



# Intel® Math Kernel Library (Intel® MKL) 10.2

## Performance Improvements Overview

Here you will find performance improvements in the Intel® Math Kernel Library (Intel® MKL) by release version. **See performance charts for each math domain in the Product In-Depth PDF.**

## Contents

<b>Performance Improvements in Version 10.2</b> .....	<b>3</b>	Intel® Pentium® 4 and Intel® Xeon® processors .....	<b>5</b>
<b>Performance Improvements in Version 10.1</b> .....	<b>3</b>	BLAS.....	<b>5</b>
BLAS.....	<b>3</b>	<b>Other Improvements</b> .....	<b>5</b>
Direct sparse solver (DSS/PARDISO):.....	<b>3</b>	Sparse BLAS.....	<b>5</b>
VML and VSL .....	<b>3</b>	FFT.....	<b>5</b>
<b>Performance Improvements in Version 10.0</b> .....	<b>4</b>	VML.....	<b>5</b>
BLAS.....	<b>4</b>	<b>Performance Improvements in Version 8.1</b> .....	<b>5</b>
LAPACK.....	<b>4</b>	Intel® Itanium® 2 processor.....	<b>5</b>
<b>Performance Improvements in Version 9.1</b> .....	<b>4</b>	BLAS.....	<b>5</b>
BLAS.....	<b>4</b>	Intel® Xeon® processor .....	<b>5</b>
LAPACK.....	<b>4</b>	Intel® Pentium® 4 processor.....	<b>5</b>
Sparse BLAS.....	<b>4</b>	BLAS.....	<b>5</b>
Direct Sparse Solver (DSS/PARDISO) .....	<b>4</b>	Intel® Core™ Duo processor.....	<b>6</b>
FFTs.....	<b>4</b>	BLAS.....	<b>6</b>
<b>Performance Improvements in Version 9.0</b> .....	<b>4</b>	<b>Performance Improvements in Version 8.0</b> .....	<b>6</b>
Intel® Itanium® 2 processor.....	<b>4</b>	Intel® Xeon® processor .....	<b>6</b>
FFT.....	<b>4</b>	Intel® Pentium® 4 processor.....	<b>6</b>
VML.....	<b>4</b>	BLAS.....	<b>6</b>
VSL.....	<b>5</b>	DFT .....	<b>6</b>
Intel® Core™2 Duo processor .....	<b>5</b>	LAPACK Optimizations .....	<b>6</b>
BLAS.....	<b>5</b>		
VSL.....	<b>5</b>		
VML.....	<b>5</b>		

## Performance Improvements in Version 10.2

- Further threading in BLAS level 1 and 2 functions for Intel® 64 architecture
  - Level 1 functions (vector-vector): (C,S,Z,D,S,D)ROT, (C,Z,S,D)COPY, and (C,Z,S,D)SWAP
    - » Increase in performance by up to 1.7-4.7 times over version 10.1 Update 1 on 4-core Intel® Core™ i7 processor depending on data location in cache
    - » Increase in performance by up to 14-130 times over version 10.1 Update 1 on 24-core Intel® Xeon® processor 7400 series system, depending on data location in cache
  - Level 2 functions (matrix-vector): (C,Z,S,D)TRMV, (S,D)SYMV, (S,D)SYR, and (S,D)SYR2
    - » Increase in performance by up to 1.9-2.9 times over version 10.1 Update 1 on 4-core Intel Core i7 processor, depending on data location in cache
    - » Increase in performance by up to 16-40 times over version 10.1 Update 1 on 24-core Intel Xeon processor 7400 series system, depending on data location in cache
- Introduced recursive algorithm in 32-bit sequential version of DSYRK for up to 20% performance improvement on Intel Core i7 processors and Intel Xeon processors in 5300, 5400, and 7400 series.
- Improved LU factorization (DGETRF) by 25% over Intel® MKL 10.1 Update 1 for large sizes on the Intel® Xeon® 7460 processor; small sizes are also dramatically improved
- BLAS\*TBMV\*/TBSV\* functions now use level 1 BLAS functions to improve performance by up to 3% on Intel Core i7 processors and up to 10% on Intel® Core™2 processor 5300 and 5400 series.
- Improved threading algorithms to increase DGEMM performance
  - up to 7% improvement on 8 threads and up to 50% on 3,5,7 threads on the Intel Core i7 processor
  - up to 50% improvement on 3 threads on Intel® Xeon® processor 7400 series.
- Threaded 1D complex-to-complex FFTs for non-prime sizes
- New algorithms for 3-D complex-to-complex transforms deliver better performance for small sizes (up to 64x64x64) on 1 or 2 threads
- Implemented high-level parallelization of out-of-core (OOC) PARDISO when operating on symmetric positive definite matrices.

- Reduced memory use by PARDISO for both in-core and out-of-core on all matrix types
  - PARDISO OOC now uses less than half the memory previously used in Intel MKL 10.1 for real symmetric, complex Hermitian, or complex symmetric matrices
- Parallelized Reordering and Symbolic factorization stage in PARDISO/DSS
- Up to two times better performance (30% improvement on average) on Intel Core i7 and Intel® Core™2 processors for the following VML functions: v(s,d)Round, v(s,d)Inv, v(s,d)Div, v(s,d)Sqrt, v(s,d)Exp, v(s,d)Ln, v(s,d)Atan, v(s,d)Atan2
- Optimized versions of the following functions available for Intel® Advanced Vector Extensions (Intel® AVX)
  - BLAS: DGEMM
  - FFTs
  - VML: exp, log, and pow
  - See important information in the Intel® MKL User's Guide regarding the mkl\_enable\_instructions() function for access to these functions

## Performance Improvements in Version 10.1

### BLAS

- 32-bit improvements
- Up to 50% improvement for (Z,C)GEMM on Quad-Core Intel® Xeon® processor 5300 series
- 10% improvement for all (D,S,Z,C)GEMM code on Quad-Core Intel® Xeon® processor 5400 series
- 64-bit improvements
- 50% improvement for SGEMM on the Intel® Core™ i7 processor
- 30% improvement for right-side cases of DTRSM on the Intel Core i7 processor

### Direct sparse solver (DSS/PARDISO):

- 35% performance improvement on average for out-of-core PARDISO.

### VML and VSL

- Optimizations on the Intel Core i7 processor:
  - Up to 17% improvement for the following VML functions: Asin, Asinh, Acos, Acosh, Atan, Atan2, Atanh, Cbrt, CIS, Cos, Cosh, Conj, Div, ErfInv, Exp, Hypot, Inv, InvCbrt, InvSqrt, Ln, Log10, MulByConj, Sin, SinCos, Sinh, Sqrt, Tanh

- Up to 67% improvement for uniform random number generation
- Up to 10% improvement for VSL distribution generators based on Wichmann-Hill, Sobol, and Niederreiter BRNGs (64-bit only)

---

## Performance Improvements in Version 10.0

### BLAS

- Threading of DGEMM was improved for small and middle sizes—outer product sizes by 10%, square sizes by 80%
- DTRSM, DTRMM, and DSYRK were improved by 5-30%

### LAPACK

- Significant improvements in Nonsymmetric Eigenproblem due to improvements made on top of LAPACK 3.1\*
- HETRD\* improved by 75% with cache optimizations
- SYTRD\* improved by 40% with cache optimizations
- Successive Bandwidth Reduction approach has been implemented in HERDB\*/SYRDB\* for a speedup of 2.8 times versus traditional HETRD/SYTRD on Dual-Core Intel® Xeon® 5100 series servers
- Improved CROT/ZROT performance by 80%
- Improved tridiagonal linear equations solver (DGTSV/SGTSV) by 8-10%
- Improved tridiagonal symmetric positive definite linear equations solver (PTSV\*) by 8-28%
- Improved generalized non-symmetric eigenproblem (GGEV\*) by 3-8%, and CGGEV by 30%
- Threaded upper triangular real 8 Cholesky\* decomposition
- Performance of (D/S/Z/C)STEDC was improved by 30% and parallelized with OpenMP\* for up to 100% performance improvement on 4 threads

---

## Performance Improvements in Version 9.1

### BLAS

- Threading of DGEMM was improved for small and middle sizes—outer product sizes by 10%, square sizes by 80%
- DTRSM, DTRMM, and DSYRK were improved by 5-30%

### LAPACK

- Significant improvements in Nonsymmetric Eigenproblem due to improvements made on top of LAPACK 3.1
- HETRD\* improved by 75% on cache
- SYTRD\* improved by 40% on cache
- Successive Bandwidth Reduction approach has been implemented

in HERDB\*/SYRDB\* for a speedup of 2.8 times versus traditional HETRD/SYTRD on Dual-Core Intel® Xeon® 5100 series servers

- Improved CROT/ZROT performance by 80%
- Improved tridiagonal linear equations solver (DGTSV/SGTSV) by 8-10%
- Improved tridiagonal symmetric positive definite linear equations solver (PTSV\*) by 8-28%
- Improved generalized non-symmetric eigenproblem (GGEV\*) by 3-8%, and CGGEV by 30%
- Threaded upper triangular real 8 Cholesky\* decomposition
- Performance of (D/S/Z/C)STEDC was improved by 30% and parallelized with OpenMP\* for up to 100% performance improvement on 4 threads

### Sparse BLAS

- Performance of triangular solver routines for the diagonal format was improved by 20-50% in serial mode affecting the following functions: mkl\_ddiasm, mkl\_ddiasv, and mkl\_ddiatrsv
- Level 3 triangular solver mkl\_ddiasm was threaded

### Direct Sparse Solver (DSS/PARDISO)

- DSS/PARDISO – performance is improved by 20-30% from last release for symmetric, positive-definite matrices with one and multiple right-hand sides

### FFTs

- Single complex forward 1D FFT size greater than  $2^{22}$  were sped up by up to 2 times on 4 threads and up to 2.4 times on 8 threads on Intel® Itanium® processors
- Double precision complex out-of-place 1D FFTs for power-of-two sizes was sped up by up to 2 times on 4 threads
- Complex 2D FFTs were sped up for double precision by up to 1.7 times and for single precision up to 1.3 times on systems with Intel® EM64T and running in 64-bit mode

---

## Performance Improvements in Version 9.0

### Intel® Itanium® 2 processor

#### FFT

- Single and double precision, real 1D FFT for  $n = 2,4,8,16,32,64$ , and 128 were sped up by up to 3 times

#### VML

- Performance of Ln, Log10, Asin, Acos, Sinh, Cosh functions was improved by 25-83%
- Performance for longvectors ( $n > 250,000$  double precision or  $n > 500,000$  single precision) was improved more than 60% on

## Intel® Math Kernel Library (Intel® MKL) 10.2: Performance Improvements

average. This brings performance to within 20% of small vector performance on a per-element basis

### VSL

- Improved performance of the Gaussian and Multivariate Gaussian distribution generator by 35% and 15% respectively

### Intel® Core™2 Duo processor

#### BLAS

- ZDOTU, ZDOTC—10-35% performance improvement for data on cache

#### VSL

- Performance improved by approximately 13%

#### VML

- Performance of Div and Inv functions in VML were improved by 2.7 times
- Performance of Asin and Acos functions was improved by 5-20%

### Intel® Pentium® 4 and Intel® Xeon® processors

#### BLAS

- CGEMV—30-80% performance improvement for aligned data
- ZGEMV—5-35% performance improvement for unaligned data on systems with Intel® 64 architecture
- DTRMV—20% performance improvement for small sizes
- SNRM2, SCNRM2—performance improvement up to 10 times for data on cache
- ZDOTU, ZDOTC—10-25% performance improvement for aligned data on cache, and 2 times speedup for unaligned data

---

## Other Improvements

### Sparse BLAS

- Performance of sparse matrix-vector multiply routines for the compressed sparse row format was improved by 20-50%. The following functions are affected: mkl\_dcsrsmv, mkl\_dcsrgemv and mkl\_dcsrsmv.
- Performance of Asin and Acos functions was improved by 5-20%

### FFT

- Improved performance of double and single precision complex 1D FFTs for power-of-two sizes larger than 17 by up to 40%
- Single and double precision, real 1D FFTs were sped up for  $n = 2, 4, 8, 16, 32$ , or 64 by up to 3 times on systems with Intel 64 architecture running in 64-bit mode

### VML

- Functions take an advantage of multiprocessor systems on vector lengths of  $n > 200$ , as well as systems supporting Intel® Hyper-Threading technology on vector lengths of  $n > 10000$  by their automatic threading with OpenMP\*. On dual-core processors and dual-processor systems, performance was improved up to 90%.

---

## Performance Improvements in Version 8.1

### Intel® Itanium® 2 processor

#### BLAS

- Level 3 BLAS
- DTRMM—100-150 percent on small sizes (up to 200)
- Level 1 BLAS
- DSWAP, SSWAP, ZSWAP, CSWAP—up to 1.5-2 times
- DASUM—30-40 percent in memory (for sizes bigger than 500,000)
- DAMAX, IDAMIN, ISAMAX, ISAMIN—30-40 percent on cache
- ZAXPY, CAXPY—up to 1.5 times on L2 and memory
- SAXPY—up to 1.5 times

### Intel® Xeon® processor

### Intel® Pentium® 4 processor

#### BLAS

- Level 3 BLAS
- DGEMM for processors with Intel® 64 architecture by 1-3 percent
- SGEMM with Intel 64 architecture by 1.5–2 times
- ZTRSM (L side) with Intel 64 architecture by 50 percent on small sizes and 5 percent on large sizes
- Improved ZGEMM by 10-40 percent
- Level 2 BLAS
- DSWAP, SSWAP, ZSWAP, CSWAP—up to 1.5-2 times
- DASUM—30-40 percent in memory (for sizes bigger than 500,000)
- DAMAX, IDAMIN, ISAMAX, ISAMIN—30-40 percent on cache
- ZAXPY, CAXPY—up to 1.5 times on L2 and memory
- SAXPY—up to 1.5 times
- Level 1 BLAS
- DDOT—35 percent on cache
- DGEMV—40 percent on cache
- ZDOTU, ZDOTC—30-90 percent
- ZAXPY—20-50 percent

- ZSCAL—20-40 percent
- VML (for processors with Intel 64 architecture)
- Performance of some variants of Inv, Div, Ln, Log10, Tan, Asin, Atan, Atan2, Acosh, Asinh, and Atanh improved by 25 percent–250 percent

### Intel® Core™ Duo processor

#### BLAS

- VML performance improved by approximately 35 percent
- VSL performance improved by approximately 35 percent

---

### Performance Improvements in Version 8.0

- Intel® Itanium® 2 processor
- Improvement for Level-2 BLAS functions:
  - Sger—20-70 percent
  - Cgeru, cgerc—50 percent
  - Dsyr—10-70 percent
  - Ssyr—10-50 percent

### Intel® Xeon® processor

### Intel® Pentium® 4 processor

#### BLAS

- Improved DGEMM for Intel 64 architecture for sizes not divisible by 4
- Improvement is approximately 30-40 percent for small sizes and 4 percent for sizes up to 1000-2000

- Improvement for Level 2 BLAS functions:
  - CGERU, CGERC—up to 2 times on L2 cache and 12 percent on memory
  - CHER—up to 2 times on L2 cache for Upper case and 40 percent for Lower case, 5-8 percent on memory
  - CHER2—8-25 percent on L2 cache and 4 percent on memory
  - CHPR—up to 2.5 times on L2 cache for Lower case, 10 percent for Upper case, 3-15 percent on memory
  - CHPR2—up to 2 times on L2 cache for Lower case, 7 percent for Upper case, 5-70 percent on memory
  - CHBMV—up to 2 times on L2 cache, 20-30 percent on memory
  - CHPMV—up to 2 times on L2 cache for lower case, 75 percent for Upper case, 50-80 percent on memory

#### DFT

- The improvement for 2D on L2 cache is 25-100 percent for complex data, up to 30 percent for single real data, up to 16 percent for double real data
- The improvement for 3D on L2 cache is 5-50 percent for complex data

#### LAPACK Optimizations

- Improved BDSQR performance by 10 percent–500 percent, depending on the number of updated singular vectors. (Fewer updated vectors means more optimization.)
- Improved DSTEQR performance by 30-500 percent

Performance tests and ratings are measured using specific computer systems and/or components and reflect the appropriate performance of Intel products as measured by those tests. Any difference in system design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. Get more information on performance tests and on the performance of Intel products.

© 2009, Intel Corporation. All rights reserved. Intel, the Intel logo, Intel Core, Itanium, Pentium, and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

\*Other names and brands may be claimed as the property of others.

0609/BLA/CMD/PDF 321514-001

