Jack Dongarra, Mathieu Faverge, Hatem Ltaief, Piotr Luszczek

High Performance Matrix Inversion Based on LU Factorization for Multicore Architectures

presented by

Piotr Luszczek
Preliminaries
Problem Statement

\( A \in \mathbb{R}^{n \times n} \)

\[ PA = LU \]

\[ U \rightarrow U^{-1} \]

\[ L \rightarrow L^{-1} \]

\[ A^{-1} \in \mathbb{R}^{n \times n} \]
To Keep in Mind...

In the vast majority of practical computational problems, it is unnecessary and inadvisable to actually compute $A^{-1}$.

_Forsythe, Malcolm, and Moler_
Data Layouts for Matrix Elements

Column-major (LAPACK and derivatives)

Tile (PLASMA)
Tasks and DAGs
**Block LU Inversion**

- For each panel  
  LU factorization
  - DGETF2( )
  - DLASWP( )
  - DLASWP( )
  - DTRSM( )
  - DGEMM( )

- For each panel  
  Invert U
  - DTRMM( )
  - DTRSM( )
  - DTRTI2( )

- For each panel  
  Invert L
  - DLACPY( )
  - DLASET( )
  - DGEMM( )
  - DTRSM( )

- DLASWP( ) column interchanges

**Tile LU Inversion**

- For each diagonal tile  
  parallel recursive LU
  - DGETRF( )
  for each tail tile panel
    - DLASWP( )
  for each tail tile
    - DGEMM( )
  for each left tile panel
    - DLASWP( )

- For each diagonal tile  
  Invert U
  for each tile in panel
    - DTRSM( )
  for each tail tile
    - DGEMM( )
  for each left panel tile
    - DTRSM( )
    - DTRTRI( )

- For each left tile  
  Invert L
  - DLACPY( )
  - DLASET( )
  ...

- ICL
Queuing Functions with QUARK

QUARK_Insert_Task(
    panel_LU_task,
    M, matrix_1, INPUT,
    N, matrix_2, INOUT,
    1, result, OUTPUT,
    K, buffer, SCRATCH,
    0);
DAGs of Tasks, Each State Separately

1 – LU Factorization

2 – Computation of $L^{-1}$

3 – Computation of $U^{-1}$

4 – Column swapping
DAGs of Tasks, All Stages Overlapped
Execution Traces

No Overlap of Stages

Overlap of Stages
The Case for Nested Parallelism
Panel Factorization as the Sequential Bottleneck

\[ x\text{GETRF-REC} \xrightarrow{\text{Swap} + x\text{TRSM}} x\text{GEMM} \xrightarrow{\text{Swap} + x\text{TRSM}} x\text{GEMM} \]

\[ x\text{GEMM} \xrightarrow{\text{Swap} + x\text{TRSM}} x\text{GEMM} \]

\[ x\text{GEMM} \xrightarrow{\text{Swap} + x\text{TRSM}} x\text{GEMM} \]

\[ x\text{GETRF-REC} \]
Panel Factorization is On Critical Path of DAG
Parallel Panel Factorization: Data Partitioning
function xGETRFR(M, N, column) {
    if N == 1 {
        idx = split_lxAMAX(…)
        gidx = combine_lxAMAX(idx)
        split_xSCAL(…)
    } else {
        xGETRFR(M, N/2, column)
        xLASWP(…)
        split_xTRSM(…)
        split_xGEMM(…)
        xGETRFR(M, N-N/2, column+N/2)
        xLASWP(…)
    }
}
Quick Performance Experiment
Results
Performance on AMD MagnyCours, 4x12=48 cores
LU Inversion's Power Profile: LAPACK

![Graph showing power consumption over time for different components: System, CPU, Memory, and Disk Motherboard.](image-url)
LU Inversion's Power Profile: MKL

![Graph showing power profile over time for system, CPU, memory, and disk motherboard components.](image-url)
LU Inversion's Power Profile: PLASMA

The graph shows the power profile over time for different components of a system. The x-axis represents time in seconds, ranging from 0 to 70. The y-axis represents power in watts, ranging from 0 to 450.

- **System**: The power usage starts high and remains mostly constant until a drop around the 30-second mark.
- **CPU**: Shows moderate power usage with some fluctuations.
- **Memory**: Consistent power usage with minor variations.
- **Disk**: Low power usage with minimal changes.
- **Motherboard**: Lowest power consumption among all components.

The graph indicates that the system's power consumption is significantly higher compared to the CPU, Memory, Disk, and Motherboard components.
This work was sponsored by NSF, DOE, and Microsoft.