A. Using O to compile the matvec.c

Table 1) time spent to compute matvec program

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.01	0.01
2000	0.02	0.09
5000	0.16	0.8
7500	0.34	1.15
10000	0.6	2.7
15000	1.44	8.13
20000	2.88	15.87
30000	6.44	38.07

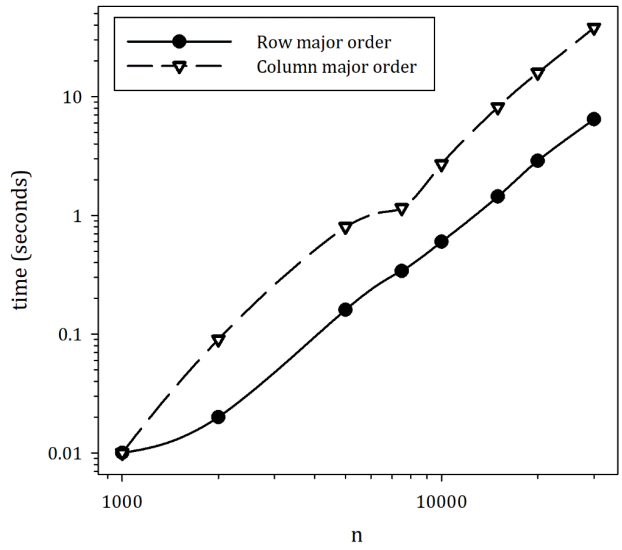

Figure 1) comparison of speed between row-major order with column-major order

Table 1 and Figure 1 indicate that in C the row-major order is much more preferable.

B. Using O1 to compile the matvec.c

Table 2) time spent to compute matvec program based on O1 compilation

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.00	0.00
2000	0.01	0.06
5000	0.03	0.42
7500	0.07	0.93
10000	0.12	1.85
15000	0.27	5.99
20000	0.48	13.11
30000	2.08	31.96

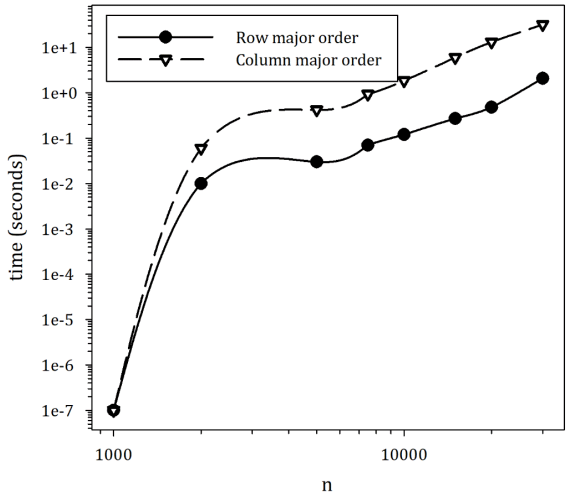

Figure 2) comparison of speed between row-major order with column-major order

Table 2 and Figure 2 indicate that while row major order should be taken into account the speed has drastically increased by O1 especially for row major order.

C. Using O2 to compile the matvec.c

Table 3) time spent to compute matvec program based on O2 compilation

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.00	0.00
2000	0.01	0.06
5000	0.03	0.41
7500	0.07	0.94
10000	0.12	1.79
15000	0.28	5.94
20000	0.5	12.79
30000	2.15	32.16

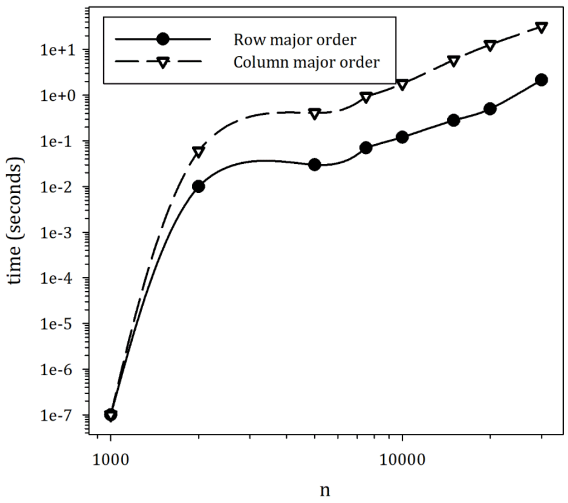

Figure 3) comparison of speed between row-major order with column-major order

Table 3 and Figure 3 indicate that while row major order should be taken into account the speed is almost identical to that of O1 where in some cases O1 has higher speed whereas, in others O2 has better results.

D. Using O3 to compile the matvec.c

Table 4) time spent to compute matvec program based on O3 compilation

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.00	0.00
2000	0.00	0.04
5000	0.04	0.21
7500	0.08	0.47
10000	0.13	0.92
15000	0.29	3.06
20000	0.53	6.71
30000	2.06	16.79

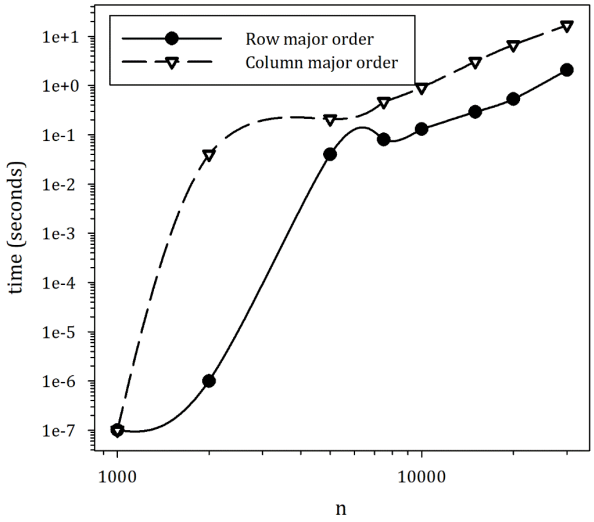

Figure 4) comparison of speed between row-major order with column-major order

Table 4 and Figure 4 indicate that while row major order should be taken into account and O3 speed is much higher than previous. Although it enables yet more optimizations. This includes optimizations that incur a space-time tradeoff in favor of time, such as loop unrolling. Your binary will almost certainly be larger and program is faster.

E. Using Os to compile the matvec.c

Table 5) time spent to compute matvec program based on Os compilation

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.00	0.00
2000	0.01	0.07
5000	0.03	0.43
7500	0.07	0.97
10000	0.12	1.99
15000	0.29	6.13
20000	0.5	14.38
30000	2.38	37.69

Table 5 indicate that while row major order should be taken into account the speed is almost identical to that of O2 and is slightly more time demanding. This is because a handful of optimizations are disabled and a handful of new ones are enabled. Binary will almost certainly be smaller than with O2. It is also likely to be slower.

F. Using Ofast to compile the matvec.c

Table 6) time spent to compute matvec program based on Os compilation

n	Cumulative time spent (Seconds) Row major order	Cumulative time spent (Seconds) Column major order
1000	0.00	0.00
2000	0.01	0.03
5000	0.04	0.22
7500	0.07	0.55
10000	0.17	1.14
15000	0.32	4.55
20000	0.6	8.56
30000	2.53	20.62

Table 6 and indicate that while row major order should be taken into account and Ofast speed is much higher than O1, O2 and Os for column major order however, its speed is lower in row major order in all cases. it includes optimizations that can generate incorrect code even for standards-compliant software. Most notably, this disables IEEE-compliant floating point

Conclusion: the O3 is the fastest optimization but might corrupt the original code, O2 is the safest way to do optimization while having a higher speed. O3, Os and Ofast may corrupt the original code unless the code is developed according to their features.

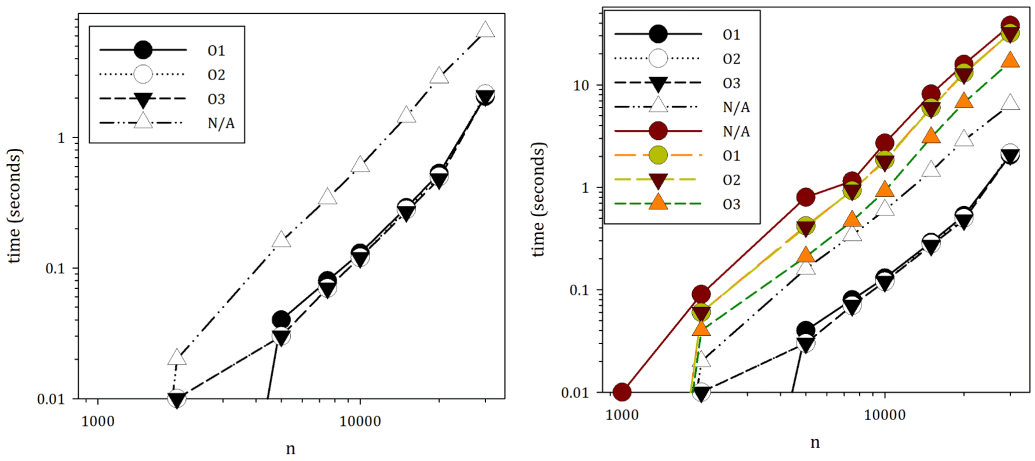

Figure 5) comparison of row major order optimization results (left figure), and comparison of all row and column major optimizations (right figure)