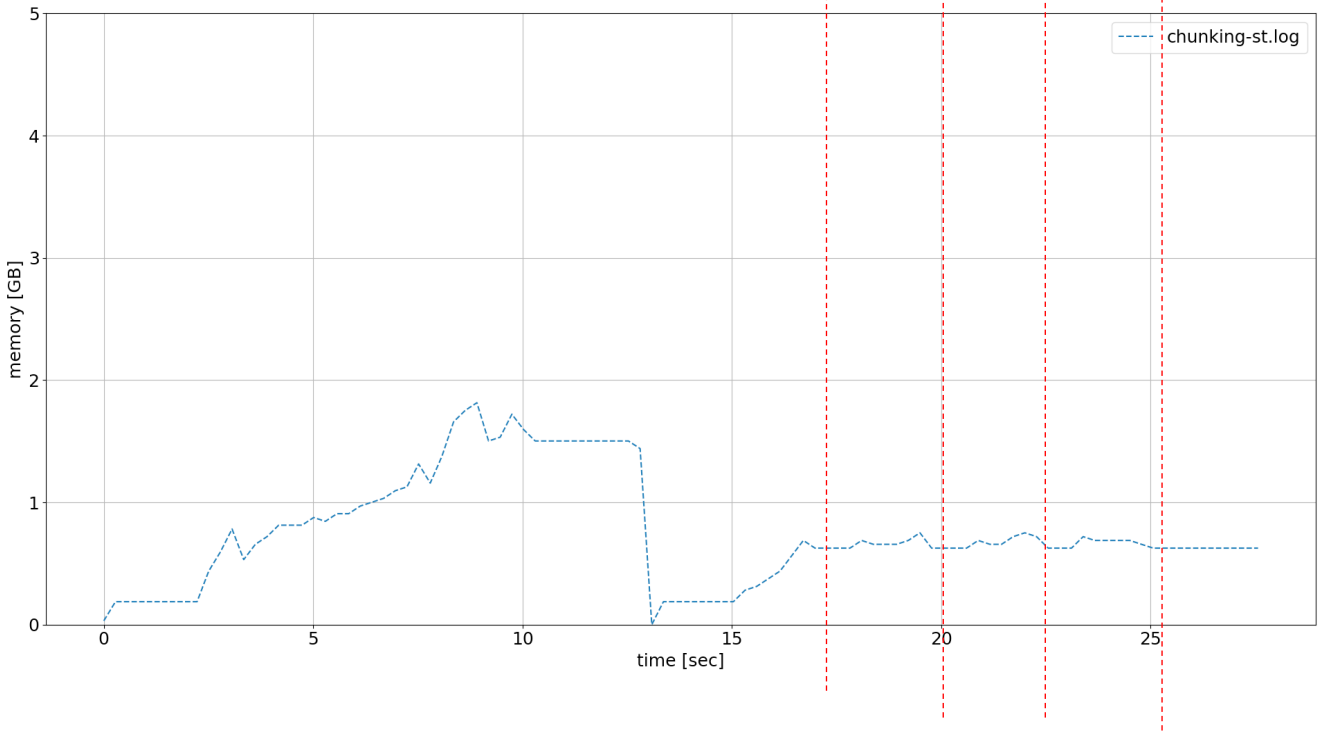




Brookhaven National Laboratory

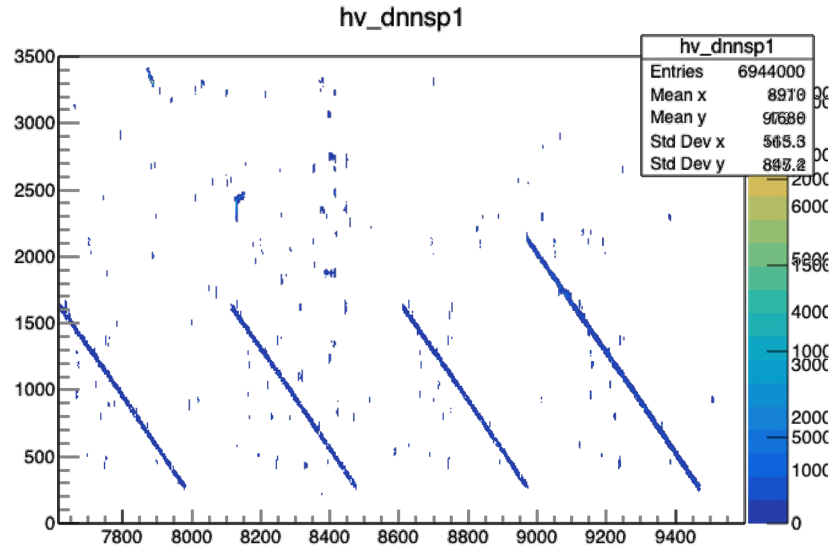
```
torch::NoGradGuard no_grad;
```





Brookhaven National Laboratory

eigen after scaling: vector of 3 of 1984x340
chunking: n x 3 x 1984/n x 340


hv_dnnsp1
hv_dnnsp1
Entries 6944000
Mean x
Mean y
Std Dev x
Std Dev y
3500
3000
2500
2000
1500
1000
500
0
7800
8000
8200
8400
8600
8800
9000
9200
9400