OPTIMIZED MATRIX MULTIPLICATION AND INVERSE

Matrix inverse

Inverse using LU factorization:

$$A * X = I$$

$$\downarrow$$

Pivoting

Rearrangement of rows

$$P * A * X = P * I$$

$$\downarrow$$

GEM

Computation of L and U

$$L * U * X = P$$

$$\downarrow$$

Forward substitution

Resolution of lower triangular systems

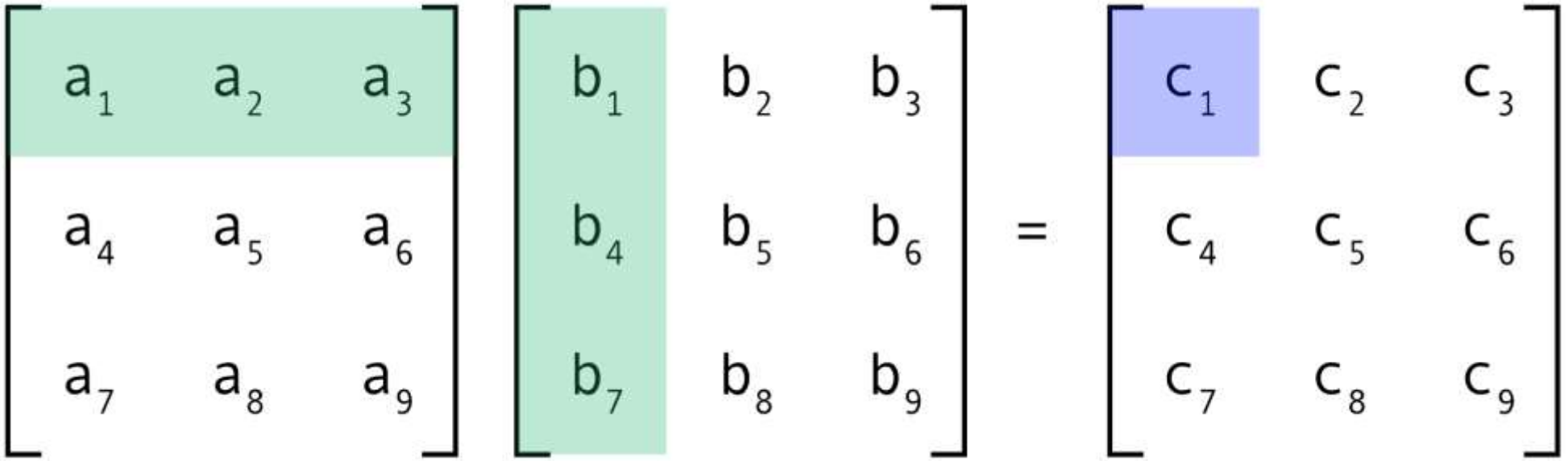
$$L * Y = P$$

$$\downarrow$$

Backward substitution

Resolution of upper triangular systems

$$U * X = Y$$



Matrix multiplication

Performance analysis

- AMDAHL'S LAW (via intel profiler)

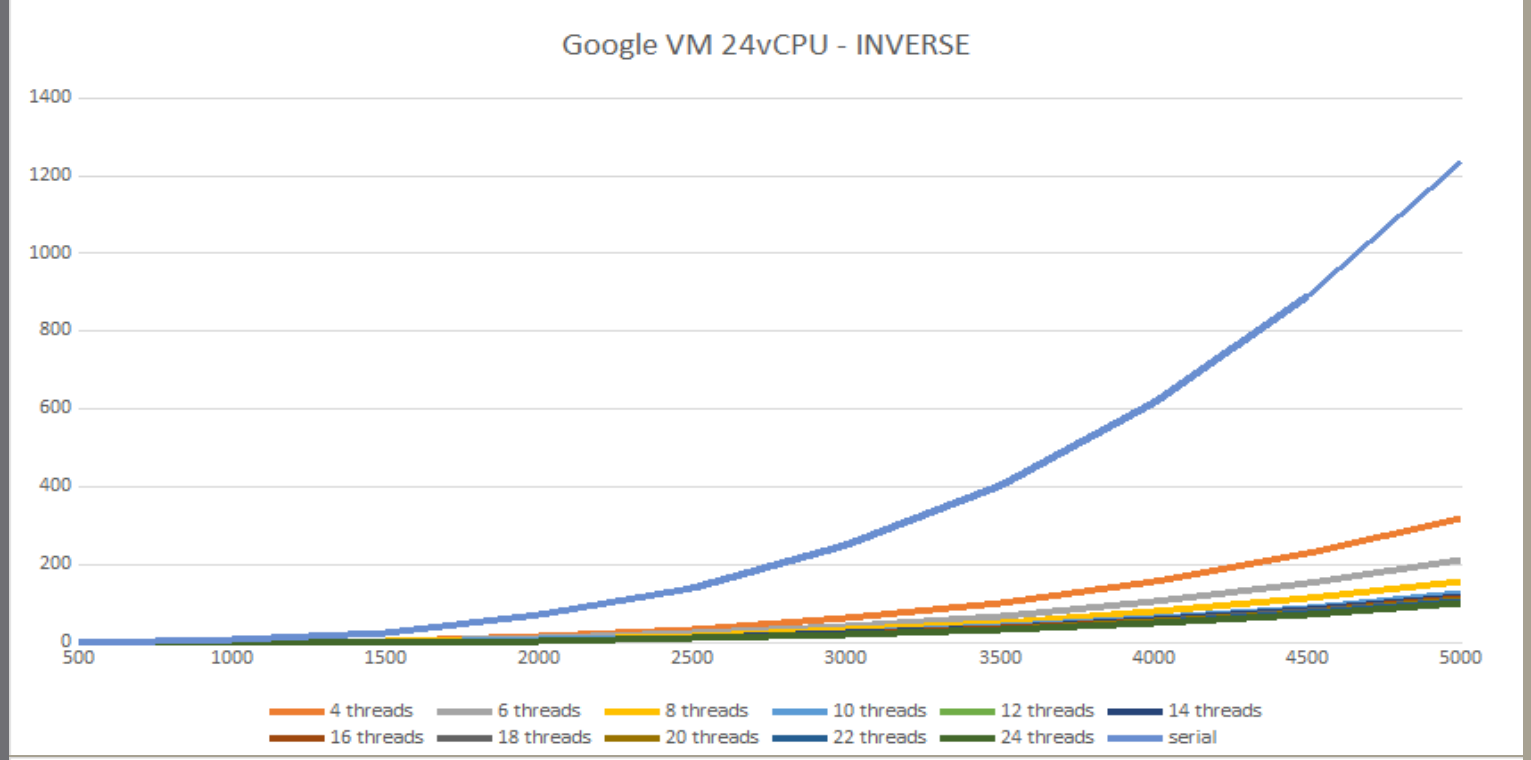
a. Speedup = 1/ ((1-P) + P/S)

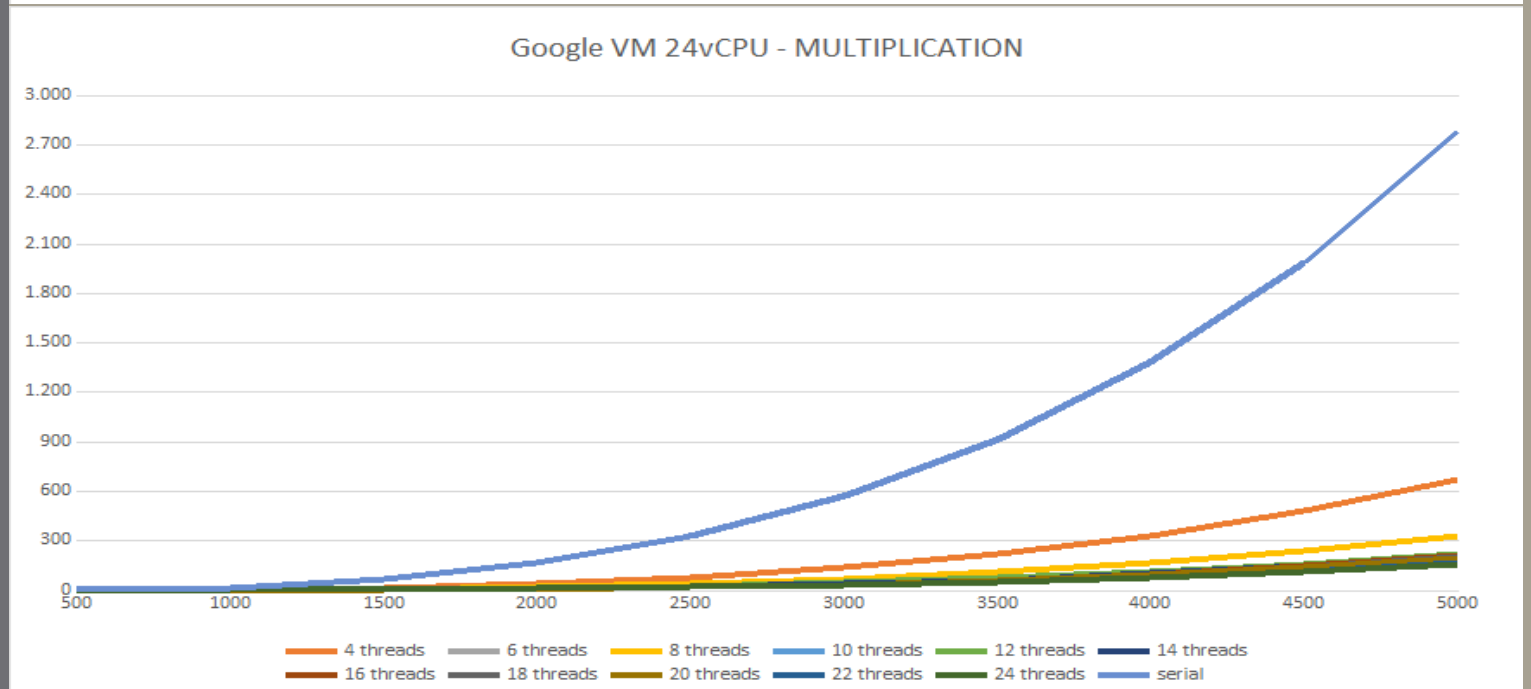
b. Speed up is 7.9

- Testing done on PC and multiple google cloud virtual machines

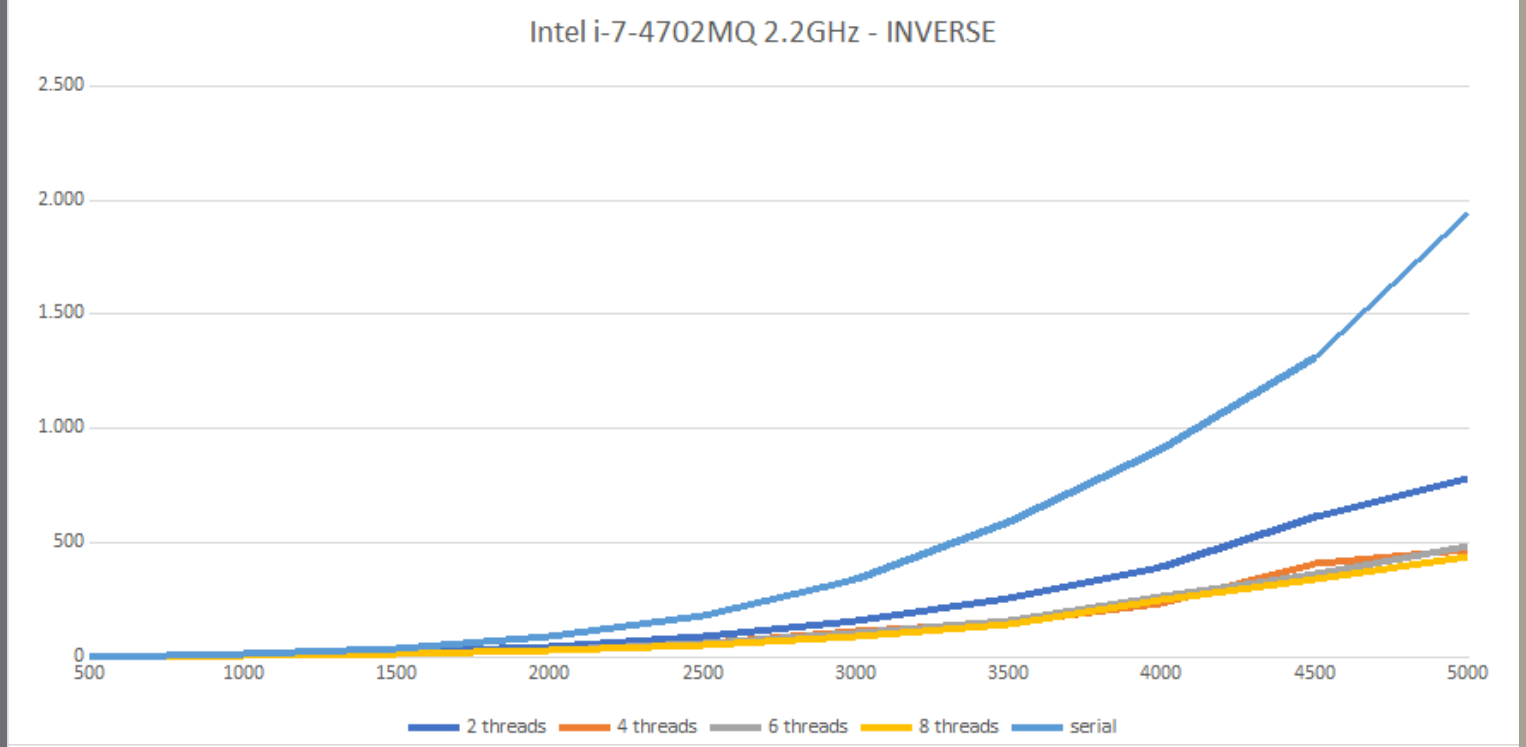
1. PC – intel i7 8 CPUs 2.4GHz
2. Google Virtual Machine – n1-highcpu-8 (8 vCPU, 7.2 GB of memory)
3. Google Virtual Machine – VM 24vCPU, 156GB of memory

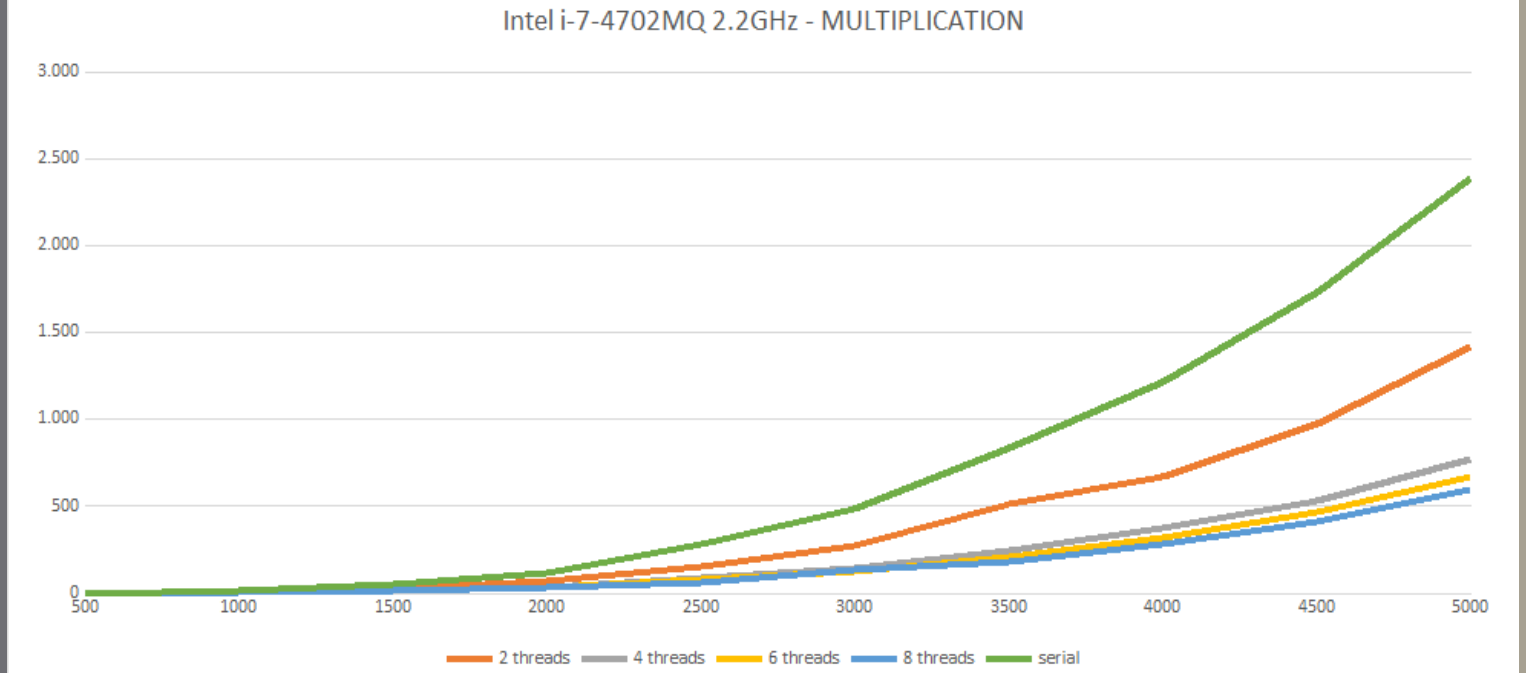
Incl.	Self	Called	Function	Location
37.35	37.35	1	backwardsub	a.out
37.29	37.29	1	forwardsub	a.out
24.91	24.91	999	gem	a.out
0.21	0.21	2 476 041	swap	a.out
0.06	0.06	2	make_diag_matrix	a.out
0.05	0.05	1 000 000	random_r	libc-2.30.so: random_r.c
25.17	0.05	1	pivoting	a.out
0.13	0.04	1	prandom	a.out
0.08	0.03	1 000 000	random	libc-2.30.so: random.c, lowlevellock.h
0.09	0.01	1 000 000	rand	libc-2.30.so: rand.c
0.09	0.00	1 000 000	0x0000000000109110	(unknown)
0.00	0.00	3 005	_int_malloc	libc-2.30.so: malloc.c
0.00	0.00	3 019	_int_free	libc-2.30.so: malloc.c
0.00	0.00	3 004	malloc	libc-2.30.so: malloc.c
0.00	0.00	2 851	systrim.isra.0.constprop.0	libc-2.30.so: malloc.c
0.00	0.00	1	_dl_addr	libc-2.30.so: dl-addr.c
0.00	0.00	3 002	unlink_chunk.isra.0	libc-2.30.so: malloc.c
0.00	0.00	3 003	free	libc-2.30.so: malloc.c
0.00	0.00	2 929	sbrk	libc-2.30.so: sbrk.c
0.00	0.00	3	allocate_matrix	a.out
0.00	0.00	3	release_mem	a.out
0.00	0.00	4	_dl_relocate_object	ld-2.30.so: dl-reloc.c, dl-machine.h, do-rel.h, ldsodefs.h
0.00	0.00	1	__GI___tunables_init	ld-2.30.so: dl-tunables.c, dl-tunables.h
0.00	0.00	98	do_lookup_x	ld-2.30.so: dl-lookup.c, ldsodefs.h
0.00	0.00	2 929	__default_morecore	libc-2.30.so: morecore.c
0.00	0.00	98	_dl_lookup_symbol_x	ld-2.30.so: dl-lookup.c
0.00	0.00	77	sysmalloc	libc-2.30.so: malloc.c, arena.c
0.00	0.00	3 003	0x00000000001090b0	(unknown)
0.00	0.00	3 003	0x0000000000109100	(unknown)
0.00	0.00	90	check_match	ld-2.30.so: dl-lookup.c
0.00	0.00	152	strcmp	ld-2.30.so: strcmp.S
0.00	0.00	2	_dl_map_object_from_fd	ld-2.30.so: dl-load.c, dl-fileid.h, dl-map-segments.h, dl-load.h, get-dynamic-info.h
0.00	0.00	4	_dl_check_map_versions	ld-2.30.so: dl-version.c
0.00	0.00	1	dl_main	ld-2.30.so: rtld.c, dl-prop.h, get-dynamic-info.h, setup-vdso.h, dl-osinfo.h
0.00	0.00	6	intel_check_word.isra.0	libc-2.30.so: cacheinfo.c, stdlib-bsearch.h
0.00	0.00	1	_dl_start	ld-2.30.so: rtld.c, dl-machine.h, get-dynamic-info.h, do-rel.h
0.00	0.00	1	__printf_fp_l	libc-2.30.so: printf_fp.c, localeinfo.h, allocalim.h, get-rounding-mode.h, rounding-mode.h
0.00	0.00	1	_dl_map_object_deps	ld-2.30.so: dl-deps.c, scratch_buffer.h
0.00	0.00	78	brk	libc-2.30.so: brk.c
0.00	0.00	1	ptmalloc_init.part.0	libc-2.30.so: arena.c, malloc.c
0.00	0.00	23	_IO_file_overflow@@GLIBC_2.2.5	libc-2.30.so: fileops.c
0.00	0.00	9	open_verify.constprop.0	ld-2.30.so: dl-load.c
0.00	0.00	1	_dl_sysdep_start	ld-2.30.so: dl-sysdep.c, dl-sysdep.c, cpu-features.c, cpu-features.c
0.00	0.00	2	__vfprintf_internal	libc-2.30.so: vfprintf-internal.c, printf-parse.h, libioP.h, lowlevellock.h
0.00	0.00	3	_dl_new_object	ld-2.30.so: dl-object.c
0.00	0.00	1	_dl_important_hwcaps	ld-2.30.so: dl-hwcaps.c, dl-hwcap.h
0.00	0.00	8	_dl_cache_libcmp	ld-2.30.so: dl-cache.c
0.00	0.00	1	open_path	ld-2.30.so: dl-load.c
0.00	0.00	22	malloc	ld-2.30.so: dl-minimal.c
0.00	0.00	13	memset	ld-2.30.so: memset-vec-unaligned-erms.S



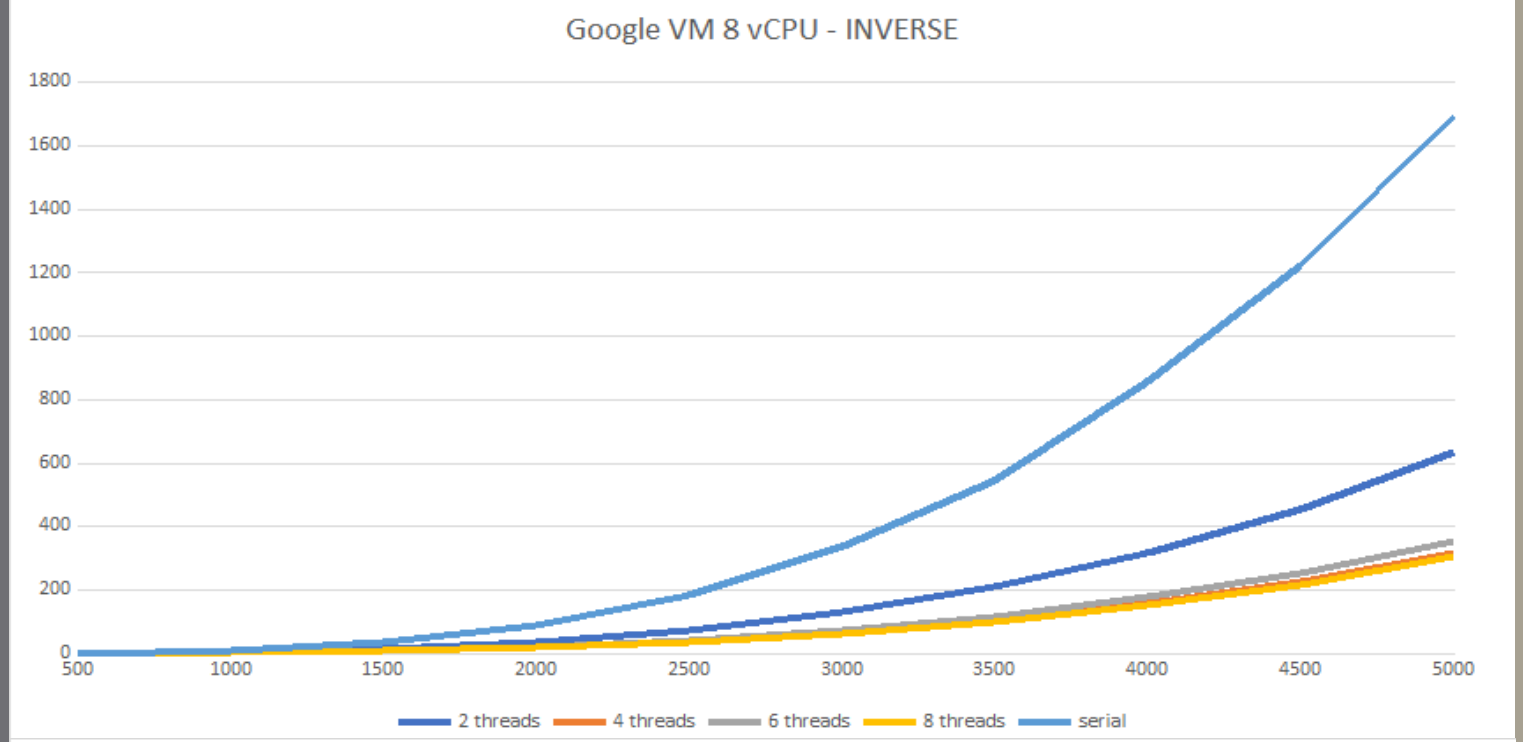


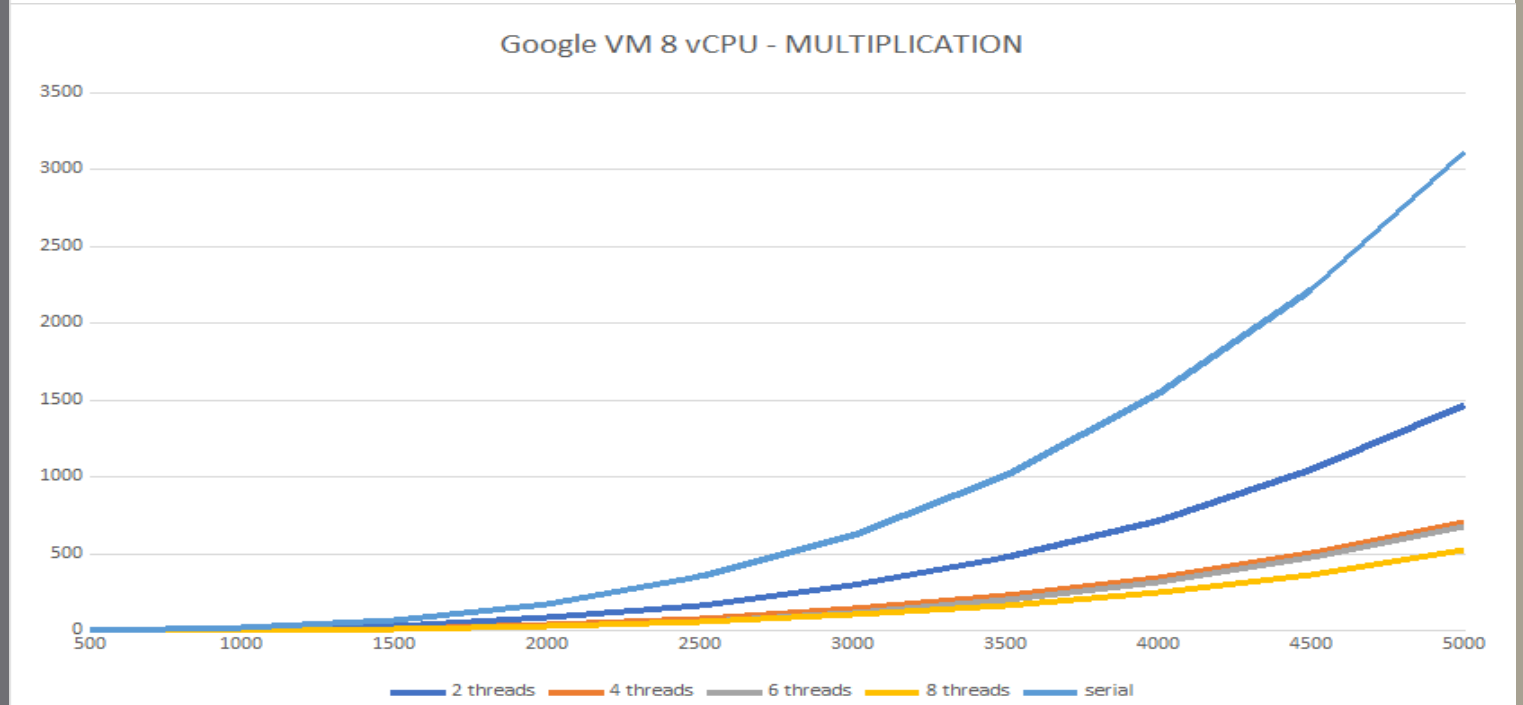
Google VM 24 CPUs





PC with 8 CPUs



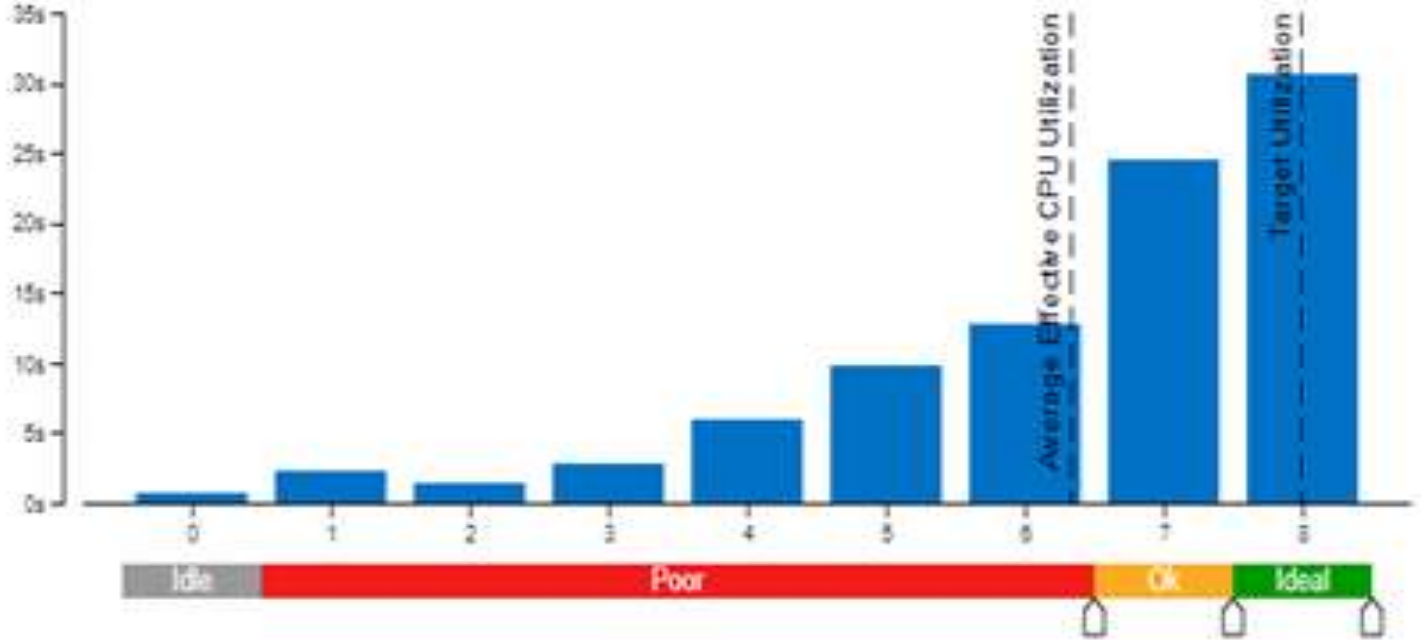


Google VM 8 CPUs

Effective CPU Utilization : 79.2% (6.332 out of 8 logical CPUs)

Effective CPU Utilization Histogram

This histogram displays a percentage of the wall time the specific number of CPUs were running simultaneously. Spin and Overhead time adds to the Idle CPU utilization value.



Results with profiler

THE END

Alessio Gullotti

and

Marko Pandža