In [28]:
```
import sys
sys.path.append("/scratch/group/csce435-f23/python-3.8.17/lib/python3.8/site-packages"
sys.path.append("/scratch/group/csce435-f23/thicket")
from glob import glob

import matplotlib.pyplot as plt
import pandas as pd

import thicket as th

pd.set_option("display.max_rows", None)
pd.set_option("display.max_columns", None)
```

Strong scaling

(same problem size, increase number of processors/nodes)

2^16

In [29]:
```
tk16 = th.Thicket.from_caliperreader(glob("cali_data_cuda/*65536*.cali"))
```

In [30]:
```
tk16.dataframe
```

Out[30]:

node	profile	nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
{'name': 'main', 'type': 'function'}	336492801	1	regionprofile	0.329976	0.329976	0.329976	0.329976	0.10
	413735523	1	regionprofile	0.525797	0.525797	0.525797	0.525797	0.19
	998640128	1	regionprofile	0.503357	0.503357	0.503357	0.503357	0.17
	1065561458	1	regionprofile	0.499803	0.499803	0.499803	0.499803	0.16
	1196434172	1	regionprofile	0.517818	0.517818	0.517818	0.517818	0.18
	1316431080	1	regionprofile	0.535747	0.535747	0.535747	0.535747	0.19
	1504292296	1	regionprofile	0.492675	0.492675	0.492675	0.492675	0.15
	1656094986	1	regionprofile	0.506167	0.506167	0.506167	0.506167	0.17
	1873224347	1	regionprofile	0.526279	0.526279	0.526279	0.526279	0.18
	2027881534	1	regionprofile	0.565978	0.565978	0.565978	0.565978	0.18
	2144587502	1	regionprofile	0.483854	0.483854	0.483854	0.483854	0.15
	2464187498	1	regionprofile	0.469428	0.469428	0.469428	0.469428	0.14
	2571354195	1	regionprofile	0.348867	0.348867	0.348867	0.348867	0.12
	2805700143	1	regionprofile	0.487954	0.487954	0.487954	0.487954	0.15
	3126055499	1	regionprofile	0.525454	0.525454	0.525454	0.525454	0.18
	3313914775	1	regionprofile	0.482080	0.482080	0.482080	0.482080	0.15
	3461656660	1	regionprofile	0.342968	0.342968	0.342968	0.342968	0.12
	3634890983	1	regionprofile	0.485699	0.485699	0.485699	0.485699	0.14
	4007850028	1	regionprofile	0.507925	0.507925	0.507925	0.507925	0.17
	4244921349	1	regionprofile	0.337806	0.337806	0.337806	0.337806	0.11
{'name': 'comm', 'type': 'function'}	336492801	3	regionprofile	0.329166	0.329166	0.329166	0.329166	0.00
	413735523	3	regionprofile	0.523700	0.523700	0.523700	0.523700	0.00
	998640128	3	regionprofile	0.502631	0.502631	0.502631	0.502631	0.00
	1065561458	3	regionprofile	0.497718	0.497718	0.497718	0.497718	0.00
	1196434172	3	regionprofile	0.517063	0.517063	0.517063	0.517063	0.00
	1316431080	3	regionprofile	0.534937	0.534937	0.534937	0.534937	0.00
	1504292296	3	regionprofile	0.491853	0.491853	0.491853	0.491853	0.00
	1656094986	3	regionprofile	0.505424	0.505424	0.505424	0.505424	0.00
	1873224347	3	regionprofile	0.525517	0.525517	0.525517	0.525517	0.00
	2027881534	3	regionprofile	0.563724	0.563724	0.563724	0.563724	0.00
	2144587502	3	regionprofile	0.483134	0.483134	0.483134	0.483134	0.00

node	profile	nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
	2464187498	3	regionprofile	0.468594	0.468594	0.468594	0.468594	0.00
	2571354195	3	regionprofile	0.346728	0.346728	0.346728	0.346728	0.00
	2805700143	3	regionprofile	0.487132	0.487132	0.487132	0.487132	0.00
	3126055499	3	regionprofile	0.523398	0.523398	0.523398	0.523398	0.00
	3313914775	3	regionprofile	0.481305	0.481305	0.481305	0.481305	0.00
	3461656660	3	regionprofile	0.342141	0.342141	0.342141	0.342141	0.00
	3634890983	3	regionprofile	0.484942	0.484942	0.484942	0.484942	0.00
	4007850028	3	regionprofile	0.507152	0.507152	0.507152	0.507152	0.00
	4244921349	3	regionprofile	0.336890	0.336890	0.336890	0.336890	0.00
{'name': 'comm_large', 'type': 'function'}	**336492801**	4	regionprofile	0.329103	0.329103	0.329103	0.329103	0.00
	413735523	4	regionprofile	0.523649	0.523649	0.523649	0.523649	0.00
	998640128	4	regionprofile	0.502579	0.502579	0.502579	0.502579	0.00
	1065561458	4	regionprofile	0.497656	0.497656	0.497656	0.497656	0.00
	1196434172	4	regionprofile	0.517010	0.517010	0.517010	0.517010	0.00
	1316431080	4	regionprofile	0.534886	0.534886	0.534886	0.534886	0.00
	1504292296	4	regionprofile	0.491801	0.491801	0.491801	0.491801	0.00
	1656094986	4	regionprofile	0.505371	0.505371	0.505371	0.505371	0.00
	1873224347	4	regionprofile	0.525464	0.525464	0.525464	0.525464	0.00
	2027881534	4	regionprofile	0.563649	0.563649	0.563649	0.563649	0.00
	2144587502	4	regionprofile	0.483082	0.483082	0.483082	0.483082	0.00
	2464187498	4	regionprofile	0.468540	0.468540	0.468540	0.468540	0.00
	2571354195	4	regionprofile	0.346671	0.346671	0.346671	0.346671	0.00
	2805700143	4	regionprofile	0.487076	0.487076	0.487076	0.487076	0.00
	3126055499	4	regionprofile	0.523348	0.523348	0.523348	0.523348	0.00
	3313914775	4	regionprofile	0.481249	0.481249	0.481249	0.481249	0.00
	3461656660	4	regionprofile	0.342071	0.342071	0.342071	0.342071	0.00
	3634890983	4	regionprofile	0.484889	0.484889	0.484889	0.484889	0.00
	4007850028	4	regionprofile	0.507093	0.507093	0.507093	0.507093	0.00
	4244921349	4	regionprofile	0.336834	0.336834	0.336834	0.336834	0.00
{'name': 'comp', 'type': 'function'}	**336492801**	5	regionprofile	0.000129	0.000129	0.000129	0.000129	0.10
	413735523	5	regionprofile	0.000095	0.000095	0.000095	0.000095	0.19

node	profile	nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
	998640128	5	regionprofile	0.000089	0.000089	0.000089	0.000089	0.17
	1065561458	5	regionprofile	0.000096	0.000096	0.000096	0.000096	0.16
	1196434172	5	regionprofile	0.000087	0.000087	0.000087	0.000087	0.18
	1316431080	5	regionprofile	0.000089	0.000089	0.000089	0.000089	0.19
	1504292296	5	regionprofile	0.000091	0.000091	0.000091	0.000091	0.15
	1656094986	5	regionprofile	0.000090	0.000090	0.000090	0.000090	0.17
	1873224347	5	regionprofile	0.000085	0.000085	0.000085	0.000085	0.18
	2027881534	5	regionprofile	0.000240	0.000240	0.000240	0.000240	0.18
	2144587502	5	regionprofile	0.000090	0.000090	0.000090	0.000090	0.15
	2464187498	5	regionprofile	0.000094	0.000094	0.000094	0.000094	0.14
	2571354195	5	regionprofile	0.000130	0.000130	0.000130	0.000130	0.12
	2805700143	5	regionprofile	0.000100	0.000100	0.000100	0.000100	0.15
	3126055499	5	regionprofile	0.000097	0.000097	0.000097	0.000097	0.18
	3313914775	5	regionprofile	0.000089	0.000089	0.000089	0.000089	0.15
	3461656660	5	regionprofile	0.000124	0.000124	0.000124	0.000124	0.12
	3634890983	5	regionprofile	0.000090	0.000090	0.000090	0.000090	0.14
	4007850028	5	regionprofile	0.000097	0.000097	0.000097	0.000097	0.17
	4244921349	5	regionprofile	0.000137	0.000137	0.000137	0.000137	0.11
{'name': 'comp_large', 'type': 'function'}	**336492801**	6	regionprofile	0.000104	0.000104	0.000104	0.000104	0.10
	413735523	6	regionprofile	0.000074	0.000074	0.000074	0.000074	0.19
	998640128	6	regionprofile	0.000067	0.000067	0.000067	0.000067	0.17
	1065561458	6	regionprofile	0.000073	0.000073	0.000073	0.000073	0.16
	1196434172	6	regionprofile	0.000067	0.000067	0.000067	0.000067	0.18
	1316431080	6	regionprofile	0.000069	0.000069	0.000069	0.000069	0.19
	1504292296	6	regionprofile	0.000070	0.000070	0.000070	0.000070	0.15
	1656094986	6	regionprofile	0.000068	0.000068	0.000068	0.000068	0.17
	1873224347	6	regionprofile	0.000065	0.000065	0.000065	0.000065	0.18
	2027881534	6	regionprofile	0.000197	0.000197	0.000197	0.000197	0.18
	2144587502	6	regionprofile	0.000069	0.000069	0.000069	0.000069	0.15
	2464187498	6	regionprofile	0.000074	0.000074	0.000074	0.000074	0.14
	2571354195	6	regionprofile	0.000107	0.000107	0.000107	0.000107	0.12

node	profile	nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
	2805700143	6	regionprofile	0.000078	0.000078	0.000078	0.000078	0.15
	3126055499	6	regionprofile	0.000075	0.000075	0.000075	0.000075	0.18
	3313914775	6	regionprofile	0.000068	0.000068	0.000068	0.000068	0.15
	3461656660	6	regionprofile	0.000102	0.000102	0.000102	0.000102	0.12
	3634890983	6	regionprofile	0.000070	0.000070	0.000070	0.000070	0.14
	4007850028	6	regionprofile	0.000074	0.000074	0.000074	0.000074	0.17
	4244921349	6	regionprofile	0.000115	0.000115	0.000115	0.000115	0.11
{'name': 'correctness_check', 'type': 'function'}	**336492801**	7	regionprofile	0.000197	0.000197	0.000197	0.000197	
	413735523	7	regionprofile	0.000192	0.000192	0.000192	0.000192	
	998640128	7	regionprofile	0.000197	0.000197	0.000197	0.000197	
	1065561458	7	regionprofile	0.000197	0.000197	0.000197	0.000197	
	1196434172	7	regionprofile	0.000191	0.000191	0.000191	0.000191	
	1316431080	7	regionprofile	0.000197	0.000197	0.000197	0.000197	
	1504292296	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
	1656094986	7	regionprofile	0.000193	0.000193	0.000193	0.000193	
	1873224347	7	regionprofile	0.000192	0.000192	0.000192	0.000192	
	2027881534	7	regionprofile	0.000193	0.000193	0.000193	0.000193	
	2144587502	7	regionprofile	0.000195	0.000195	0.000195	0.000195	
	2464187498	7	regionprofile	0.000193	0.000193	0.000193	0.000193	
	2571354195	7	regionprofile	0.000195	0.000195	0.000195	0.000195	
	2805700143	7	regionprofile	0.000195	0.000195	0.000195	0.000195	
	3126055499	7	regionprofile	0.000192	0.000192	0.000192	0.000192	
	3313914775	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
	3461656660	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
	3634890983	7	regionprofile	0.000196	0.000196	0.000196	0.000196	
	4007850028	7	regionprofile	0.000193	0.000193	0.000193	0.000193	
	4244921349	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
{'name': 'data_init', 'type': 'function'}	**336492801**	2	regionprofile	0.000217	0.000217	0.000217	0.000217	
	413735523	2	regionprofile	0.001587	0.001587	0.001587	0.001587	
	998640128	2	regionprofile	0.000221	0.000221	0.000221	0.000221	
	1065561458	2	regionprofile	0.001564	0.001564	0.001564	0.001564	

node	profile	nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
	1196434172	2	regionprofile	0.000249	0.000249	0.000249	0.000249	
	1316431080	2	regionprofile	0.000307	0.000307	0.000307	0.000307	
	1504292296	2	regionprofile	0.000310	0.000310	0.000310	0.000310	
	1656094986	2	regionprofile	0.000243	0.000243	0.000243	0.000243	
	1873224347	2	regionprofile	0.000257	0.000257	0.000257	0.000257	
	2027881534	2	regionprofile	0.001564	0.001564	0.001564	0.001564	
	2144587502	2	regionprofile	0.000220	0.000220	0.000220	0.000220	
	2464187498	2	regionprofile	0.000334	0.000334	0.000334	0.000334	
	2571354195	2	regionprofile	0.001561	0.001561	0.001561	0.001561	
	2805700143	2	regionprofile	0.000298	0.000298	0.000298	0.000298	
	3126055499	2	regionprofile	0.001555	0.001555	0.001555	0.001555	
	3313914775	2	regionprofile	0.000274	0.000274	0.000274	0.000274	
	3461656660	2	regionprofile	0.000269	0.000269	0.000269	0.000269	
	3634890983	2	regionprofile	0.000244	0.000244	0.000244	0.000244	
	4007850028	2	regionprofile	0.000252	0.000252	0.000252	0.000252	
	4244921349	2	regionprofile	0.000344	0.000344	0.000344	0.000344	

In [31]:
```
tk16.metadata
```

Out[31]:

profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
336492801	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
413735523	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
998640128	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1065561458	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1196434172	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1316431080	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1504292296	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1656094986	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1873224347	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2027881534	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2144587502	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2464187498	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2571354195	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2805700143	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3126055499	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3313914775	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3461656660	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3634890983	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
4007850028	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
4244921349	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

In [32]:
```
tk16.statsframe.dataframe["time"] = 1
print(tk16.tree())
```

```
  _____ _     _      _        _
 |_   _| |__ (_) ___| | _____| |_
   | | | '_ \| |/ __| |/ / _ \ __|
   | | | | | | | (__|   <  __/ |_
   |_| |_| |_|_|\___|_|\_\___|\__|  v2023.3.0

1.000 main
├─ 1.000 comm
│  └─ 1.000 comm_large
├─ 1.000 comp
│  └─ 1.000 comp_large
├─ 1.000 correctness_check
└─ 1.000 data_init

Legend (Metric: time Min: 1.00 Max: 1.00)
  1.00 - 1.00
  1.00 - 1.00
  1.00 - 1.00
  1.00 - 1.00
  1.00 - 1.00
  1.00 - 1.00

name User code    ◀ Only in left graph    ▶ Only in right graph
```

In [33]:
```
gb16 = tk16.groupby("InputType")
```

```
4  thickets created...
{'1% Perturbed': <thicket.thicket.Thicket object at 0x2b25c47d7970>, 'Random': <thicket.thicket.Thicket object at 0x2b25c3c1dfa0>, 'Reverse Sorted': <thicket.thicket.Thicket object at 0x2b25c3ce5550>, 'Sorted': <thicket.thicket.Thicket object at 0x2b25c4800370>}
```

In [34]:
```
ctk16 = th.Thicket.concat_thickets(
    thickets=list(gb16.values()),
    headers=list(gb16.keys()),
    axis='columns',
    metadata_key='num_threads'
)
```

In [35]:
```
ctk16.dataframe
```

Out[35]:

		nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
node	**num_threads**							
{'name': 'main', 'type': 'function'}	**2**	1	regionprofile	0.517818	0.517818	0.517818	0.517818	0.18
	4	1	regionprofile	0.342968	0.342968	0.342968	0.342968	0.12
	8	1	regionprofile	0.507925	0.507925	0.507925	0.507925	0.17
	16	1	regionprofile	0.482080	0.482080	0.482080	0.482080	0.15
	32	1	regionprofile	0.506167	0.506167	0.506167	0.506167	0.17
{'name': 'comm', 'type': 'function'}	**2**	3	regionprofile	0.517063	0.517063	0.517063	0.517063	0.00
	4	3	regionprofile	0.342141	0.342141	0.342141	0.342141	0.00
	8	3	regionprofile	0.507152	0.507152	0.507152	0.507152	0.00
	16	3	regionprofile	0.481305	0.481305	0.481305	0.481305	0.00
	32	3	regionprofile	0.505424	0.505424	0.505424	0.505424	0.00
{'name': 'comm_large', 'type': 'function'}	**2**	4	regionprofile	0.517010	0.517010	0.517010	0.517010	0.00
	4	4	regionprofile	0.342071	0.342071	0.342071	0.342071	0.00
	8	4	regionprofile	0.507093	0.507093	0.507093	0.507093	0.00
	16	4	regionprofile	0.481249	0.481249	0.481249	0.481249	0.00
	32	4	regionprofile	0.505371	0.505371	0.505371	0.505371	0.00
{'name': 'comp', 'type': 'function'}	**2**	5	regionprofile	0.000087	0.000087	0.000087	0.000087	0.18
	4	5	regionprofile	0.000124	0.000124	0.000124	0.000124	0.12
	8	5	regionprofile	0.000097	0.000097	0.000097	0.000097	0.17
	16	5	regionprofile	0.000089	0.000089	0.000089	0.000089	0.15
	32	5	regionprofile	0.000090	0.000090	0.000090	0.000090	0.17
{'name': 'comp_large', 'type': 'function'}	**2**	6	regionprofile	0.000067	0.000067	0.000067	0.000067	0.18
	4	6	regionprofile	0.000102	0.000102	0.000102	0.000102	0.12
	8	6	regionprofile	0.000074	0.000074	0.000074	0.000074	0.17
	16	6	regionprofile	0.000068	0.000068	0.000068	0.000068	0.15
	32	6	regionprofile	0.000068	0.000068	0.000068	0.000068	0.17
{'name': 'correctness_check', 'type': 'function'}	**2**	7	regionprofile	0.000191	0.000191	0.000191	0.000191	
	4	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
	8	7	regionprofile	0.000193	0.000193	0.000193	0.000193	
	16	7	regionprofile	0.000194	0.000194	0.000194	0.000194	
	32	7	regionprofile	0.000193	0.000193	0.000193	0.000193	

		nid	spot.channel	Min time/rank	Max time/rank	Avg time/rank	Total time	Avg time/
node	num_threads							
{'name': 'data_init', 'type': 'function'}	2	2	regionprofile	0.000249	0.000249	0.000249	0.000249	
	4	2	regionprofile	0.000269	0.000269	0.000269	0.000269	
	8	2	regionprofile	0.000252	0.000252	0.000252	0.000252	
	16	2	regionprofile	0.000274	0.000274	0.000274	0.000274	
	32	2	regionprofile	0.000243	0.000243	0.000243	0.000243	

In [36]:
```
ctk16.dataframe = ctk16.dataframe.reset_index().drop([("node", ""), ('1% Perturbed', '
                                                      ('Random', "nid"), ('Random', "s
                                                      ('Reverse Sorted', "nid"), ('Rev
                                                      ('Sorted', "nid"), ('Sorted', "s
                                                      ('1% Perturbed', 'Avg GPU time/r
                                                      ('Random', 'Avg GPU time/rank'),
                                                      ('Reverse Sorted', "Avg GPU time
                                                      ('Sorted', "Avg GPU time/rank"),
                                                     ], axis=1)
```

In [37]:
```
ctk16.dataframe
```

Out[37]:

	num_threads	1% Perturbed	Random	Reverse Sorted	Sorted	name
		Avg time/rank	Avg time/rank	Avg time/rank	Avg time/rank	
0	2	0.517818	0.565978	0.492675	0.483854	main
1	4	0.342968	0.348867	0.337806	0.329976	main
2	8	0.507925	0.525454	0.487954	0.485699	main
3	16	0.482080	0.525797	0.535747	0.503357	main
4	32	0.506167	0.499803	0.469428	0.526279	main
5	2	0.517063	0.563724	0.491853	0.483134	comm
6	4	0.342141	0.346728	0.336890	0.329166	comm
7	8	0.507152	0.523398	0.487132	0.484942	comm
8	16	0.481305	0.523700	0.534937	0.502631	comm
9	32	0.505424	0.497718	0.468594	0.525517	comm
10	2	0.517010	0.563649	0.491801	0.483082	comm_large
11	4	0.342071	0.346671	0.336834	0.329103	comm_large
12	8	0.507093	0.523348	0.487076	0.484889	comm_large
13	16	0.481249	0.523649	0.534886	0.502579	comm_large
14	32	0.505371	0.497656	0.468540	0.525464	comm_large
15	2	0.000087	0.000240	0.000091	0.000090	comp
16	4	0.000124	0.000130	0.000137	0.000129	comp
17	8	0.000097	0.000097	0.000100	0.000090	comp
18	16	0.000089	0.000095	0.000089	0.000089	comp
19	32	0.000090	0.000096	0.000094	0.000085	comp
20	2	0.000067	0.000197	0.000070	0.000069	comp_large
21	4	0.000102	0.000107	0.000115	0.000104	comp_large
22	8	0.000074	0.000075	0.000078	0.000070	comp_large
23	16	0.000068	0.000074	0.000069	0.000067	comp_large
24	32	0.000068	0.000073	0.000074	0.000065	comp_large
25	2	0.000191	0.000193	0.000194	0.000195	correctness_check
26	4	0.000194	0.000195	0.000194	0.000197	correctness_check
27	8	0.000193	0.000192	0.000195	0.000196	correctness_check
28	16	0.000194	0.000192	0.000197	0.000197	correctness_check
29	32	0.000193	0.000197	0.000193	0.000192	correctness_check
30	2	0.000249	0.001564	0.000310	0.000220	data_init
31	4	0.000269	0.001561	0.000344	0.000217	data_init
32	8	0.000252	0.001555	0.000298	0.000244	data_init

	num_threads	1% Perturbed	Random	Reverse Sorted	Sorted	name
		Avg time/rank	Avg time/rank	Avg time/rank	Avg time/rank	
33	16	0.000274	0.001587	0.000307	0.000221	data_init
34	32	0.000243	0.001564	0.000334	0.000257	data_init

In [39]:
```
ctk16.dataframe = ctk16.dataframe.rename({("name", ""): "name", ("num_threads", ""): '
ctk16.dataframe
```

Out[39]:

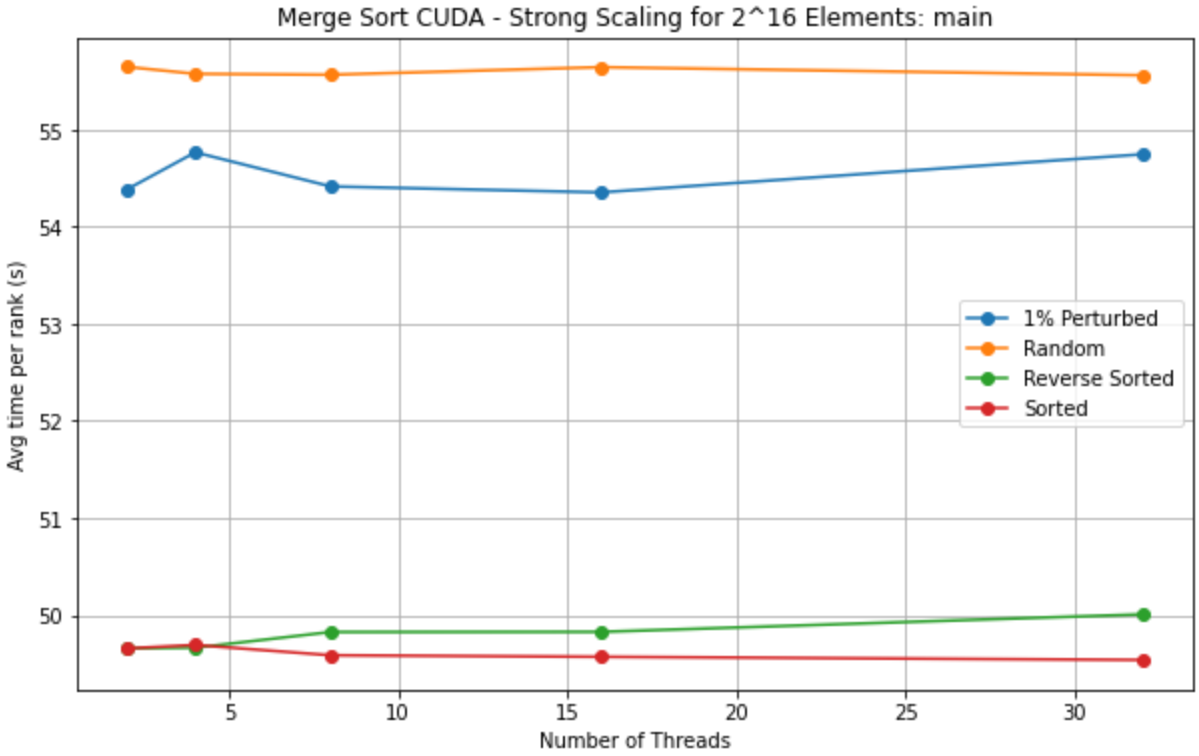
		1% Perturbed	Random	Reverse Sorted	Sorted
		Avg time/rank	Avg time/rank	Avg time/rank	Avg time/rank
name	num_threads				
main	2	0.517818	0.565978	0.492675	0.483854
	4	0.342968	0.348867	0.337806	0.329976
	8	0.507925	0.525454	0.487954	0.485699
	16	0.482080	0.525797	0.535747	0.503357
	32	0.506167	0.499803	0.469428	0.526279
comm	2	0.517063	0.563724	0.491853	0.483134
	4	0.342141	0.346728	0.336890	0.329166
	8	0.507152	0.523398	0.487132	0.484942
	16	0.481305	0.523700	0.534937	0.502631
	32	0.505424	0.497718	0.468594	0.525517
comm_large	2	0.517010	0.563649	0.491801	0.483082
	4	0.342071	0.346671	0.336834	0.329103
	8	0.507093	0.523348	0.487076	0.484889
	16	0.481249	0.523649	0.534886	0.502579
	32	0.505371	0.497656	0.468540	0.525464
comp	2	0.000087	0.000240	0.000091	0.000090
	4	0.000124	0.000130	0.000137	0.000129
	8	0.000097	0.000097	0.000100	0.000090
	16	0.000089	0.000095	0.000089	0.000089
	32	0.000090	0.000096	0.000094	0.000085
comp_large	2	0.000067	0.000197	0.000070	0.000069
	4	0.000102	0.000107	0.000115	0.000104
	8	0.000074	0.000075	0.000078	0.000070
	16	0.000068	0.000074	0.000069	0.000067
	32	0.000068	0.000073	0.000074	0.000065
correctness_check	2	0.000191	0.000193	0.000194	0.000195
	4	0.000194	0.000195	0.000194	0.000197
	8	0.000193	0.000192	0.000195	0.000196
	16	0.000194	0.000192	0.000197	0.000197
	32	0.000193	0.000197	0.000193	0.000192
data_init	2	0.000249	0.001564	0.000310	0.000220
	4	0.000269	0.001561	0.000344	0.000217

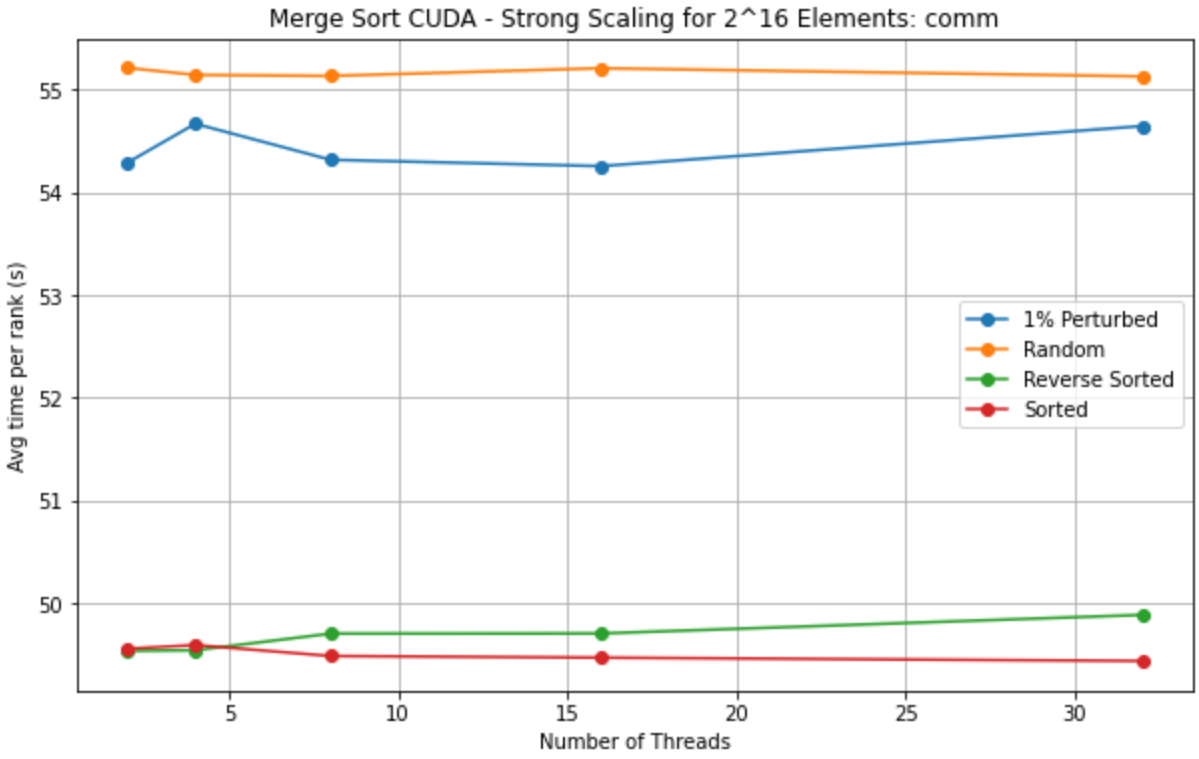
		1% Perturbed	Random	Reverse Sorted	Sorted
		Avg time/rank	Avg time/rank	Avg time/rank	Avg time/rank
name	num_threads				
	8	0.000252	0.001555	0.000298	0.000244
	16	0.000274	0.001587	0.000307	0.000221
	32	0.000243	0.001564	0.000334	0.000257

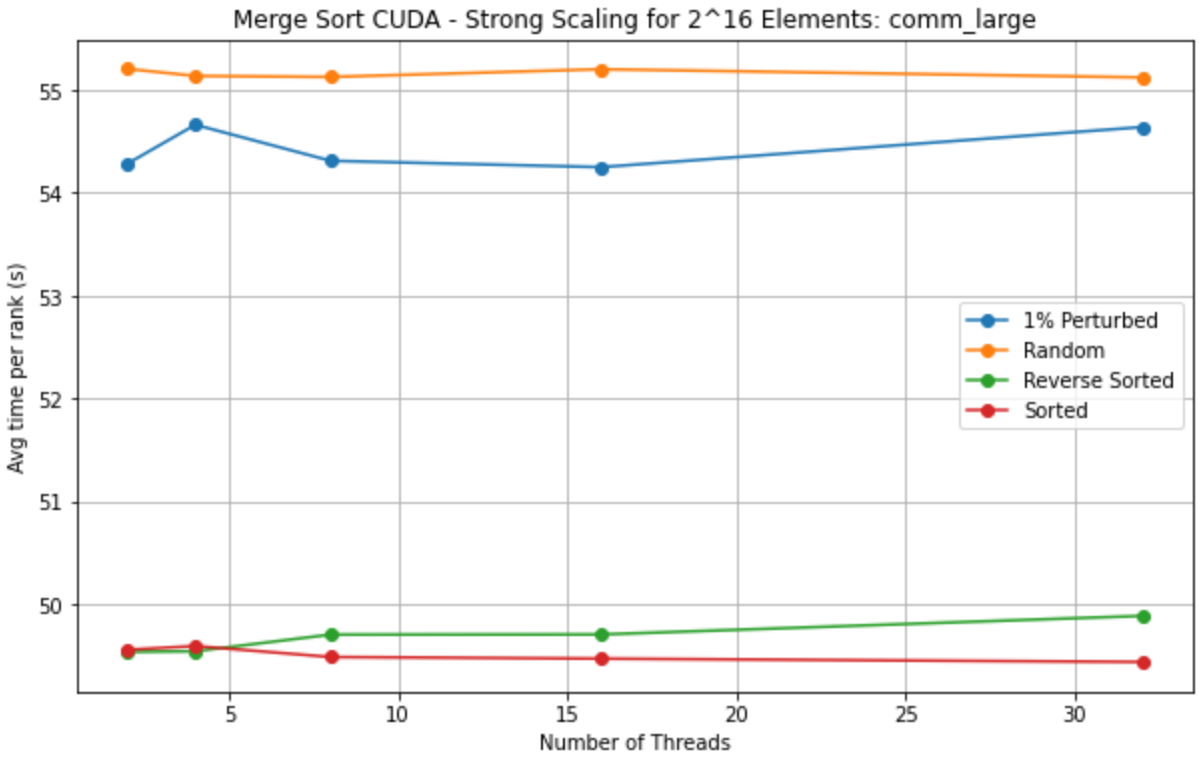
In [41]:
```
main = ctk16.dataframe.loc["main"]
comm = ctk16.dataframe.loc["comm"]
comm_large = ctk16.dataframe.loc["comm_large"]
comp = ctk16.dataframe.loc["comp"]
comp_large = ctk16.dataframe.loc["comp_large"]
correctness_check = ctk16.dataframe.loc["correctness_check"]
data_init = ctk16.dataframe.loc["data_init"]
```

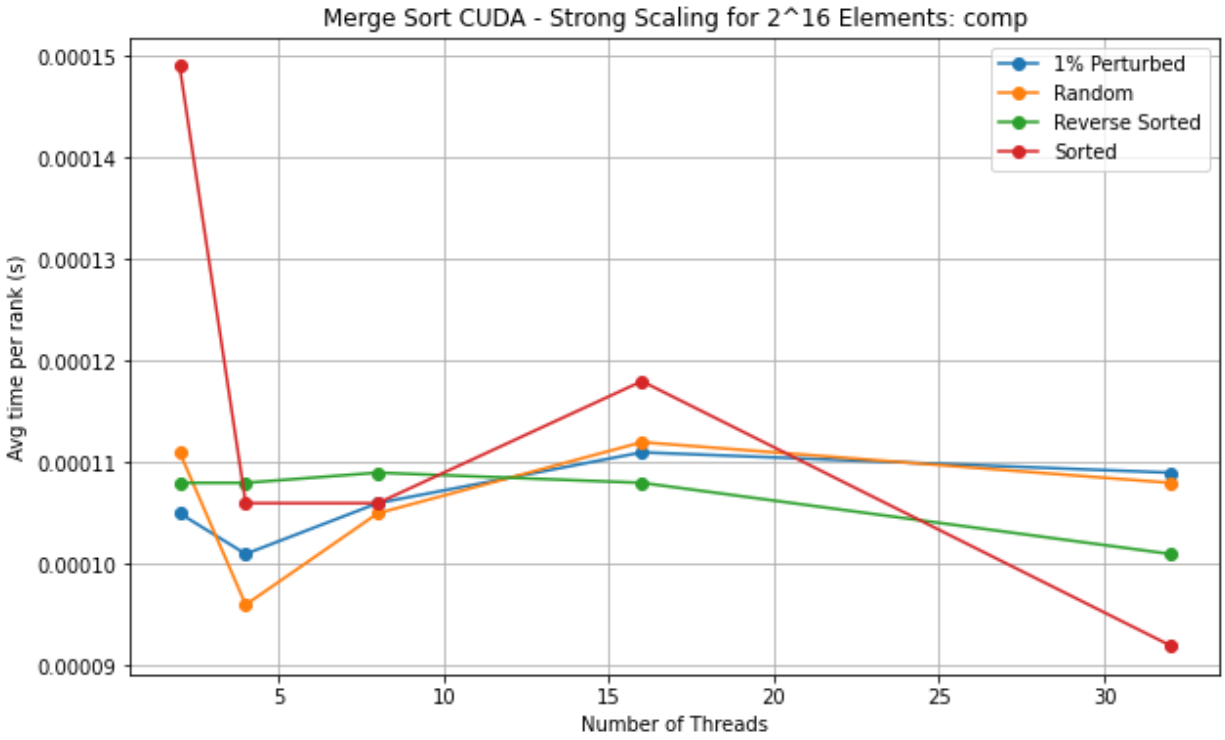
In [42]:
```
regions = [main, comm, comm_large, comp, comp_large, correctness_check, data_init]
names = ["main", "comm", "comm_large", "comp", "comp_large", "correctness_check", "dat
```

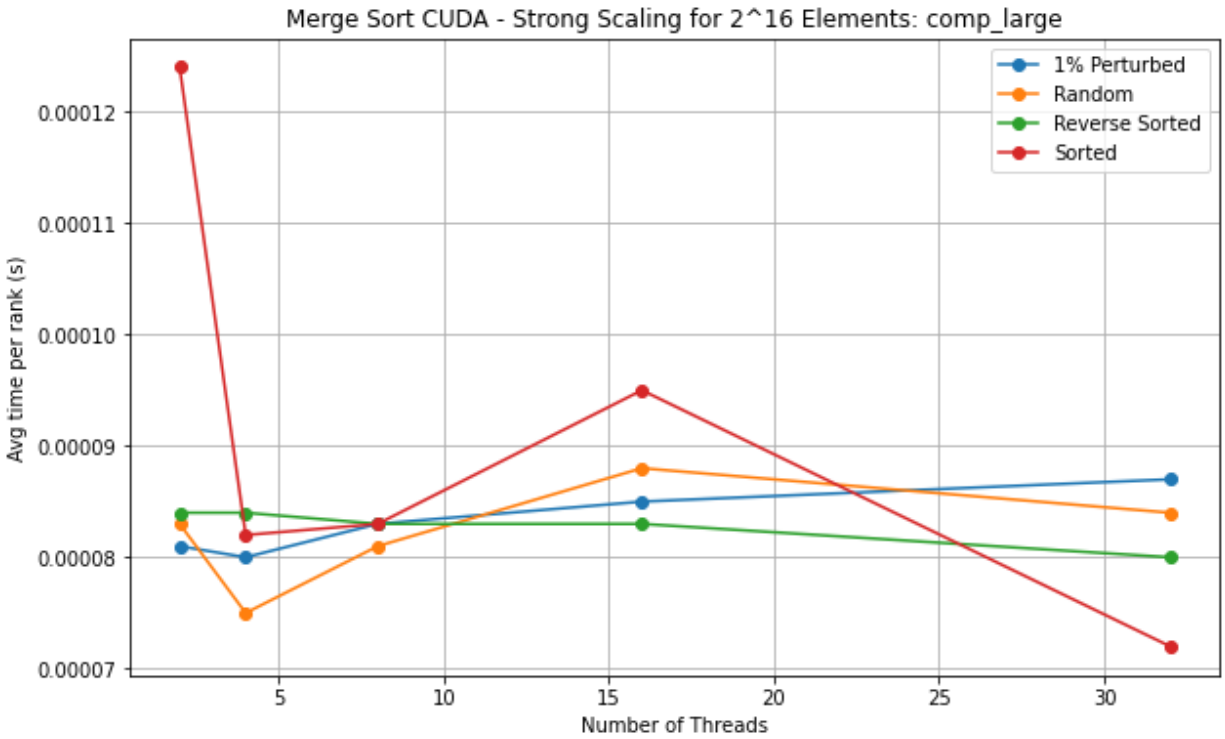
In [63]:
```
for region, name in zip(regions, names):
    plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
    legend_labels = []
    for column in region.columns:
        first_index = column[0]  # Extract the first index
        legend_labels.append(first_index)
        plt.plot(region.index, region.xs(column, axis=1), marker='o', label=column)

    plt.xlabel('Number of Threads')
    plt.ylabel('Avg time per rank (s)')
    plt.title(f'Merge Sort CUDA - Strong Scaling for 2^16 Elements: {name}')
    plt.legend(legend_labels)
    plt.grid(True)
    plt.show()
```

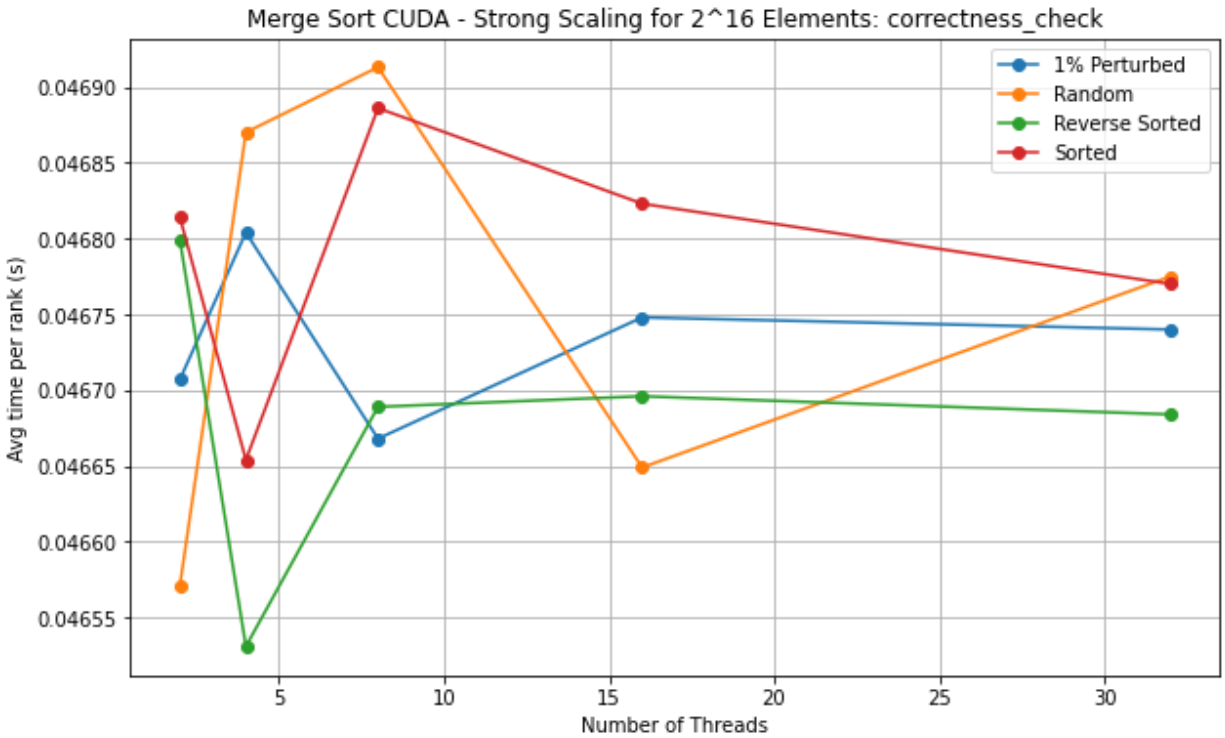

Merge Sort CUDA - Strong Scaling for 2^16 Elements: main
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted

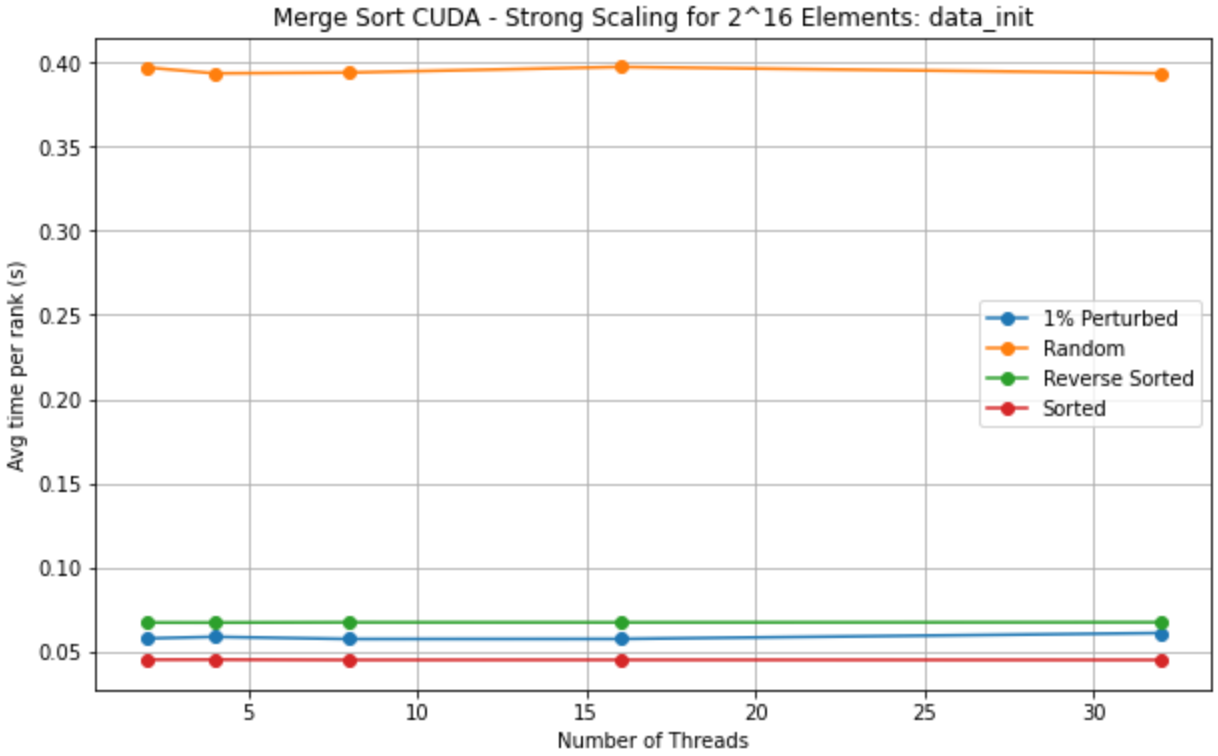

Merge Sort CUDA - Strong Scaling for 2^16 Elements: comm
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^16 Elements: comm_large
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^16 Elements: comp
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^16 Elements: comp_large
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.00012
0.00011
0.00010
0.00009
0.00008
0.00007
5
10
15
20
25
30
Number of Threads


Merge Sort CUDA - Strong Scaling for 2^16 Elements: correctness_check
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.04690
0.04685
0.04680
0.04675
0.04670
0.04665
0.04660
0.04655
5
10
15
20
25
30
Number of Threads



2^18

In [45]:
```
tk18 = th.Thicket.from_caliperreader(glob("cali_data_cuda/*262144*.cali"))
tk18.metadata
```

Out[45]:

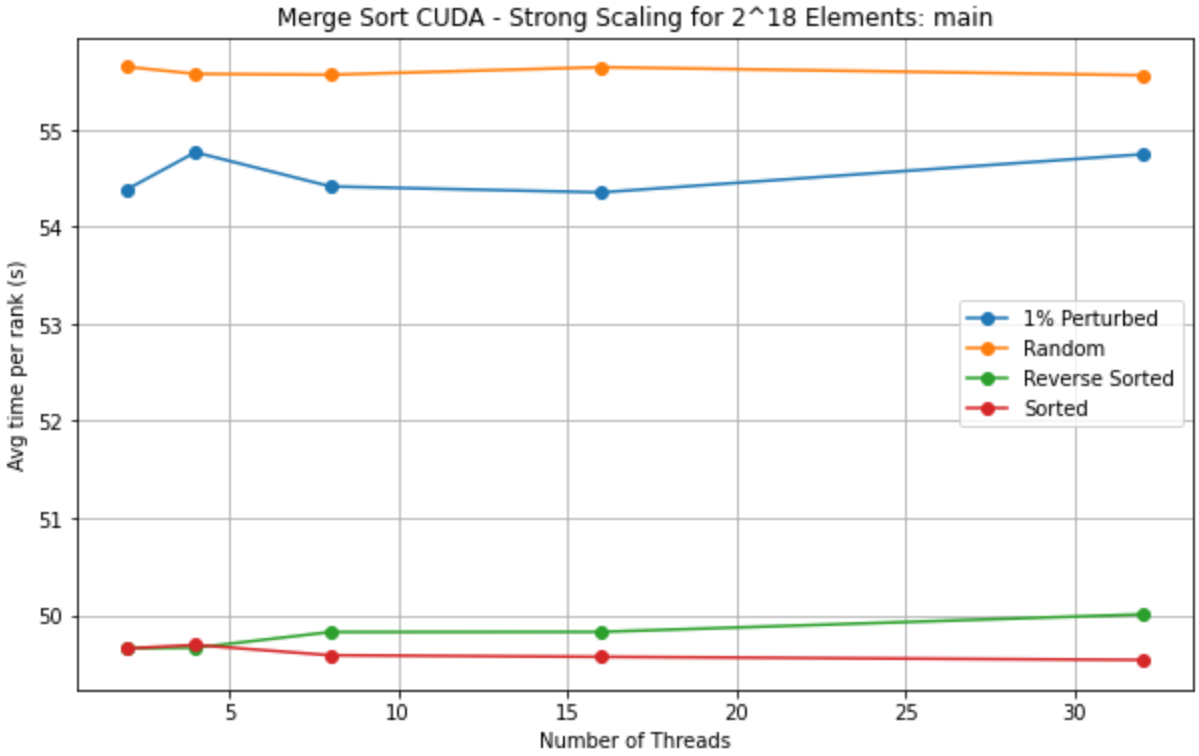
profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
234945630	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
338536597	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
437300641	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
699378644	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
718458009	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
744446730	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
865460097	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
926025040	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1017687023	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1275449112	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2153393909	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2338961288	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2346345698	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2575277462	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2602276143	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2913184065	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2936061741	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3300489053	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3442526181	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3950360809	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

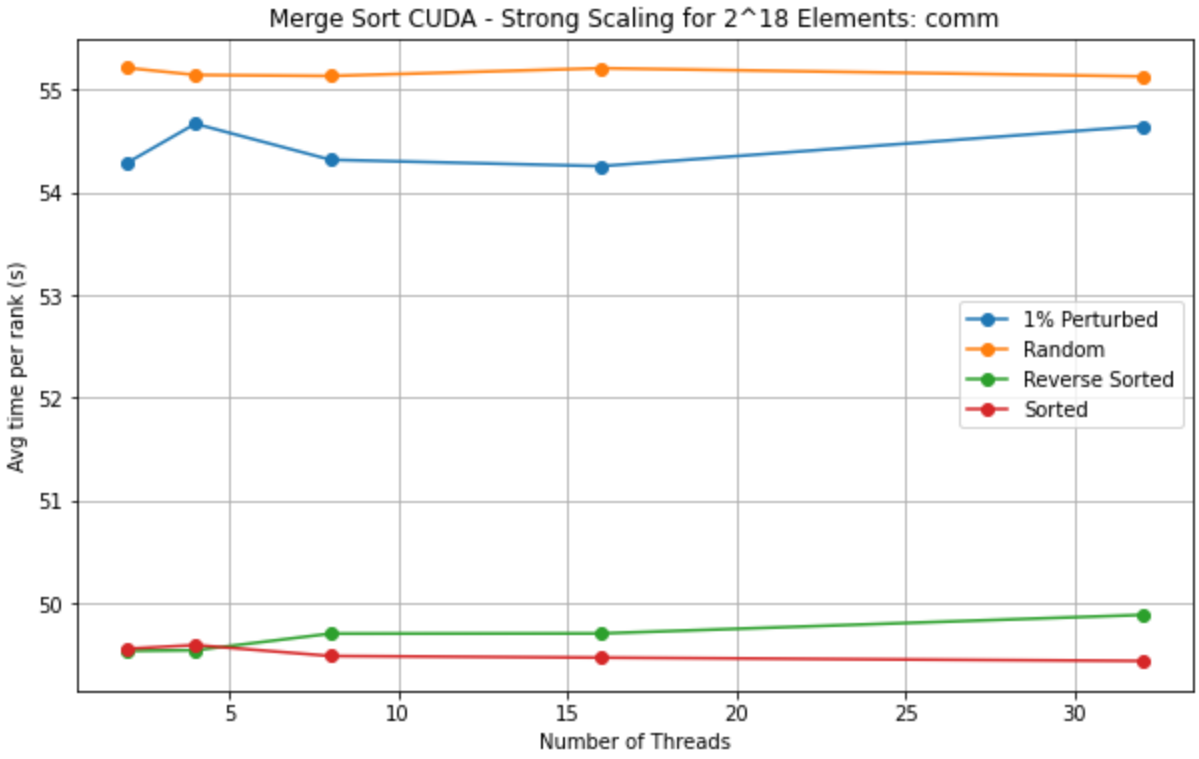
In [46]:
```
gb18 = tk18.groupby("InputType")

ctk18 = th.Thicket.concat_thickets(
    thickets=list(gb18.values()),
    headers=list(gb18.keys()),
    axis='columns',
    metadata_key='num_threads'
)

ctk18.dataframe = ctk18.dataframe.reset_index().drop([("node", ""), ('1% Perturbed', '
                                                      ('Random', "nid"), ('Random', "s
                                                      ('Reverse Sorted', "nid"), ('Rev
                                                      ('Sorted', "nid"), ('Sorted', "s
                                                      ('1% Perturbed', 'Avg GPU time/r
                                                      ('Random', 'Avg GPU time/rank'),
                                                      ('Reverse Sorted', "Avg GPU time
                                                      ('Sorted', "Avg GPU time/rank"),
                                                     ], axis=1)
ctk18.dataframe = ctk18.dataframe.rename({("name", ""): "name", ("num_threads", ""): '
```

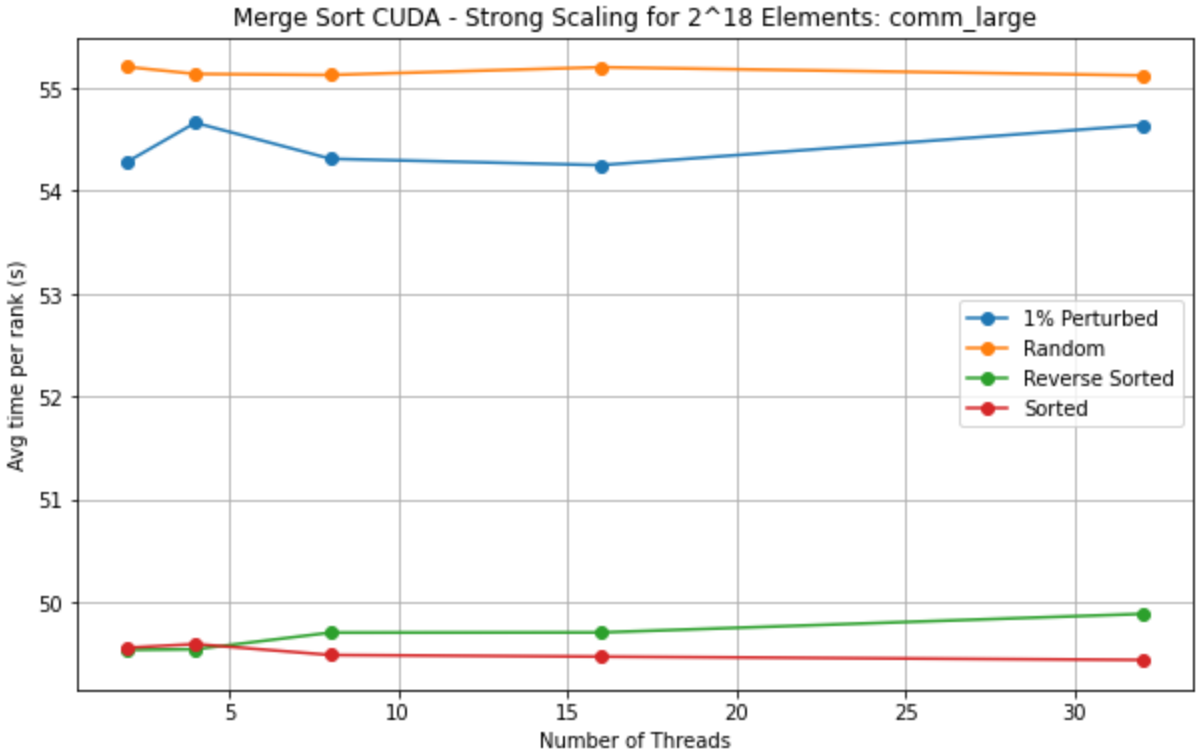
```
4  thickets created...
{'1% Perturbed': <thicket.thicket.Thicket object at 0x2b25c4859400>, 'Random': <thick
et.thicket.Thicket object at 0x2b25c3c85640>, 'Reverse Sorted': <thicket.thicket.Thic
ket object at 0x2b25c6d77fa0>, 'Sorted': <thicket.thicket.Thicket object at 0x2b25c6d
f1460>}
```

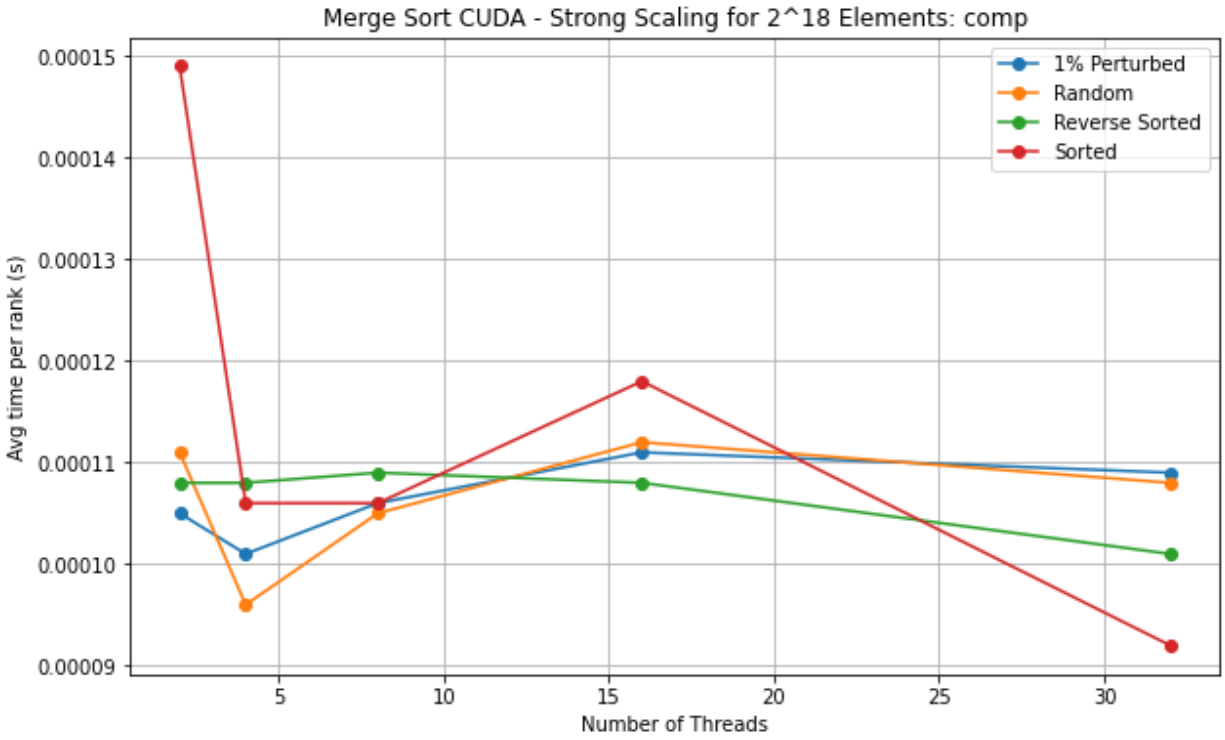
In [47]:
```
main = ctk18.dataframe.loc["main"]
comm = ctk18.dataframe.loc["comm"]
comm_large = ctk18.dataframe.loc["comm_large"]
comp = ctk18.dataframe.loc["comp"]
comp_large = ctk18.dataframe.loc["comp_large"]
correctness_check = ctk18.dataframe.loc["correctness_check"]
data_init = ctk18.dataframe.loc["data_init"]

regions = [main, comm, comm_large, comp, comp_large, correctness_check, data_init]
names = ["main", "comm", "comm_large", "comp", "comp_large", "correctness_check", "dat
```

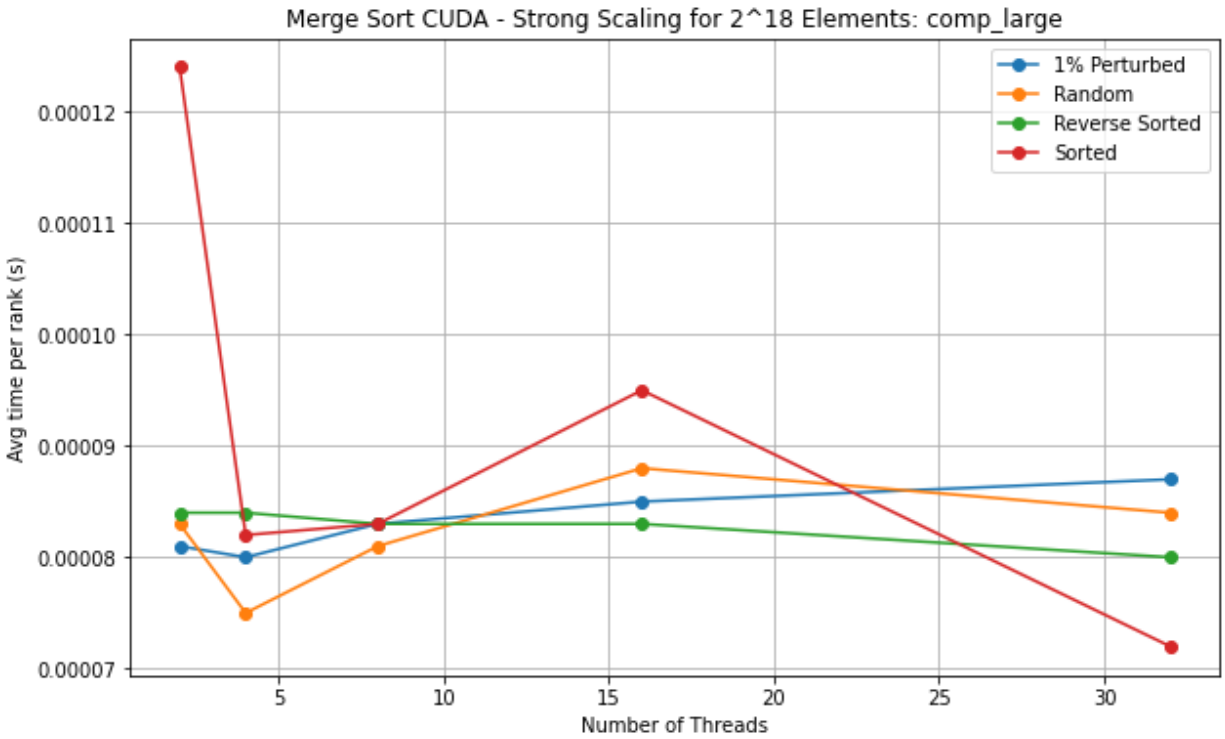
In [62]:
```
for region, name in zip(regions, names):
    plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
    legend_labels = []
    for column in region.columns:
        first_index = column[0]  # Extract the first index
        legend_labels.append(first_index)
        plt.plot(region.index, region.xs(column, axis=1), marker='o', label=column)

    plt.xlabel('Number of Threads')
    plt.ylabel('Avg time per rank (s)')
    plt.title(f'Merge Sort CUDA - Strong Scaling for 2^18 Elements: {name}')
    plt.legend(legend_labels)
    plt.grid(True)
    plt.show()
```

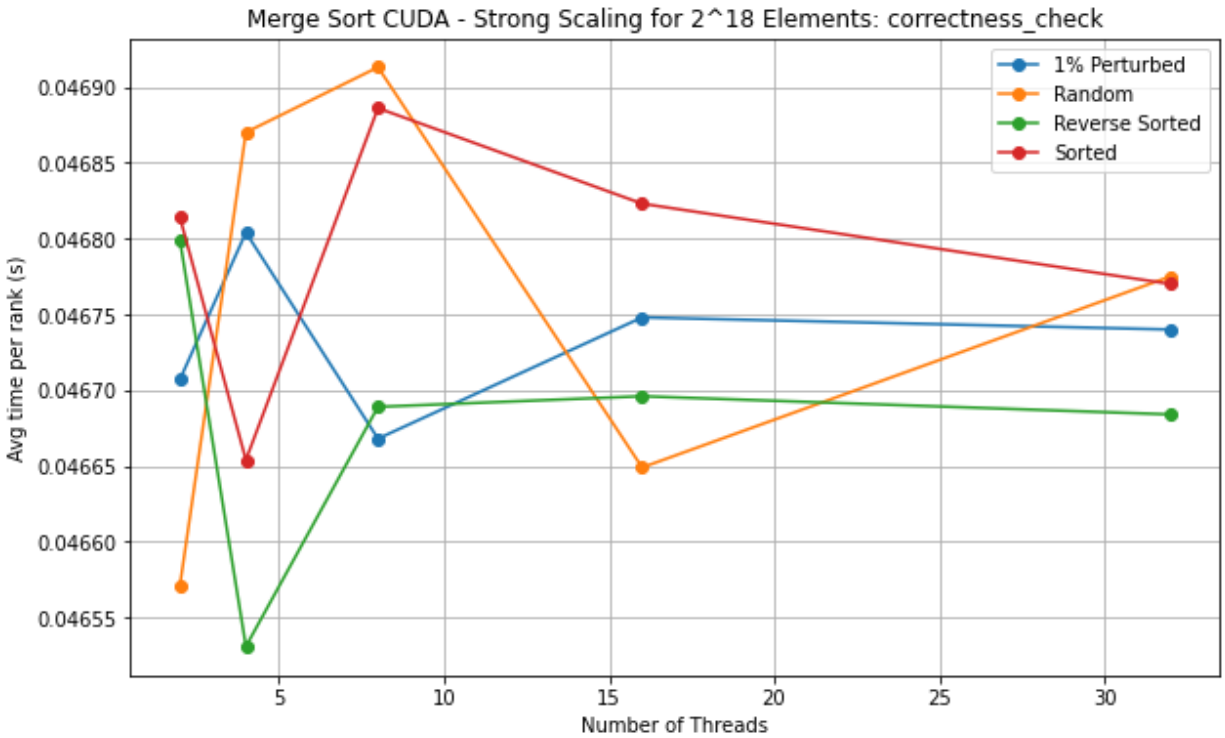

Merge Sort CUDA - Strong Scaling for 2^18 Elements: main
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted

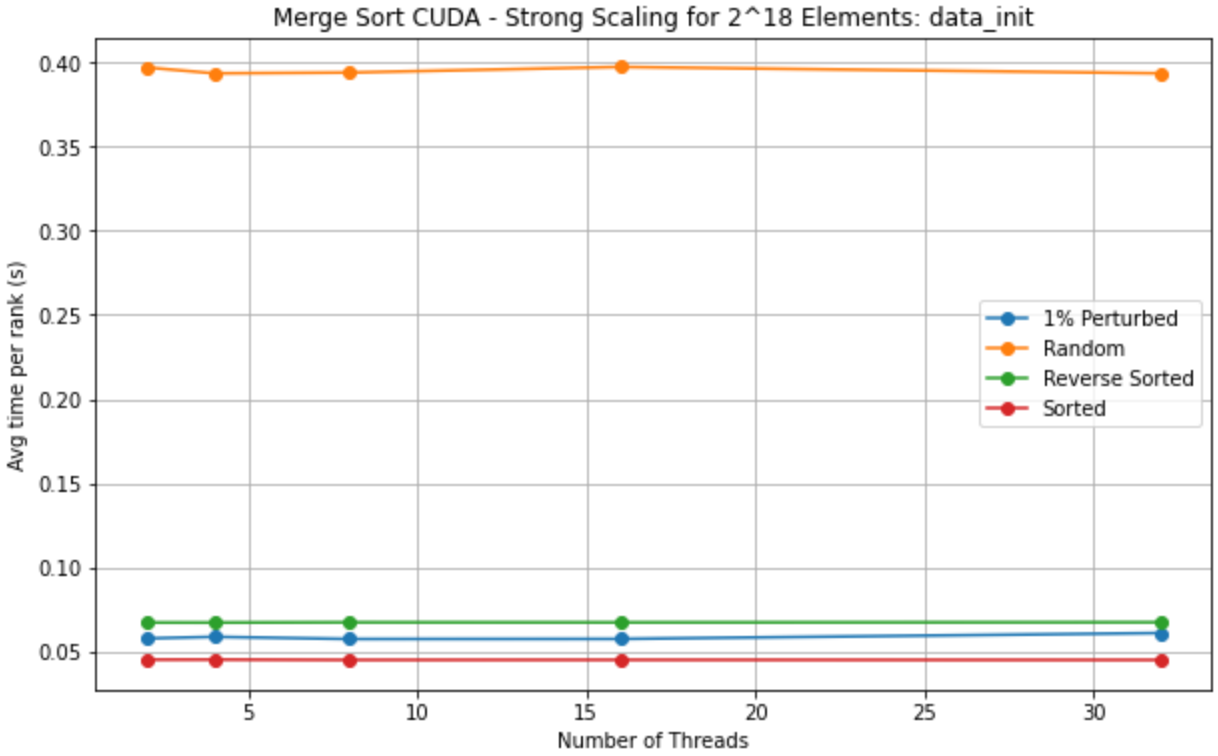

Merge Sort CUDA - Strong Scaling for 2^18 Elements: comm
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^18 Elements: comm_large
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^18 Elements: comp
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^18 Elements: comp_large
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.00012
0.00011
0.00010
0.00009
0.00008
0.00007
5
10
15
20
25
30
Number of Threads


Merge Sort CUDA - Strong Scaling for 2^18 Elements: correctness_check
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.04690
0.04685
0.04680
0.04675
0.04670
0.04665
0.04660
0.04655
5
10
15
20
25
30
Number of Threads



2^20

In [49]:
```
tk20 = th.Thicket.from_caliperreader(glob("cali_data_cuda/*1048576*.cali"))
tk20.metadata
```

Out[49]:

profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
126873939	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
275130569	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
394070192	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
499237816	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1048949714	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1106171182	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1282151141	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1364802912	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1620400637	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1726556461	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1901037037	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1906796618	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2045376319	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2117898895	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2754120943	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3167015544	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3276412449	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3359832445	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
4144639890	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
4292452911	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

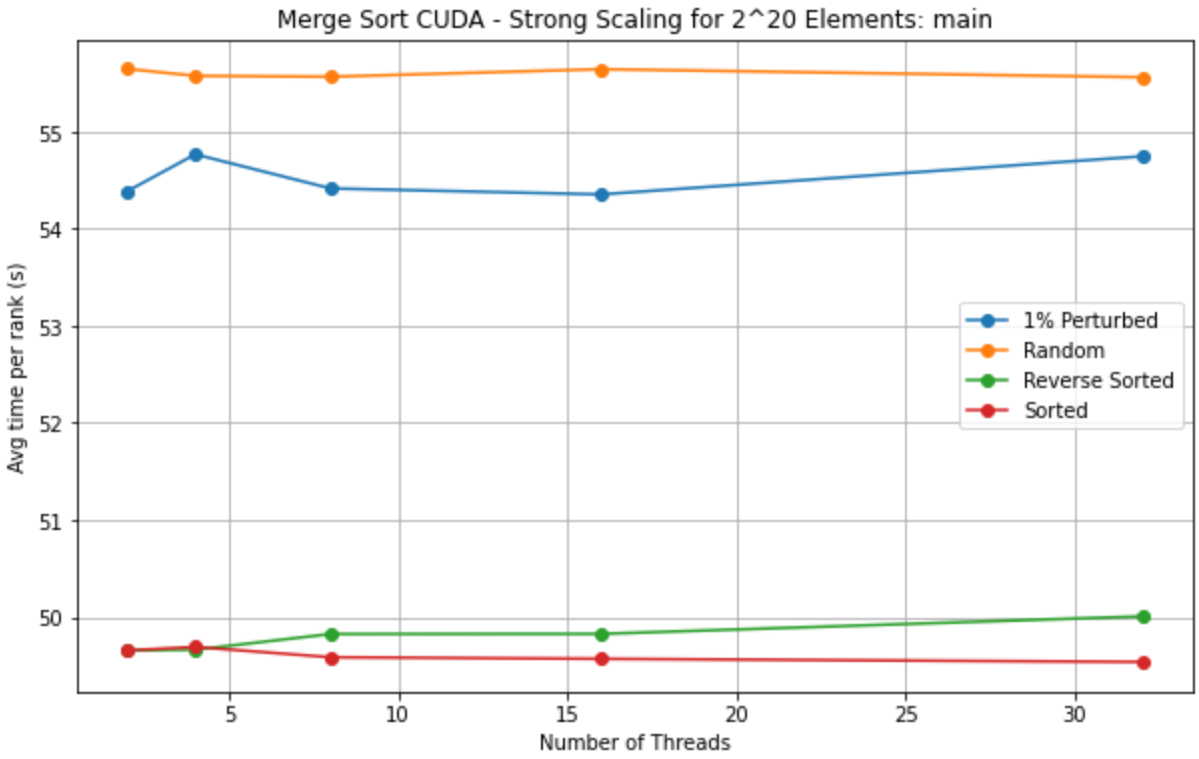
In [50]:
```
gb20 = tk20.groupby("InputType")

ctk20 = th.Thicket.concat_thickets(
    thickets=list(gb20.values()),
    headers=list(gb20.keys()),
    axis='columns',
    metadata_key='num_threads'
)

ctk20.dataframe = ctk20.dataframe.reset_index().drop([("node", ""), ('1% Perturbed', '
                                                      ('Random', "nid"), ('Random', "s
                                                      ('Reverse Sorted', "nid"), ('Rev
                                                      ('Sorted', "nid"), ('Sorted', "s
                                                      ('1% Perturbed', 'Avg GPU time/r
                                                      ('Random', 'Avg GPU time/rank'),
                                                      ('Reverse Sorted', "Avg GPU time
                                                      ('Sorted', "Avg GPU time/rank"),
                                                     ], axis=1)
ctk20.dataframe = ctk20.dataframe.rename({("name", ""): "name", ("num_threads", ""): '
```

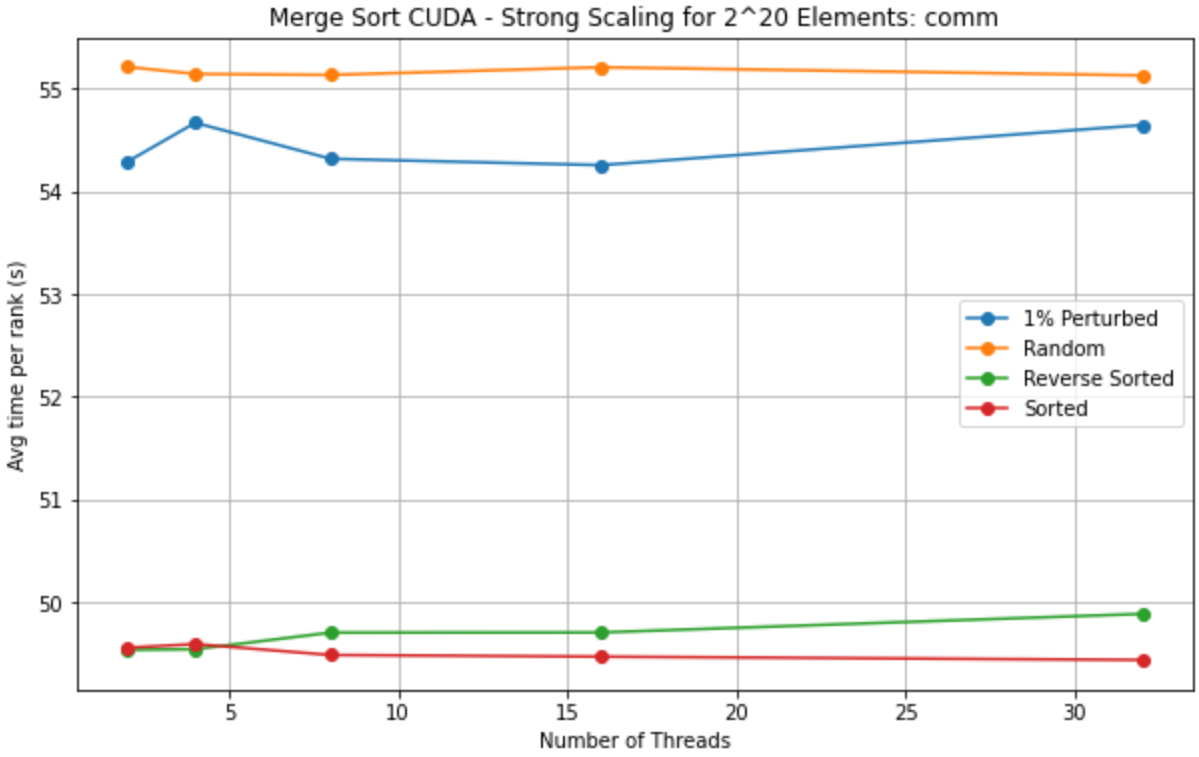
```
4  thickets created...
{'1% Perturbed': <thicket.thicket.Thicket object at 0x2b25c4791a00>, 'Random': <thick
et.thicket.Thicket object at 0x2b25c6f274f0>, 'Reverse Sorted': <thicket.thicket.Thic
ket object at 0x2b25c4791730>, 'Sorted': <thicket.thicket.Thicket object at 0x2b25c6f
3cf40>}
```

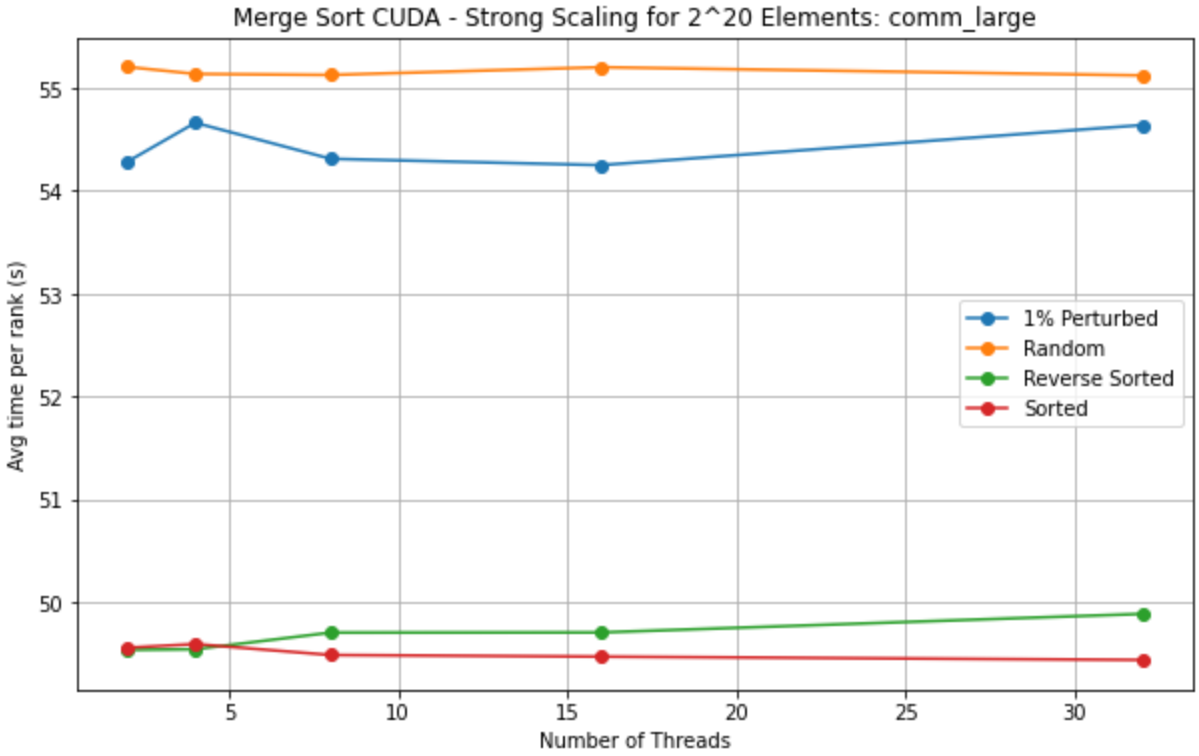
In [51]:
```
main = ctk20.dataframe.loc["main"]
comm = ctk20.dataframe.loc["comm"]
comm_large = ctk20.dataframe.loc["comm_large"]
comp = ctk20.dataframe.loc["comp"]
comp_large = ctk20.dataframe.loc["comp_large"]
correctness_check = ctk20.dataframe.loc["correctness_check"]
data_init = ctk20.dataframe.loc["data_init"]

regions = [main, comm, comm_large, comp, comp_large, correctness_check, data_init]
names = ["main", "comm", "comm_large", "comp", "comp_large", "correctness_check", "dat
```

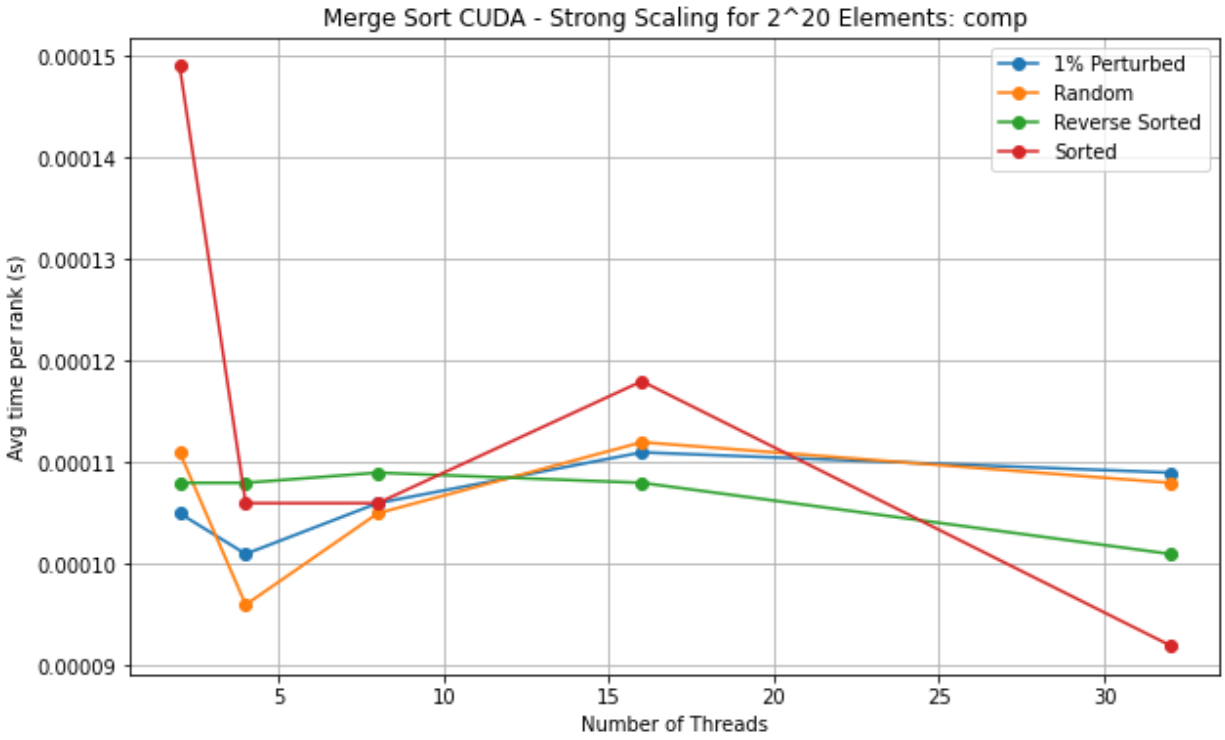
In [61]:
```
for region, name in zip(regions, names):
    plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
    legend_labels = []
    for column in region.columns:
        first_index = column[0]  # Extract the first index
        legend_labels.append(first_index)
        plt.plot(region.index, region.xs(column, axis=1), marker='o', label=column)

    plt.xlabel('Number of Threads')
    plt.ylabel('Avg time per rank (s)')
    plt.title(f'Merge Sort CUDA - Strong Scaling for 2^20 Elements: {name}')
    plt.legend(legend_labels)
```

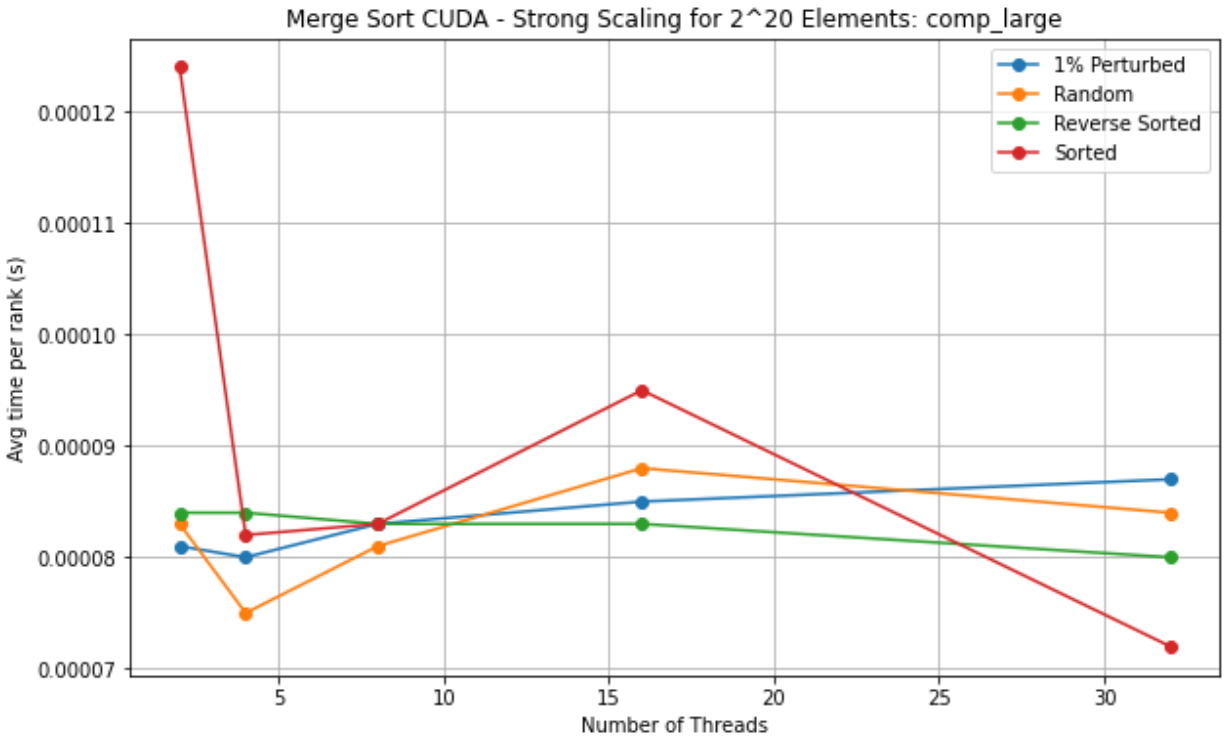
```
    plt.grid(True)
    plt.show()
```

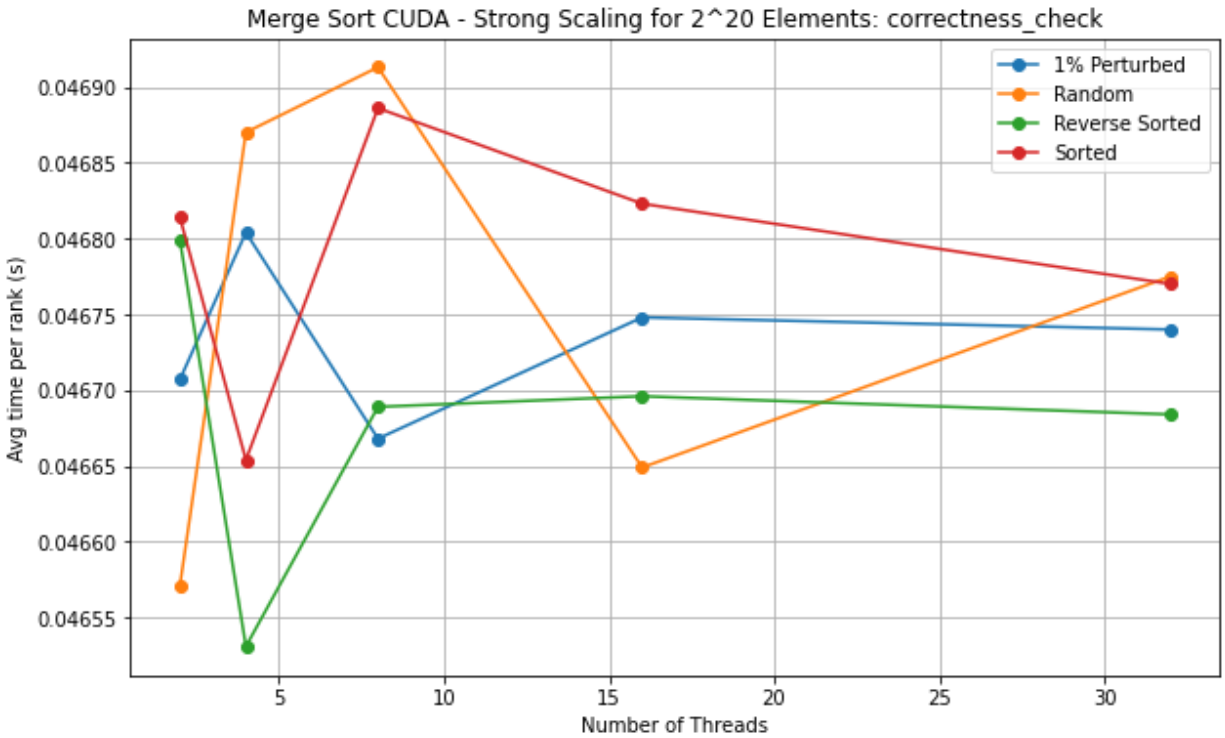


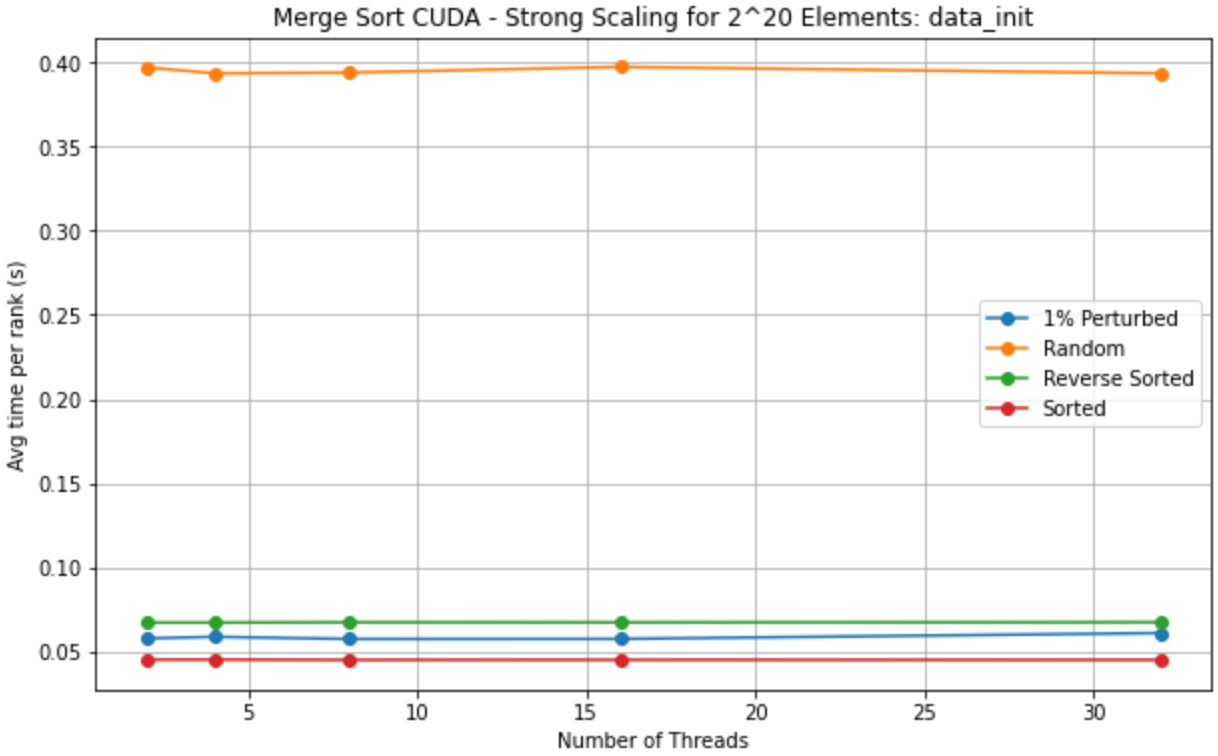



Merge Sort CUDA - Strong Scaling for 2^20 Elements: comm_large
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^20 Elements: comp
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^20 Elements: comp_large
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.00012
0.00011
0.00010
0.00009
0.00008
0.00007
5
10
15
20
25
30
Number of Threads


Merge Sort CUDA - Strong Scaling for 2^20 Elements: correctness_check
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.04690
0.04685
0.04680
0.04675
0.04670
0.04665
0.04660
0.04655
5
10
15
20
25
30
Number of Threads



2^22

In [53]:
```
tk22 = th.Thicket.from_caliperreader(glob("cali_data_cuda/*4194304*.cali"))
tk22.metadata
```

Out[53]:

profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
106929339	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
326037894	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
552869812	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
593281061	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1249895049	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1253943117	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1315551820	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1331496442	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1351151667	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2290947788	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2313735545	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2398174332	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2624507269	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2624724649	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3238077121	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3260250304	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3518408279	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3547221075	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

	cali.caliper.version	spot.metrics	spot.timeseries.metrics	profile
3845686239	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...		
3948053733	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...		

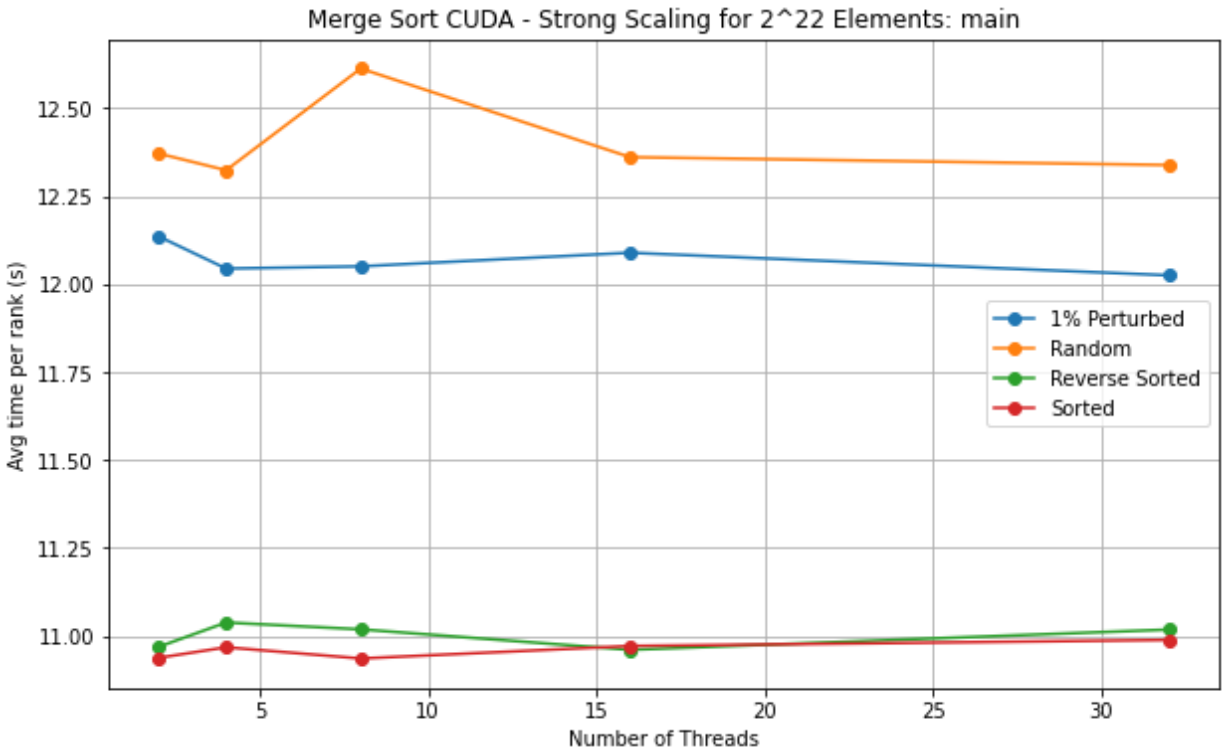
In [54]:
```
gb22 = tk22.groupby("InputType")

ctk22 = th.Thicket.concat_thickets(
    thickets=list(gb22.values()),
    headers=list(gb22.keys()),
    axis='columns',
    metadata_key='num_threads'
)

ctk22.dataframe = ctk22.dataframe.reset_index().drop([("node", ""), ('1% Perturbed', '
                                                      ('Random', "nid"), ('Random', "s
                                                      ('Reverse Sorted', "nid"), ('Rev
                                                      ('Sorted', "nid"), ('Sorted', "s
                                                      ('1% Perturbed', 'Avg GPU time/r
                                                      ('Random', 'Avg GPU time/rank'),
                                                      ('Reverse Sorted', "Avg GPU time
                                                      ('Sorted', "Avg GPU time/rank"),
                                                     ], axis=1)
ctk22.dataframe = ctk22.dataframe.rename({("name", ""): "name", ("num_threads", ""): '
```

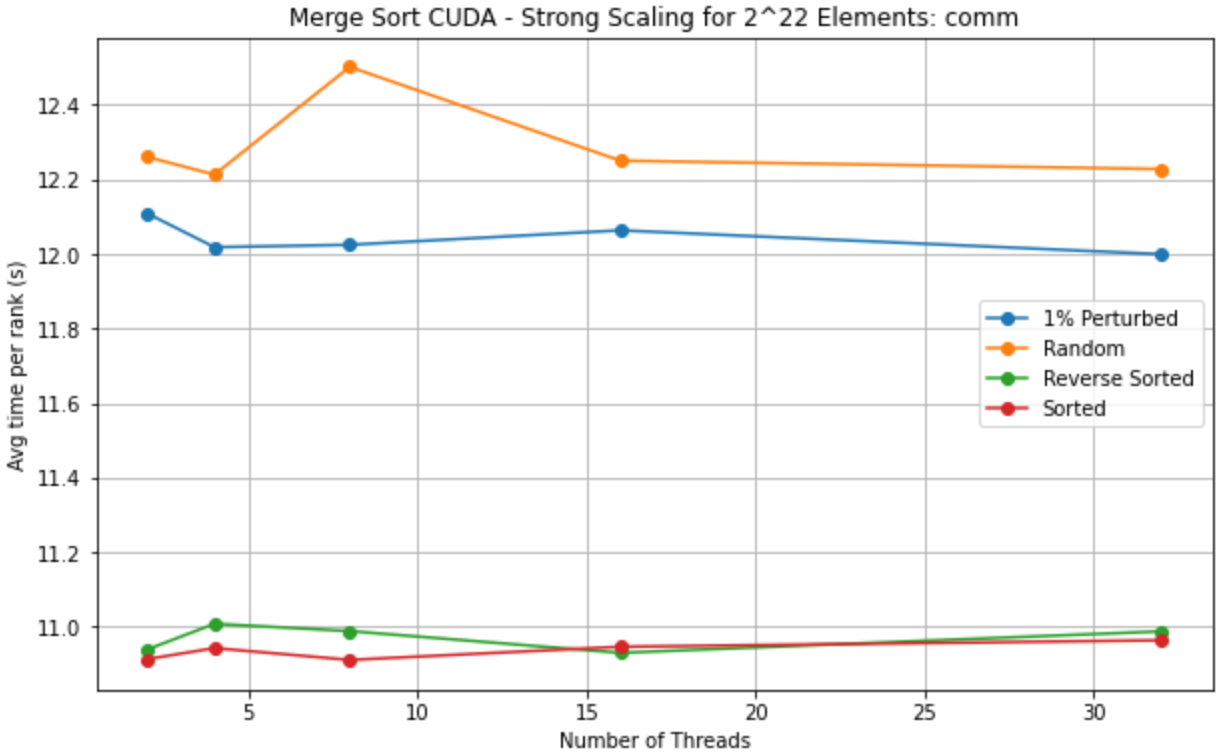
```
4  thickets created...
{'1% Perturbed': <thicket.thicket.Thicket object at 0x2b25c3e11b20>, 'Random': <thick
et.thicket.Thicket object at 0x2b25c6dbf7c0>, 'Reverse Sorted': <thicket.thicket.Thic
ket object at 0x2b25c474c460>, 'Sorted': <thicket.thicket.Thicket object at 0x2b25c48
6ff10>}
```

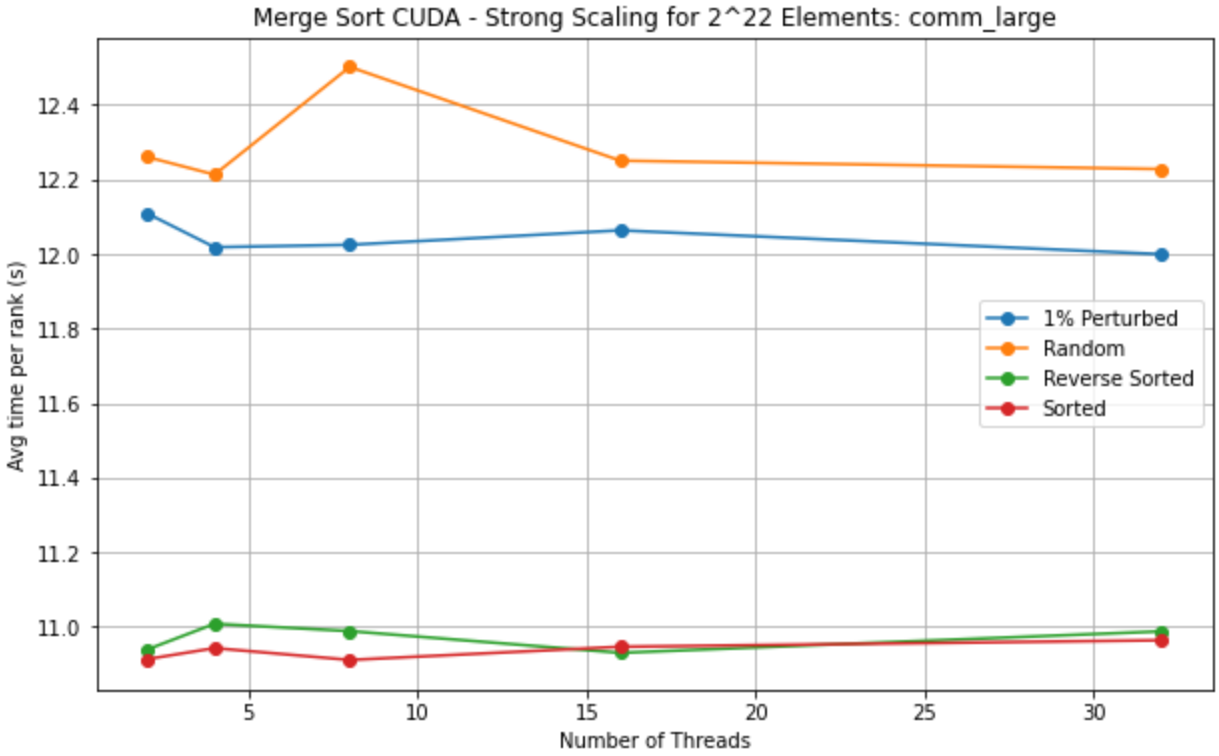
In [55]:
```
main = ctk22.dataframe.loc["main"]
comm = ctk22.dataframe.loc["comm"]
comm_large = ctk22.dataframe.loc["comm_large"]
comp = ctk22.dataframe.loc["comp"]
comp_large = ctk22.dataframe.loc["comp_large"]
correctness_check = ctk22.dataframe.loc["correctness_check"]
data_init = ctk22.dataframe.loc["data_init"]

regions = [main, comm, comm_large, comp, comp_large, correctness_check, data_init]
names = ["main", "comm", "comm_large", "comp", "comp_large", "correctness_check", "dat
```

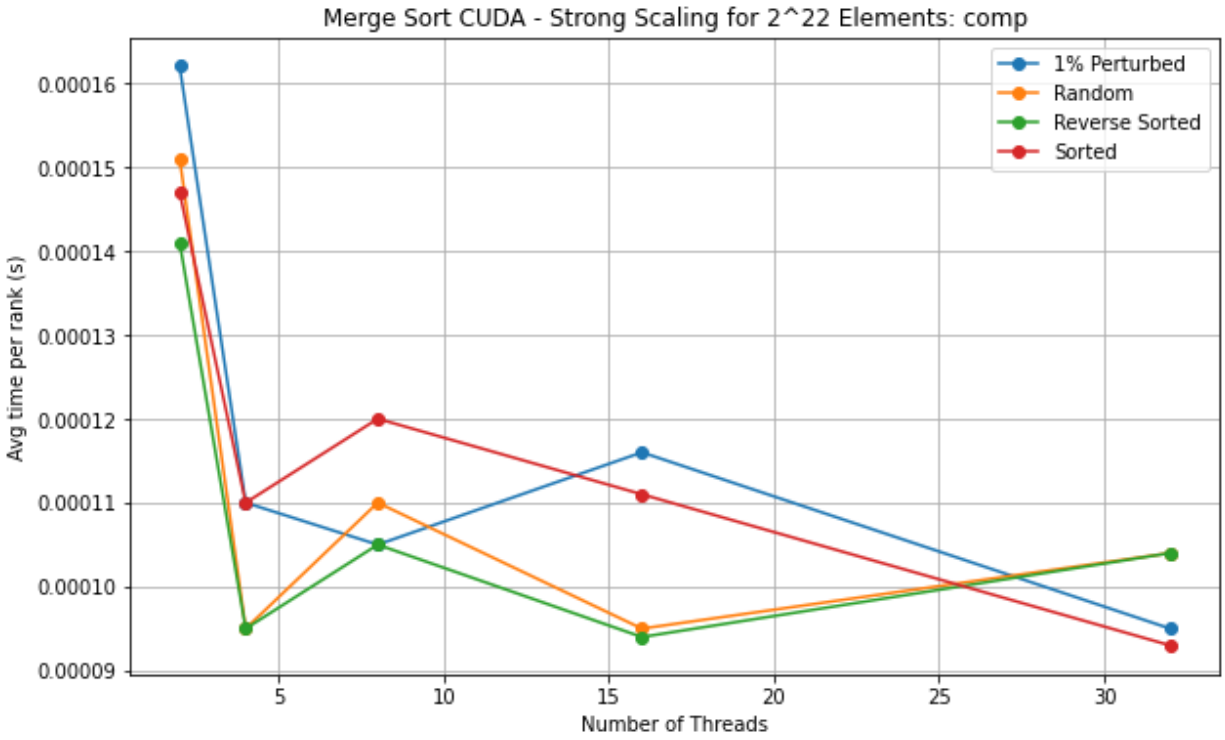
In [56]:
```
for region, name in zip(regions, names):
    plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
    legend_labels = []
    for column in region.columns:
        first_index = column[0]  # Extract the first index
        legend_labels.append(first_index)
        plt.plot(region.index, region.xs(column, axis=1), marker='o', label=column)

    plt.xlabel('Number of Threads')
    plt.ylabel('Avg time per rank (s)')
    plt.title(f'Merge Sort CUDA - Strong Scaling for 2^22 Elements: {name}')
    plt.legend(legend_labels)
```

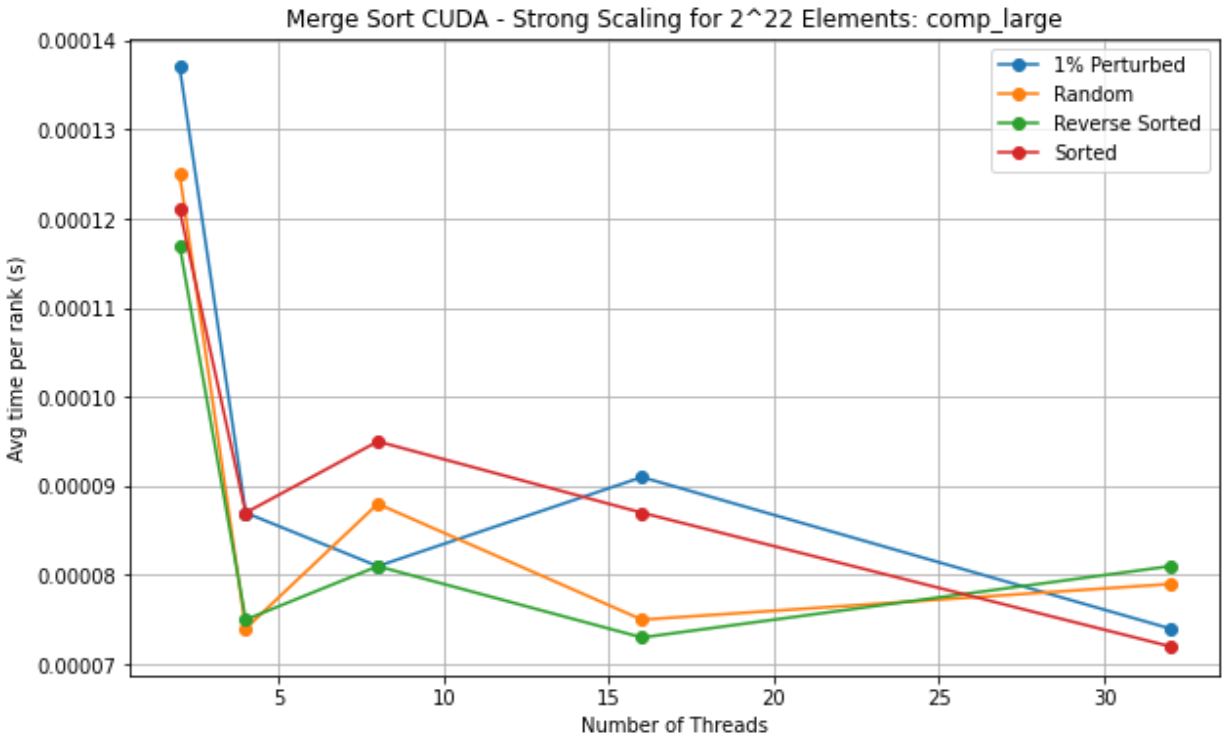
```
    plt.grid(True)
    plt.show()
```

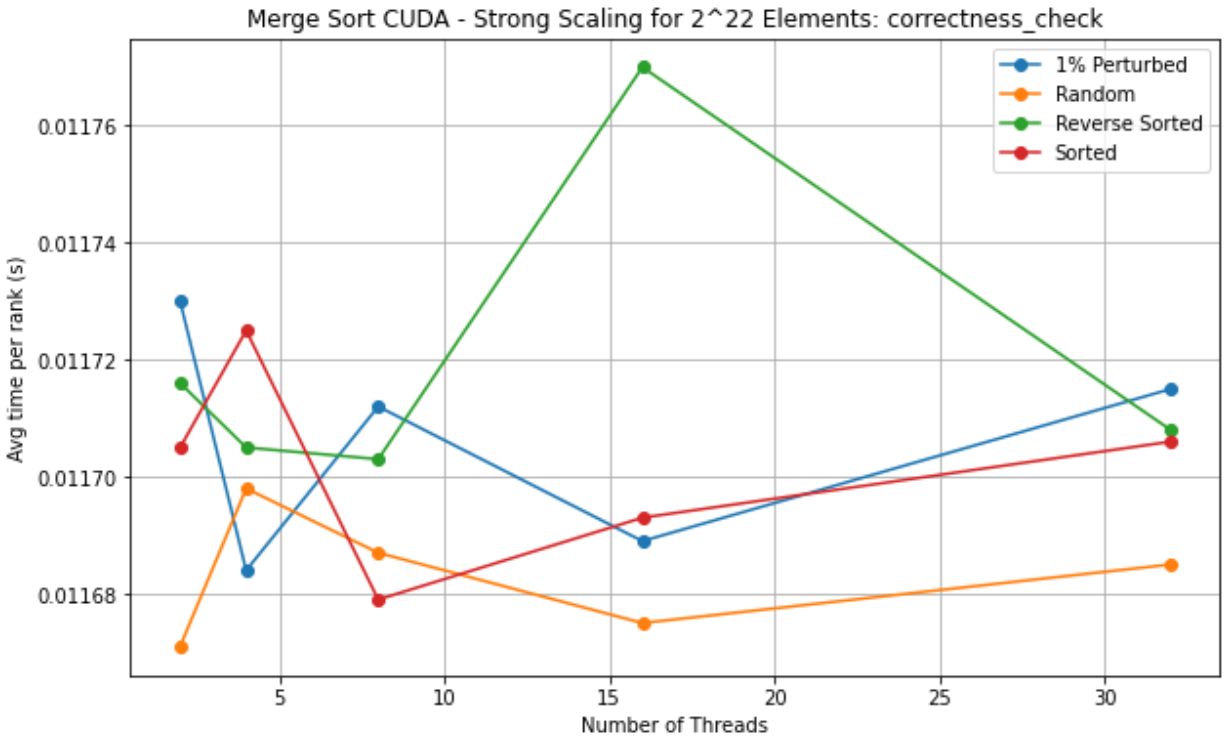


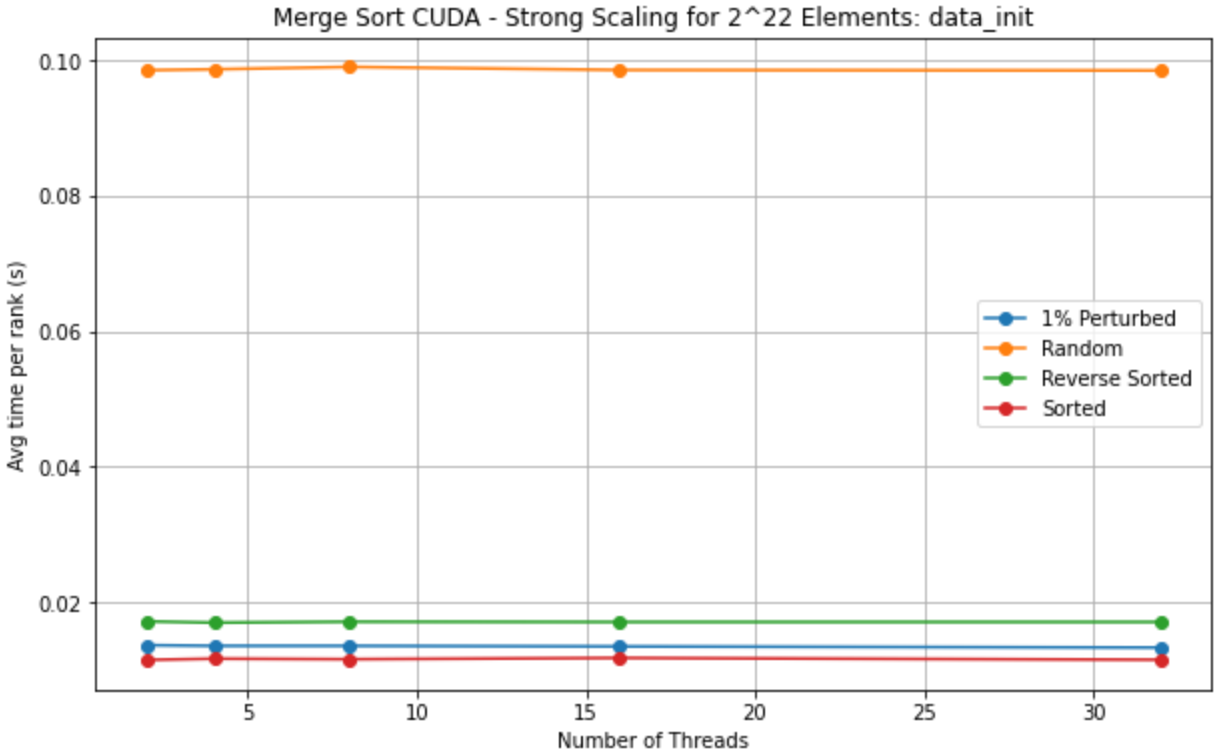



Merge Sort CUDA - Strong Scaling for 2^22 Elements: comm_large
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^22 Elements: comp
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^22 Elements: comp_large
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted
0.00014
0.00013
0.00012
0.00011
0.00010
0.00009
0.00008
0.00007
5
10
15
20
25
30


Merge Sort CUDA - Strong Scaling for 2^22 Elements: correctness_check
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted
0.01176
0.01174
0.01172
0.01170
0.01168
5
10
15
20
25
30



2^24

In [57]:
```
tk24 = th.Thicket.from_caliperreader(glob("cali_data_cuda/*16777216*.cali"))
tk24.metadata
```

Out[57]:

profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
405066056	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1052014983	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1108269707	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1172034348	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1328978594	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1397320877	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1484327580	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1640714657	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
1861528624	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2151748933	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2576479918	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
2762938203	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3106083997	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3217139869	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3264747243	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

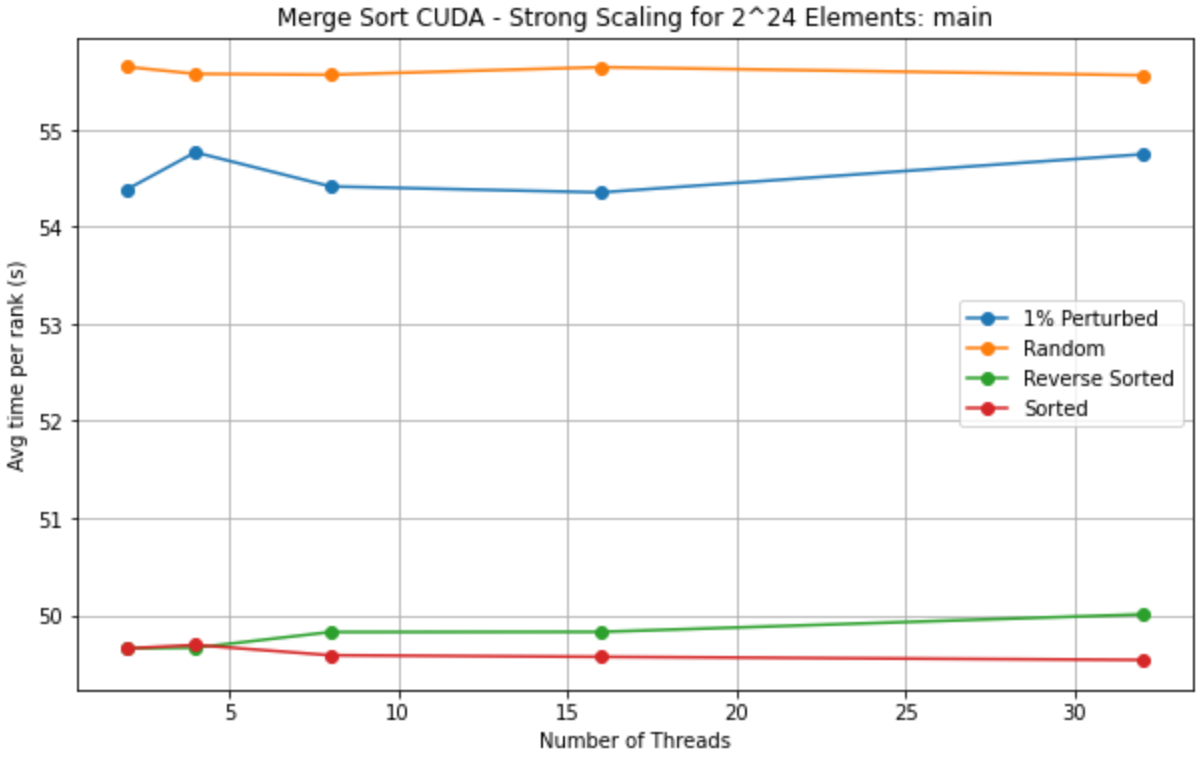
profile	cali.caliper.version	spot.metrics	spot.timeseries.metrics
3672410656	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3698302230	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
3795537088	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
4205446648	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	
4273584183	2.11.0-dev	min#inclusive#sum#time.duration,max#inclusive#...	

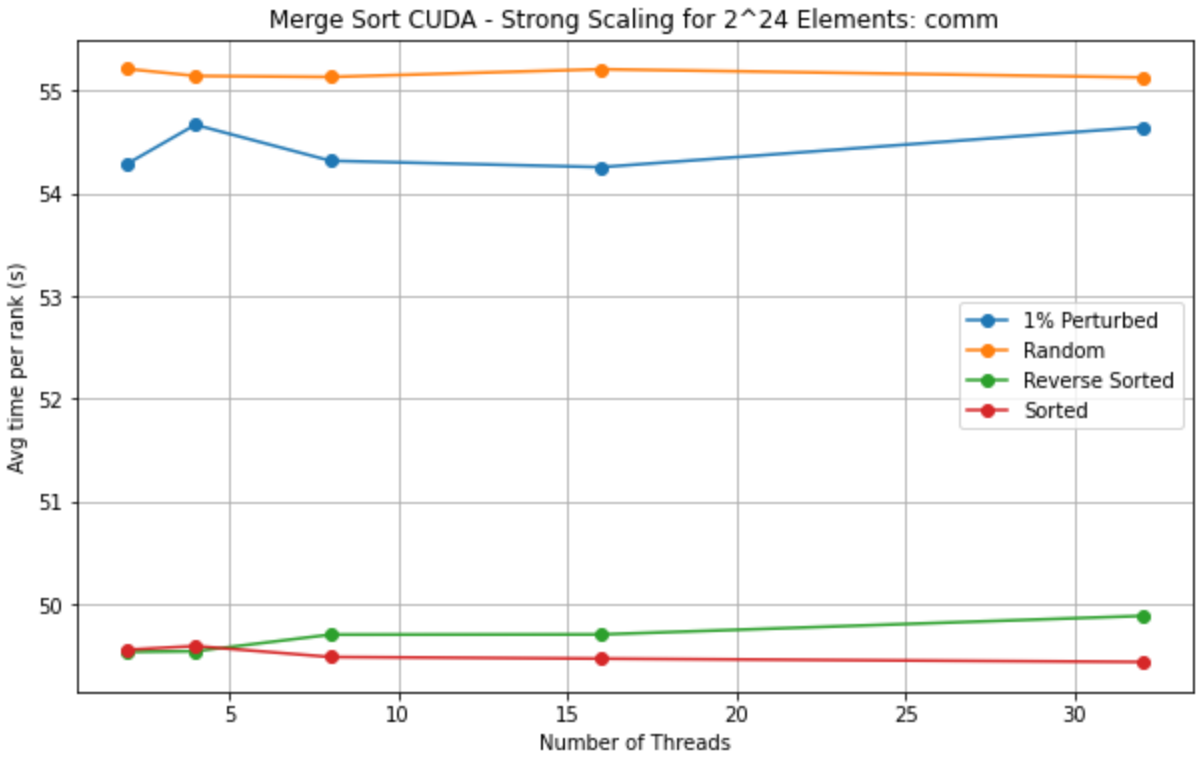
In [58]:
```
gb24 = tk24.groupby("InputType")

ctk24 = th.Thicket.concat_thickets(
    thickets=list(gb24.values()),
    headers=list(gb24.keys()),
    axis='columns',
    metadata_key='num_threads'
)

ctk24.dataframe = ctk24.dataframe.reset_index().drop([("node", ""), ('1% Perturbed', '
                                                      ('Random', "nid"), ('Random', "s
                                                      ('Reverse Sorted', "nid"), ('Rev
                                                      ('Sorted', "nid"), ('Sorted', "s
                                                      ('1% Perturbed', 'Avg GPU time/r
                                                      ('Random', 'Avg GPU time/rank'),
                                                      ('Reverse Sorted', "Avg GPU time
                                                      ('Sorted', "Avg GPU time/rank"),
                                                     ], axis=1)
ctk24.dataframe = ctk24.dataframe.rename({("name", ""): "name", ("num_threads", ""): '
```

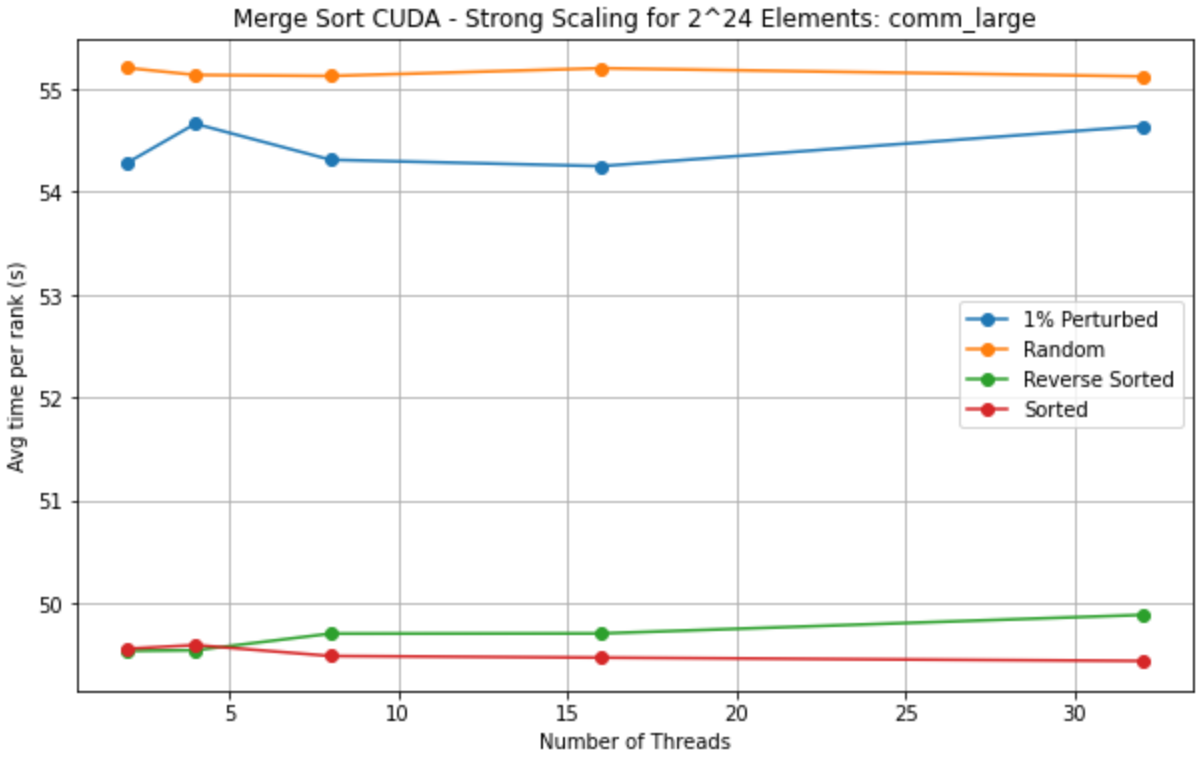
```
4  thickets created...
{'1% Perturbed': <thicket.thicket.Thicket object at 0x2b25c6eff970>, 'Random': <thick
et.thicket.Thicket object at 0x2b25c6f3c490>, 'Reverse Sorted': <thicket.thicket.Thic
ket object at 0x2b25c6eff310>, 'Sorted': <thicket.thicket.Thicket object at 0x2b25c6f
1c9a0>}
```

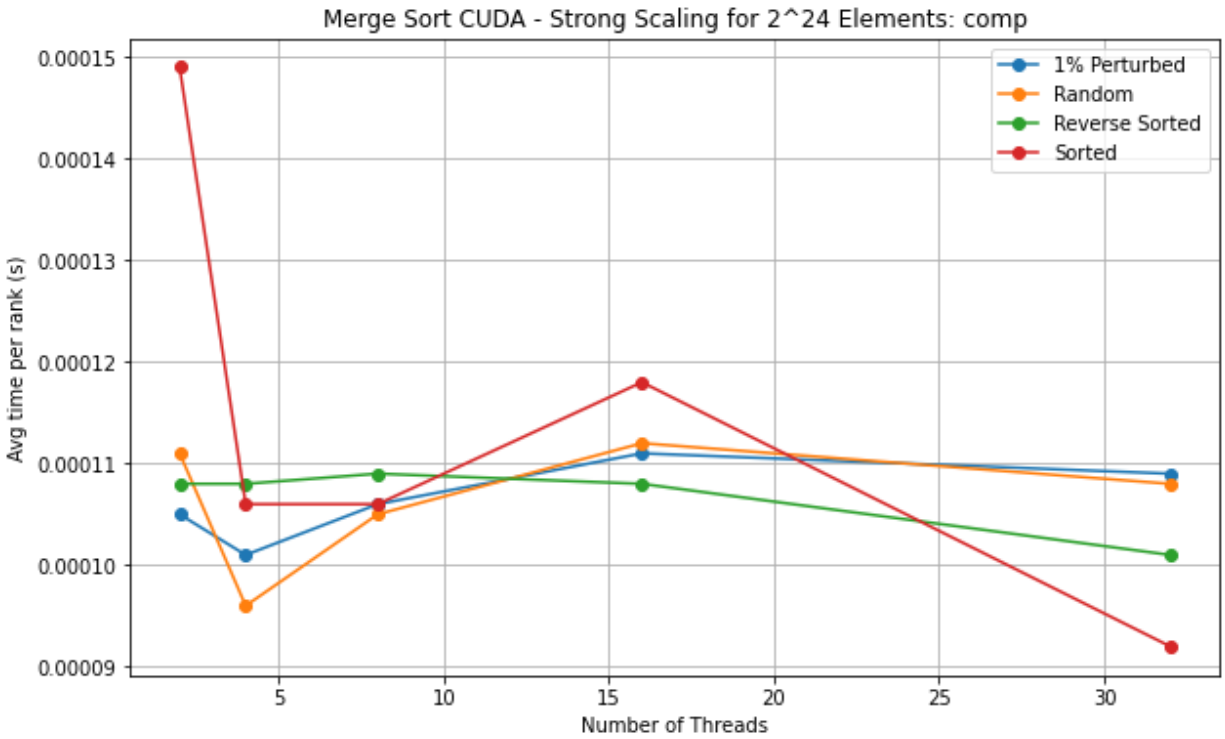
In [59]:
```
main = ctk24.dataframe.loc["main"]
comm = ctk24.dataframe.loc["comm"]
comm_large = ctk24.dataframe.loc["comm_large"]
comp = ctk24.dataframe.loc["comp"]
comp_large = ctk24.dataframe.loc["comp_large"]
correctness_check = ctk24.dataframe.loc["correctness_check"]
data_init = ctk24.dataframe.loc["data_init"]

regions = [main, comm, comm_large, comp, comp_large, correctness_check, data_init]
names = ["main", "comm", "comm_large", "comp", "comp_large", "correctness_check", "dat
```

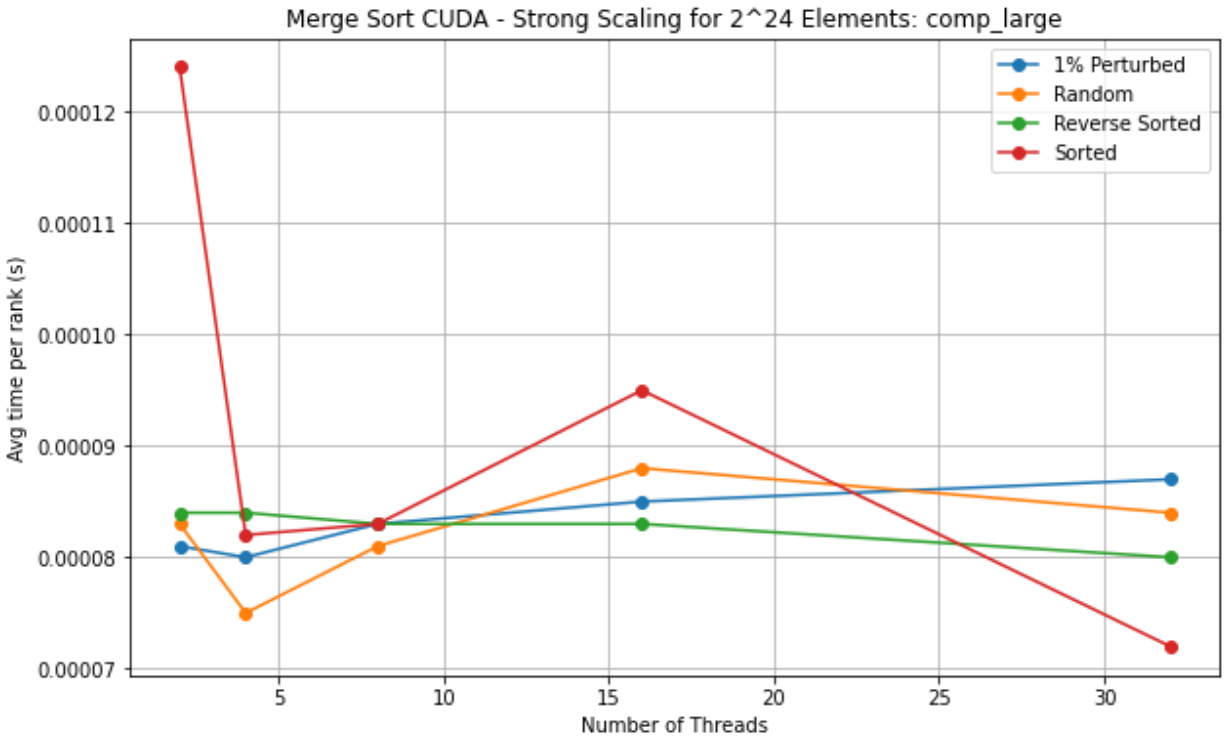
In [60]:
```
for region, name in zip(regions, names):
    plt.figure(figsize=(10, 6))  # Adjust the figure size if needed
    legend_labels = []
    for column in region.columns:
        first_index = column[0]  # Extract the first index
        legend_labels.append(first_index)
        plt.plot(region.index, region.xs(column, axis=1), marker='o', label=column)

    plt.xlabel('Number of Threads')
    plt.ylabel('Avg time per rank (s)')
    plt.title(f'Merge Sort CUDA - Strong Scaling for 2^24 Elements: {name}')
    plt.legend(legend_labels)
    plt.grid(True)
    plt.show()
```

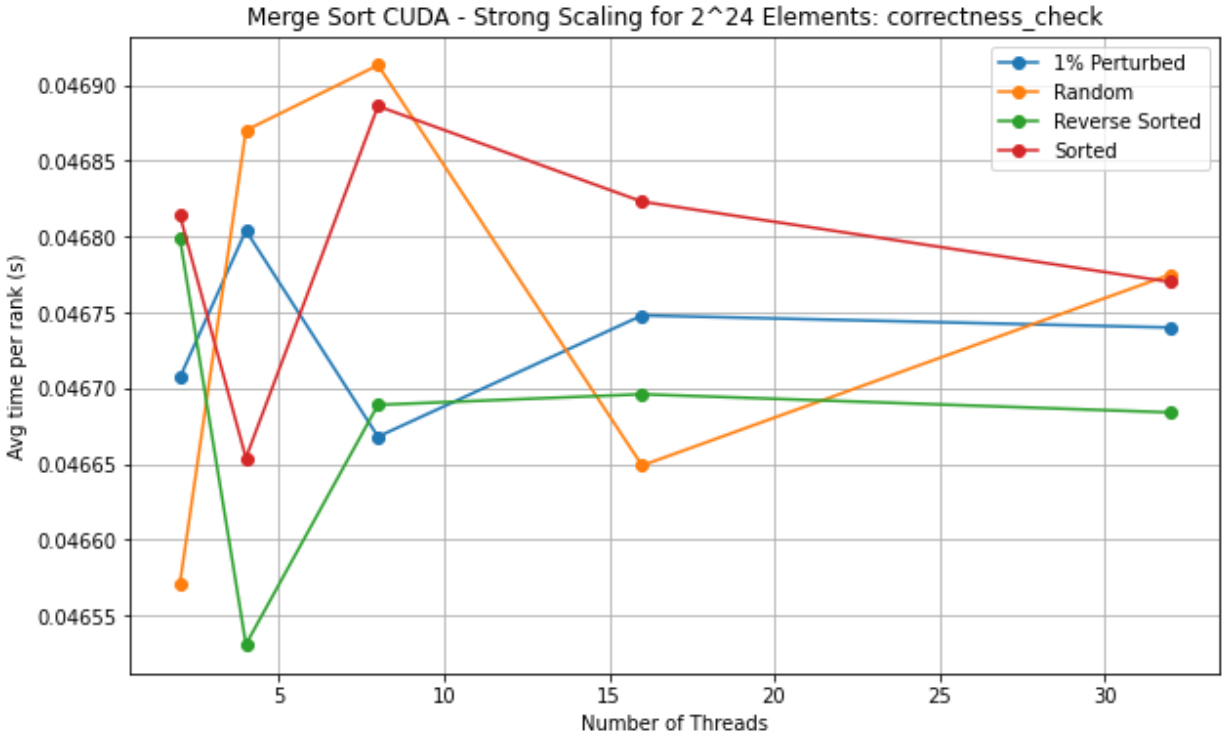


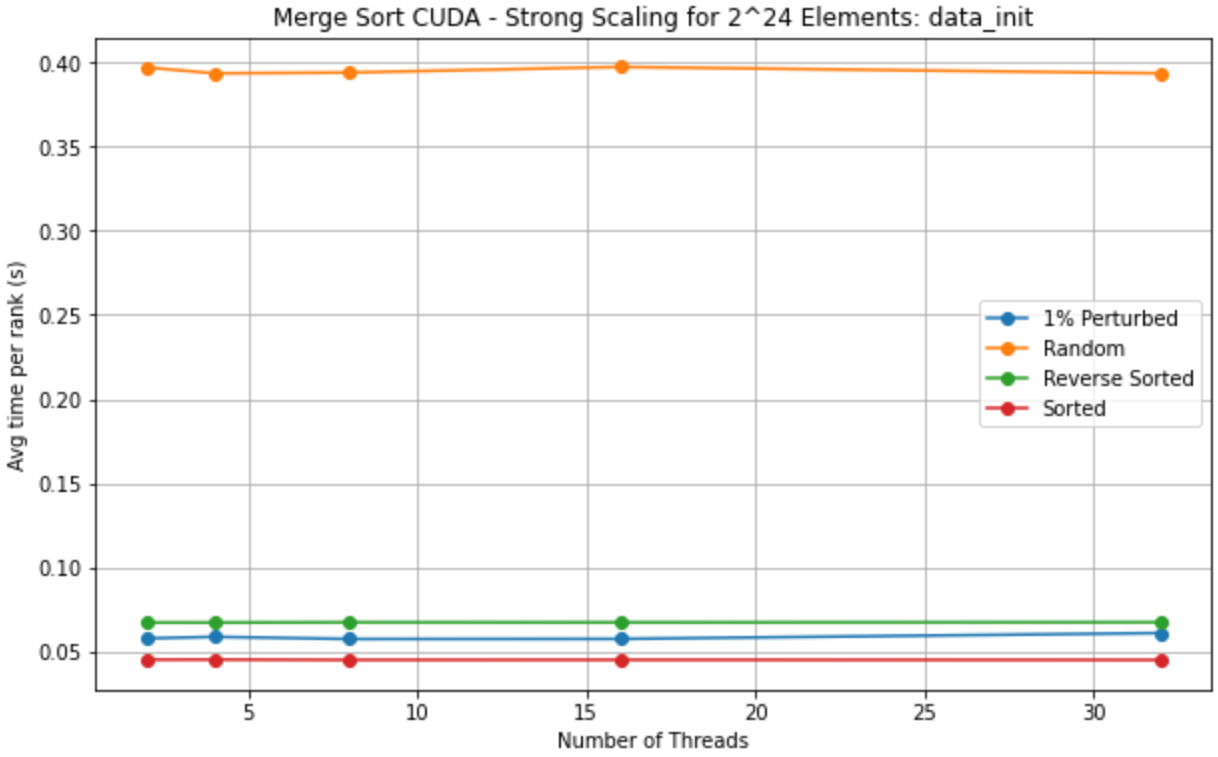

Merge Sort CUDA - Strong Scaling for 2^24 Elements: comm
Avg time per rank (s)
Number of Threads
55
54
53
52
51
50
5
10
15
20
25
30
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^24 Elements: comm_large
Avg time per rank (s)
Number of Threads
55
54
53
52
51
50
5
10
15
20
25
30
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^24 Elements: comp
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.00015
0.00014
0.00013
0.00012
0.00011
0.00010
0.00009
5
10
15
20
25
30
Number of Threads


Merge Sort CUDA - Strong Scaling for 2^24 Elements: comp_large
1% Perturbed
Random
Reverse Sorted
Sorted
Avg time per rank (s)
0.00012
0.00011
0.00010
0.00009
0.00008
0.00007
5
10
15
20
25
30
Number of Threads


Merge Sort CUDA - Strong Scaling for 2^24 Elements: correctness_check
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted


Merge Sort CUDA - Strong Scaling for 2^24 Elements: data_init
Avg time per rank (s)
Number of Threads
1% Perturbed
Random
Reverse Sorted
Sorted

In [ ]: