MAGMA MIC

MAGMA on Intel Xeon Phi (MIC)

Innovative Computing Laboratory
Electrical Engineering and Computer Science
University of Tennessee

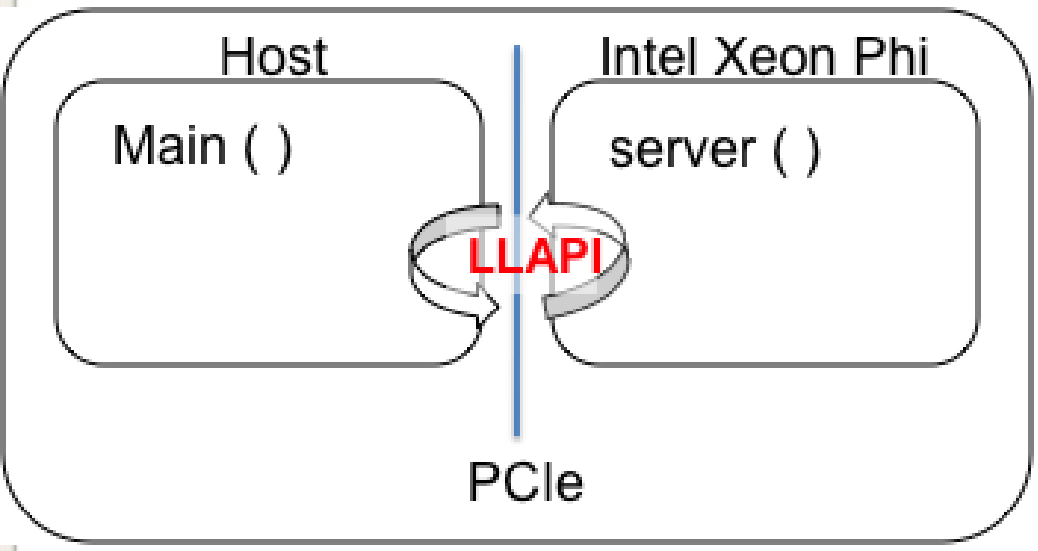
Piotr Luszczek (presenter)

web.eecs.utk.edu/~luszczek/conf/

Intel Xeon Phi specific considerations

- Intel Xeon Phi coprocessors (vs GPUs) are less dependent on host
  - can login on the coprocessor, develop, and run programs in native mode
- There is no high-level API similar to CUDA/OpenCL facilitating Intel Xeon Phi's use from the host
  - OpenCL 1.2 support is available for Intel Xeon Phi as of Intel SDK XE 2013 Beta
- There is Phi-specific syntax for pragma programming
  - Syntax: `#pragma offload target(mic:0) in(a,b) out(c) wait(x)`
  - It may be too high-level for some HPC applications and numerical libraries
- Compiler offload interface (COI)
- We used Intel Xeon Phi's Low Level API (LLAPI) to develop MAGMA API
  - Allows us to uniformly handle hybrid systems
  - Messaging API (SCIF) is platform-specific

MAGMA MIC programming model



- Intel Xeon Phi acts as coprocessor
- On the Intel Xeon Phi, MAGMA runs a "server"
- Communications are implemented using LLAPI and SCIF

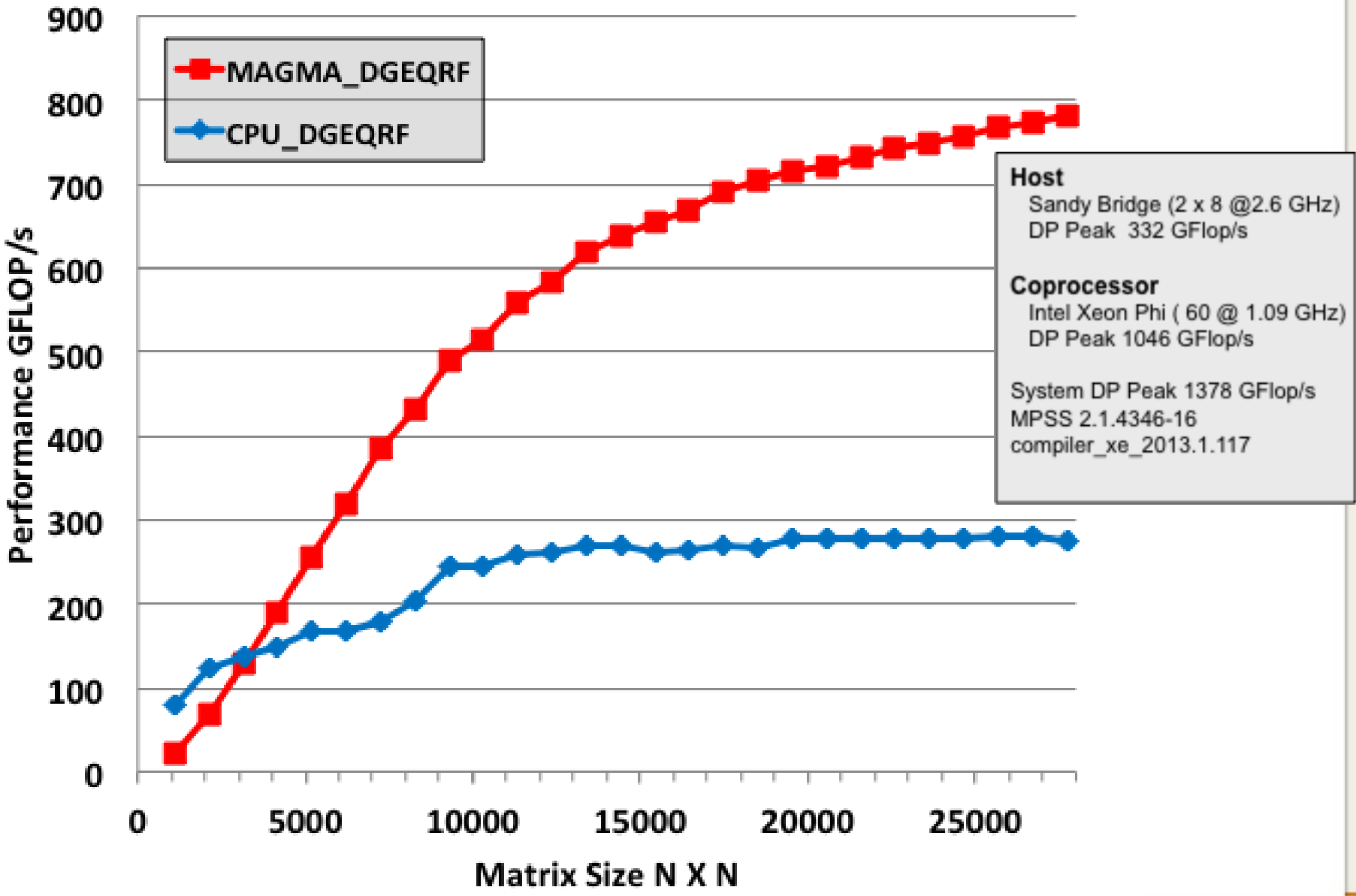
MAGMA MIC Cholesky Code

```
for ( j=0; j<n; j += nb)  {
      jb = min(nb, n – j);
      magma_zherk( MagmaUpper, MagmaConjTrans,
                           jb, j, m_one, dA(0, j), ldda, one, dA(j, j), ldda
      magma_zgetmatrix_async( jb, jb, dA(j,j), ldda, work, 0, jb, q
      if ( j+jb < n )
          magma_zgemm( MagmaConjTrans, MagmaNoTrans, jb,
                               dA(0, j ), ldda, dA(0, j+jb), ldda, z_one,
      magma_event_sync( event );
      lapackf77_zpotrf( MagmaUpperStr, &jb, work, &jb, info );
      if ( *info != 0 )
           *info += j;
      magma_zsetmatrix_async( jb, jb, work, 0, jb, dA(j,j), ldda, q
      if ( j+jb < n ) {
          magma_event_sync( event );
          magma_ztrsm( MagmaLeft, MagmaUpper, MagmaConjT
                               jb, n-j-jb, z_one, dA(j, j), ldda, dA(j, j+jb), ldda,
      }
}
```

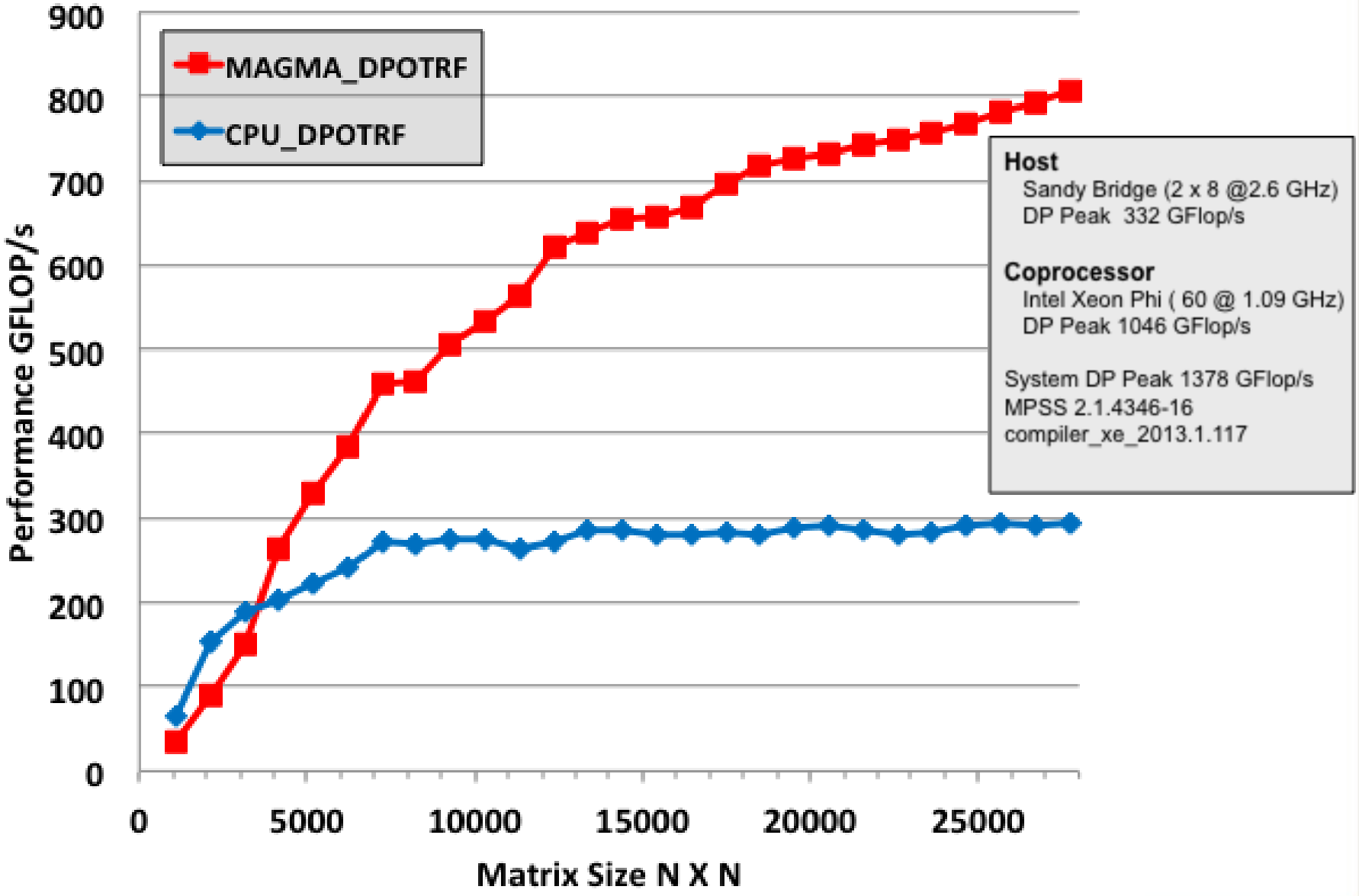
```
// ========================================
// BLAS functions
magma_err_t
magma_zgemm(
    magma_trans_t transA, magma_trans_t transB,
    magma_int_t m, magma_int_t n, magma_int_t k,
    magmaDoubleComplex alpha, magmaDoubleComplex_const_ptr dA, size_t dA_offset, magma_int_t lda,
                              magmaDoubleComplex_const_ptr dB, size_t dB_offset, magma_int_t ldb,
    magmaDoubleComplex beta,  magmaDoubleComplex_ptr       dC, size_t dC_offset, magma_int_t ldc,
    magma_queue_t handle )
{
    int err;
    magma_mic_zgemm_param gemm_param;

    gemm_param.transa = transA;
    gemm_param.transb = transB;
    gemm_param.m      = m;
    gemm_param.n      = n;
    gemm_param.k      = k;
    gemm_param.alpha  = alpha;
    gemm_param.a      = dA + dA_offset;
    gemm_param.lda    = lda;
    gemm_param.b      = dB + dB_offset;
    gemm_param.ldb    = ldb;
    gemm_param.beta   = beta;
    gemm_param.c      = dC + dC_offset;
    gemm_param.ldc    = ldc;

    int control_msg = magma_mic_ZGEMM;

    if ((err = scif_send(gEpd, &control_msg, sizeof(control_msg), 1)) <= 0) {
```

Send asynchronous requests to the MIC;
Queued & Executed on the MIC

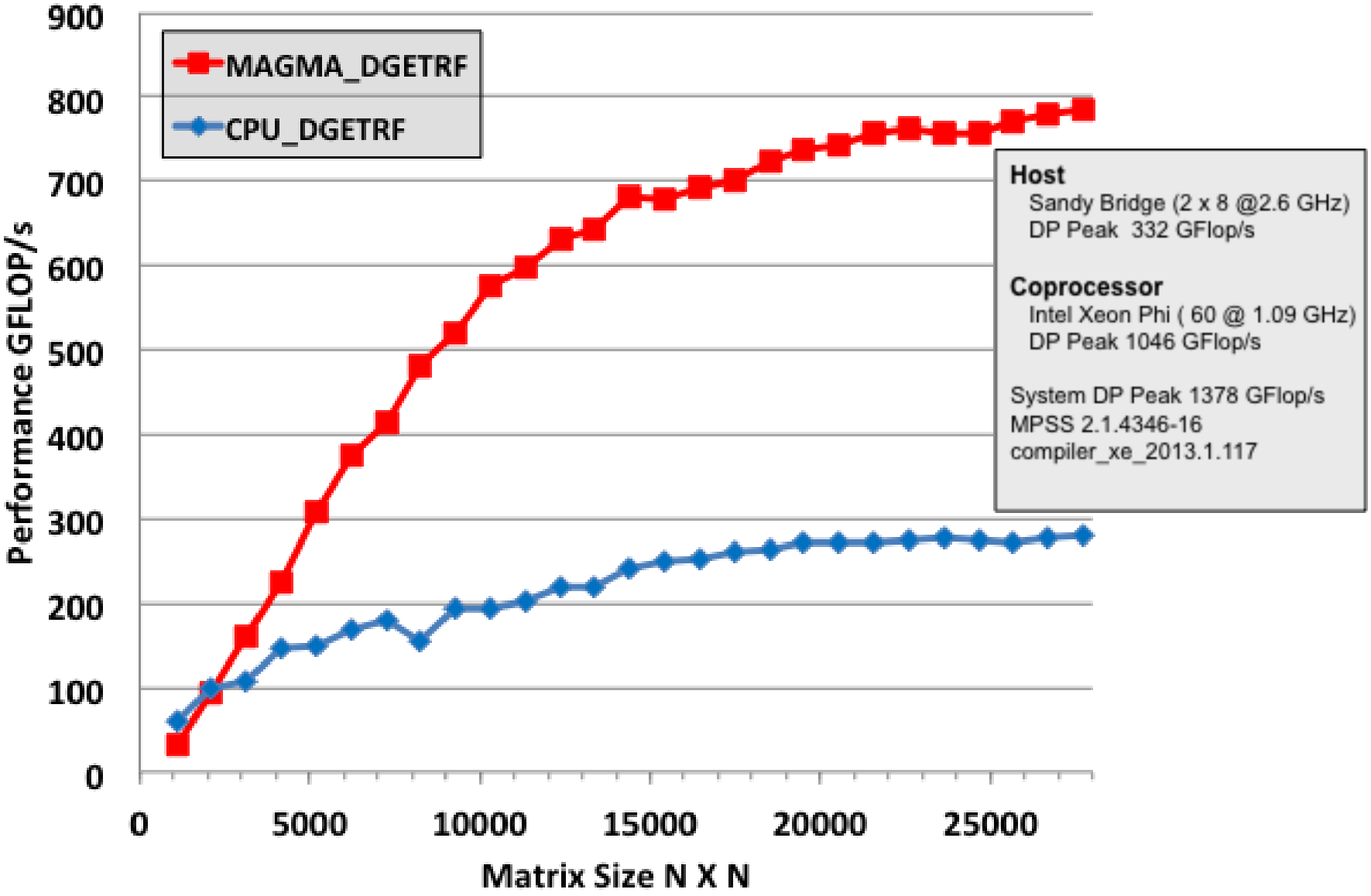
MAGMA MIC Performance (QR)



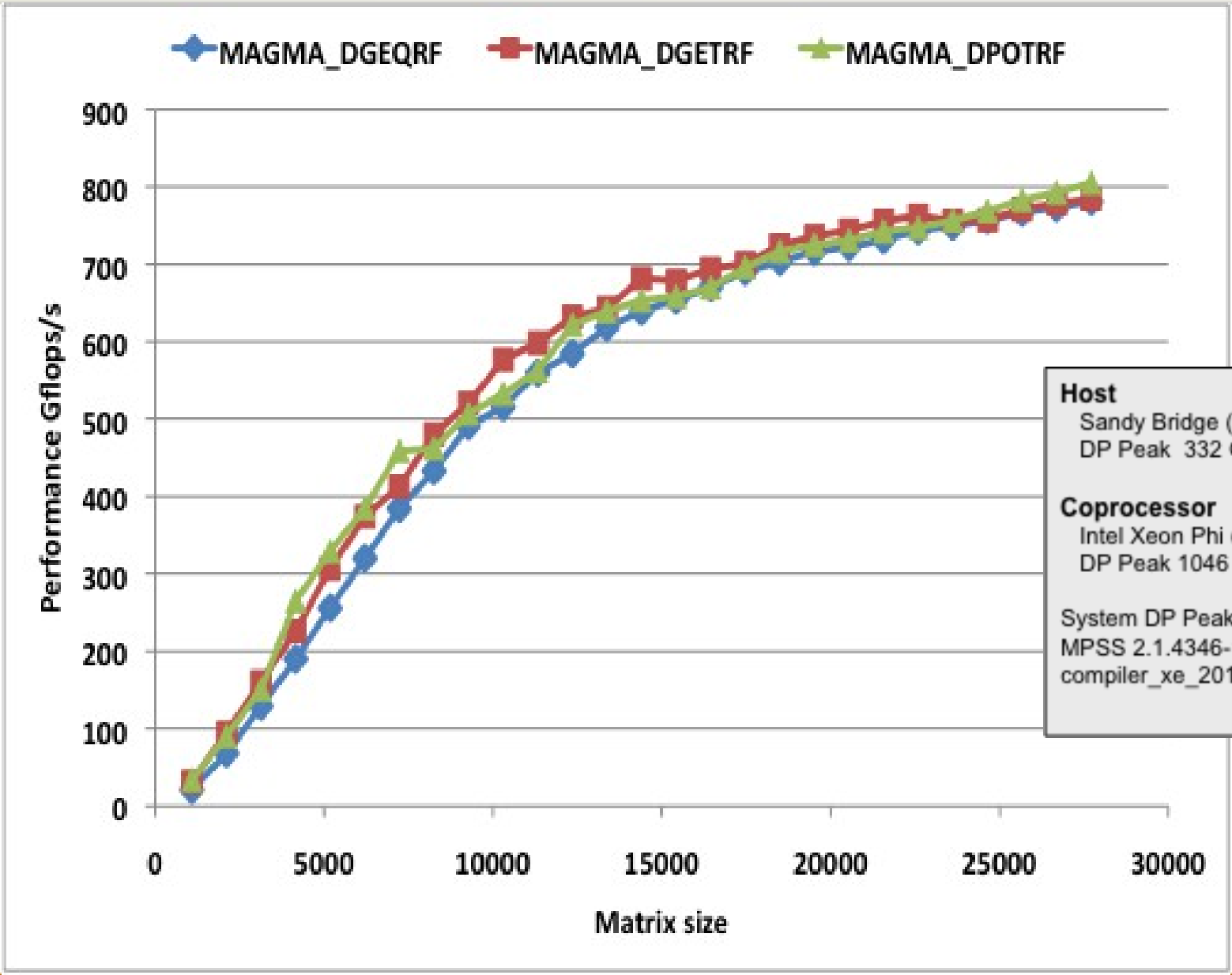
MAGMA MIC Performance (Cholesky)



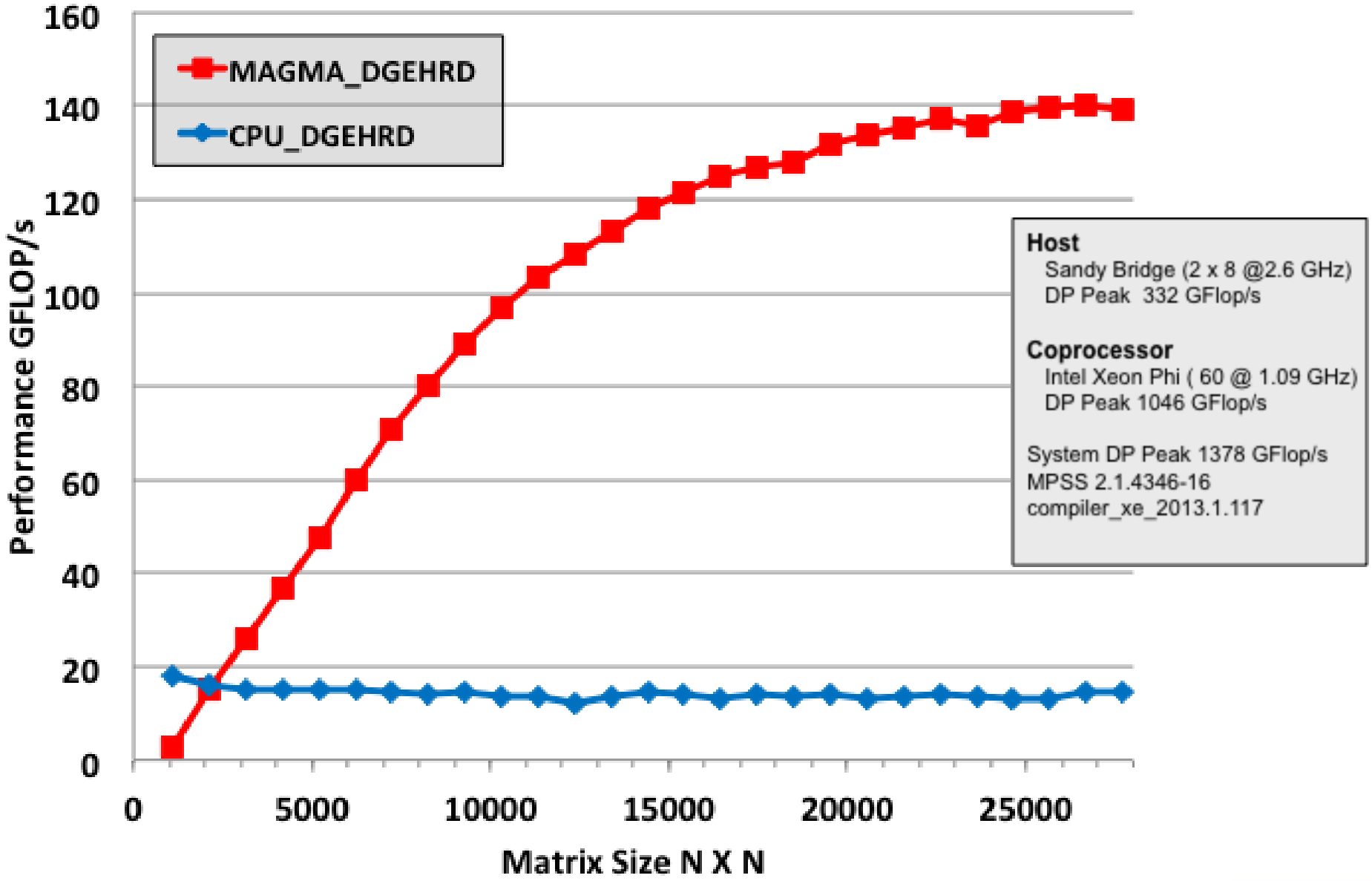
MAGMA MIC Performance (LU)



MAGMA MIC Performance (LU, QR, Chol.)

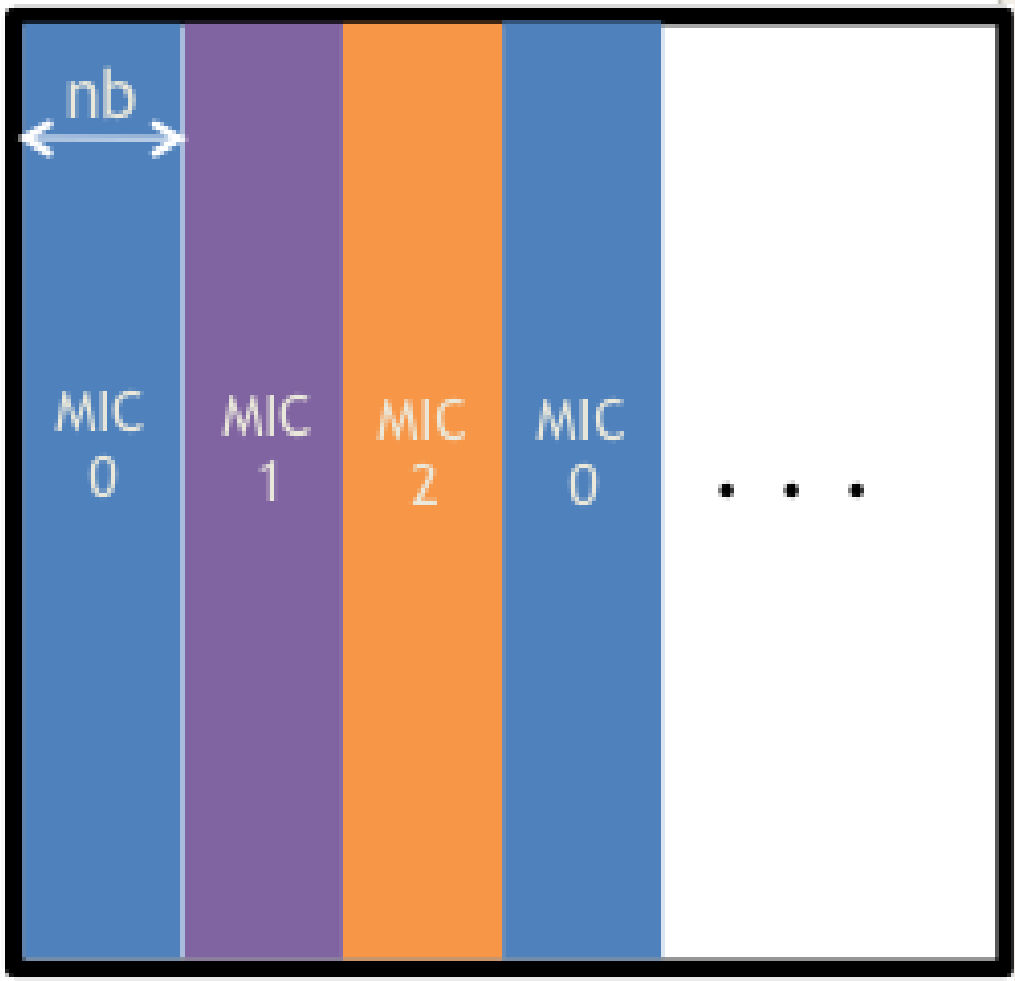


MAGMA MIC Performance (Hessenberg)

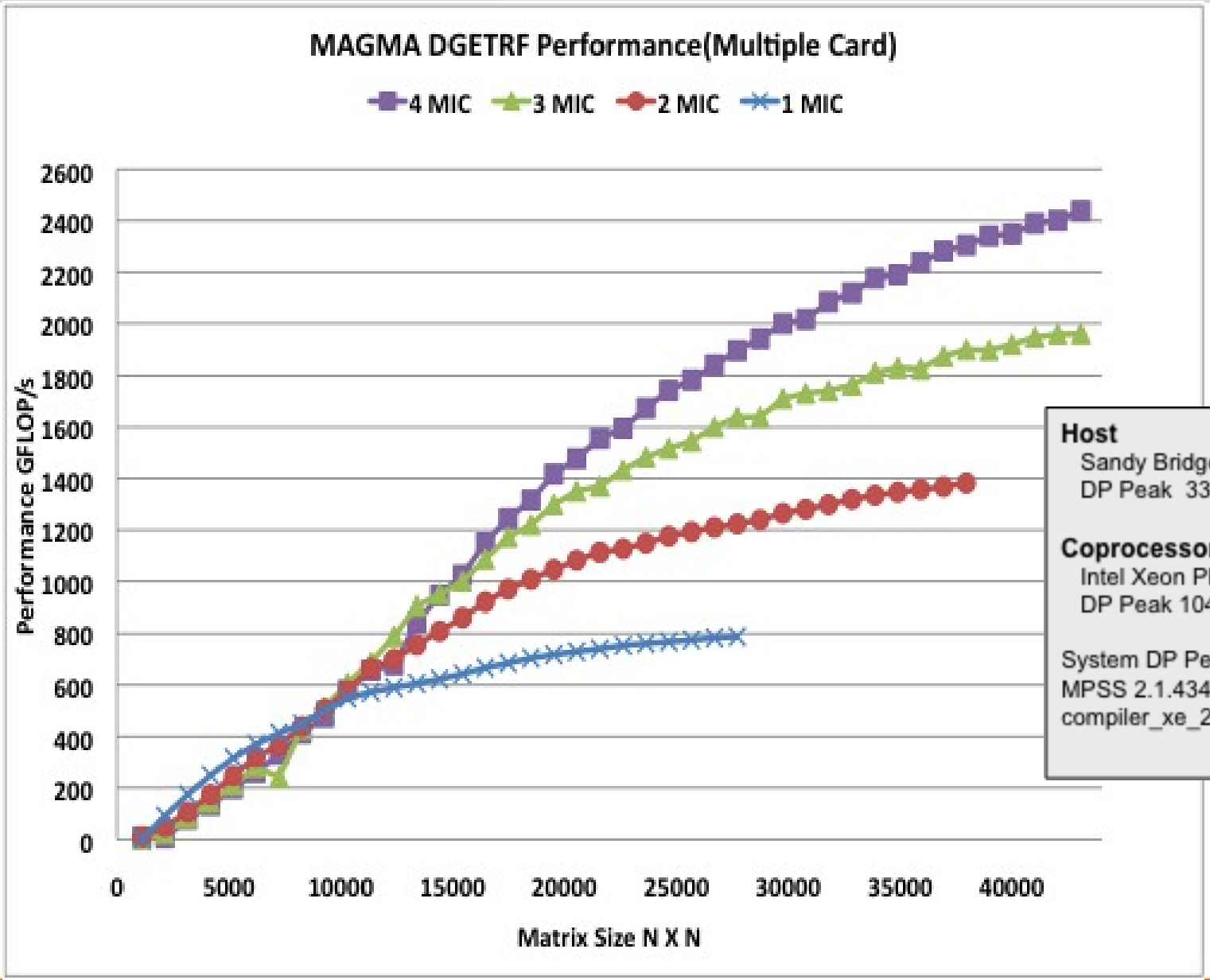


From Single to Multi-MIC Support

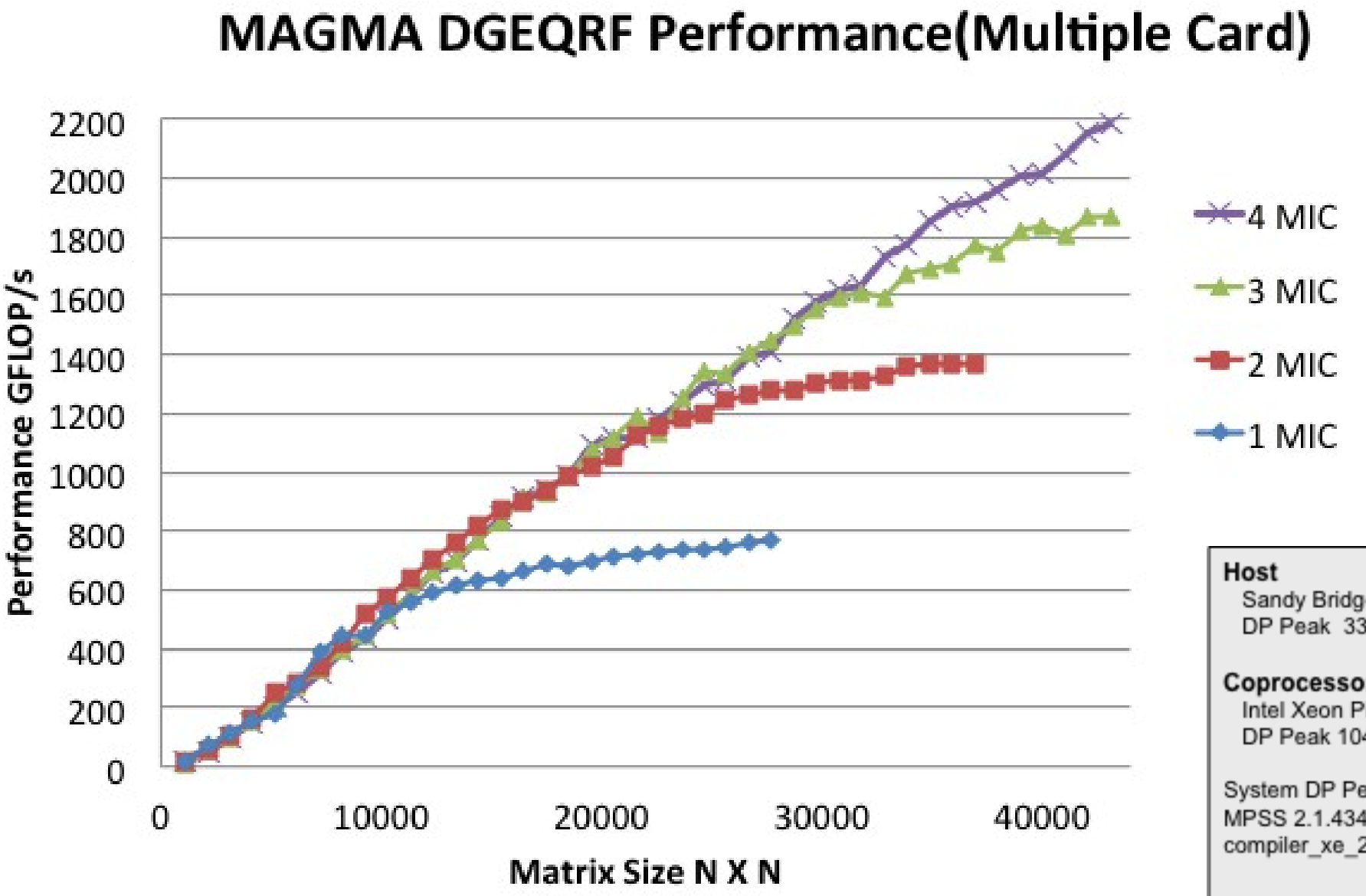
- Data distribution
  - 1-D block-cyclic distribution
- Algorithm
  - MIC holding current panel is sending it to CPU
  - All updates are done in parallel on the MICs
  - Look-ahead is done with MIC holding the next panel



MAGMA MIC Scalability for LU (real64)



MAGMA MIC Scalability for QR (real64)

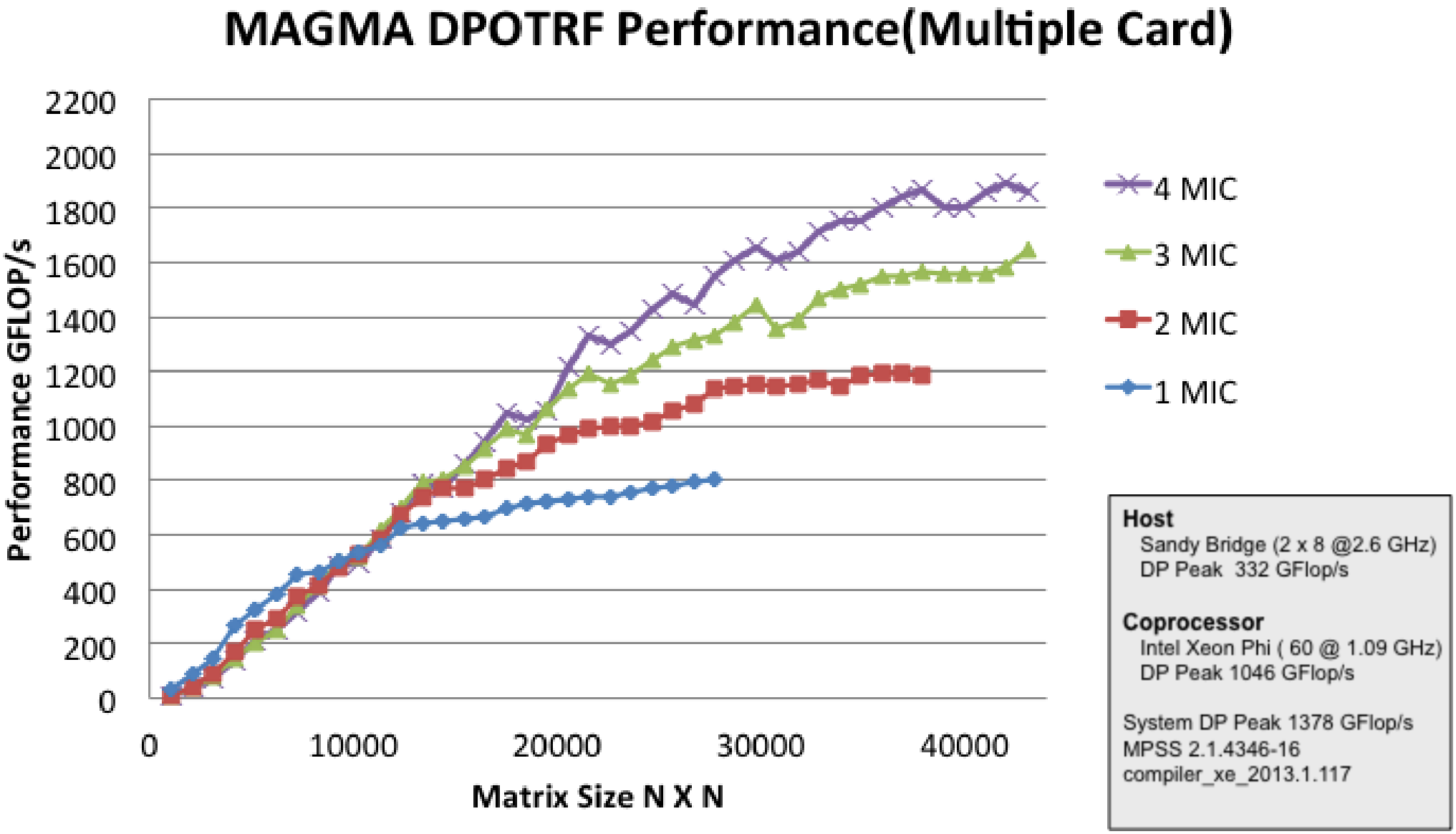


Host
Sandy Bridge (2 x 8 @2.6 GHz)
DP Peak 332 GFlop/s

Coprocessor
Intel Xeon Phi (60 @ 1.09 GHz)
DP Peak 1046 GFlop/s

System DP Peak 1378 GFlop/s
MPSS 2.1.4346-16
compiler_xe_2013.1.117

MAGMA MIC Scalability for Cholesky (real64)



Contact Information and Generous Sponsors

Stan Tomov

tomov@eecs.utk.edu

MAGMA team

http://icl.cs.utk.edu/magma/

PLASMA team

http://icl.cs.utk.edu/plasma/

Collaborating partners

- University of Tennessee, Knoxville
- University of California, Berkeley
- University of Colorado, Denver
- INRIA, France (StarPU team)
- KAUST, Saudi Arabia

MAGMA

PLASMA