Comparative Study of One-Sided Factorizations with Multiple Software Packages on Multi-Core Hardware

Emmanuel AGULLO
Jack DONGARRA
Bilel HADRI
Jakub KURZAK
Hatem LTAIEF
Piotr LUSZCZEK

Scheduling for Large-Scale Systems, Knoxville, TN, May 13-15, 2009
1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Tile Cholesky Factorization

\[
\begin{align*}
\text{FOR } k &= 0..\text{TILES}-1 \\
\text{FOR } n &= 0..k-1 \\
&A[k][k] \leftarrow \text{DSYRK}(A[k][n], A[k][k]) \\
&A[k][k] \leftarrow \text{DPOTRF}(A[k][k]) \\
\text{FOR } m &= k+1..\text{TILES}-1 \\
\text{FOR } n &= 0..k-1 \\
&A[m][k] \leftarrow \text{DGEMM}(A[k][n], A[m][n], A[m][k]) \\
&A[m][k] \leftarrow \text{DTRSM}(A[k][k], A[m][k])
\end{align*}
\]

- Basically identical to the block algorithm (LAPACK).
- Input matrix stored and processed by square tiles.
- Complex DAG.
Tile Cholesky Factorization - Static pipeline

- Work partitioned in one dimension (by block-rows).
- Cyclic assignment of work across all steps of the factorization (pipelining of factorization steps).
- Process tracking by a global progress table.
- Stall on dependencies (busy waiting).

```c
void dsyrk(double *A, double *T);
void dpotrf(double *T);
void dgemm(double *A, double *B, double *C);
void dtrsm(double *T, double *C);

k = 0; m = my_core_id;
while (m >= TILES) {
    k++; m = m-TILES+k;
} n = 0;

while (k < TILES && m < TILES) {
    next_n = n; next_m = m; next_k = k;
    next_n++;
    if (next_n > next_k) {
        next_m += cores_num;
        while (next_m >= TILES && next_k < TILES) {
            next_k++; next_m = next_m-TILES+next_k;
        }
        next_n = 0;
    }
    if (m == k) {
        if (n == k) {
            dpotrf(A[k][k]);
            core_progress[k][k] = 1;
        } else {
            while(core_progress[k][n] != 1);
            dsyrk(A[k][n], A[k][k]);
        }
    } else {
        if (n == k) {
            while(core_progress[k][k] != 1);
            dtrsm(A[k][k], A[m][k]);
            core_progress[m][k] = 1;
        } else {
            while(core_progress[k][n] != 1);
            while(core_progress[m][n] != 1);
            dgemm(A[k][n], A[m][n], A[m][k]);
        }
    }
}
```

[Diagram showing the execution and dependency tracking in a matrix format]
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Tile Algorithms

**Tile QR (&LU) Factorization**

```plaintext
FOR k = 0..TILES-1
    A[k][k], T[k][k] ← DGRQRT(A[k][k])
FOR m = k+1..TILES-1
    A[k][k], A[m][k], T[m][k] ← DTSQRT(A[k][k], A[m][k], T[m][k])
FOR n = k+1..TILES-1
    A[k][n] ← DLARFB(A[k][k], T[k][k], A[k][n])
    A[k][n], A[m][n] ← DSSRFB(A[m][k], T[m][k], A[k][n], A[m][n])
```

- Different from the block algorithm.
- Derived from out-of-core algorithm.
- Input matrix stored and processed by square tiles.
- Complex DAG.
Tile QR Factorization - Static pipeline

void dgeqrt(double *RV1, double *T);
void dtsqrt(double *R, double *V2, double *T);
void dlarfb(double *V1, double *T, double *C1);
void dssrfb(double *V2, double *T, double *C1, double *C2);

k = 0; n = my_core_id;
while (n >= TILES) {
    k++;
    n = n-TILES+k;
} m = k;

while (k < TILES && n < TILES) {
    next_n = n; next_m = m; next_k = k;
    next_m++;
    if (next_m == TILES) {
        next_n += cores_num;
        while (next_n >= TILES && next_k < TILES) {
            next_k++;
            next_n = next_n-TILES+next_k;
        } next_m = next_k;
    }
    if (n == k) {
        if (m == k) {
            while(progress[k][k] != k-1);
dgeqrt(A[k][k], T[k][k]);
progress[k][k] = k;
        } else {
            while(progress[m][k] != k-1);
dtsqrt(A[k][k], A[m][k], T[m][k]);
progress[m][k] = k;
        }
    } else {
        while(progress[k][k] != k);
        while(progress[k][n] != k-1);
dlarfb(A[k][k], T[k][k], A[k][n]);
    }
else {
    while(progress[m][k] != k);
    while(progress[m][n] != k-1);
dssrfb(A[m][k], T[m][k], A[m][n], A[m][n]);
progress[m][n] = k;
} n = next_n; m = next_m; k = next_k;

★ Work partitioned in one dimension (by block-rows).
★ Cyclic assignment of work across all steps of the factorization (pipelining of factorization steps).
★ Process tracking by a global progress table.
★ Stall on dependencies (busy waiting).

![Diagram of Tile QR Factorization - Static pipeline]

PLASMA group

Comparative Study of One-Sided Factorizations
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
 Libraries

- **LAPACK:**
  - LAPACK 3.2 on Intel machine;
  - LAPACK 3.1.1 on IBM machine;

- **SCALAPACK:**
  - SCALAPACK 1.8.0;

- **Vendor libraries:**
  - Intel MKL 10.1;
  - IBM ESSL 4.3;
  - IBM PESSL 3.3;

- **Tile algorithms:**
  - PLASMA;
  - TBLAS.
**Libraries**

- **LAPACK:**
  - LAPACK 3.2 on Intel machine;
  - LAPACK 3.1.1 on IBM machine;

- **SCALAPACK:**
  - SCALAPACK 1.8.0;

- **Vendor libraries:**
  - Intel MKL 10.1;
  - IBM ESSL 4.3;
  - IBM PESSL 3.3;

- **Tile algorithms:**
  - PLASMA
  - TBLAS.
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Experimental environment

**Hardware**

- **Node:**
  - quad-socket quad-core Intel64 processors (16 cores).

- **Intel Xeon processor:**
  - quad-core;
  - Frequency: 2.4 GHz.

- **Theoretical peak:**
  - 9.6 Gflop/s/core;
  - 153.6 Gflop/s/node.

- **System and compilers:**
  - Linux 2.6.25;
  - Intel Compilers 11.0.
IBM Power6 - 32 cores machine

- **Node:**
  - 16 dual-core Power6 processors (32 cores).

- **Power6 processor:**
  - dual-core;
  - each core 2-way SMT;
  - L1: 64kB data + 64 kB instructions;
  - L2: 4 MB per core, accessible by the other core;
  - L3: 32 MB per processor, one controller per core (80 MB/s).
  - Frequency: 4.7 GHz.

- **Theoretical peak:**
  - 18.8 Gflop/s/core;
  - 601.6 Gflop/s/node.

- **System and compilers:**
  - AIX 5.3;
  - xlf version 12.1;
  - xlc version 10.1.
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Performance metrics (How to read the graphs)

- **Performance**: Gflop/s (y-axis).
- **Plots** scaled to the theoretical peak.
- **Parallel DGEMM**.
- **Upper bound**: embarrassingly parallel fastest core kernel:
  - \( \text{DPOTRF} (L L^T) \to \text{dgemm} \);
  - \( \text{DGEQRF} (Q R) \to \text{dssrfb} \);
  - \( \text{DGETRF} (L U) \to \text{dssssm} \).

![Graphs showing performance metrics for Intel64 and Power6 DGEMM over different matrix sizes and core counts.](image-url)
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Input parameters of the serial core kernels:

- **NB**: tile size;
- **IB**: internal blocking (for dssrfb and dssssm only).
Impact of NB - DPOTRF - Intel64 - 16 cores

Matrix size vs. Performance for different NB values

- NB=196
- NB=168
- NB=120
- NB=84
- NB=60

16xdgemm-seq
Impact of NB - DPOTRF - Power6 - 32 cores

Matrix size
Impact of NB/IB - DGEQRF - Intel64 - 16 cores

Matrix size vs Performance for different NB/IB configurations:
- NB=256 IB=64
- NB=200 IB=40
- NB=168 IB=56
- NB=120 IB=40
- NB=84 IB=28
- NB=60 IB=20

16xdsrflb-seq
Impact of NB/IB - \texttt{DGEQRF} - Power6 - 32 cores

Comparative Study of One-Sided Factorizations
Impact of NB/IB - DGETRF - Intel64 - 16 cores

- NB=252 IB=28
- NB=196 IB=28
- NB=168 IB=28
- NB=120 IB=24
- NB=84 IB=28
- NB=60 IB=20

Matrix size vs. Performance

- 16xdssssm-seq
Impact of NB/IB - DGETRF - Power6 - 32 cores

Matrix size

0 2000 4000 6000 8000 10000 12000

NB=480 IB=60
NB=340 IB=68
NB=280 IB=56
NB=240 IB=60
NB=196 IB=28
NB=168 IB=28
NB=120 IB=40
NB=80 IB=20

32xdssssm-seq
Exhaustive search

For "each" matrix size and number of cores:

1. Time PLASMA on all NB/IB samples;
2. Select the best sample.

Number of samples

⋆ \(|\{(IB, NB) \mid IB|NB, 40 \leq NB \leq 500, 4 \leq IB \leq NB\}|=1352;\)
⋆ all combinations cannot be explored on large executions;

→ need for a pruned search.
Exhaustive search

For “each” matrix size and number of cores:

1. Time PLASMA on all NB/IB samples;
2. Select the best sample.

Number of samples

- $|\{(IB, NB) \mid IB|NB, 40 \leq NB \leq 500, 4 \leq IB \leq NB\}| = 1352$;
- all combinations cannot be explored on large executions;

→ need for a pruned search.
Method

1. Time serial core kernels (dgemm, dssrfb, dssssm).

2. Pick up the "best" NB or NB/IB samples (pruning);

3. Select one per matrix size and number of cores.

Intel64 - dgemm  

Power6 - dssrfb
Pruned search

Method

1. Time serial core kernels (dgemm, dssrfb, dssssm).

   ![Graph 1: Intel64 - dgemm](image1)
   ![Graph 2: Power6 - dssrfb](image2)

2. Pick up the ”best” NB or NB/IB samples (pruning);
3. Select one per matrix size and number of cores.
Pruned search

### Method

1. Time serial core kernels (dgemm, dssrfb, dssssm).

   ![Graph showing performance for different NB values for Intel64 - dgemm and Power6 - dssrfb.]

   - **Intel64 - dgemm**
   - **Power6 - dssrfb**

2. Pick up the "best" NB or NB/IB samples (pruning);

3. Select one per matrix size and number of cores.
Exhaustive search VS pruned search

Intel64 - 16 cores - DPOTRF
Exhaustive search VS pruned search

Intel64 - 16 cores - DGEQRF
Exhaustive search VS pruned search

Intel 64 - 16 cores - DGETRF

Comparison of EXPLASMA and PLASMA with different matrix sizes.
Other software

- **PLASMA**: pruned search.

- **TBLAS**: exhaustive search.

- **SCALAPACK, PESSL**: exhaustive search.

- **LAPACK, MKL, ESSL**: tuned by vendor.
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Comparison against other libraries
Experiments on few cores

DPOTRF - Intel64 - 4 cores

![Graph comparing DPOTRF against other libraries on 4 cores.](image)

- **DPOTRF**
- Intel64
- 4 cores

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on few cores

DPOTRF - Power 6 - 2 cores

Matrix size

DGEMM
PLASMA
TBLAS
ESL
PESSL
SCALAPACK
LAPACK

2xdgemm-seq
Comparison against other libraries

Experiments on few cores

DGEQRF - Intel64 - 4 cores

Comparison of performance for DGEQRF compared to other libraries (DGEMM, PLASMA, TBLAS, MKL, SCALAPACK, LAPACK) across different matrix sizes for Intel64 with 4 cores.

Matrix size vs. Performance (in seconds)
Comparison against other libraries

Experiments on few cores

DGEQRF - Power6 - 2 cores
Comparison against other libraries

Experiments on few cores

DGETRF - Intel64 - 4 cores

Matrix size

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on few cores

DGETRF - Power6 - 2 cores

Matrix size vs. performance for different libraries:
- DGEMM
- PLASMA
- ESSL
- PESSL
- SCALAPACK
- LAPACK

Graph shows performance across different matrix sizes for 2 cores on Power6 architecture.
1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Comparison against other libraries

Experiments on a large number of cores

DPOTRF- Intel64- 16 cores

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on a large number of cores

**DPOTRF - Power6 - 32 cores**

![Graph comparing various libraries](image)

- **DGEMM**
- **PLASMA**
- **TBLAS**
- **ESSL**
- **PESSL**
- **SCALAPACK**
- **LAPACK**

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on a large number of cores

DGEQRF - Intel64 - 16 cores

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on a large number of cores

DGEQRF - Power6 - 32 cores

PLASMA group

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on a large number of cores

DGETRF - Intel64 - 16 cores

Matrix size

Comparative Study of One-Sided Factorizations
Comparison against other libraries

Experiments on a large number of cores

**DGETRF - Power6 - 32 cores**

![Graph comparing libraries for matrix factorization](image)

- **DGEMM**
- **PLASMA**
- **ESSL**
- **PESSL**
- **SCALAPACK**
- **LAPACK**

Matrix size vs. performance comparison for different libraries.

Comparative Study of One-Sided Factorizations
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Comparison against other libraries

PLASMA - DPOTRF - Intel64

Matrix size vs. Performance

- 16 cores
- 14 cores
- 12 cores
- 10 cores
- 8 cores
- 6 cores
- 4 cores
- 2 cores
- 1 core

16xdgemm-seq
Comparison against other libraries

PLASMA scalability

PLASMA - DGEQRF - Intel64

Comparative Study of One-Sided Factorizations
Comparison against other libraries

PLASMA scalability

PLASMA - DGETRF - Intel64

Matrix size

16 cores
14 cores
12 cores
10 cores
8 cores
6 cores
4 cores
2 cores
1 core

16xdsssssm-seq
Comparison against other libraries

PLASMA - DPOTRF - Power6

Comparative Study of One-Sided Factorizations
Comparison against other libraries

PLASMA scalability

PLASMA- DGEQRF- Power6

Comparative Study of One-Sided Factorizations
Outline

1. Tile Algorithms
   - Cholesky Factorization
   - QR (&LU) Factorizations

2. Experimental environment
   - Libraries
   - Hardware
   - Metrics

3. Tuning
   - PLASMA

4. Comparison against other libraries
   - Experiments on few cores
   - Experiments on a large number of cores
   - PLASMA scalability

5. Conclusion and current work
Conclusion

★ Performance brought by tile algorithms:
  ☹ Possible overheads:
    • extra-flops;
    • kernels not optimized.
  😊 Benefits:
    • better data reuse;
    • better scheduling opportunities.

★ Better scalability.

★ Importance of tuning:
  → efficient pruned search.
Current work

- **Compute-intensive kernels:**
  successive BLAS-3 calls → single BLAS-3 call.

- **Dynamic scheduling:**
  → Piotr’s presentation.

- **Improve scalability for small matrix sizes:**
  → increase parallelism (tile TSQR).

- **Generalization to other linear algebra algorithms:**
  → two-sided factorizations.
Questions?
1. Scalability of other libraries
   - TBLAS
   - MKL- ESSL
   - SCALAPACK- PESSL
   - LAPACK
1. Scalability of other libraries

- TBLAS
- MKL- ESSL
- SCALAPACK- PESSL
- LAPACK
Scalability of other libraries

TBLAS - DPOTRF - Intel64

Comparative Study of One-Sided Factorizations
Comparative Study of One-Sided Factorizations
Scalability of other libraries

TBLAS - DPOTRF - Power6

Comparative Study of One-Sided Factorizations
Scalability of other libraries

TBLAS - DGEQRF - Power6

Comparative Study of One-Sided Factorizations
Outline

1. Scalability of other libraries
   - TBLAS
   - MKL- ESSL
   - SCALAPACK- PESSL
   - LAPACK
Scalability of other libraries

MKL - ESSL

MKL - DPOTRF - Intel64

Comparative Study of One-Sided Factorizations
Scalability of other libraries

MKL - ESSL

MKL - DGEQRF - Intel64

Comparative Study of One-Sided Factorizations
Scalability of other libraries: MKL and ESSL.

**MKL - DGETRF - Intel64**

Comparative Study of One-Sided Factorizations

The graph shows the scalability of the DGETRF library for different matrix sizes and core counts.

- **16 cores**: The green dashed line shows the performance of the 16xdssrfb-seq algorithm with 16 cores.
- **8 cores**: The blue dotted line shows the performance with 8 cores.
- **4 cores**: The magenta dash-dotted line shows the performance with 4 cores.
- **2 cores**: The orange dash-dot-dotted line shows the performance with 2 cores.
- **1 core**: The black dotted line shows the performance with 1 core.

The x-axis represents the matrix size, and the y-axis represents the performance metric (in arbitrary units), which is consistent across all core counts.
Scalability of other libraries

ESSL - DPOTRF - Power 6

Comparative Study of One-Sided Factorizations
Comparative Study of One-Sided Factorizations

ESSL - DGEQRF - Power6

Scalability of other libraries

Matrix size

0 2000 4000 6000 8000 10000 12000

0 100 200 300 400 500 600

32 cores
16 cores
8 cores
4 cores
2 cores
1 core

32xdssrfb-seq

PLASMA group
Scalability of other libraries

Comparative Study of One-Sided Factorizations
1. Scalability of other libraries

- TBLAS
- MKL- ESSL
- SCALAPACK- PESSL
- LAPACK
Comparative Study of One-Sided Factorizations
Scalability of other libraries

Comparative Study of One-Sided Factorizations

Matrix size

Power6

ScalAPACK- DPOTRF- Power6

PLASMA group

Comparative Study of One-Sided Factorizations
Scalability of other libraries

PESSL - DPOTRF - Power6

Comparative Study of One-Sided Factorizations
Scalability of other libraries

SCALAPACK - PESSL

SCALAPACK - DGEQRF - Intel 64

Comparative Study of One-Sided Factorizations

PLASMA group

Matrix size

0 2000 4000 6000 8000 10000 12000

0 20 40 60 80 100 120 140

16 cores
8 cores
4 cores
2 cores
1 core

16xdsrfb-seq
Scalability of other libraries

PESSL-DGEQRF-Power6

Comparative Study of One-Sided Factorizations

Matrix size vs. time for different core counts (1 to 32 cores) and matrix sizes (0 to 12000).
Scalability of other libraries

SCALAPACK - PESSL

SCALAPACK - DGETRF - Intel64

Comparative Study of One-Sided Factorizations
Comparative Study of One-Sided Factorizations
Comparative Study of One-Sided Factorizations
1. Scalability of other libraries

- TBLAS
- MKL- ESSL
- SCALAPACK- PESSL
- LAPACK
Scalability of other libraries

**LAPACK - DPOTRF**

Comparative Study of One-Sided Factorizations
Scalability of other libraries

LAPACK

LAPACK - DGEQRF

Comparative Study of One-Sided Factorizations
Scalability of other libraries

LAPACK

LAPACK - DGETRF

Comparative Study of One-Sided Factorizations