

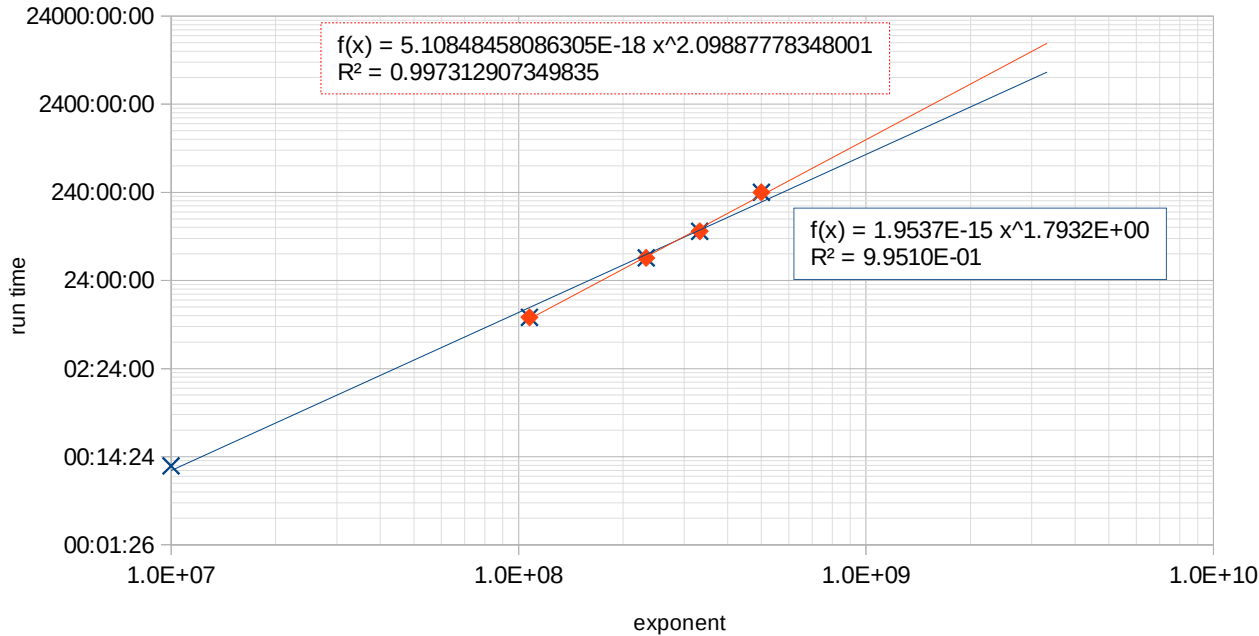
Mlucas V20.1.1 (built 2021-11-12) P-1 run time scaling Test system ostrich dual 8-core&HT Xeon e5-2690, 64 GiB ram, Win10 pro x64, WSL1, Ubuntu 20.04.2 LTS
 Windows remote desktop also running on the system during scaling measurements. (No prime95, Mfactor, web browser, etc.)

Exponent	B1	B2	Stage 1 time	gcd1 time	Buffer init	Stage 2 time	gcd2 time	total time	s2/s1	notes	factor if found
10000831	65000	1500000	00:04:23.55	00:00:02.47	00:03:46.40	00:03:04.53	00:00:02.29	00:11:19.24	0.702	1	646560662529991467527
107511749	700000	24000000	04:32:27.51	00:00:41.21	00:04:23.01	04:31:41.23	00:00:41.72	09:09:54.68	.997	1	none
233000099	1300000	50000000	19:16:53.40	00:01:49.37	00:03:40.84	23:52:11.76	00:01:46.28	43:16:21.65	1.238	1	none
332221607	1900000	80000000	36:56:26.87	00:02:45.74	00:04:00.35	49:39:59.64	00:02:44.67	86:45:57.27	1.344	1	none
500000777	3000000	130000000	96:19:13.13	00:04:33.77	00:03:55.99	143:18:21.90	00:04:22.35	239:50:27.13	1.487	1, 2	none
3321928307	17000000	1000000000									

1 run with -cpu 0:15 (16 cores) -shift 0

2 stage 2 observed using 59.9G virtual size, 49G resident

Mlucas 20.1.1 P-1 run time scaling on WSL1 16 cores dual e5-2690



Extrapolate to gigadigit

$T = a p^b$

a 1.95E-15

b 1.7932

p 3321928307

T 231.5 (days)

$T = a p^b$

a 5.11E-18

b 2.09887778

p 3321928307

T 492.6 (days)

Above extrapolations =

0.63

1.35

years

One year =

365 days

8760 hours

3.65 times 2400 hours

Usage in above was 16 mlucas threads, across 16 cores / 32 hyperthread capable hardware;

Initial timing of 1Gdigit S1 is 2.13 hours for 0.04%; 2.13 hours / 0.0004 * (1 + 2) / 24 / 365 =

GCDs at stage ends are at 500M 0.0621% of total yielding

1.87 years s1&s2

1.873 years with other loading described above.

Estimated stage 1 time = 7.49 months; yrs ~ 0.62

Running without remote desktop would help performance slightly.

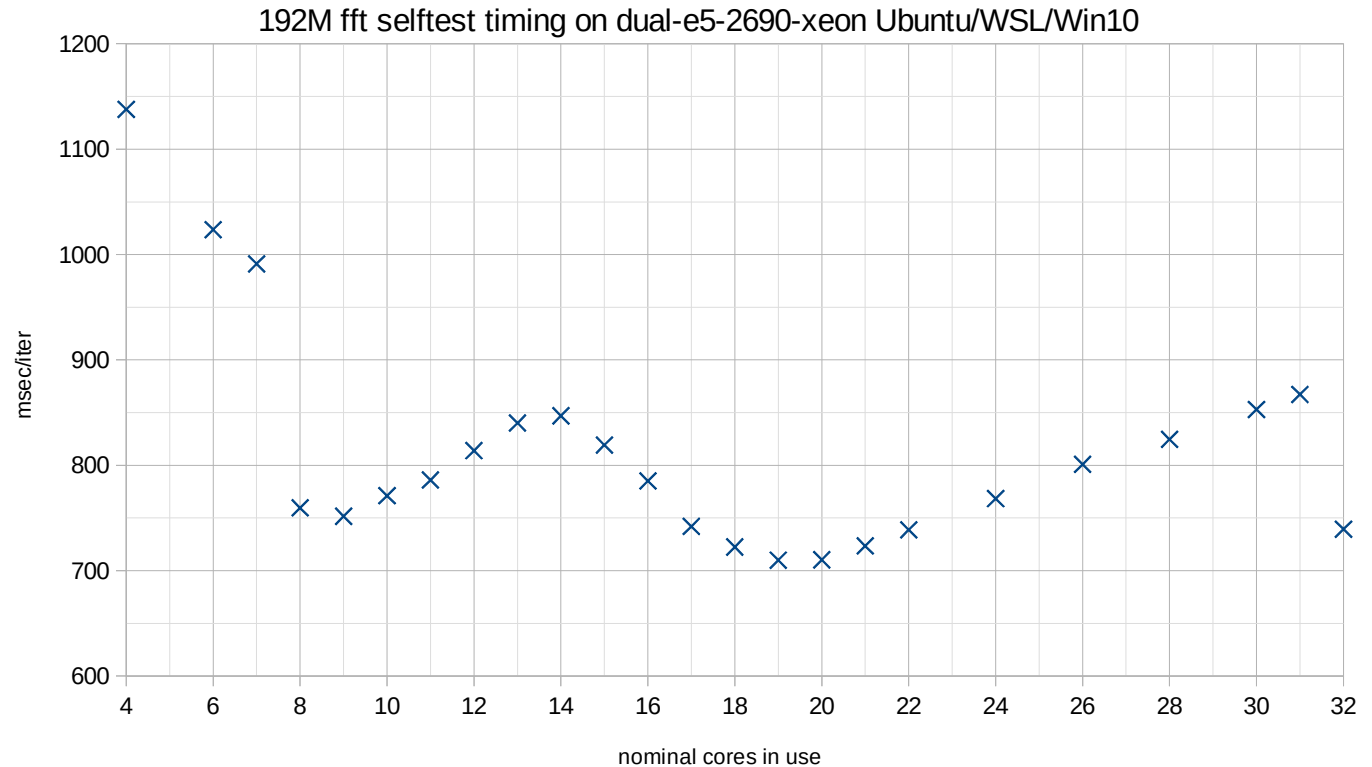
Update to Mlucas past V20.1.1 (when available) might also help.
 Slightly higher -maxalloc than the implicit default 90% used may help stage 2 performance..
 Possibly using fewer than the total number of cores would help.
 Splitting stage 2 with a slightly faster system would allow completion within a year.
 Additional performance past that could probably be obtained by a Linux native boot.

1.240 years of two equal systems in stage 2 extrapolated from stage 1 timing
 0.893 years of two equal systems in stage 2 extrapolated from red power fit
 1.067 average

192M fft length self-test timings versus number of "cores" on Ubuntu/WSL1/Win10

Single-instance performance optimization

"cores"	ms/iter	retest
16	785.14	786.41
15	819.18	
14	846.98	
13	840.09	
12	813.88	
11	785.97	
10	771.12	
9	751.59	
8	759.55	
7	991.18	
6	1023.68	
32	739.37	
24	768.34	
30	852.94	
18	722.42	
20	710.3	
22	738.68	
19	709.93	
21	723.47	
26	800.89	
28	824.61	
31	867.19	
4	1137.77	
17	741.93	



estimated speedup 9 vs 16 cores

4.55%

19 vs 16 10.59%

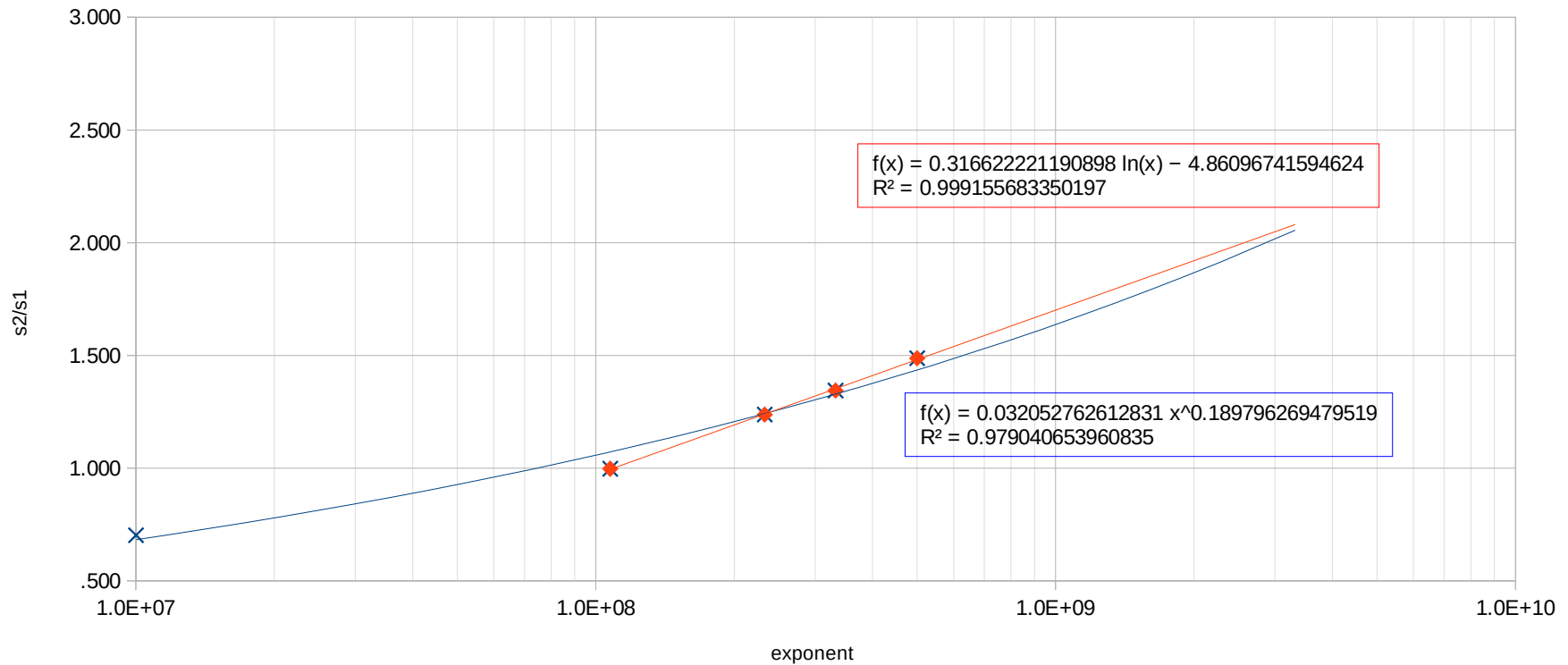
Equal systems at 9 cores in stage 2

total time est 1.020 years with tandem stage 2 of equal systems

Optimal -cpu **0.964 years with tandem stage 2 of equal systems**

Mlucas -cpu 0:15 on Ubuntu atop WSL1 appeared in Task Manager to primarily be using all hyperthreads of Xeon #2, not all 16 real cores of both Xeon packages in the initial 16-thread run-time scaling measurements.

duration ratio s2/s1 vs. exponent



Extrapolation to OBD s2/s1 duration ratio

2.0805984 2.0558423796

I used 2.08 in run time estimation above; ~2.056 would give slightly less overall duration.

Xeon 1 on s1 p1, xeon 2 on s2 p2 8 cores each & affinity specified ~2 year latency & ~1/year throughput estimated from above, passing some to other systems for s2
 (One of the Xeons in the system could complete 2 stage 1 in about the time that the other plus most of the ECC ram completes one stage 2)
 Running two instances 16-core -fft 192M -iters 100 gave 886.44 ms/iter and 839.02 ms/iter in separate folders/patch versions (2021-11-06 and 2021-12-02 respectively).
 Windows Task Manager indicated both Xeons being used on all hyperthreads. $1/(1/886.44+1/839.02)=$ 431.039 msec/iter aggregate throughput.

-cpu 0:14:2 and 16:30:2 also shows all logical processors being used significantly, but gives better 192M timings 766.1 and 744.61 **377.60** ms/iter aggregate throughput

Before ending testing on WSL, sudo apt install numactl etc.

Experiments with numactl & taskset on Ubuntu/WSL/Win10 were unfruitful.

All the following were obtained with Mlucas V20.1.1 2021-12-02 with the 2022-02-08 patch added.

Linux native boot try to skip virtualization overhead, WSL logical processor thrashing	WSL 16 core	WSL 8-core	WSL 9-core	WSL 19-core	for comparison
Centos 8 Stream install, -cpu 0:7 selftest 192M fft msec / iter result:	672.78	785.14	759.55	751.59	709.93
	1	116.70%	112.90%	111.7%	105.5% longer
Centos 8 Stream install, -cpu 0:14:2 selftest 192M fft msec/iter result: (which should be one thread per physical core of one Xeon)	559.33	140.37%	135.80%	134.37%	126.9%
	1.335 years				
Centos 8 Stream install, -cpu 0:30:2 selftest 192M fft msec/iter result: (which should be one thread per physical core of both Xeons)	610.41	128.63%			
	slower than 0:14:2				

Centos 8 Stream install, -cpu 0:15 selftest 192M fft msec/iter result: (which should be one thread per hyperthread of one Xeon)	505.34	estimated S1	estimated tandem S2	estimated tandem duration
	1.206 years solo	0.391	0.407	0.799

Centos 8 Stream install, -cpu 0:31 selftest 192M fft msec/iter result:
(which should be one thread per hyperthread of both Xeons)

Dual-instance Centos 8 Stream selftest 192M fft msec / iter results:
16 threads per instance, segregated per Xeon: 0:15, 16:31

effective combined throughput	832.14	836.83	417.24	110.50% longer than best WSL throughput
-------------------------------	--------	--------	---------------	---

Dual-instance Centos 8 Stream selftest 192M fft msec / iter results:
8 cores per instance, segregated per Xeon: 0:14:2, 16:30:2

effective combined throughput	1062.99	1063.70	531.67
-------------------------------	---------	---------	---------------

Centos 8 Stream install, -cpu 0:12:2 selftest 192M fft msec/iter result:
7 core

Centos 8 Stream install, -cpu 0:10:2 selftest 192M fft msec/iter result:
6 core

Centos 8 Stream install, -cpu 0:19 selftest 192M fft msec/iter result:
20 thread, all hyperthreads of one Xeon, 4 of the other on 2 cores

Centos 8 Stream install, -cpu 0:23 selftest 192M fft msec/iter result:
24 thread, all hyperthreads of one Xeon, 8 of the other on 4 cores

Centos 8 Stream install, -cpu 0:17 selftest 192M fft msec/iter result:
18 thread, all hyperthreads of one Xeon, 2 of the other on 1 core

Summary:	(in limited testing, only 1 or 2 instances)	
Best overall solo completion time, years	1.206	CentOS native boot, using all hyperthreads on one Xeon, other Xeon idle
Best overall tandem time estimate, yrs	0.799	CentOS native boot, using all hyperthreads on one Xeon, other Xeon idle
Best overall throughput, equiv duration	0.901	Ubuntu/WSL/Win10, 1 thread / core, two instances (note, higher instance count not tested)