Università degli Studi di Salerno
DIEM, a.a 2021-22
Course: HIGH PERFORMANCE COMPUTING
Lecturer: Francesco Moscato

PARALLELIZATION AND PERFORMANCES EVALUATION OF ***COUNTING SORT*** ALGORITHM

Salvatore Grimaldi	0622701742	s.grimaldi29@studenti.unisa.it
Enrico Maria Di Mauro	0622701706	e.dimauro5@studenti.unisa.it
Allegra Cuzzocrea	0622701707	a.cuzzocrea2@studenti.unisa.it

Sommario

Problem description

Parallelize and Evaluate Performances of "COUNTING SORT" Algorithm, by using OpenMP.

Introduction

The purpose of this report consists of improving counting sort algorithm's performances through parallelization using OpenMP (Open Multi-Processing), which is an Application Programming Interface (API) that supports multiplatform shared memory multiprocessing programming.

Parallelization introduces an overhead. In fact, when OMP is used in a program, a completely transparent mechanism is introduced. It handles threads, shared memory, synchronization, cache, pipeline.

It implies a delay (not dependent on the particular algorithm) which is crucial in order to distinguish between theoretical and real speedup.

Experimental setup

Information provided below refers to hardware and software used during the evaluation of performances.

Hardware

CPU

```
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 4
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 4
apicid          : 1
initial apicid  : 1
```

```
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 2
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 1
cpu cores       : 4
apicid          : 2
initial apicid  : 2
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 3
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
```

```
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 1
cpu cores       : 4
apicid          : 3
initial apicid  : 3
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 4
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 2
cpu cores       : 4
apicid          : 4
initial apicid  : 4
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
```

```
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 5
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 2
cpu cores       : 4
apicid          : 5
initial apicid  : 5
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 6
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 3
cpu cores       : 4
apicid          : 6
initial apicid  : 6
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
```

```
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 7
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz
stepping        : 12
microcode       : 0xffffffff
cpu MHz         : 1992.006
cache size      : 8192 KB
physical id     : 0
siblings        : 8
core id         : 3
cpu cores       : 4
apicid          : 7
initial apicid  : 7
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm
constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq vmx ssse3 fma cx16 pcid
sse4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm
3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vnmi
ept vpid ept_ad fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap
clflushopt xsaveopt xsavec xgetbv1 xsaves flush_l1d arch_capabilities
vmx flags       : vnmi invvpid ept_x_only ept_ad ept_1gb tsc_offset vtpr ept
vpid unrestricted_guest ept_mode_based_exec
bugs            : spectre_v1 spectre_v2 spec_store_bypass swapgs itlb_multihit
srbds
bogomips        : 3984.01
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:
```

RAM

```
MemTotal:         3930548 kB
MemFree:          3276532 kB
MemAvailable:     3272192 kB
Buffers:            15300 kB
Cached:            168732 kB
SwapCached:             0 kB
Active:             63876 kB
Inactive:          327516 kB
Active(anon):         204 kB
Inactive(anon):    207440 kB
Active(file):       63672 kB
Inactive(file):    120076 kB
Unevictable:            0 kB
Mlocked:                0 kB
SwapTotal:        1048576 kB
SwapFree:         1048576 kB
Dirty:                  4 kB
Writeback:              0 kB
AnonPages:         207396 kB
Mapped:             67260 kB
Shmem:                284 kB
KReclaimable:       24688 kB
Slab:               57384 kB
SReclaimable:       24688 kB
SUnreclaim:         32696 kB
KernelStack:         3712 kB
PageTables:          7708 kB
NFS_Unstable:           0 kB
Bounce:                 0 kB
WritebackTmp:           0 kB
CommitLimit:      3013848 kB
Committed_AS:      737008 kB
VmallocTotal:   34359738367 kB
VmallocUsed:        23564 kB
VmallocChunk:           0 kB
Percpu:              2560 kB
AnonHugePages:      40960 kB
ShmemHugePages:         0 kB
ShmemPmdMapped:         0 kB
FileHugePages:          0 kB
FilePmdMapped:          0 kB
HugePages_Total:        0
HugePages_Free:         0
HugePages_Rsvd:         0
HugePages_Surp:         0
Hugepagesize:        2048 kB
Hugetlb:                0 kB
DirectMap4k:        19456 kB
DirectMap2M:      1984512 kB
DirectMap1G:     10485760 kB
```

Software

```
Ubuntu       : 20.04.3
GCC          : 9.3.0
```

Performance, Speedup & Efficiency

Case Study n.1 – WIKI Counting Sort

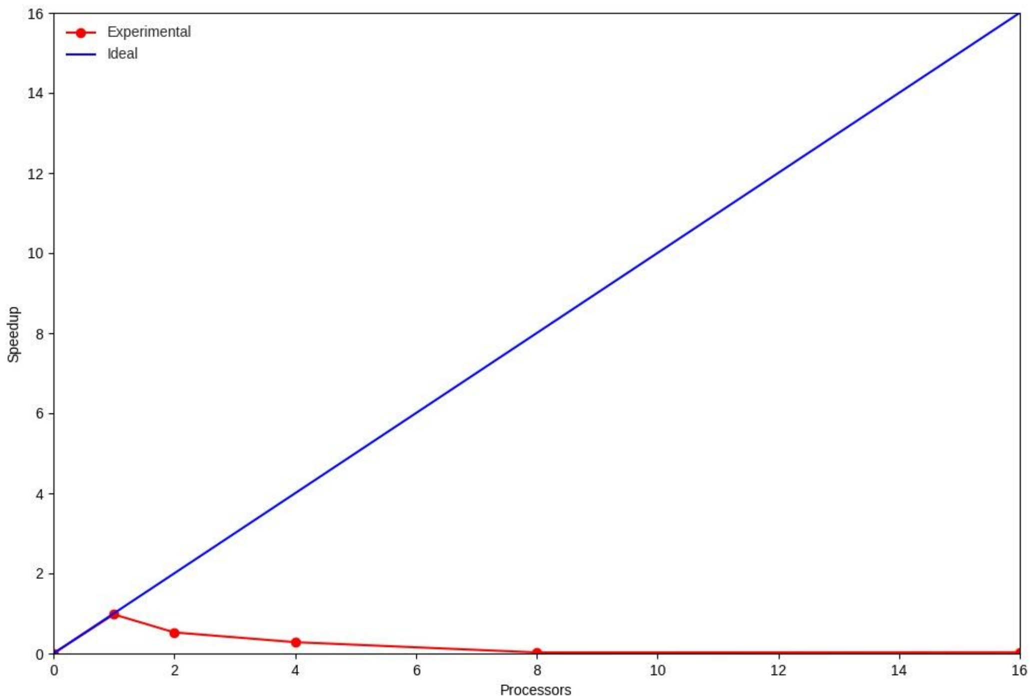
In this case study the main purpose is to analyse the performances of a first version of the counting sort algorithm, that can be found here. The build setup is:

Both the sequential and the parallel programs are compiled with the gcc optimization -Ox, where x = 0, 1, 2, 3 (O0 means without any optimization).

We want to highlight the difference between a sequential program compared to a parallel one, compiled with the same optimizations. The case study is done on multiple size: 10000, 100000, 1000000, and with different number of threads (0, 1, 2, 4, 8, 16).

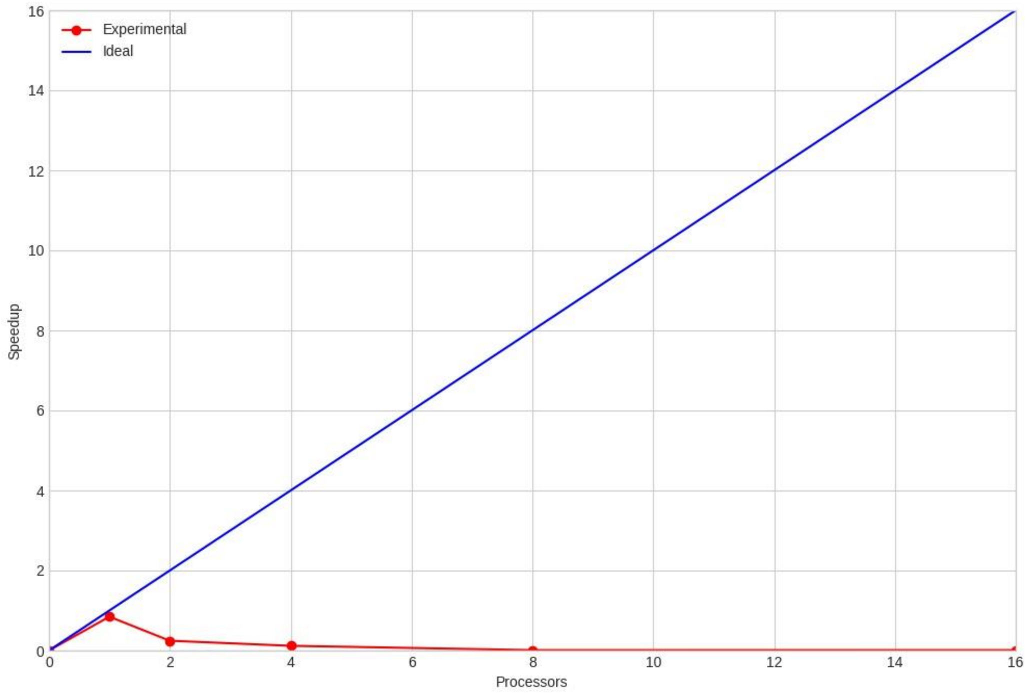
SIZE-10.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00014	0,00376	0,00002	0,01027	1,00000	1,00000
Parallel	1	0,00014	0,00378	0,00002	0,00985	0,97628	0,97628
Parallel	2	0,00026	0,00380	0,00003	0,00977	0,52642	0,26321
Parallel	4	0,00048	0,00428	0,00006	0,01083	0,28224	0,07056
Parallel	8	0,00529	0,02616	0,00230	0,01534	0,02565	0,00321
Parallel	16	0,00453	0,00088	0,01343	0,01496	0,02994	0,00187



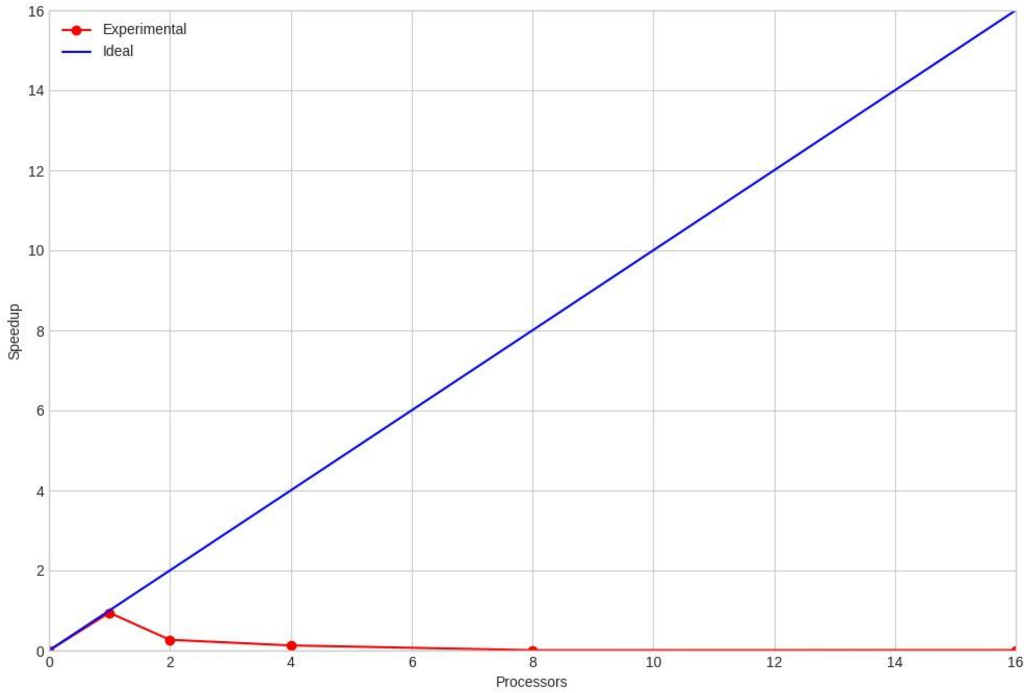
SIZE-10.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00005	0,00367	0,00004	0,00937	1,00000	1,00000
Parallel	1	0,00006	0,00368	0,00003	0,01023	0,84648	0,84648
Parallel	2	0,00020	0,00380	0,00001	0,00989	0,24453	0,12227
Parallel	4	0,00041	0,00415	0,00010	0,00987	0,11802	0,02950
Parallel	8	0,00454	0,02325	0,00154	0,01523	0,01058	0,00132
Parallel	16	0,00440	0,00107	0,00834	0,01463	0,01090	0,00068



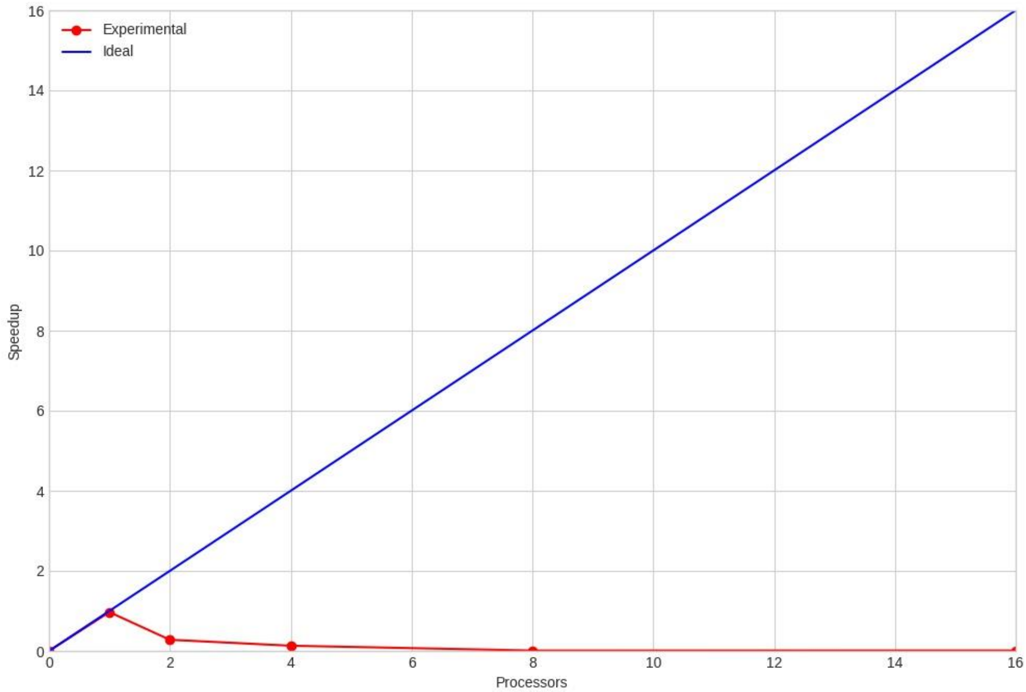
SIZE-10.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00005	0,00368	0,00003	0,00984	1,00000	1,00000
Parallel	1	0,00005	0,00368	0,00001	0,00941	0,94490	0,94490
Parallel	2	0,00019	0,00371	0,00005	0,00988	0,26858	0,13429
Parallel	4	0,00040	0,00377	0,00009	0,00979	0,12763	0,03191
Parallel	8	0,00542	0,02658	0,00304	0,01568	0,00945	0,00118
Parallel	16	0,00453	0,00100	0,01379	0,01494	0,01130	0,00071



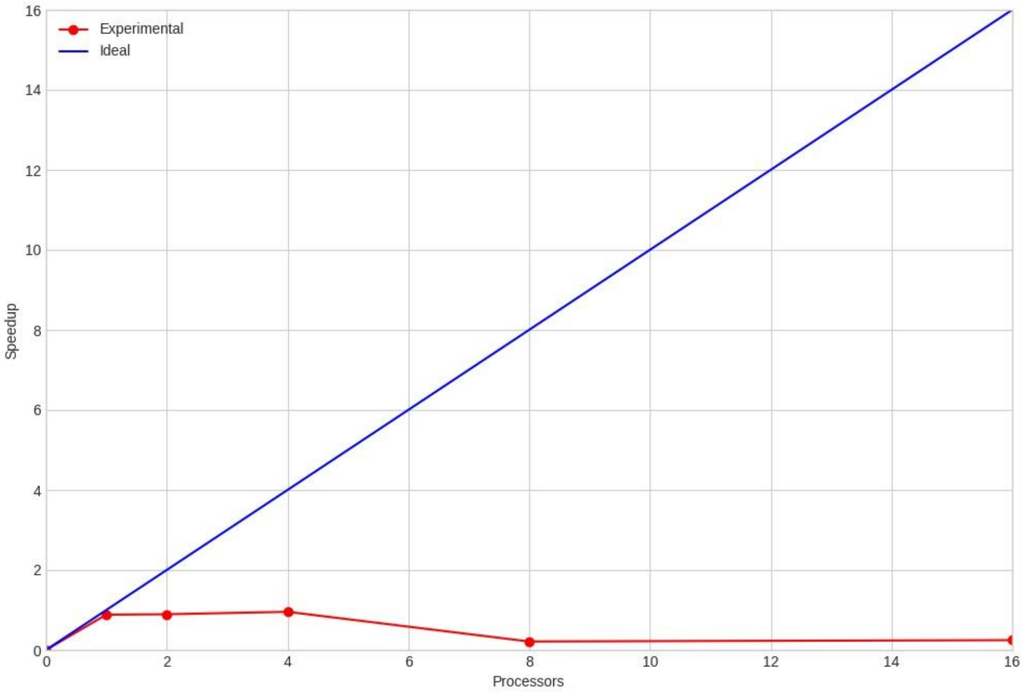
SIZE-10.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00006	0,00367	0,00002	0,00943	1,00000	1,00000
Parallel	1	0,00006	0,00367	0,00003	0,00953	0,97153	0,97153
Parallel	2	0,00020	0,00377	0,00001	0,00950	0,28079	0,14039
Parallel	4	0,00042	0,00418	0,00008	0,01045	0,13193	0,03298
Parallel	8	0,00510	0,02425	0,00222	0,01528	0,01099	0,00137
Parallel	16	0,00464	0,00126	0,01376	0,01529	0,01207	0,00075



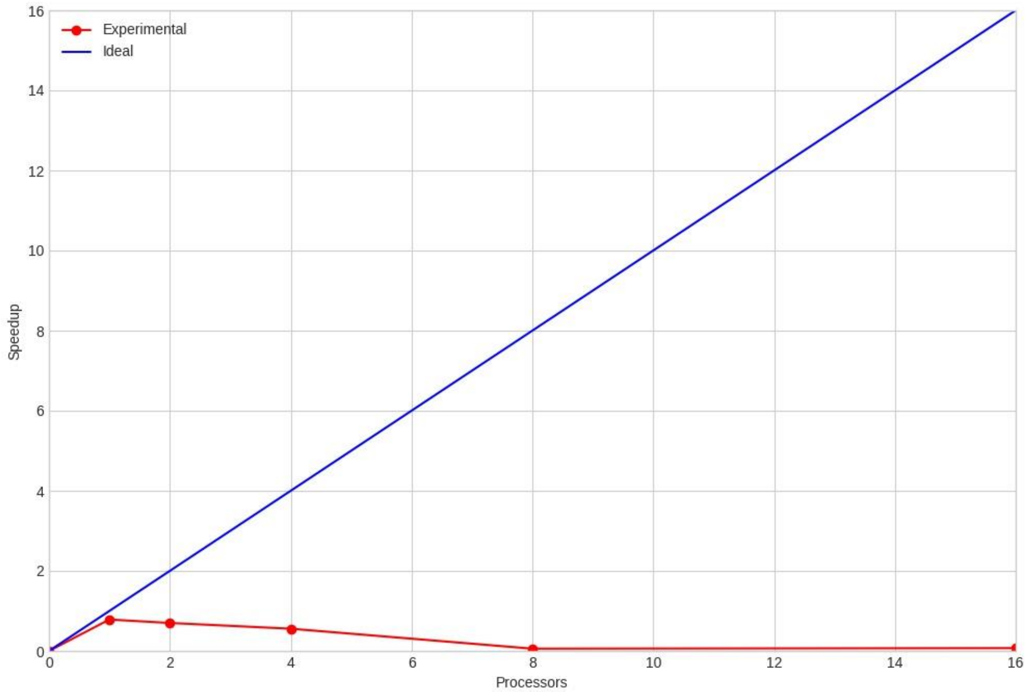
SIZE-100.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00118	0,00620	0,00017	0,01280	1,00000	1,00000
Parallel	1	0,00133	0,00687	0,00015	0,01386	0,88377	0,88377
Parallel	2	0,00132	0,00777	0,00036	0,01617	0,89404	0,44702
Parallel	4	0,00123	0,00664	0,00033	0,01288	0,95550	0,23888
Parallel	8	0,00552	0,02872	0,00244	0,01646	0,21343	0,02668
Parallel	16	0,00477	0,00164	0,01470	0,01585	0,24695	0,01543



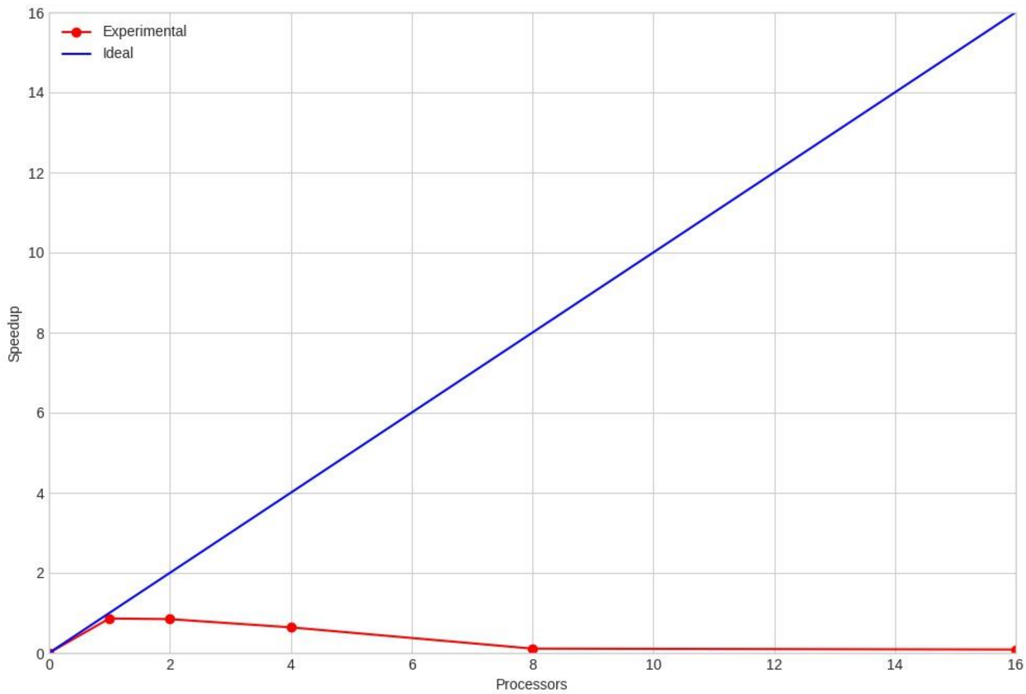
SIZE-100.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00032	0,00548	0,00007	0,01188	1,00000	1,00000
Parallel	1	0,00041	0,00671	0,00012	0,01438	0,78571	0,78571
Parallel	2	0,00046	0,00635	0,00017	0,01355	0,69956	0,34978
Parallel	4	0,00058	0,00567	0,00008	0,01191	0,55723	0,13931
Parallel	8	0,00554	0,03157	0,00237	0,01593	0,05821	0,00728
Parallel	16	0,00451	0,00175	0,01466	0,01573	0,07153	0,00447



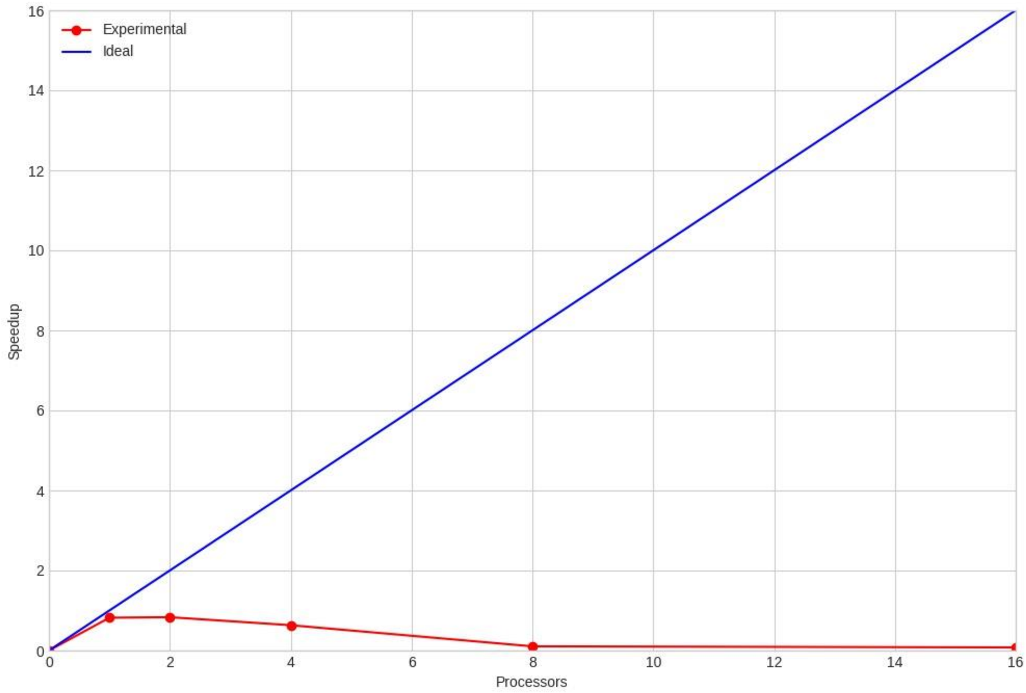
SIZE-100.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00038	0,00606	0,00008	0,01288	1,00000	1,00000
Parallel	1	0,00044	0,00671	0,00022	0,01444	0,86219	0,86219
Parallel	2	0,00045	0,00609	0,00007	0,01302	0,84569	0,42284
Parallel	4	0,00059	0,00582	0,00008	0,01160	0,64060	0,16015
Parallel	8	0,00346	0,01797	0,00159	0,01419	0,10993	0,01374
Parallel	16	0,00439	0,00054	0,01525	0,01575	0,08661	0,00541



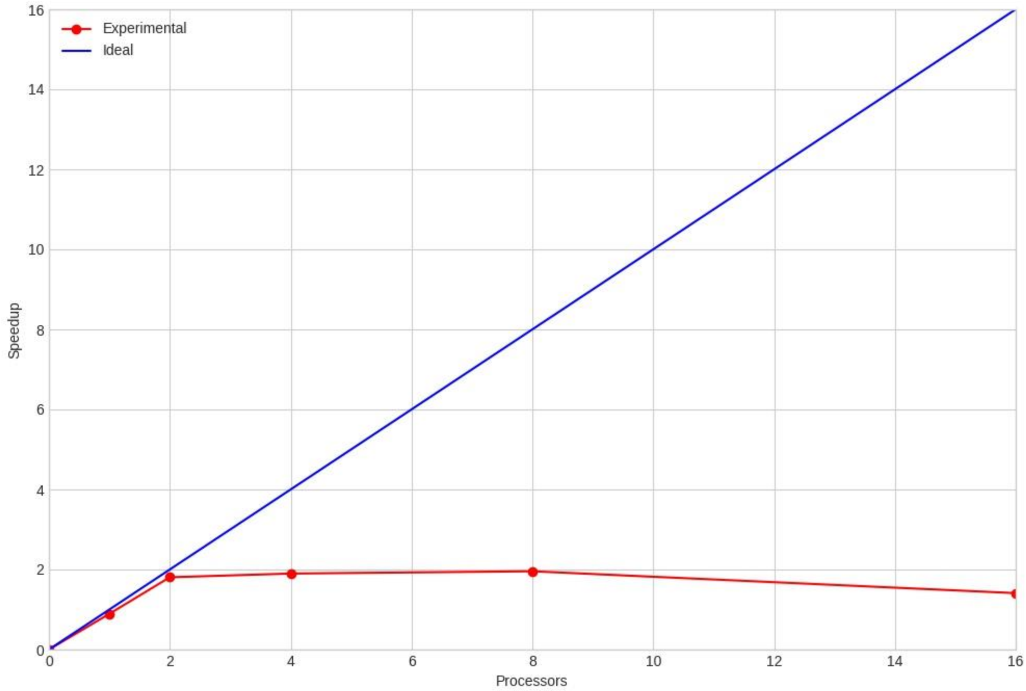
SIZE-100.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00034	0,00552	0,00015	0,01248	1,00000	1,00000
Parallel	1	0,00042	0,00693	0,00023	0,01495	0,81990	0,81990
Parallel	2	0,00041	0,00584	0,00007	0,01278	0,83276	0,41638
Parallel	4	0,00054	0,00528	0,00007	0,01142	0,63309	0,15827
Parallel	8	0,00328	0,01280	0,00226	0,01371	0,10481	0,01310
Parallel	16	0,00443	0,00289	0,01496	0,01606	0,07757	0,00485



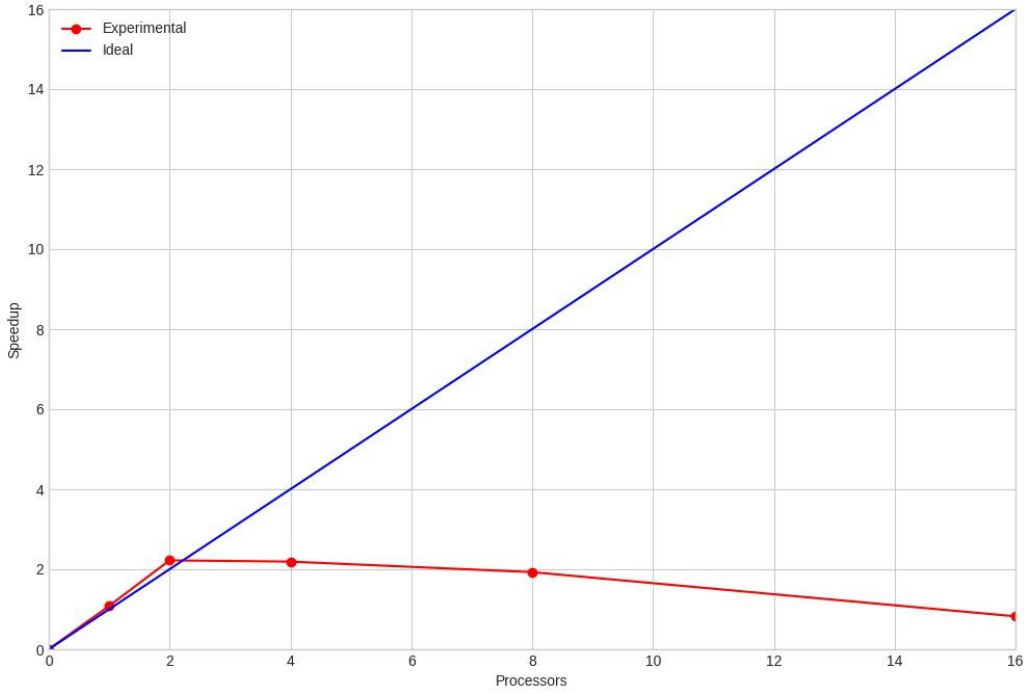
SIZE-1.000.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,01109	0,02752	0,00007	0,03472	1,00000	1,00000
Parallel	1	0,01243	0,03048	0,00002	0,03818	0,89238	0,89238
Parallel	2	0,00614	0,02432	0,00068	0,02583	1,80607	0,90303
Parallel	4	0,00584	0,03181	0,00053	0,02503	1,89874	0,47469
Parallel	8	0,00567	0,04476	0,00232	0,02581	1,95582	0,24448
Parallel	16	0,00787	0,03133	0,00660	0,02836	1,40965	0,08810



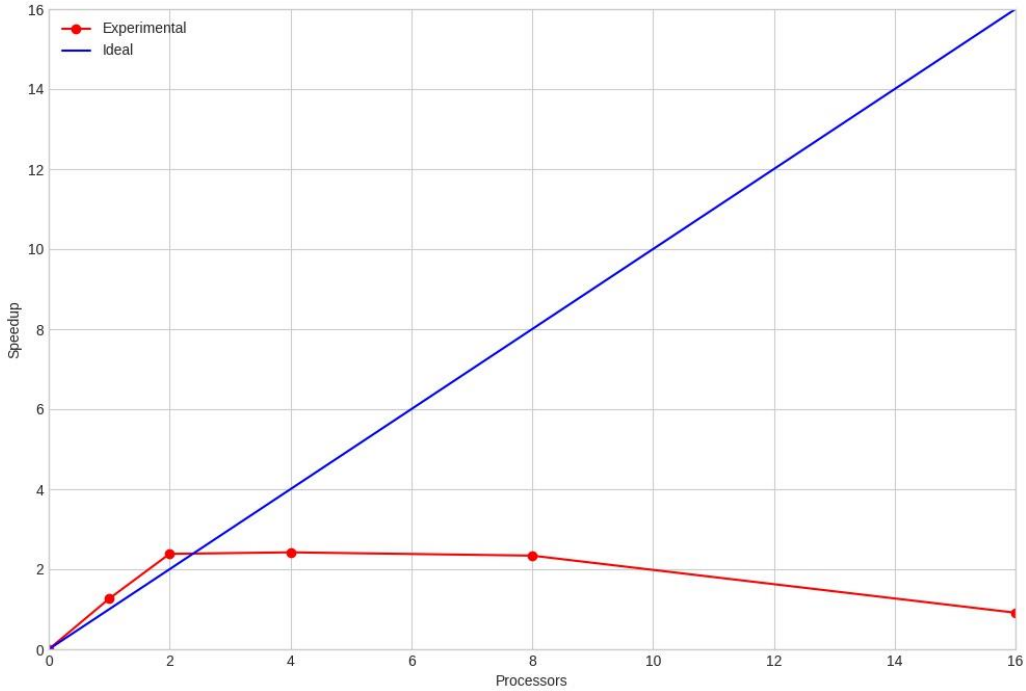
SIZE-1.000.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00372	0,02229	0,00093	0,03062	1,00000	1,00000
Parallel	1	0,00344	0,02010	0,00090	0,02791	1,08342	1,08342
Parallel	2	0,00168	0,01510	0,00025	0,02096	2,22232	1,11116
Parallel	4	0,00170	0,01549	0,00006	0,02141	2,19125	0,54781
Parallel	8	0,00193	0,01723	0,00126	0,02173	1,92831	0,24104
Parallel	16	0,00452	0,01659	0,01354	0,02545	0,82294	0,05143



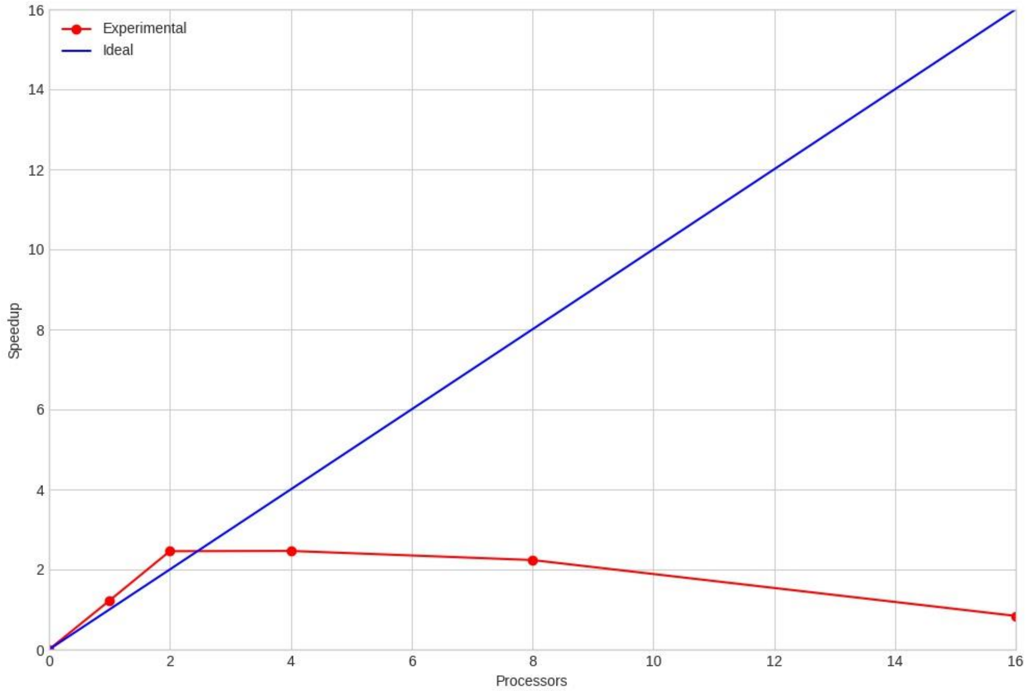
SIZE-1.000.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00431	0,02328	0,00139	0,03279	1,00000	1,00000
Parallel	1	0,00341	0,02009	0,00015	0,02646	1,26374	1,26374
Parallel	2	0,00181	0,01550	0,00031	0,02150	2,38411	1,19206
Parallel	4	0,00178	0,01545	0,00035	0,02132	2,42287	0,60572
Parallel	8	0,00184	0,01599	0,00074	0,02174	2,34061	0,29258
Parallel	16	0,00472	0,02335	0,00654	0,02620	0,91380	0,05711



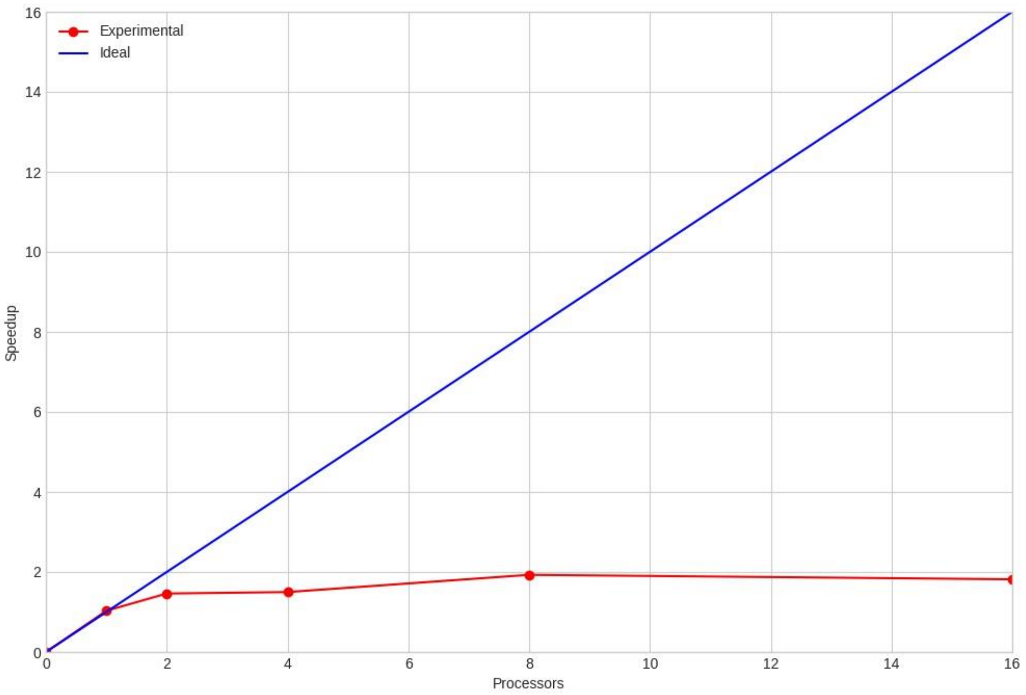
SIZE-1.000.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00390	0,02215	0,00158	0,03120	1,00000	1,00000
Parallel	1	0,00320	0,01792	0,00004	0,02454	1,22148	1,22148
Parallel	2	0,00159	0,01469	0,00030	0,02047	2,45760	1,22880
Parallel	4	0,00158	0,01501	0,00009	0,02084	2,46435	0,61609
Parallel	8	0,00174	0,01543	0,00160	0,02104	2,23684	0,27960
Parallel	16	0,00464	0,02213	0,00672	0,02646	0,84033	0,05252



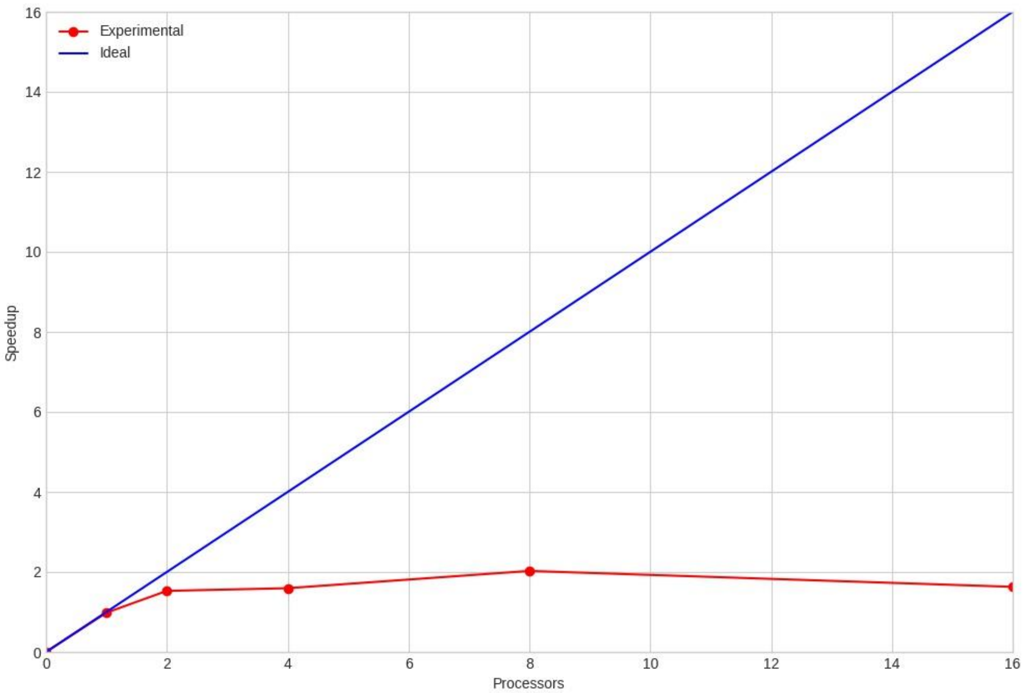
SIZE-10.000.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,07448	0,15889	0,00431	0,16843	1,00000	1,00000
Parallel	1	0,07226	0,15423	0,00354	0,16287	1,03075	1,03075
Parallel	2	0,05093	0,16498	0,00383	0,14710	1,46232	0,73116
Parallel	4	0,04969	0,19986	0,00323	0,14846	1,49879	0,37470
Parallel	8	0,03861	0,22799	0,00479	0,13604	1,92922	0,24115
Parallel	16	0,04097	0,20926	0,00673	0,13672	1,81783	0,11361



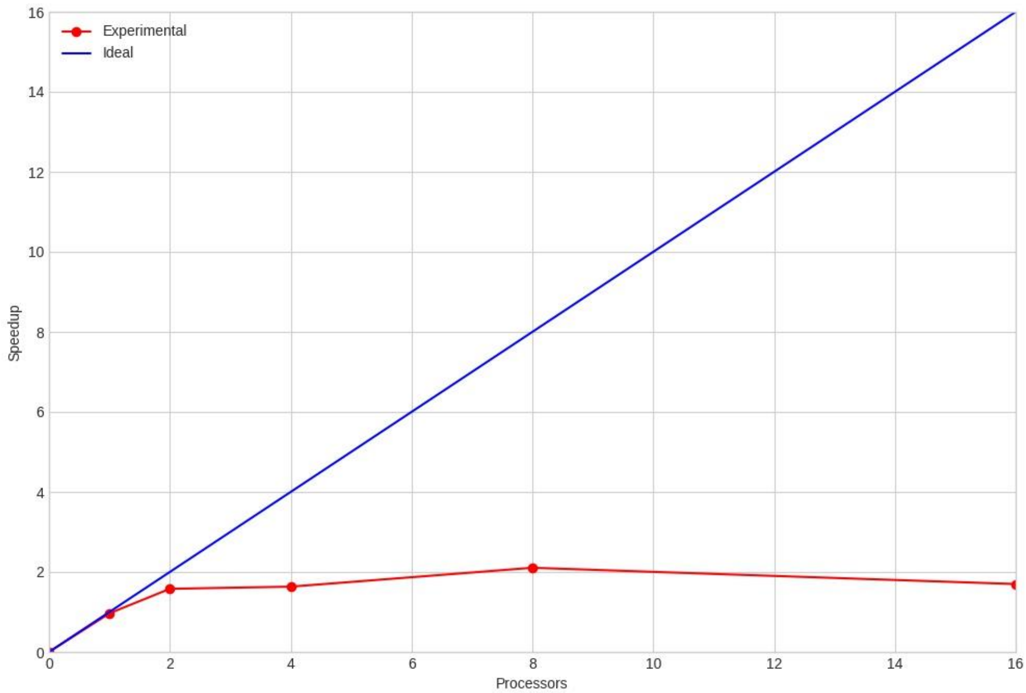
SIZE-10.000.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,02314	0,10729	0,00301	0,11548	1,00000	1,00000
Parallel	1	0,02357	0,10897	0,00055	0,11784	0,98195	0,98195
Parallel	2	0,01514	0,11317	0,00393	0,11283	1,52797	0,76399
Parallel	4	0,01450	0,12888	0,00257	0,10847	1,59539	0,39885
Parallel	8	0,01142	0,15010	0,00287	0,10736	2,02715	0,25339
Parallel	16	0,01420	0,13956	0,00557	0,11283	1,62917	0,10182



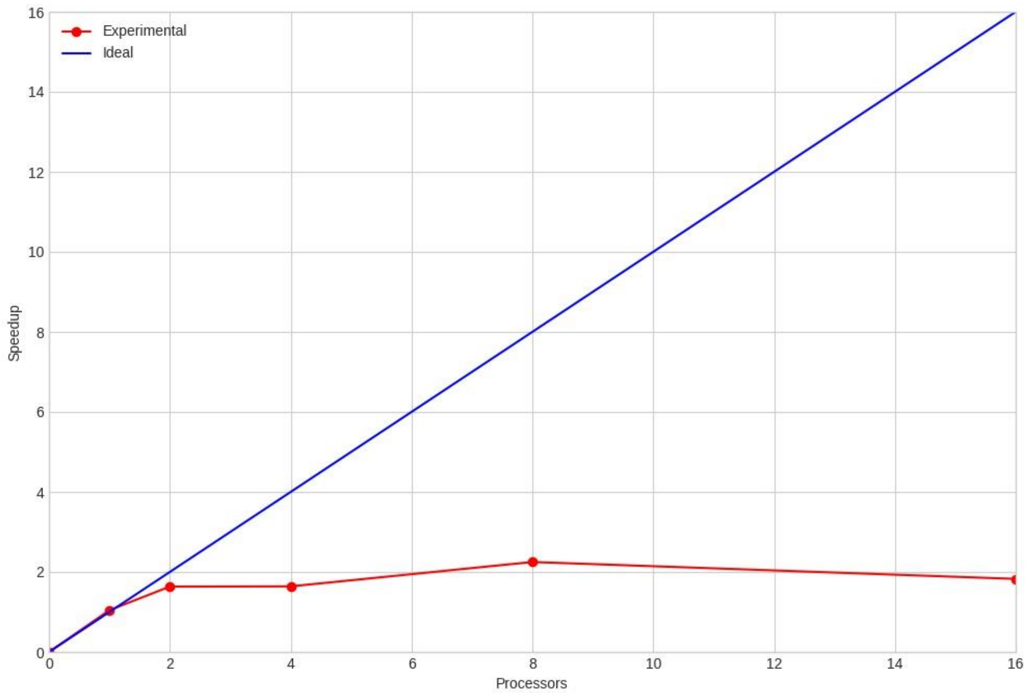
SIZE-10.000.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,02501	0,10996	0,00317	0,11730	1,00000	1,00000
Parallel	1	0,02588	0,11211	0,00376	0,12183	0,96627	0,96627
Parallel	2	0,01583	0,11386	0,00302	0,10988	1,57961	0,78981
Parallel	4	0,01531	0,13120	0,00282	0,11038	1,63371	0,40843
Parallel	8	0,01188	0,15872	0,00397	0,11304	2,10530	0,26316
Parallel	16	0,01472	0,14405	0,00642	0,11426	1,69913	0,10620



SIZE-10.000.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,02608	0,11151	0,00344	0,12066	1,00000	1,00000
Parallel	1	0,02496	0,10647	0,00337	0,11368	1,04498	1,04498
Parallel	2	0,01595	0,11056	0,00350	0,10807	1,63483	0,81742
Parallel	4	0,01589	0,13261	0,00322	0,11188	1,64077	0,41019
Parallel	8	0,01159	0,15045	0,00408	0,10714	2,24967	0,28121
Parallel	16	0,01427	0,13609	0,00535	0,10688	1,82760	0,11422



Case Study n.2 – HM Counting Sort

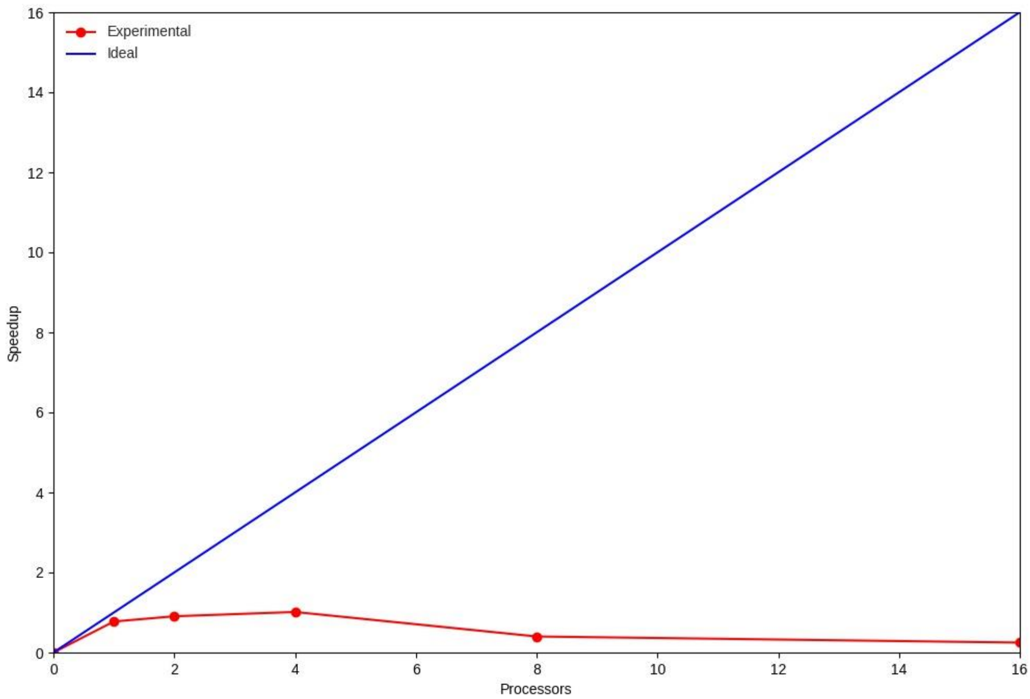
In this case study the main purpose is to analyse the performances of a second version of the counting sort algorithm, that is explained here. The build setup is:

Both the sequential and the parallel programs are compiled with the gcc optimization -Ox, where x = 0, 1, 2, 3 (O0 means without any optimization).

We want to highlight the difference between a sequential program compared to a parallel one, compiled with the same optimizations. The case study is done on multiple size: 10000, 100000, 1000000, and with different number of threads (0, 1, 2, 4, 8, 16).

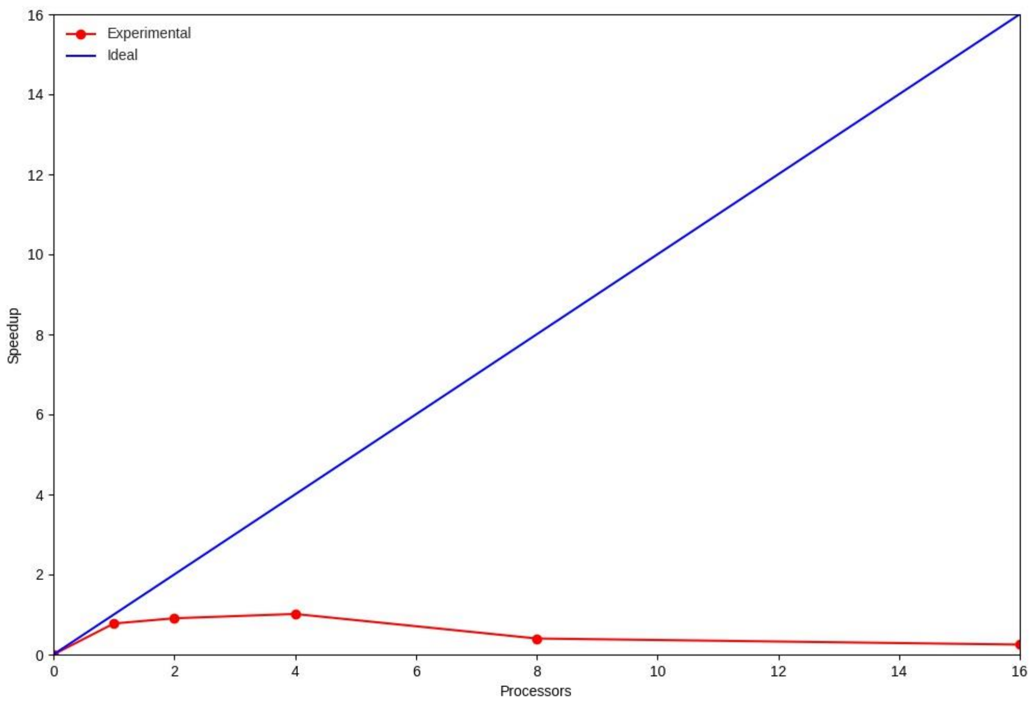
SIZE-10.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00119	0,00370	0,00000	0,00880	1,00000	1,00000
Parallel	1	0,00153	0,00475	0,00001	0,00976	0,77493	0,77493
Parallel	2	0,00131	0,00453	0,00001	0,00943	0,90592	0,45296
Parallel	4	0,00117	0,00436	0,00004	0,00942	1,01390	0,25348
Parallel	8	0,00297	0,01011	0,00013	0,01113	0,40091	0,05011
Parallel	16	0,00479	0,00354	0,01510	0,01347	0,24854	0,01553



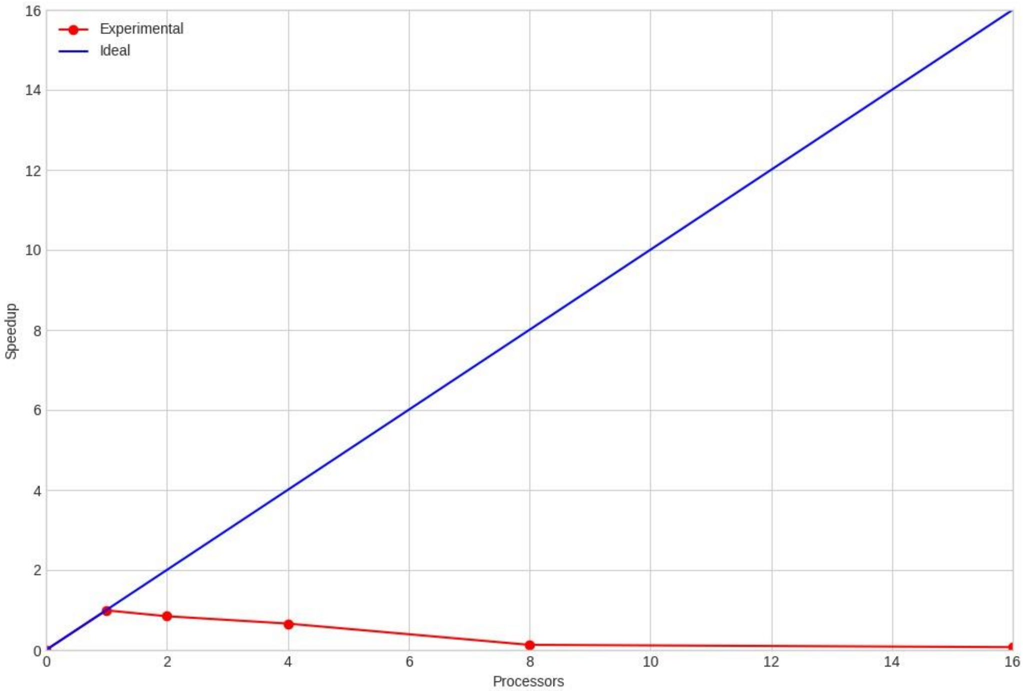
SIZE-10.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00078	0,00359	0,00001	0,00795	1,00000	1,00000
Parallel	1	0,00099	0,00417	0,00000	0,00935	0,78351	0,78351
Parallel	2	0,00090	0,00421	0,00002	0,00935	0,86639	0,43320
Parallel	4	0,00079	0,00407	0,00001	0,00880	0,97870	0,24468
Parallel	8	0,00267	0,00919	0,00026	0,01064	0,29124	0,03640
Parallel	16	0,00466	0,00266	0,01383	0,01347	0,16665	0,01042



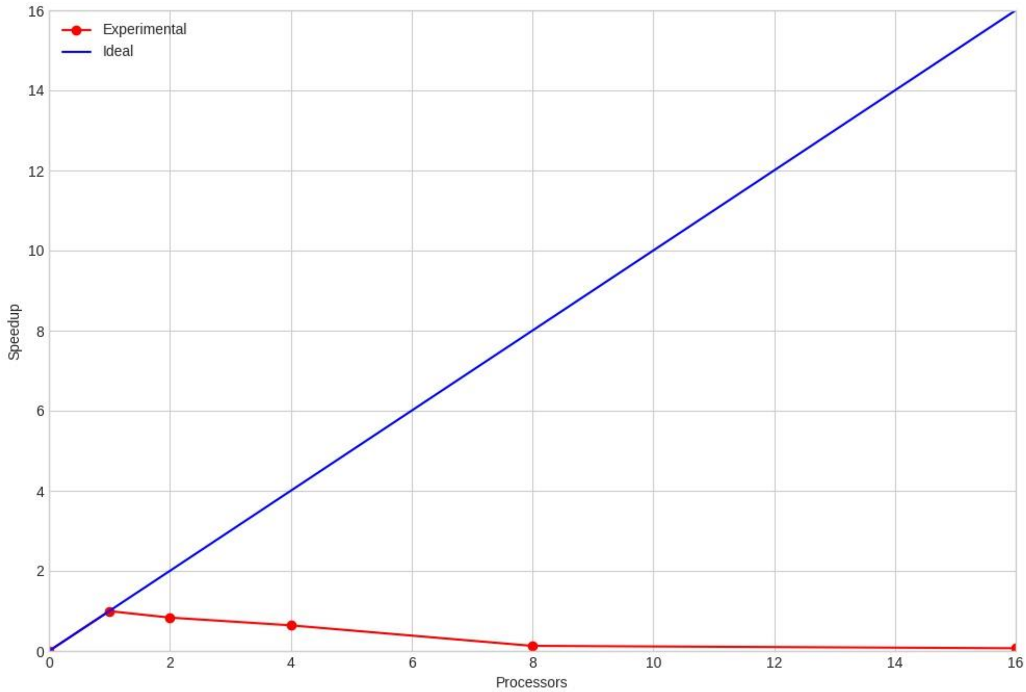
SIZE-10.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00033	0,00358	0,00000	0,00868	1,00000	1,00000
Parallel	1	0,00033	0,00339	0,00000	0,00831	0,99173	0,99173
Parallel	2	0,00039	0,00361	0,00000	0,00868	0,84450	0,42225
Parallel	4	0,00050	0,00372	0,00005	0,00882	0,66098	0,16525
Parallel	8	0,00255	0,00848	0,00019	0,01065	0,12963	0,01620
Parallel	16	0,00459	0,00176	0,01342	0,01343	0,07209	0,00451



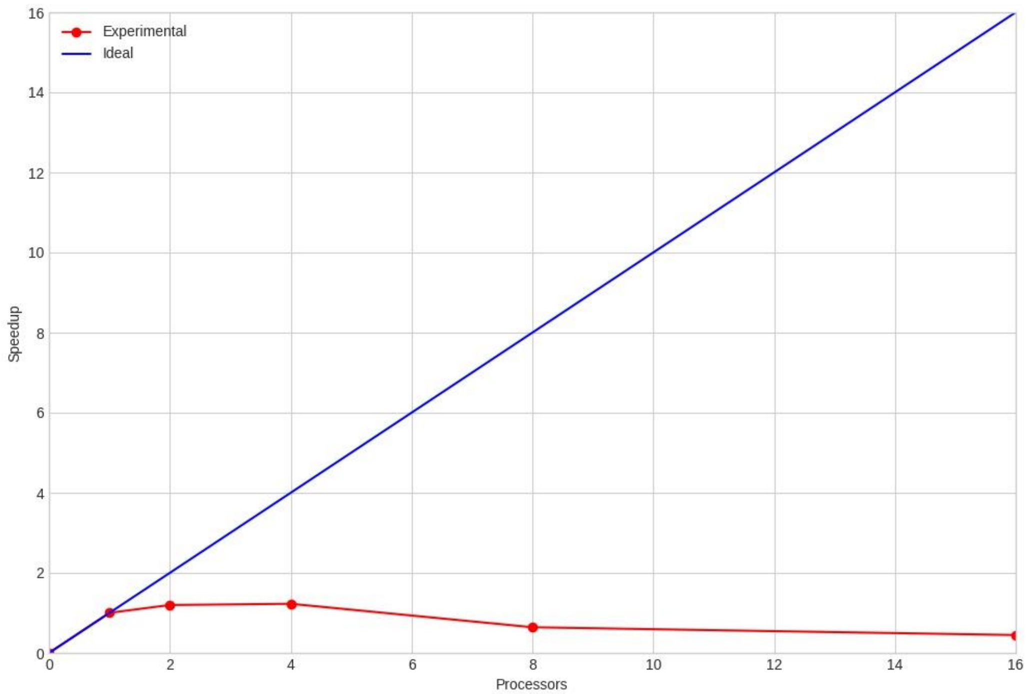
SIZE-10.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00032	0,00350	0,00002	0,00846	1,00000	1,00000
Parallel	1	0,00032	0,00354	0,00001	0,00867	0,99610	0,99610
Parallel	2	0,00038	0,00345	0,00000	0,00837	0,83457	0,41728
Parallel	4	0,00050	0,00369	0,00007	0,00839	0,64116	0,16029
Parallel	8	0,00248	0,00865	0,00024	0,01057	0,12884	0,01611
Parallel	16	0,00467	0,00046	0,01412	0,01336	0,06859	0,00429



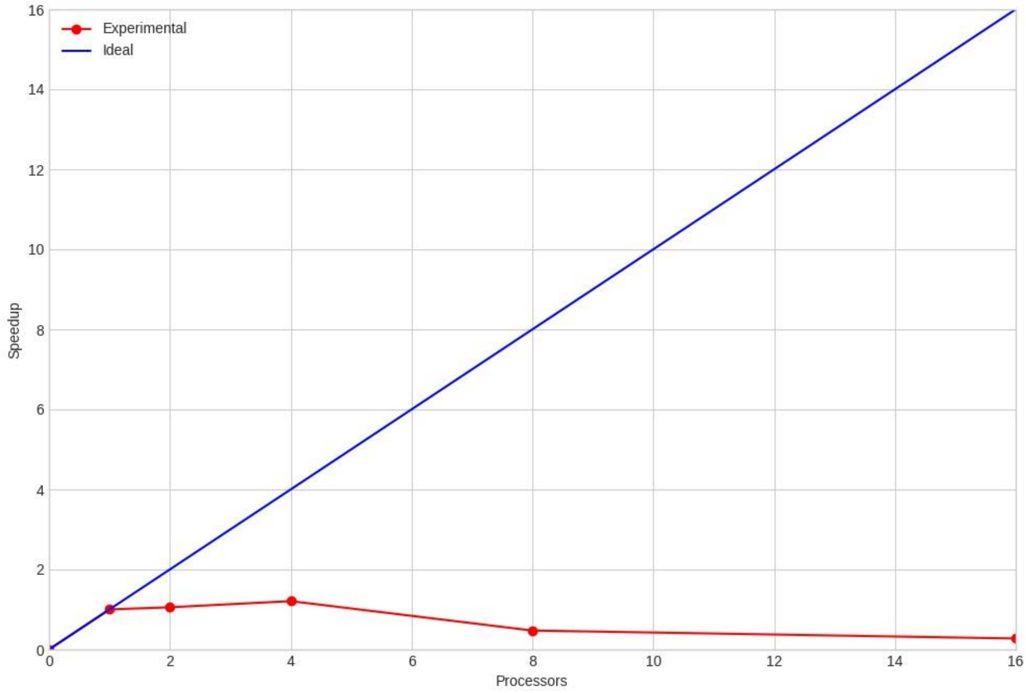
SIZE-100.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00250	0,00704	0,00009	0,01247	1,00000	1,00000
Parallel	1	0,00250	0,00693	0,00008	0,01243	1,00244	1,00244
Parallel	2	0,00209	0,00663	0,00002	0,01154	1,19776	0,59888
Parallel	4	0,00204	0,00643	0,00011	0,01152	1,22872	0,30718
Parallel	8	0,00389	0,01538	0,00041	0,01313	0,64271	0,08034
Parallel	16	0,00558	0,00228	0,01743	0,01588	0,44850	0,02803



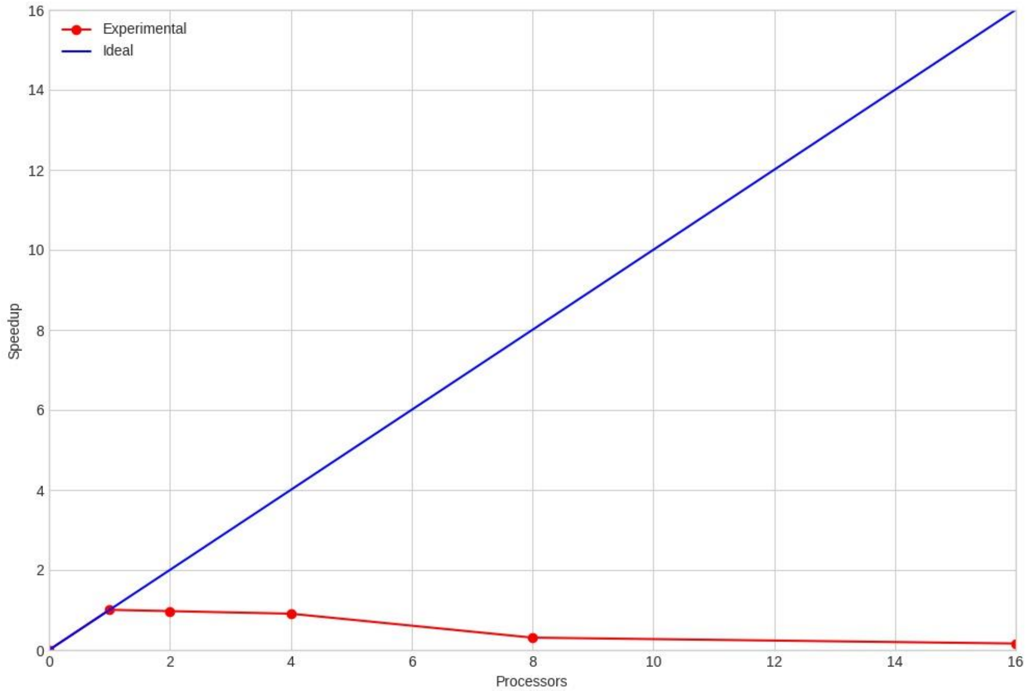
SIZE-100.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00142	0,00595	0,00002	0,01142	1,00000	1,00000
Parallel	1	0,00141	0,00587	0,00002	0,01145	1,00254	1,00254
Parallel	2	0,00134	0,00593	0,00007	0,01090	1,05495	0,52748
Parallel	4	0,00117	0,00551	0,00013	0,01058	1,20951	0,30238
Parallel	8	0,00299	0,01240	0,00053	0,01190	0,47450	0,05931
Parallel	16	0,00514	0,01455	0,01469	0,01549	0,27562	0,01723



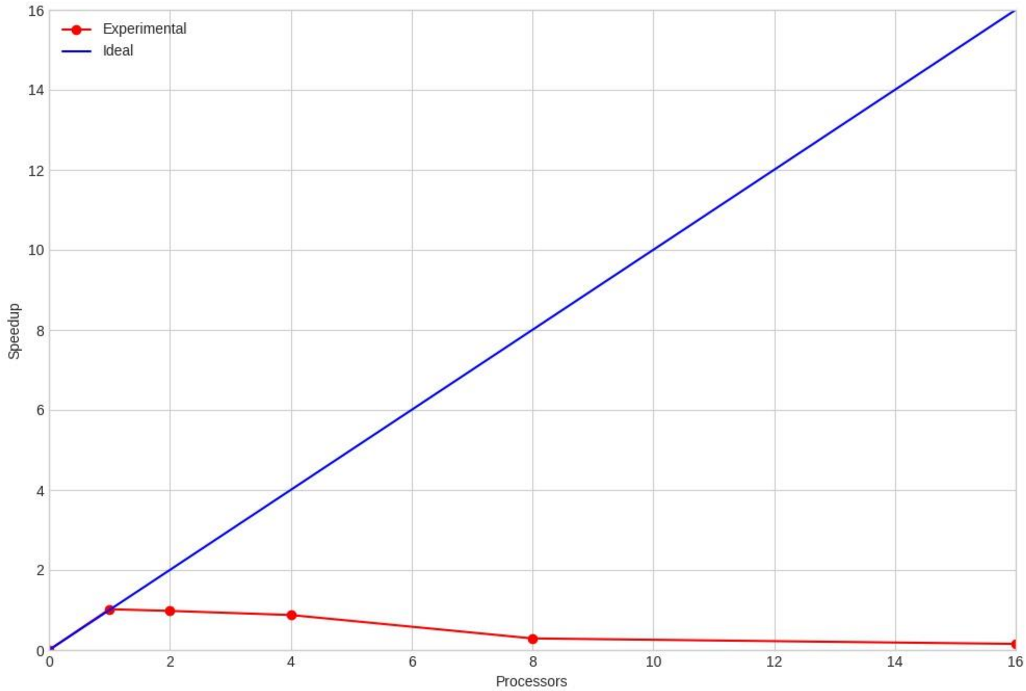
SIZE-100.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00084	0,00525	0,00004	0,01057	1,00000	1,00000
Parallel	1	0,00084	0,00523	0,00000	0,01050	1,00588	1,00588
Parallel	2	0,00087	0,00525	0,00002	0,01055	0,97271	0,48635
Parallel	4	0,00093	0,00517	0,00003	0,01049	0,90850	0,22713
Parallel	8	0,00272	0,01054	0,00056	0,01193	0,30979	0,03872
Parallel	16	0,00516	0,00071	0,01642	0,01494	0,16343	0,01021



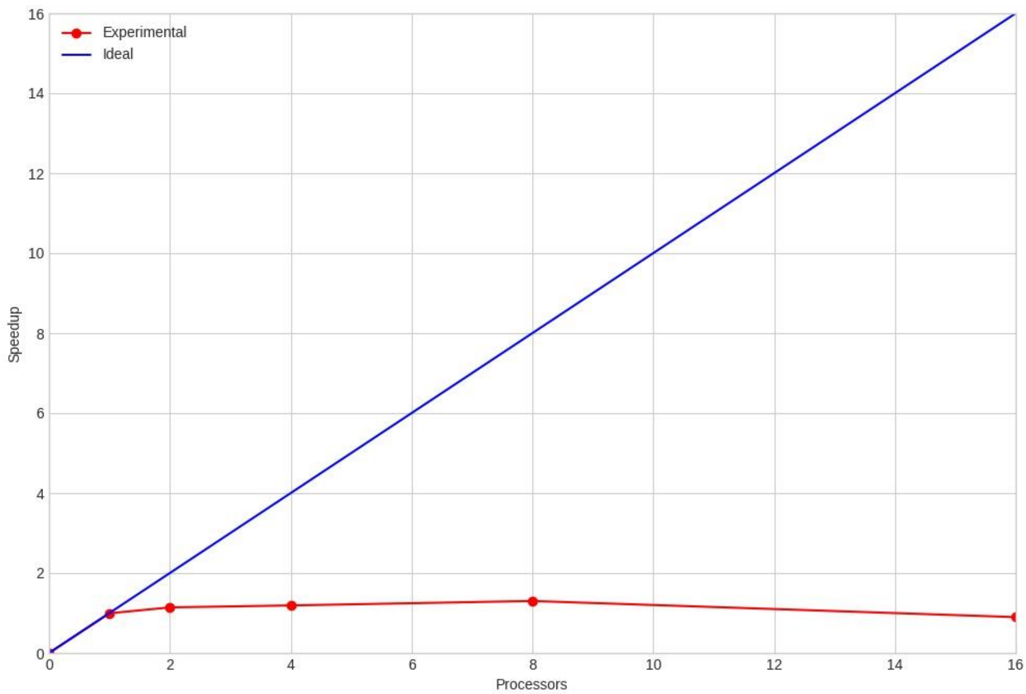
SIZE-100.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00079	0,00528	0,00000	0,01052	1,00000	1,00000
Parallel	1	0,00078	0,00521	0,00002	0,01053	1,02042	1,02042
Parallel	2	0,00081	0,00515	0,00000	0,01037	0,97857	0,48929
Parallel	4	0,00090	0,00538	0,00011	0,01139	0,87570	0,21892
Parallel	8	0,00272	0,01046	0,00049	0,01180	0,29058	0,03632
Parallel	16	0,00513	0,00079	0,01679	0,01547	0,15421	0,00964



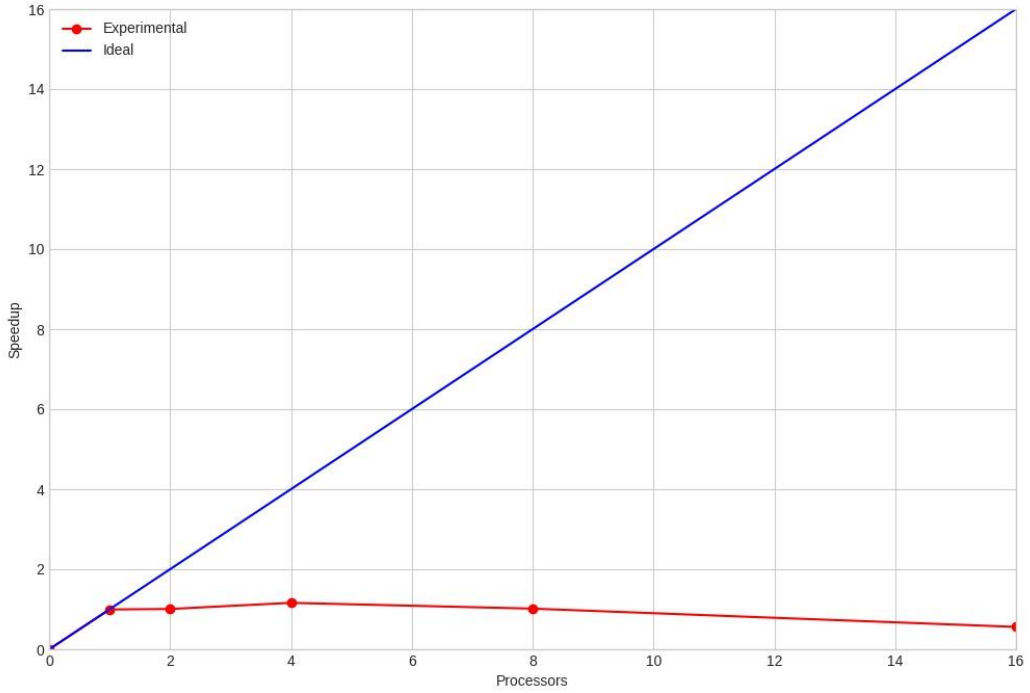
SIZE-1.000.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,01213	0,02516	0,00104	0,03141	1,00000	1,00000
Parallel	1	0,01220	0,02559	0,00054	0,03180	0,99418	0,99418
Parallel	2	0,01062	0,03090	0,00090	0,03205	1,14201	0,57101
Parallel	4	0,01019	0,03799	0,00178	0,03177	1,19100	0,29775
Parallel	8	0,00931	0,05061	0,00187	0,02986	1,30263	0,16283
Parallel	16	0,01350	0,03355	0,01705	0,03694	0,89884	0,05618



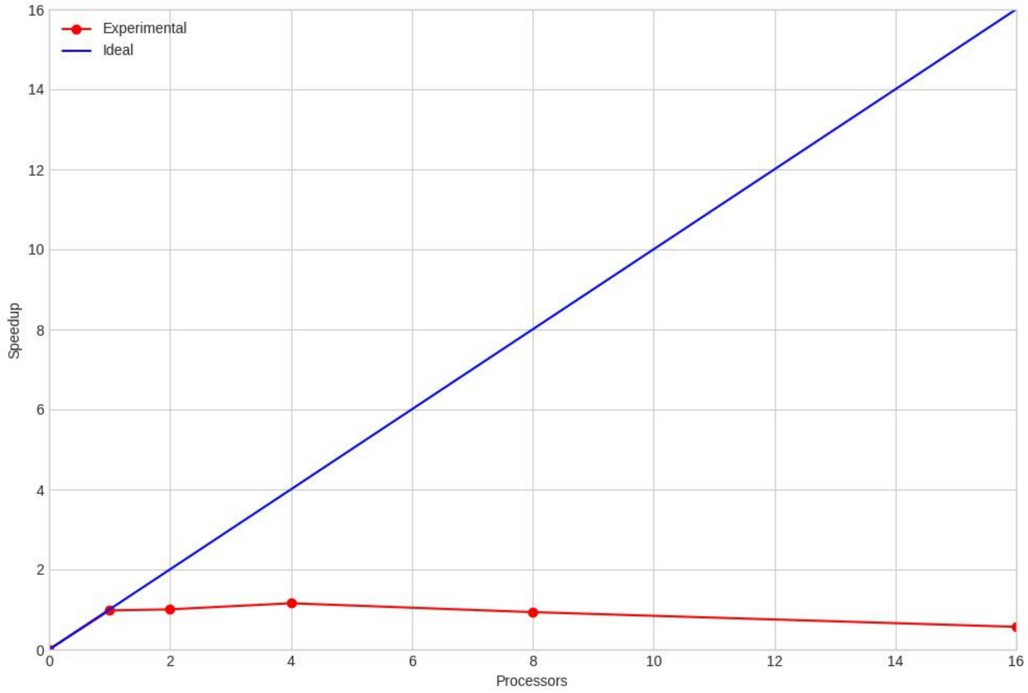
SIZE-1.000.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00572	0,01906	0,00022	0,02490	1,00000	1,00000
Parallel	1	0,00575	0,01853	0,00040	0,02501	0,99562	0,99562
Parallel	2	0,00568	0,02443	0,00058	0,02753	1,00711	0,50355
Parallel	4	0,00493	0,02796	0,00099	0,02547	1,16041	0,29010
Parallel	8	0,00563	0,04015	0,00338	0,02630	1,01613	0,12702
Parallel	16	0,01023	0,02376	0,01762	0,03490	0,55942	0,03496



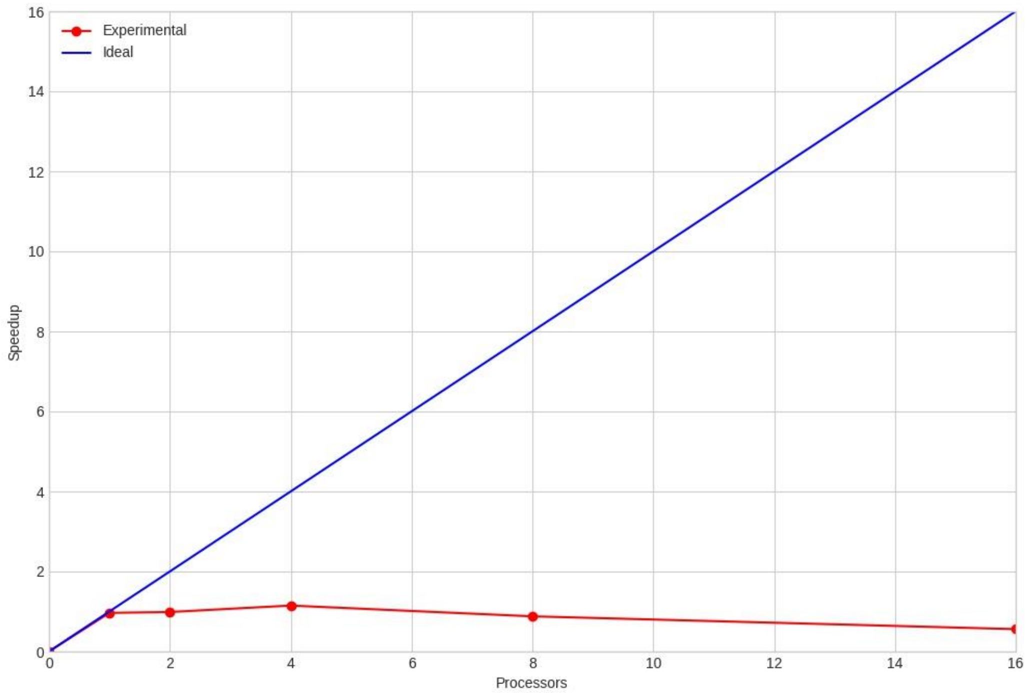
SIZE-1.000.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00566	0,01845	0,00036	0,02476	1,00000	1,00000
Parallel	1	0,00579	0,01870	0,00126	0,02539	0,97744	0,97744
Parallel	2	0,00563	0,02377	0,00139	0,02793	1,00416	0,50208
Parallel	4	0,00489	0,02701	0,00091	0,02485	1,15596	0,28899
Parallel	8	0,00605	0,04215	0,00309	0,02716	0,93502	0,11688
Parallel	16	0,00997	0,02356	0,01725	0,03323	0,56722	0,03545



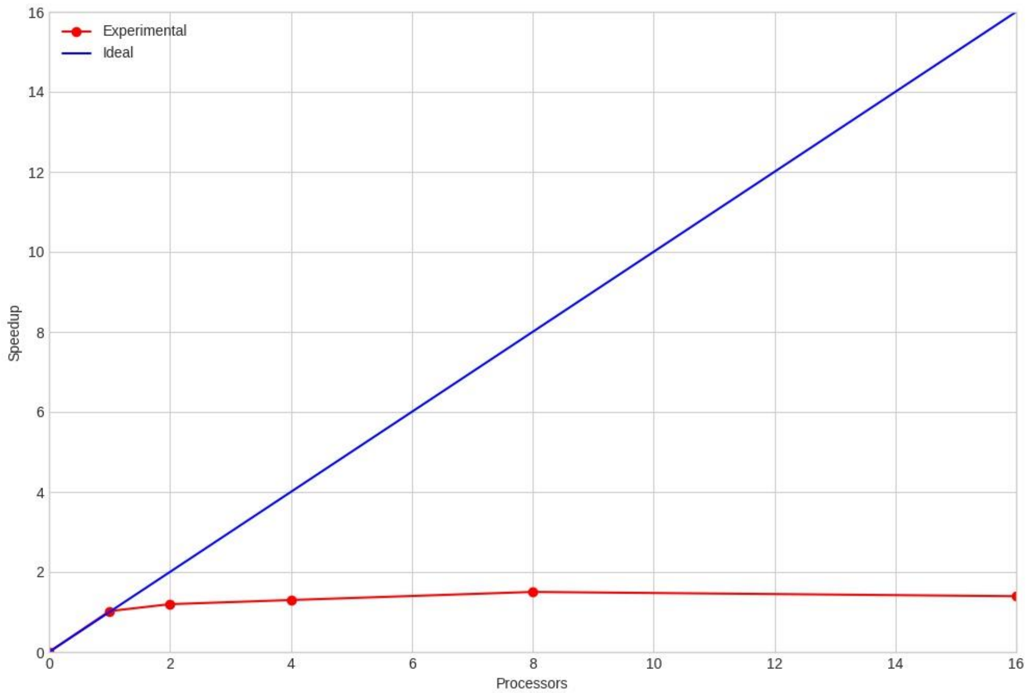
SIZE-1.000.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,00515	0,01787	0,00072	0,02465	1,00000	1,00000
Parallel	1	0,00533	0,01918	0,00014	0,02536	0,96654	0,96654
Parallel	2	0,00521	0,02309	0,00090	0,02707	0,98786	0,49393
Parallel	4	0,00448	0,02412	0,00165	0,02413	1,14855	0,28714
Parallel	8	0,00585	0,04410	0,00189	0,02780	0,87997	0,11000
Parallel	16	0,00920	0,02296	0,01517	0,03084	0,55982	0,03499



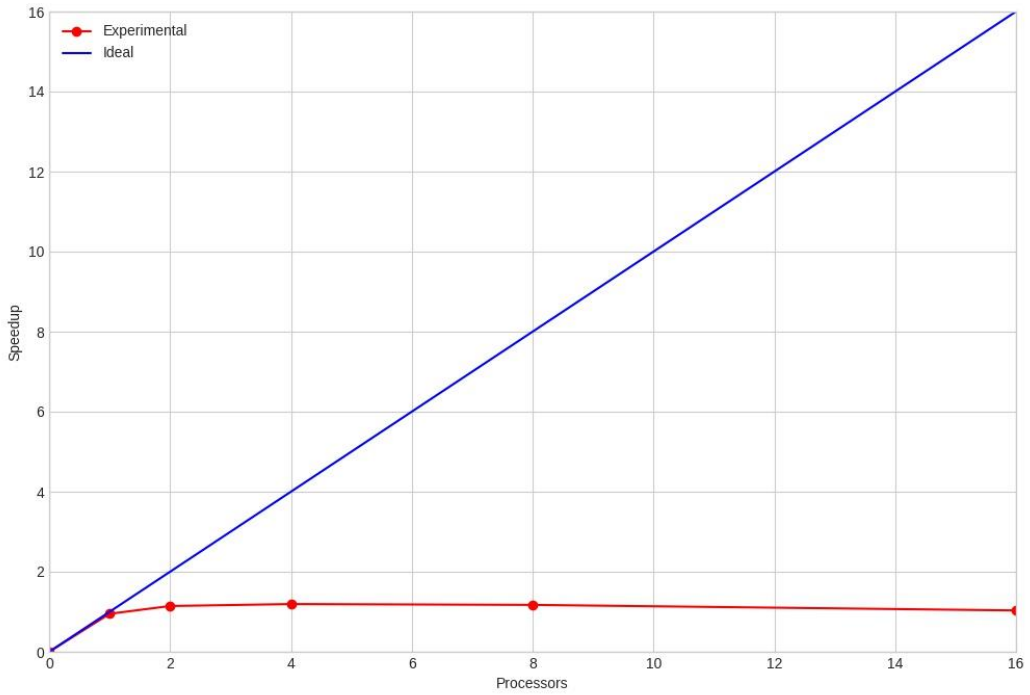
SIZE-10.000.000-OPT-O0

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,09247	0,17293	0,00555	0,18504	1,00000	1,00000
Parallel	1	0,09045	0,16853	0,01003	0,18160	1,02223	1,02223
Parallel	2	0,07732	0,19096	0,01097	0,17798	1,19588	0,59794
Parallel	4	0,07116	0,20828	0,00723	0,16966	1,29934	0,32483
Parallel	8	0,06156	0,23258	0,01151	0,15774	1,50198	0,18775
Parallel	16	0,06627	0,21713	0,01747	0,16401	1,39530	0,08721



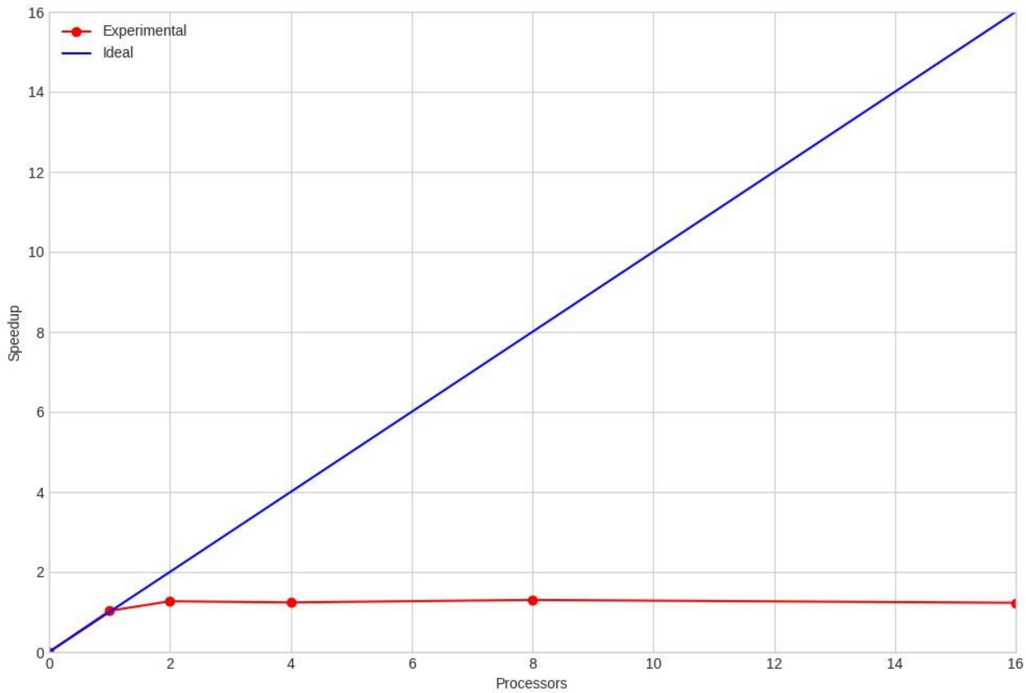
SIZE-10.000.000-OPT-O1

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,04037	0,11815	0,01005	0,12957	1,00000	1,00000
Parallel	1	0,04251	0,12449	0,01016	0,13681	0,94970	0,94970
Parallel	2	0,03530	0,12263	0,01025	0,12692	1,14381	0,57190
Parallel	4	0,03385	0,13485	0,01025	0,12453	1,19261	0,29815
Parallel	8	0,03445	0,16679	0,00792	0,13020	1,17184	0,14648
Parallel	16	0,03907	0,16060	0,01594	0,13825	1,03336	0,06459



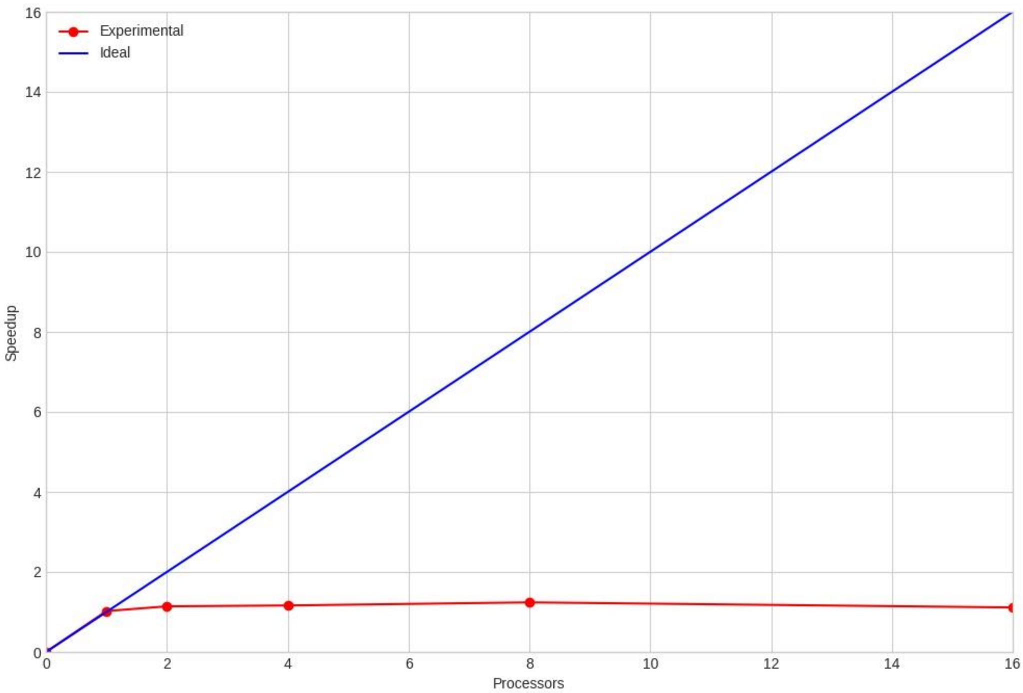
SIZE-10.000.000-OPT-O2

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,04665	0,12401	0,01003	0,13760	1,00000	1,00000
Parallel	1	0,04535	0,12104	0,01014	0,13394	1,02880	1,02880
Parallel	2	0,03677	0,12357	0,01001	0,12563	1,26889	0,63445
Parallel	4	0,03764	0,14618	0,01127	0,13149	1,23950	0,30988
Parallel	8	0,03584	0,17515	0,01298	0,13411	1,30162	0,16270
Parallel	16	0,03791	0,15725	0,01467	0,12890	1,23057	0,07691



SIZE-10.000.000-OPT-O3

Version	Threads	TimeInt	User	Sys	Real	Speedup	Efficiency
Serial	1	0,04151	0,11870	0,01100	0,13231	1,00000	1,00000
Parallel	1	0,04056	0,11473	0,00996	0,12858	1,02331	1,02331
Parallel	2	0,03636	0,12723	0,00989	0,13132	1,14141	0,57071
Parallel	4	0,03568	0,14179	0,01255	0,13234	1,16325	0,29081
Parallel	8	0,03345	0,16193	0,00781	0,12437	1,24099	0,15512
Parallel	16	0,03726	0,15115	0,01610	0,12884	1,11404	0,06963



Considerations

Case study n.1 – WIKI Counting Sort

It is possible to observe that for a small load[1] (10.000), overhead introduced by OMP determines worse performances than the sequential version ones, even when optimizations[2] are introduced.

For a mid-load (100.000), parallel version performances keep being below sequential version ones. Differently from what happens in previous case, there isn't an immediate collapse of real speedup: the curve remains stable for a number of threads between 1 and 4.

Starting from a large load (1.000.000), performances improvement is visible. The reason is that parallelization's benefits overcome OMP overhead and produce a higher speedup compared to previous cases. All measures on a load of 1.000.000 present a phenomenon well known in parallel computing field: Superlinear Speedup.

The real speedup exceeds the theoretical one due to cache effect. By increasing elaboration units number, data access time dramatically reduces because a part of data has already been stored in the cache.

This produces an extra speedup in addition to the one determined by parallelization, reaching a peak of about 2.45 with 2 threads and O3 optimization in the case taken in consideration. In the latter, superlinear speedup occurs only when the number of threads is 1 or 2, since the increasing of threads number moves away real speedup from theoretical one. When threads number exceeds 2, cache effect is not able to keep the real curve over the theoretical one anymore. Without optimizations, as expected, the maximum speedup is obtained with 8 threads, that corresponds to the effective number of threads that characterizes the machine used for measures. In presence of optimizations, best performances are reached with 4 threads, in fact a speedup decrease is produced starting from 8 threads (and not 16 threads) in opposition to what expected. Moreover, it is possible to observe a more and more marked speedup reduction because the threads number is higher than the machine's ones.

For a very large load (10.000.000), performances increase in most of the cases and are better than the ones obtained in previous cases.

In conclusion, the best speedup trend occurs when load is 10.000.000 with O3 optimization (with a peak for 8 threads as expected). However, major speedup values are obtained when load is 1.000.000 with O2 and O3 optimizations due to superlinear speedup previously analysed.

[1] **Load**: input data quantity (number of elements of the array to sort).

[2] **Optimization**: performance improvements realized by compiler, e.g., based on detection of lack of dependences among data and/or indices used in successive iterations of a loop. Compilers can manage different kinds of optimizations. In particular, GCC is characterized by 4 levels of optimizations named Ox with x = {0, 1, 2, 3} where O0 means that there are no optimizations. The joint use of GCC optimizations and OMP could produce different results from the expected ones, e.g., O2 could be better than O3 since the latter is more hardware oriented.

Case study n.2 – HM Counting Sort

It is possible to observe that for a small load* (10.000), overhead introduced by OMP determines worse performances than the sequential version ones.

For a mid-load (100.000), parallel version performances are on average similar to sequential version ones. In particular, it is observable a decrease of performances with the introduction of O2 and O3 optimizations.

For a large load (1.000.000), although speedup values are similar to sequential version ones, it is possible to identify a light performances improvement.

For a very large load (10.000.000), performances are always better than sequential version ones, reaching a speedup peak value of about 1.5 with 8 threads and without optimizations.

In conclusion, the best speedup trend occurs when load is 10.000.000 without optimization (with a peak for 8 threads as expected).

Comparison between WIKI & HM versions

As regards speedup performances, HM version is better than WIKI version when the load is low, while for higher loads WIKI is better than HM.

As regards time execution, WIKI is faster than HM. In particular, WIKI is on average about 1.7 faster than HM:

- WIKI elapses an average of $\simeq$ 0.009s
- HM elapses an average of $\simeq$ 0.015s

Finally, speed difference increases when load and optimizations rise.

API - WIKI Counting Sort

Public Docs also available here

Public functions

Type	Name
void	**countingSort**(int *a, int N) *This function sorts the array 'a' using Counting Sort Algorithm.*
void	**gen_rand**(int *a, int N) *This function generates 'N' random integers and insert them into the array 'a'.*

Public functions documentation

function countingSort

```
void countingSort(
        int *a,
        int N
)
```

Parameters:

- a pointer to the unsorted array.
- N number of array elements.

function gen_rand

```
void gen_rand(
        int *a,
        int N
)
```

Parameters:

- a pointer to the array where to insert elements.
- N number of random integers that have to be generated.

API - HM Counting Sort

Public Docs also available here

Public functions

Type	Name
void	**countingSort**(int *a, int *c, int N) This function sorts the array 'a' using Counting Sort Algorithm and returns the corresponding sorted array 'c'
void	**gen_rand**(int *a, int N) *This function generates 'N' random integers and insert them into the array 'a'.*

Public functions documentation

function countingSort

```
void countingSort(
        int *a,
        int *c,
        int N
)
```

Parameters:

- a pointer to the unsorted array.
- c pointer to the sorted array.
- N number of array elements.

function gen_rand

```
void gen_rand(
        int *a,
        int N
)
```

Parameters:

- a pointer to the array where to insert elements.
- N number of random integers that have to be generated.

How to run

Since 2 versions of Counting Sort are provided, you must use command `cd WIKI_Counting_Sort` or `cd HM_Counting_Sort` in order to choose the version to run and:

1. Create a build directory and launch cmake
   `mkdir build`
   `cd build`
   `cmake ..`
2. Generate executables with `make`
3. To generate measures, run make `generate_measures`
   Attention: it takes a lot of time. This is the reason why our measures are already included, so you should skip this step.
4. To extract mean times and speedup curves from them run `make extract_measures`

Results can be found in the `measure/YYYY-MM-DD.hh:mm:ss` directory, divided by problem size and the gcc optimization option used.