SuperLU_DIST와 PaScaL_TDMA를 이용한 대형희소행렬의 병렬성능 벤치마크

Parallel performance benchmark of large sparse matrix using SuperLU_DIST and PaScaL_TDMA

한국과학기술정보연구원 국가슈퍼컴퓨팅본부

안찬영(coldyoung@kisti.re.kr), 권오경(okkwon@kisti.re.kr), 강지훈(jhkang@kisti.re.kr)

Introduction

Sparse matrix and its parallel solver

- **Sparse matrix**
  - A matrix in which most of the elements are zero.
  - Originate from systems that are modeled with equations describing sparse relationships between variables.
  - Finds various scientific and engineering applications : Scientific computing, optimization, and machine learning.
- **High-performance parallel solver**
  - Efficient handling and solving of large sparse matrices are crucial.
  - Various solvers according to the problems and applications : Direct, iterative, and multi-grid based solvers.
  - Performance is highly dependent on numerical algorithm, sparse. matrix representation, and parallel programming models.

Motivation and purpose

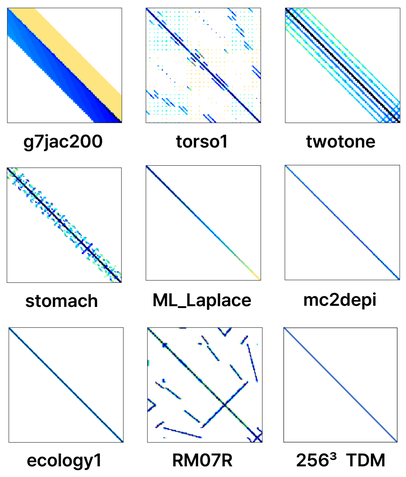
- **Inefficiency of general sparse matrix solver**
  - Well-known general-purpose sparse matrix solvers are versatile and convenient for solving any kinds of large sparse matrix system.
  - However, general solvers can not be as efficient as specialized solvers that only target a specific sparse matrix.
- **Comparative performance benchmarks between general and specialized solvers**
  - Benchmarks on parallel performance of Ax=b by using large sparse matrix files.
  - Highlight on suitable sparse matrix solvers for target problems for computational performance.

Methodology

Experimental setup

- **Hardware - KISTI Nurion Supercomputer**
  - 8,305 compute nodes interconnected with high-speed network.
  - Each node has a many-core processors, 68-Core Intel Xeon Phi 7250.
- **Benchmark tests**
  - Comparative performance benchmark of three parallel solvers: SuperLU-dist, Strumpack, and PaScaL_TDMA
  - Benchmark total and factor times with nine sparse matrices:
    - Matrix dimensions: $10^4 \sim 10^7$
    - Matrix nonzero ratio: $10^{-4} \sim 10^{-7}$
    - Select nine matrices with diagonal component structures
  - Tested in KISTI Nurion using 1 ~ 4096 cores of KNL nodes

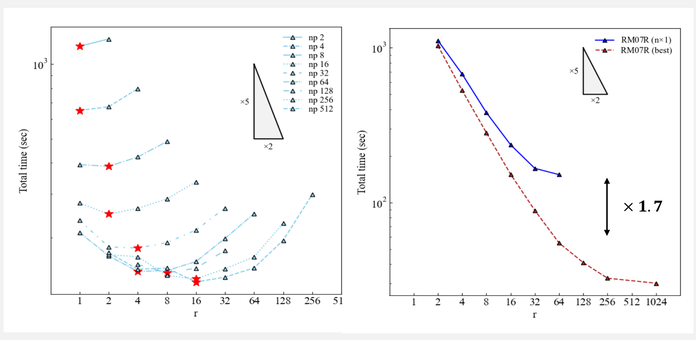
	r/c	nnz	ratio
g7jac200	59,310	717,620	2.38E-04
torso1	116,158	8,516,500	6.31E-04
twotone	120,750	1,206,265	8.40E-05
stomach	213,360	3,021,648	6.64E-05
ML_Laplace	377,002	27,582,698	1.95E-04
RM07R	381,689	37,464,962	2.57E-04
mc2depi	525,825	2,100,225	7.60E-06
ecology1	1,000,000	4,996,000	5.00E-06
256³ TDM	16,777,216	50,331,646	1.79E-07



Results and Discussion

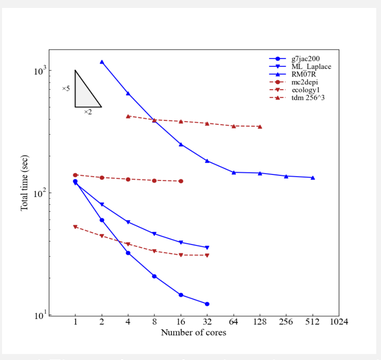
Benchmark test results

- **SuperLU-DIST**
  - Process-grid affects on parallel performance
  - 1.7x reduction in total computing time by optimizing parameters



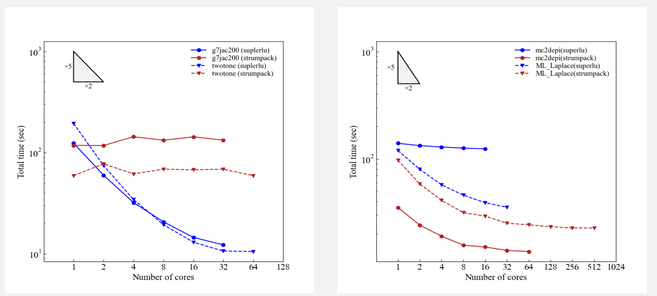
  - **None-zero ratio effect**

    Parallelization performance of lower non-zero ratio is 3~9 times worse than that of dense sparse matrix



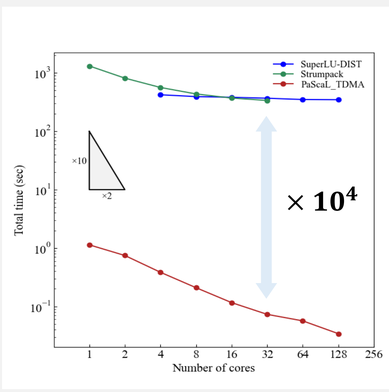
	r/c	total time	nnz ratio	speed up (2^5)
g7jac200	59,310	12.25377	2.38E-04	10.1
RM07R	381,689	132.56982	2.57E-04	8.1
ML_Laplace	377,002	35.44098	1.95E-04	3.4
mc2depi	525,825	124.06596	7.60E-06	1.1
ecology1	1,000,000	30.64655	5.00E-06	1.7
256³ TDM	16,777,216	348.09223	1.79E-07	1.2

- **SuperLU-dist and STRUMPACK**
  - SuperLU_dist tends to outperform STRUMPACK for the matrices that have broader distribution near the diagonal components.



(a) SuperLU > STRUMPACK (b) SuperLU < STRUMPACK

- **Comparison of three solvers: tridiagonal matrix (256³ TDM)**



  - PaScaL_TDMA outperforms 10^4x faster than two baseline solvers for performing tridiagonal matrix computations .
  - Parallel efficiency of PaScaL_TDMA remains stable: Continuous reduction on execution time as increasing cores

    (SuperLU-DIST: over 16 ncpus, STRUMPACK: 64 ncpus)

Conclusion and Plans

- **Parallel performance benchmark**
  - Performance benchmarks highlight the significance of specialized solvers for target problem and specific matrix structure.
  - PaScaL_TDMA that is dedicated for distributed tridiagonal systems, is much faster the other two general and versatile sparse matrix solvers.
- **Strong/weak scalability test planned**
  - Compare strong/weak scalability of three linear solvers until thousands of compute nodes on KISTI Nurion system.
  - Extend the test cases to high-dimensional sparse matrices for real applications.