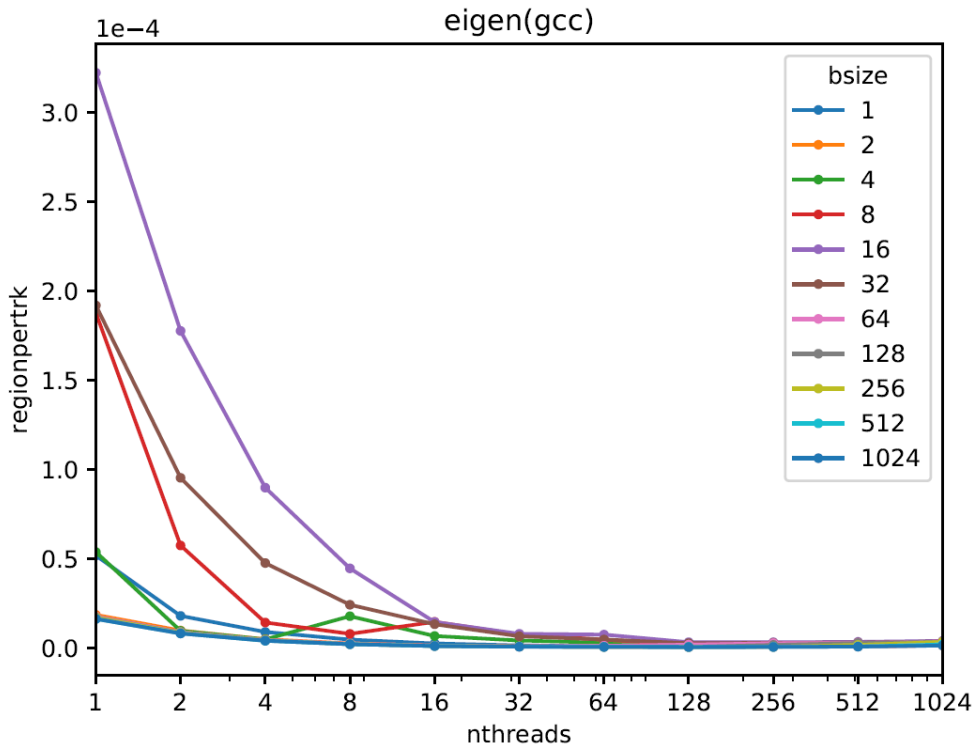
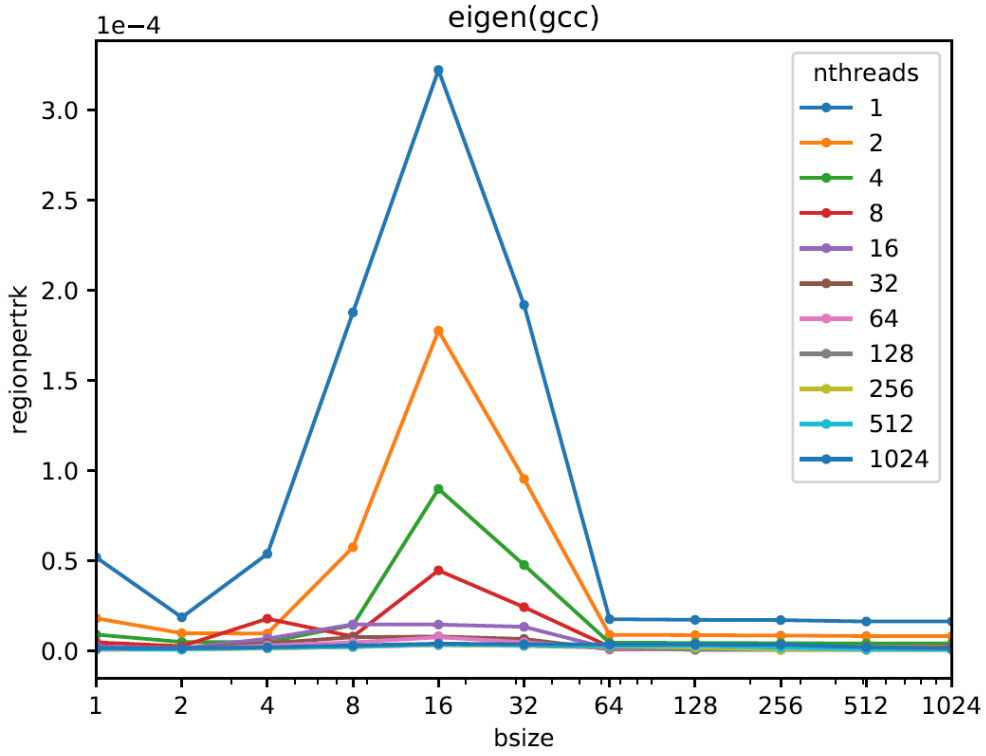
Thread scaling

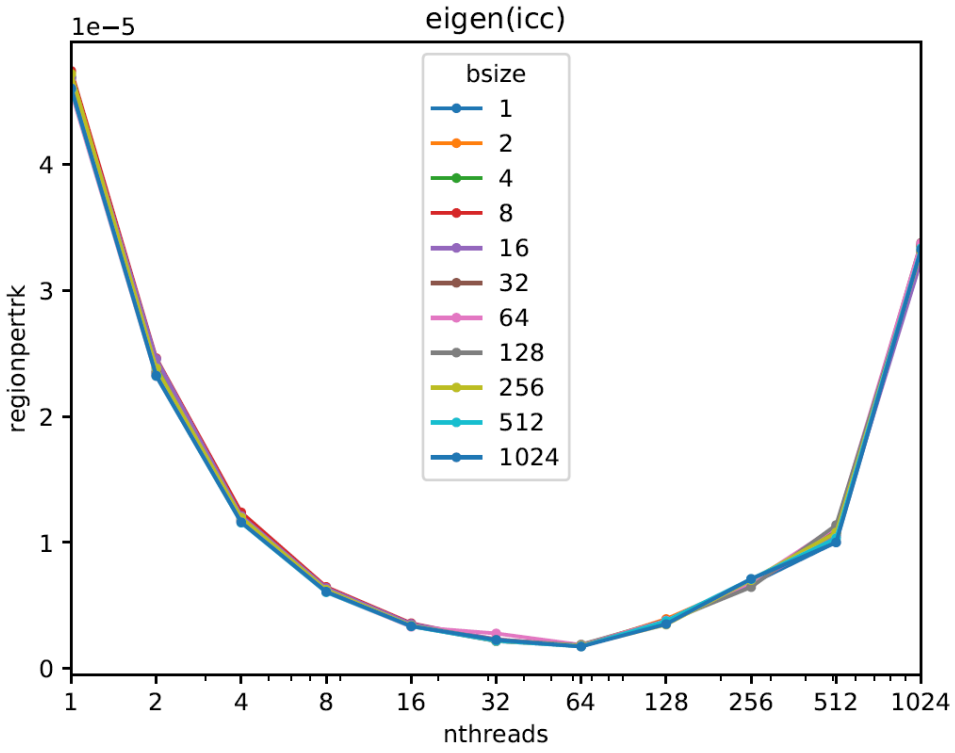
Tres Reid

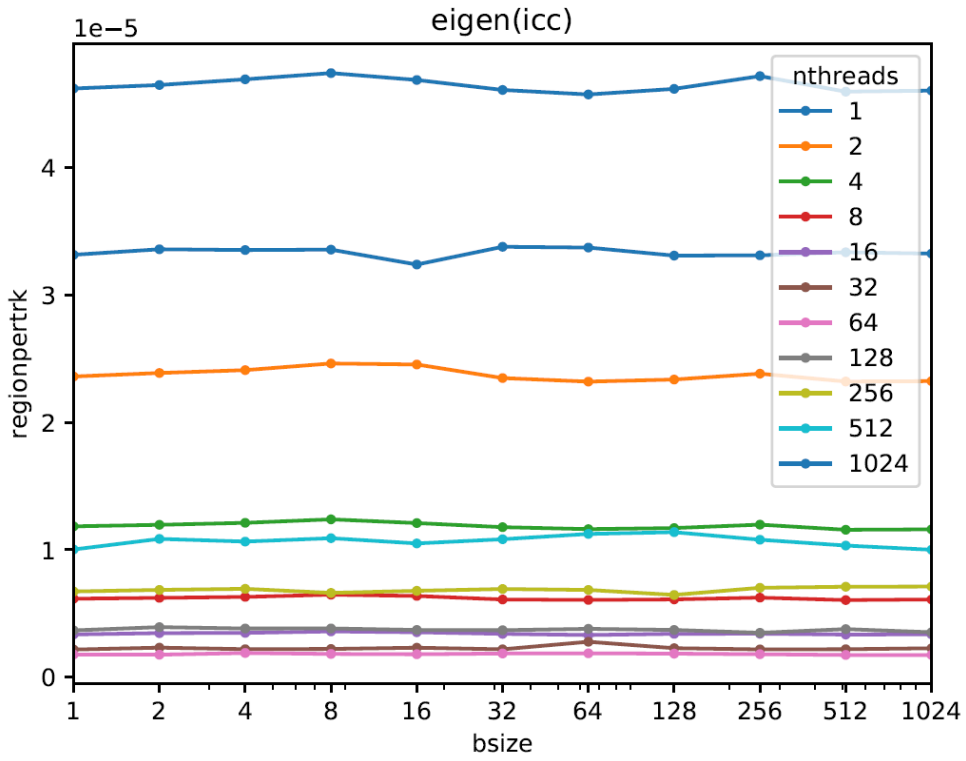
Eigen(gcc)


eigen(gcc)
1e−4
regionpertrk
nthreads
bsize
1
2
4
8
16
32
64
128
256
512
1024

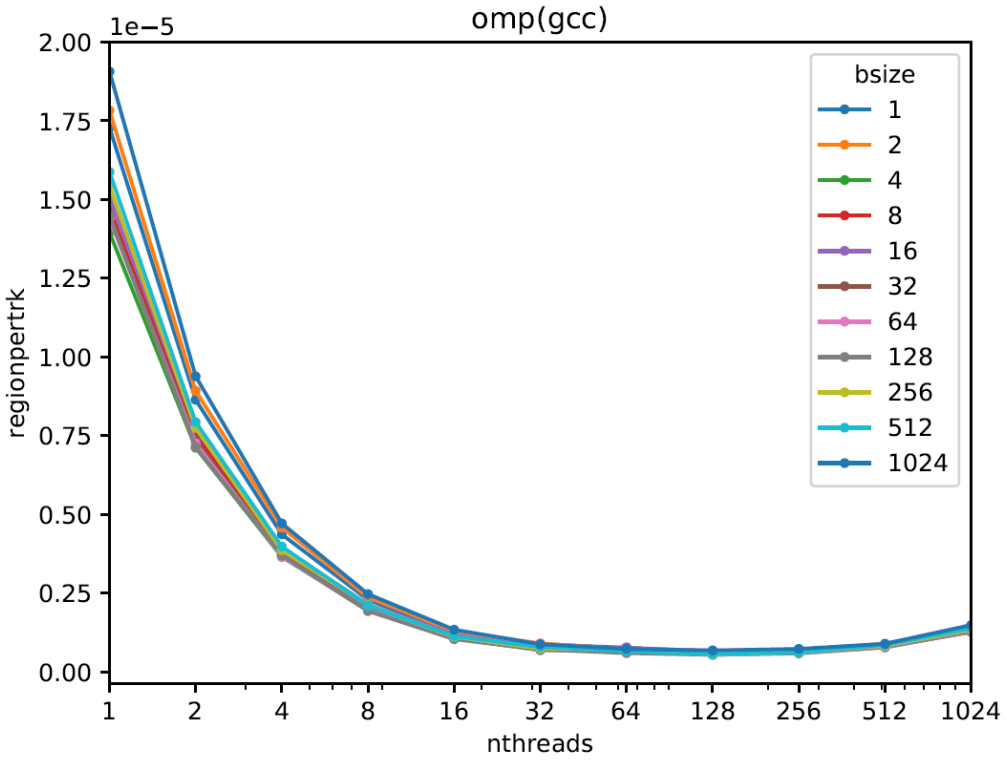

eigen(gcc)
1e−4
regionpertrk
bsize
nthreads
1
2
4
8
16
32
64
128
256
512
1024

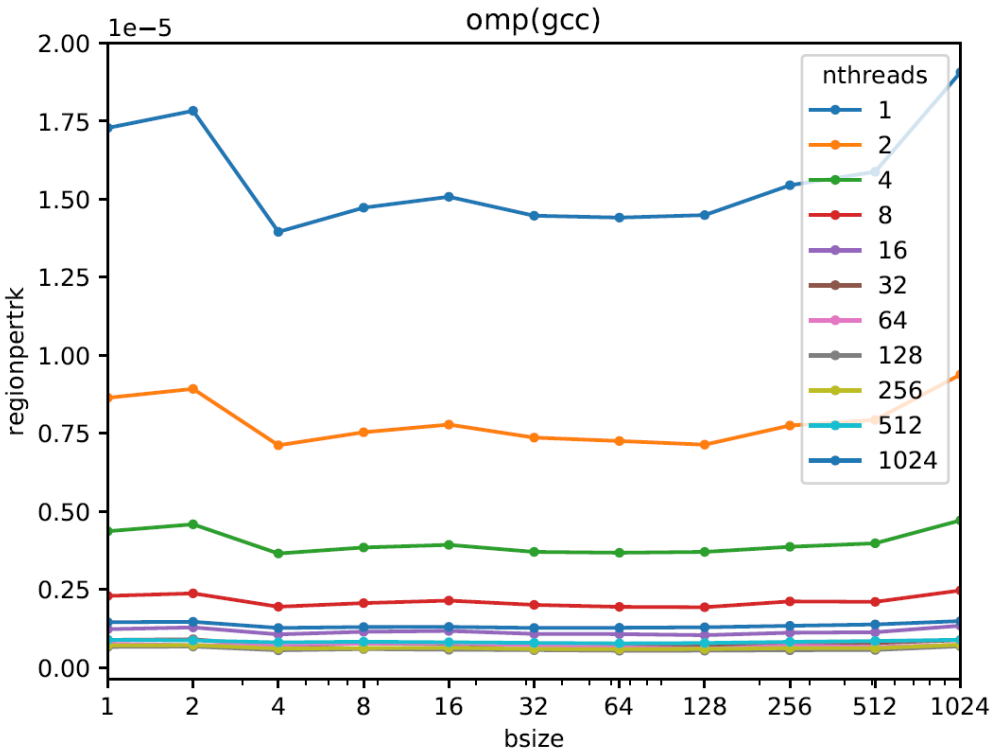
Eigen(icc)


eigen(icc)
1e−5
bsize
1
2
4
8
16
32
64
128
256
512
1024
regionpertrk
nthreads
0
1
2
3
4
1
2
4
8
16
32
64
128
256
512
1024

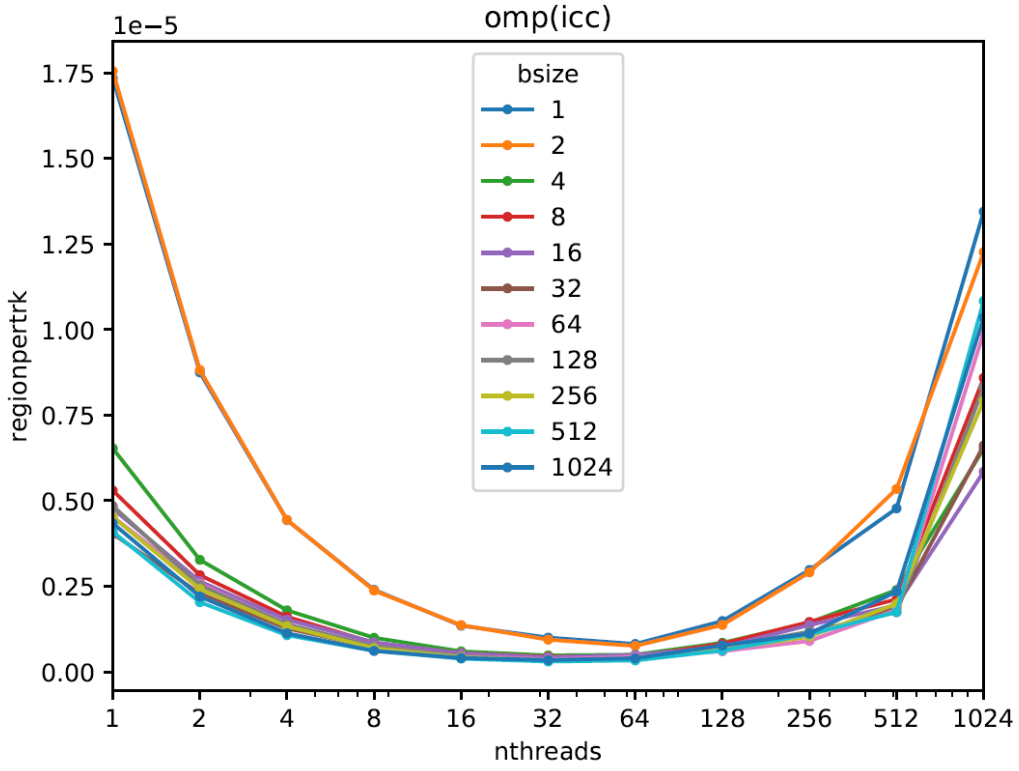

eigen(icc)
1e−5
nthreads
1
2
4
8
16
32
64
128
256
512
1024
regionpertrk
bsize
0
1
2
3
4
1
2
4
8
16
32
64
128
256
512
1024

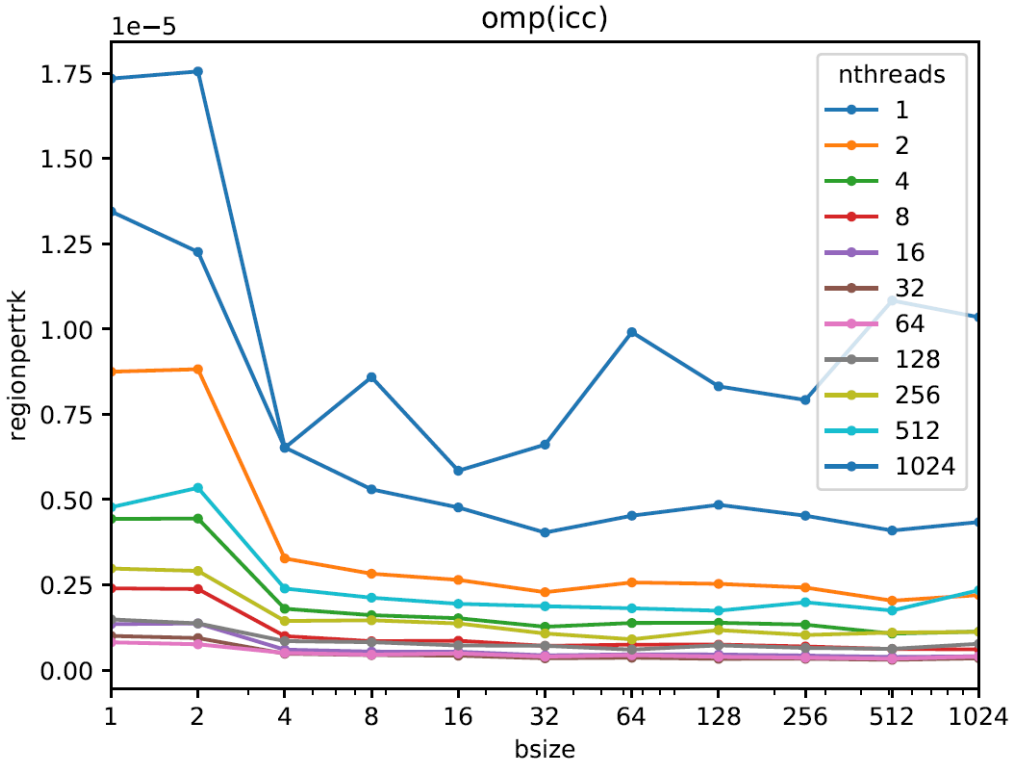
OMP(gcc)


omp(gcc)
1e−5
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
regionpertrk
1
2
4
8
16
32
64
128
256
512
1024
nthreads
bsize
1
2
4
8
16
32
64
128
256
512
1024

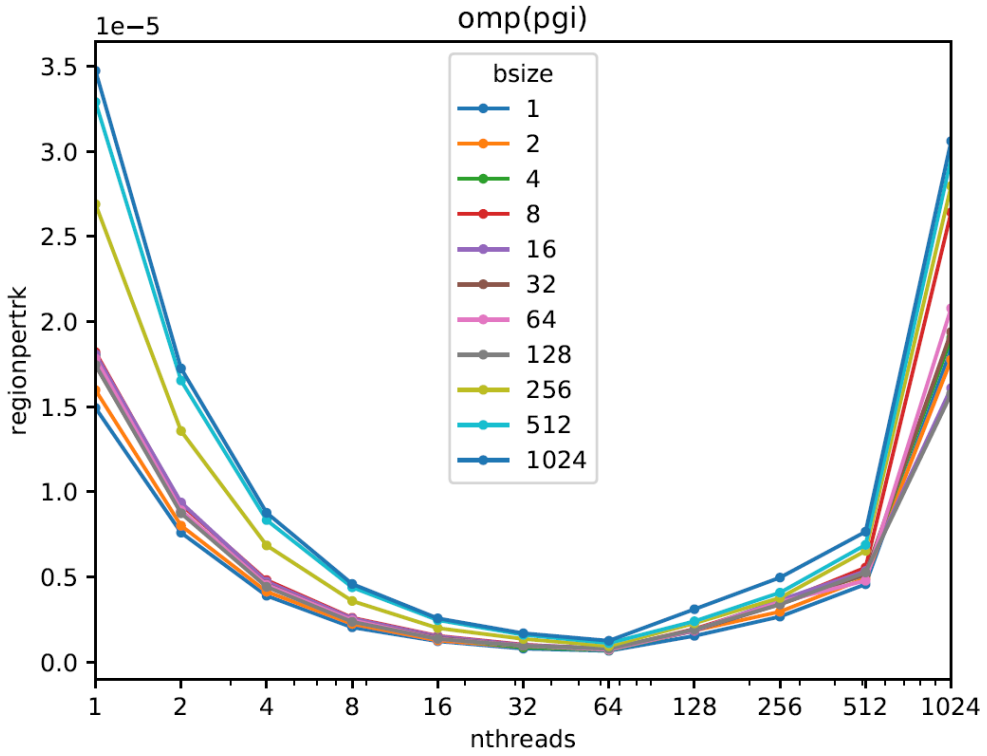

omp(gcc)
1e−5
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
regionpertrk
1
2
4
8
16
32
64
128
256
512
1024
bsize
nthreads
1
2
4
8
16
32
64
128
256
512
1024

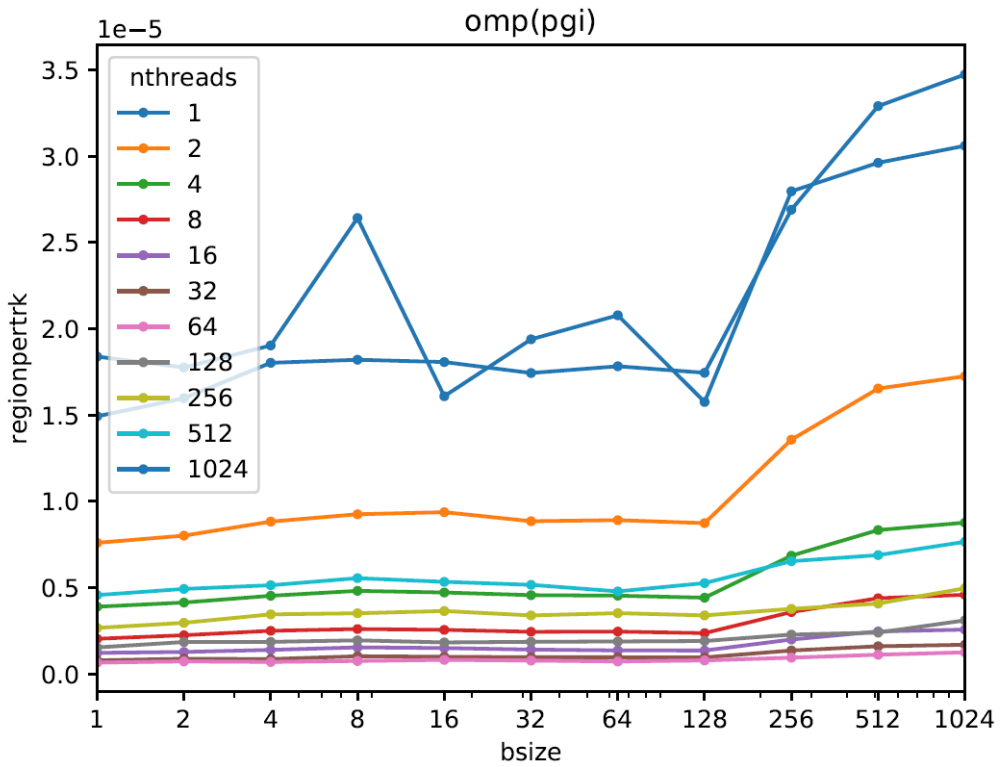
OMP(icc)


omp(icc)
1e−5
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
regionpertrk
bsize
1
2
4
8
16
32
64
128
256
512
1024
1
2
4
8
16
32
64
128
256
512
1024
nthreads

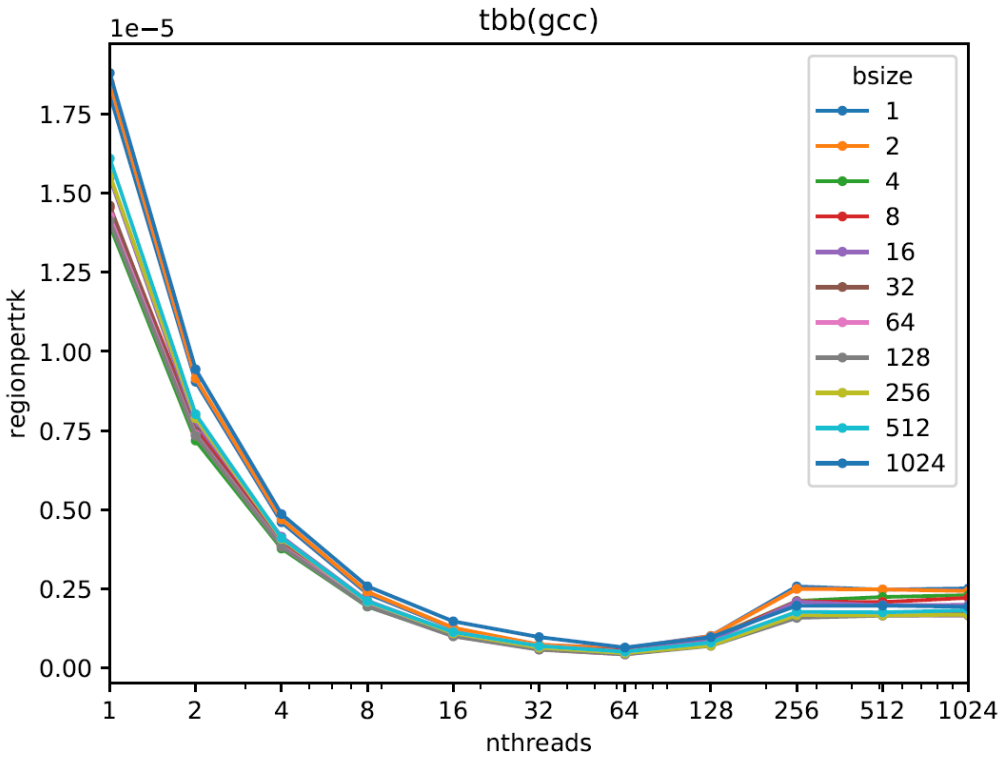

omp(icc)
1e−5
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
regionpertrk
nthreads
1
2
4
8
16
32
64
128
256
512
1024
1
2
4
8
16
32
64
128
256
512
1024
bsize

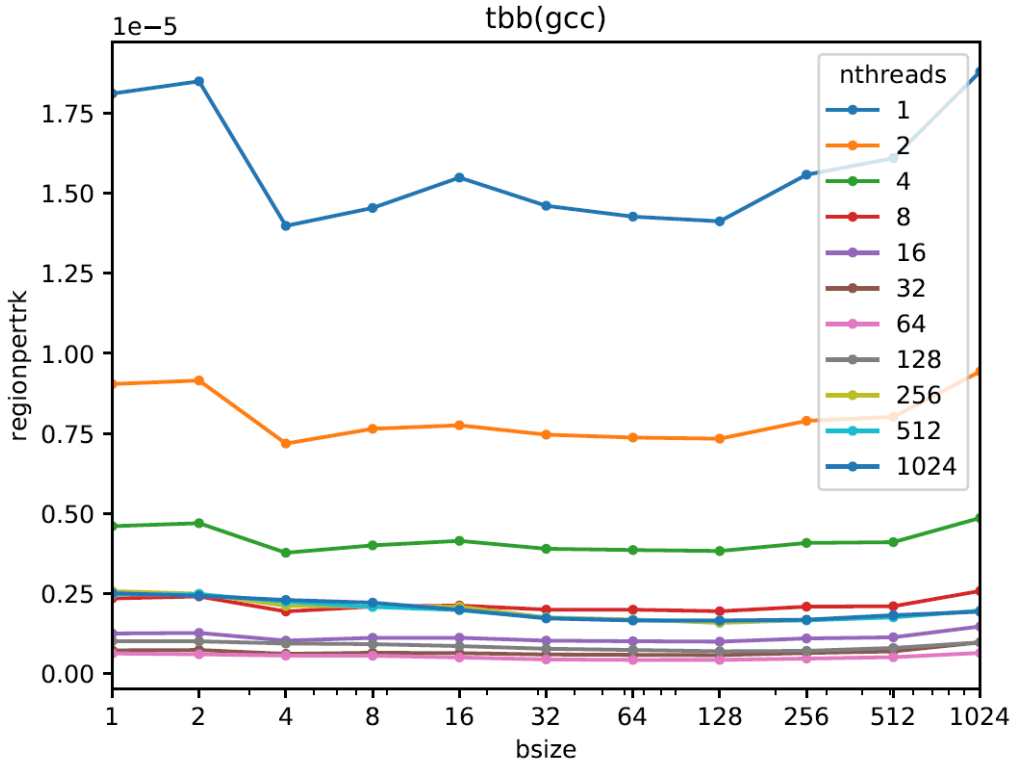
OMP(pgi)


omp(pgi)
1e−5
regionpertrk
nthreads
bsize
1
2
4
8
16
32
64
128
256
512
1024

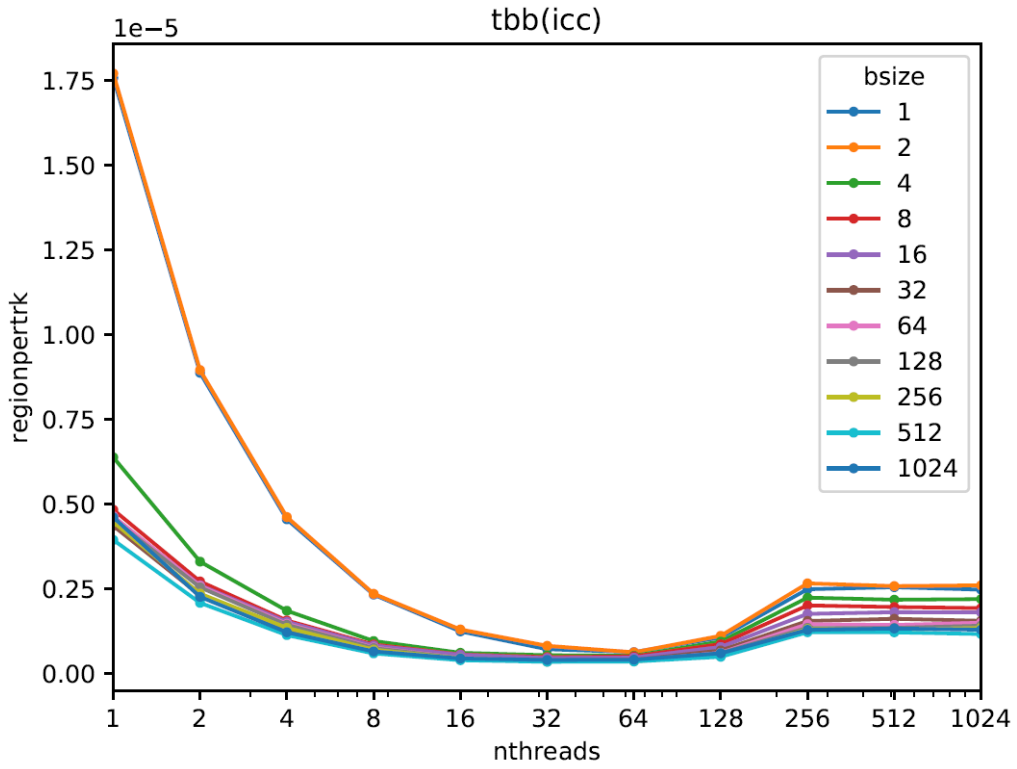

omp(pgi)
1e−5
regionpertrk
bsize
nthreads
1
2
4
8
16
32
64
128
256
512
1024

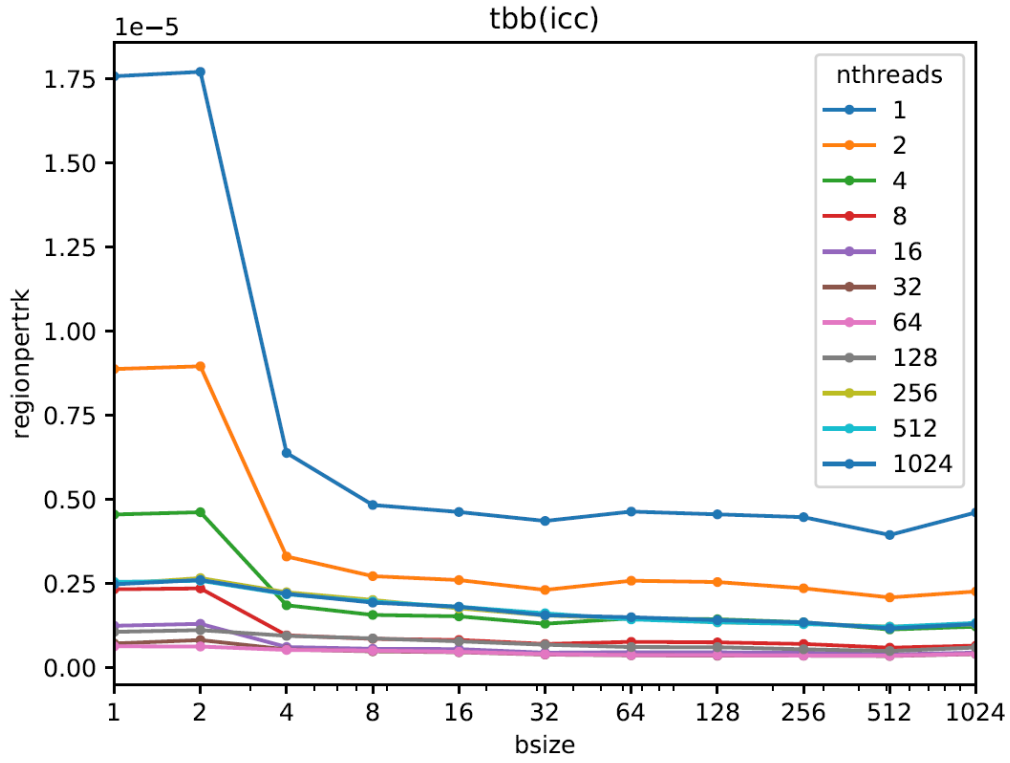
TBB(gcc)

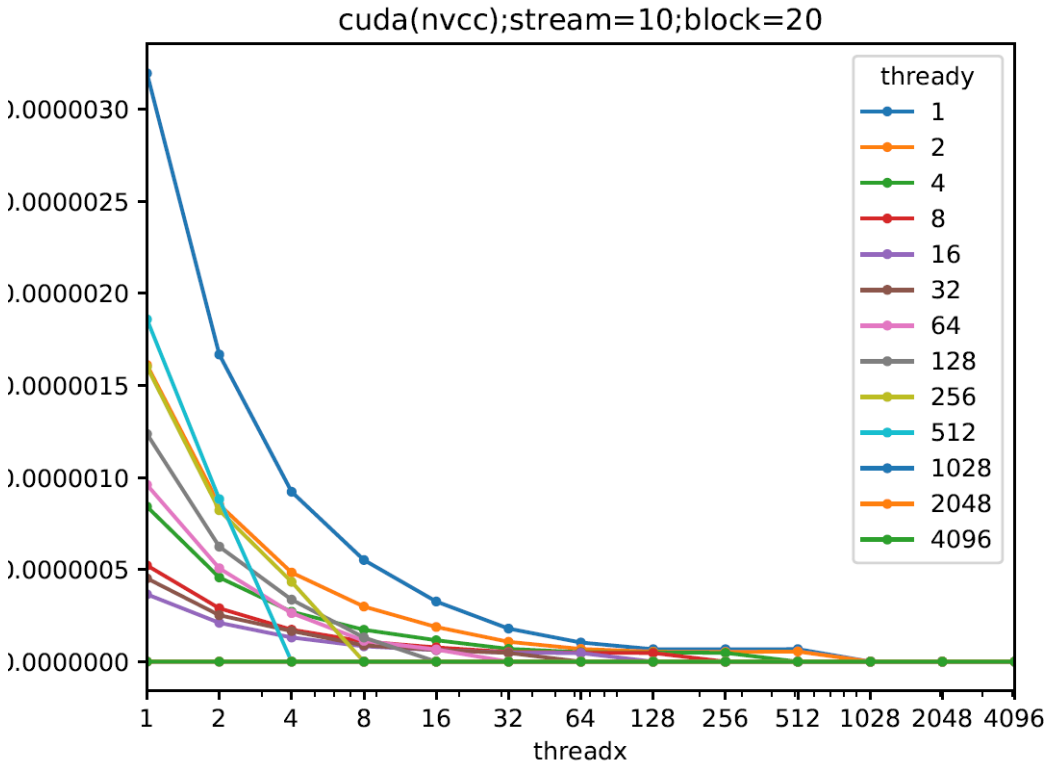

tbb(gcc)
1e−5
regionpertrk
nthreads
bsize
1
2
4
8
16
32
64
128
256
512
1024


tbb(gcc)
1e−5
regionpertrk
bsize
nthreads
1
2
4
8
16
32
64
128
256
512
1024

TBB(icc)


tbb(icc)
1e−5
regionpertrk
nthreads
bsize
1
2
4
8
16
32
64
128
256
512
1024

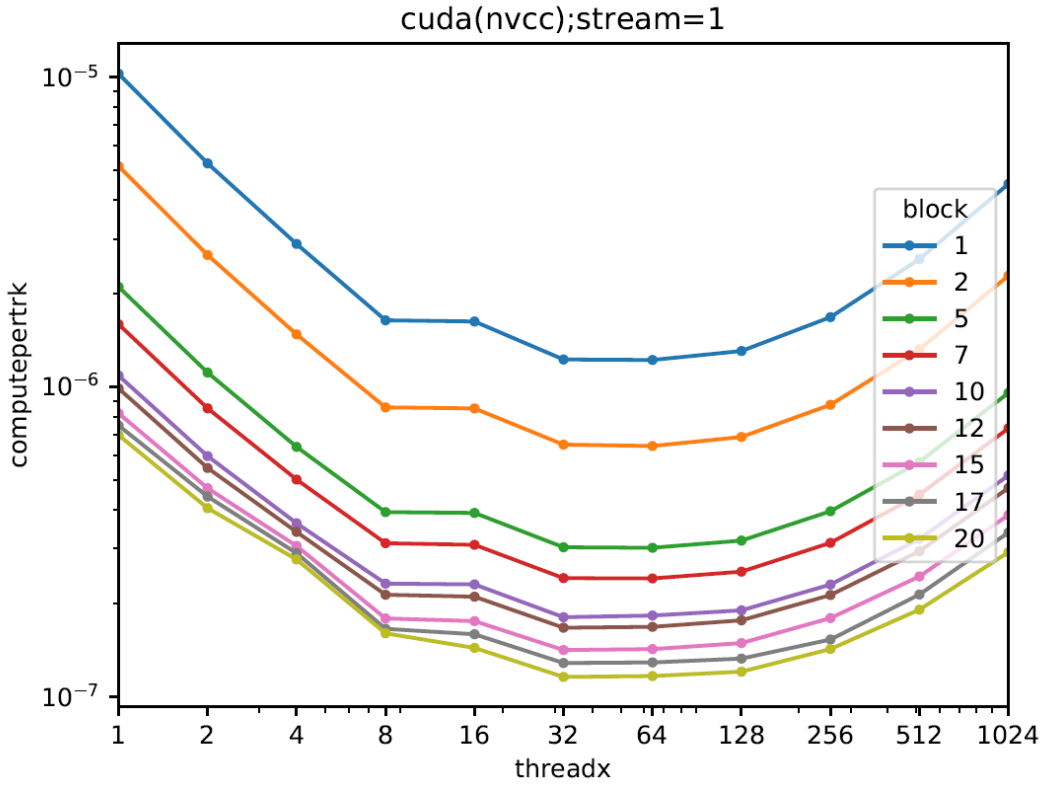

tbb(icc)
1e−5
regionpertrk
bsize
nthreads
1
2
4
8
16
32
64
128
256
512
1024

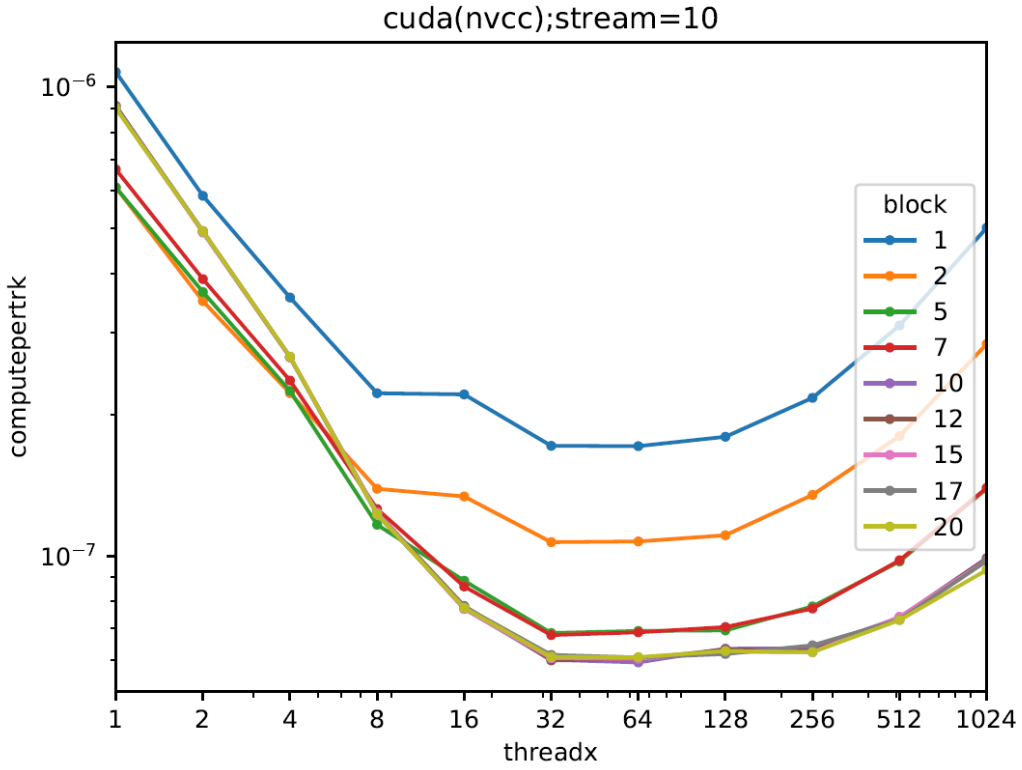



```
RUNNING CUDA!!
Streams: 10, blocks: 10, threads(x,y): (32,32)
track in pos: -12.806847, -7.723825, 38.130142
track in cov: 6.29e-07, 7.53e-07, 9.63e-08
hit in pos: -20.782465 -12.241503 57.806763
produce nevts=100 ntrks=122880 smearing by=0.100000
NITER=100
done preparing!
Size of struct MPTRK trk[] = 2457600000
Size of struct MPTRK outtrk[] = 2457600000
Size of struct struct MPHIT hit[] = 442368000
done ntracks=12288000 tot time=0.599035 (s) time/trk=4.874963e-08 (s)
data region time=1.488774 (s)
memory transfer time=0.889739 (s)
setup time time=20.465000 (s)
formatted 12288000 0.599035 4.874963e-08 1.488774 0.889739 20.465000 1.488000
wall region time=1.488000 (s)
track x avg=-17.902779 std/avg=0.164711
track y avg=-11.078340 std/avg=0.207770
track z avg=61.212490 std/avg=0.108644
track dx/x avg=-0.123526 std=0.204147
track dy/y avg=-0.132613 std=2.795923
track dz/z avg=0.000000 std=0.000000
```

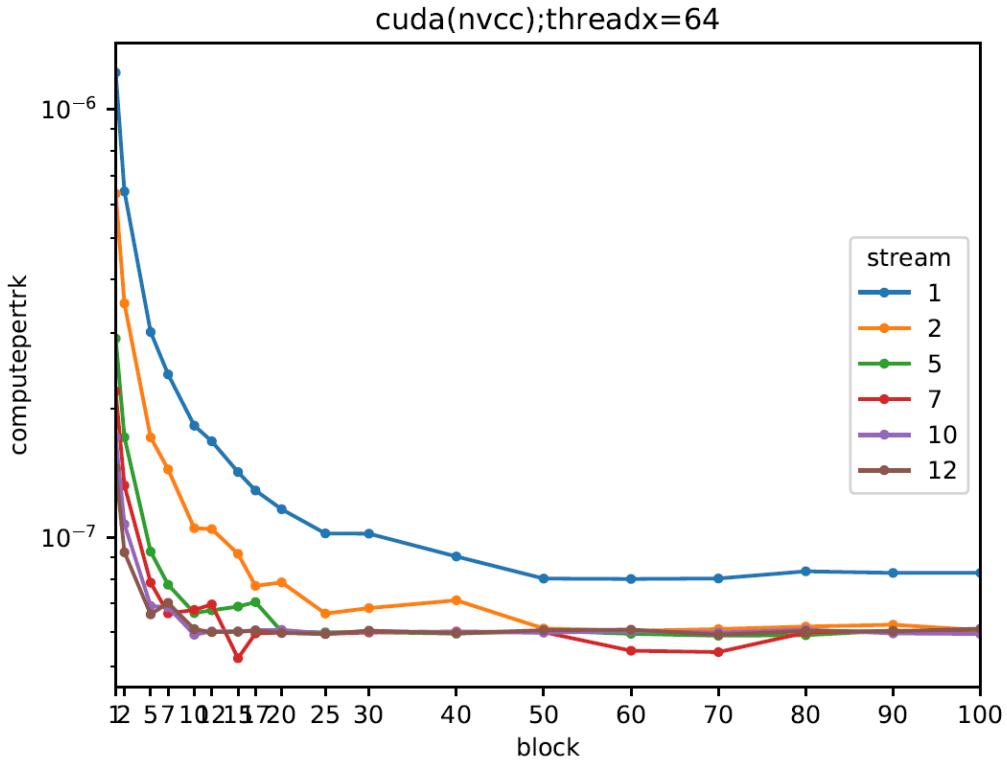
```
RUNNING CUDA!!
Streams: 10, blocks: 10, threads(x,y): (32,33)
track in pos: -12.806847, -7.723825, 38.130142
track in cov: 6.29e-07, 7.53e-07, 9.63e-08
hit in pos: -20.782465 -12.241503 57.806763
produce nevts=100 ntrks=122880 smearing by=0.100000
NITER=100
done preparing!
Size of struct MPTRK trk[] = 2457600000
Size of struct MPTRK outtrk[] = 2457600000
Size of struct struct MPHIT hit[] = 442368000
done ntracks=12288000 tot time=0.011005 (s) time/trk=8.956146e-10 (s)
data region time=1.142916 (s)
memory transfer time=1.131910 (s)
setup time time=20.457000 (s)
formatted 12288000 0.011005 8.956146e-10 1.142916 1.131910 20.457000 1.143000
wall region time=1.143000 (s)
track x avg=0.000000 std/avg=nan
track y avg=0.000000 std/avg=nan
track z avg=0.000000 std/avg=nan
track dx/x avg=inf std=-nan
track dy/y avg=inf std=-nan
track dz/z avg=-inf std=-nan
[mgr85@lnx7188 p2z-tests]$
```

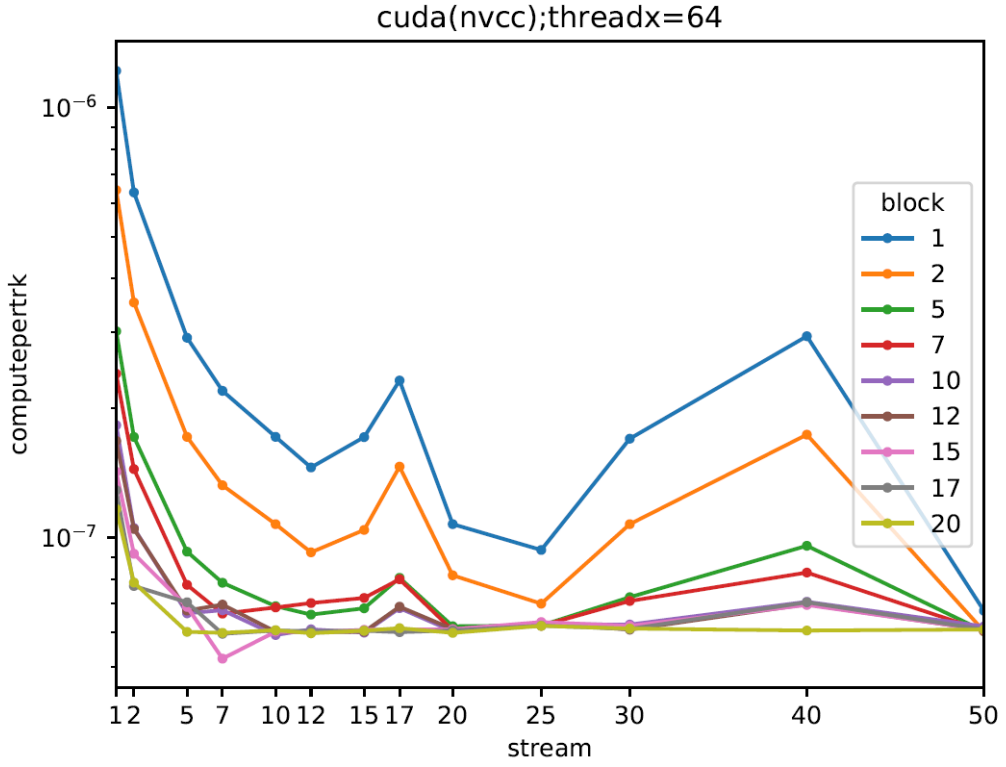
Fixed total threads to 1024/block





CUDA


cuda(nvcc);threadx=64
computepertrk
10^{-6}
10^{-7}
block
stream
1
2
5
7
10
12


cuda(nvcc);threadx=64
computepertrk
10^{-6}
10^{-7}
stream
block
1
2
5
7
10
12
15
17
20