Interfacing Epetra to the RSB sparse matrix format for shared-memory performance

Michele Martone

High Level Support Team
Max Planck Institute for Plasma Physics
Garching bei München, Germany

5th European Trilinos User Group Meeting
Garching bei München, Germany
April 19, 2016
### Background: sparse kernels can be performance critical

<table>
<thead>
<tr>
<th>Sparse kernel</th>
<th>Operation definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textit{SpMV}: Matrix-Vector Multiply \n\textit{SpMV-T}: (Transposed)</td>
<td>“( y \leftarrow \beta y + \alpha A x )” \n“( y \leftarrow \beta y + \alpha A^T x )”</td>
</tr>
<tr>
<td>\textit{SpSV}: Triangular Solve \n\textit{SpSV-T}: (Transposed)</td>
<td>“( x \leftarrow \alpha L^{-1} x )” \n“( x \leftarrow \alpha L^{-T} x )”</td>
</tr>
</tbody>
</table>

- **Epetra\_CrsMatrix** can rely on Intel MKL
- **CRS (Compressed Row Storage)** versatile and straightforward

---

\( A \) is a sparse matrix, \( L \) sparse triangular, \( x, y \) vectors or multi-vectors, \( \alpha, \beta \) scalars.
Background: optimized sparse matrix classes

- OSKI (Optimized Sparse Kernels Interface) library, by R.Vuduc
  - kernels for iterative methods, e.g. SpMV, SpSV
  - autotuning framework
- OSKI + Epetra = class Epetra_OskiMatrix, by I.Karlin
Background: librsb

- a stable “Sparse BLAS”-like kernels library:
  - Sparse BLAS API (blas_sparse.h)
  - own API (rsb.h)
- LGPLv3 licensed
- librsb-1.2 at http://librsb.sourceforge.net/
- OpenMP threaded
- C/C++/FORTRAN\(^2\) interface

\(^2\)ISO-C-BINDING for own and F90 for Sparse BLAS
What can librsb do

- can be multiple times faster than Intel MKL’s CRS in\(^3\)
  - symmetric/transposed sparse matrix multiply (SpMV)
  - multi-RHS (SpMM)
  - large matrices
- threaded SpSV/SpSM (sparse triangular solve)
- iterative methods ops: scaling, extraction, update, ...
- usable from GNU Octave (sparsersb plugin+keyword)

\(^3\)Please see:
Auto-tuning shared memory parallel Sparse BLAS operations in librsb-1.2
Poster presented at Sparse Solvers for Exascale: From Building Blocks to Applications, Greifswald, Germany
http://hdl.handle.net/11858/00-001M-0000-0029-A8D3-E
Focus of this talk: class Epetra_RsbMatrix

- librsb + Epetra =
  class Epetra_RsbMatrix: public Epetra_CrsMatrix

- analog to class Epetra_OskiMatrix
RSB sparse matrix format: Recursive Sparse Blocks

- CSR (=CRS) and COO blocks
- 32 or 16 (sHort) bit indices
- with fewer, or more than average nonzeroes
- inter-block Z-Morton order
- adaptive (cache blocking) to cache size + threads count

Figure: RSB instance of bayer02
Example results\textsuperscript{4}: RSB vs MKL CRS Speedup, \textit{SymSpMV}

A matrix per sector, four BLAS types on 2x8 threaded Sandy Bridge

\textsuperscript{4}From poster \textit{Auto-tuning shared memory parallel Sparse BLAS operations in librsb-1.2, http://hdl.handle.net/11858/00-001M-0000-0029-A8D3-E}
RSB and autotuning

- similar concept as in Epetra_OskiMatrix
- empirical autotuning:

```c
A_Rsb.tune_spmm(... transA, alpha, nrhs,
    order, B, ldB, beta, C, ldC);
```
- adapts data structure to operation
- tuning can cost e.g. $10^2 \approx 10^3$ operations
- iterated \texttt{faster} operations shall \texttt{amortize} tuning cost
Example of autotuning

5

Sample instances of symmetric \textit{audikw.1} (ca. 1M \times 1M, 39M nonzeroes), node with 256 KB sized L2 cache, 16 threads.

Left (625 blocks, avg 491 KB) before tuning.

Middle tuning merged blocks (271, avg 1133 KB).

Right tuning subdivided blocks (1319, avg 233 KB).

\textsuperscript{5}From poster \textit{Auto-tuning shared memory parallel Sparse BLAS operations in librsb-1.2}, http://hdl.handle.net/11858/00-001M-0000-0029-A8D3-E
Example results\textsuperscript{6}: SymSpMV autotuning speedup: 19 $\approx$ 23%

\textsuperscript{6}From poster *Auto-tuning shared memory parallel Sparse BLAS operations in librsb-1.2*, http://hdl.handle.net/11858/00-001M-0000-0029-A8D3-E
Example results\textsuperscript{7}: Blocking operations favours multi-RHS

Currently: reuse of sparse blocks in \textit{SpMM} (and can be improved)

\textsuperscript{7}From poster \textit{Auto-tuning shared memory parallel Sparse BLAS operations in librsb-1.2}, http://hdl.handle.net/11858/00-001M-0000-0029-A8D3-E
What is librsb being used for?

- **Plasma Physics**
  A.K. Stegmeir
  GRILLIX: A 3D turbulence code for magnetic fusion devices based on Field line map
  Max-Planck-Institut fuer Plasmaphysik, Garching, Germany
  http://dx.doi.org/10.17617/2.2085490

- **Environmental Modelling**
  P.A. Browne, S. Wilson
  A simple method for integrating a complex model into an ensemble data assimilation system using MPI
  Environmental Modelling & Software, Volume 68, June 2015, Pages 122-128
  http://dx.doi.org/10.1016/j.envsoft.2015.02.003

- **Machine Learning**
  Fast Matrix-vector Multiplications for Large-scale Logistic Regression on Shared-memory Systems
  M-C. Lee, W-L. Chiang, C-J. Lin; National Taiwan University
  15th IEEE International Conference on Data Mining (ICDM 2015), Atlantic City, NJ, U.S.A., November 14-17, 2015
  https://www.csie.ntu.edu.tw/~cjlin/papers/multicore_liblinear_icdm.pdf

- **Singular Value Decomposition, Big Data**
  High-Performance Algorithms for Large-Scale Singular Value Problems and Big Data Applications
  L. Wu, A. Stathopoulos, College of William and Mary, Williamsburg, Virginia, USA
Prototype class Epetra_RsbMatrix

Epetra_CsrMatrix A_Csr(...);
...
/* Populate, set options. */
...
Epetra_RsbMatrix A_Rsb(A_Csr);
/* just as Epetra_CsrMatrix: */
A_Rsb.Multiply(...);
A_Rsb.Solve(...);
A_Rsb.ReplaceMyValues(...);
A_Rsb.ExtractDiagonalCopy(...);
/* ...and so on. */

/* But additionally: */
A_Rsb.tune_spmm(... transA, alpha, nrhs,
    order, B, ldB, beta, C, ldC);
A_Rsb.tune_spsm(... transA, alpha, nrhs,
    order, B, ldB, beta, C, ldC);
Backbone of class Epetra_RsbMatrix

- **class Rsb_Matrix**: a proper C++ interface to librsb
- soon part of librsb
- base class providing many internals to `class Epetra_RsbMatrix`
Differences with OSKI/Epetra_OskiMatrix

Epetra_RsbMatrix

- needs no install time tuning
- memory occupation:
  the three arrays of COO for $i, j, val$ (so, no fill-in) and the quad-tree structure, $O(\text{nonzeroes})$
- works with Epetra_Vector and Epetra_MultiVector (faster if ConstantStride() == true in multi-RHS)
- offers nearly all operations of what class Epetra_CrsMatrix does
Differences with Epetra\_CrsMatrix

- construction can cost time of a dozen of SpMV
- access / modify functions:
  ReplaceMyValues(), SumIntoMyValues(),
  ExtractDiagonalCopy(), ReplaceDiagonalValues()...
  cost a little more (need tree traversal)
class Epetra_RsbMatrix: Conclusions and outlook

- transparent access to OpenMP parallel kernels of librsb
- best for symmetric large matrices, multi-RHS
- not really reliant on class Epetra_CrsMatrix (so far, a transitory solution)
- potential for:
  - distributed matrices
  - Tpetra,
    64 bit indices,
    further numerical types (e.g. long long double, ...)
- expertise of Trilinos developers beneficial here!