Solving sparse linear systems of equations using CAF

Ambra Abdullahi Hassan [a] Valeria Cardellini [a] and Salvatore Filippone [b]

[a] *University of Rome Tor Vergata*
[b] *Cranfield University*

Keywords. CAF, MPI, numerical linear algebra, sparse linear systems

1. Abstract

Fortran has been and still is a dominant language in High Performance Computing. In past decades, Fortran has greatly developed, and now supports class abstraction, object-oriented programming, pure functions an coarrays. Fortran Coarrays (CAF) has been part of the Fortran standard since Fortran 2008 and it provides a syntactic extension of Fortran to support parallel programming. Coarrays are fully supported by Cray and Intel compilers. Although GNU compilers provide only partial coarray support (only single image), the openCoarrays project provides two coarray transport layers based on MPI and on GASNet [1]. Following Partitioned Global Adress Space programming model, with CAF the program is replicated among a certain number of images, executing asynchronously. A variable can be declared as a coarray, through the use of codimension and coindices: it implies that variable is accessible from all images. Combining, elegancy and simplicity, CAF allows easily for one-sided communications, potentially reducing shynchronization and code complexity. CAF is a languade-based approach, which means it is bounds on specific compilers: when a good implementation is available, compiler can concurrently consider serial aspects and communication for optimization. Additionally, if vendors compiler provides support for OpenAcc, the user can easily get a dual benefit form the combined use of CAF and GPUs. MPI (Message Passing Interface) defines a suite of functions for communication exchange between processes. Differently from CAF, it is library-based: it means that while independent from a specific compiler, the user is bound to the vendor's fine-tuning of the MPI library. Although MPI is the de facto standard for parallel programs running on distributed memory systems and few scientific softwares are written in CAF, we think that many scientific applications could benefit from the use of CAF. The aim of this paper is to present the migration from MPI to CAF of two libraries (PSBLAS [2] and MLD2P4[3]) for the solution of large systems of equations using iterative methods and preconditioners. We point out that solution of large sparse linear systems is the most computationally expensive part of many scientific applications, thus the study of the impact of CAF on performances and quality of software on this kind of problems could be extremely valuable. PSBLAS is a library of Basic Linear Algebra Subroutines that implements iterative solvers for sparse linear sys-

tems and includes subroutines for multiplying sparse matrices by dense matrices, solving sparse triangular systems, and preprocessing sparse matrices, as well as additional routines for dense matrix operations. MLD2P4 (Multi-Level Domain Decomposition Parallel Preconditioners) is a package based on PSBLAS and it implements various algebraic multi-level preconditioners. They are both implemented in Fortran 2003 and the current versions use message passing to address a distributed memory execution model. We converted the code gradually from MPI to coarrays, thus having MPI and CAF coexisting in the same code. We detected three communication patterns that had to be modified: the halo exchange, the collective subroutines and the point-to-point communication. Halo exchange is performed everytime a parallel matrix-vector multiplication is computed, and for this reason it is probably the most expensive communication in solving sparse linear systems. Since most of communication in MLD2P4 is based on PSBLAS routines, translation of MLD2P4 after PSBLAS comes with little effort. Several iterative solvers and preconditioners are implemented in PSBLAS and MLD2P4 using MPI. This allows us to explore different communication patterns involved in solution of linear systems. For example, it is possible to compare performances of multigrid and domain decomposition methods in MPI and in CAF. Few attempts to implement multigrid solvers in CAF have been made, for example in [4] (in which early versions of CAF and MPI have been used). In [5] a comparison between MPI3 and CAF is presented for spectral techniques, multigrid techniques and for applications drawn from computational fluid dynamics. It shows that Cray implementations of MPI3 and CAF performances in message passing are comparable (with CAF slightly outperforms MPI3 in some cases), and it points out the advantages of CAF when it comes to ease of implementation. Preliminary tests have been carried for the halo exchange in PSBLAS, using GNU compiler and opencoarrays library, showing that performances of CAF and MPI are comparable. Further tests will show if there is any change in performance in translating collective subroutines and point-to-point communication, as well as to show whether any gain in performance is possible using CAF with other compilers. Additionally, since halo exchanges ,collective subroutines and point-to-point communications are all common pattern exchange in parallel softwares, we describe the best-usage startegies implemented in PSBLAS and MLD2P4.

References

[1] Fanfarillo, Alessandro, et al. "OpenCoarrays: open-source transport layers supporting coarray Fortran compilers." Proceedings of the 8th International Conference on Partitioned Global Address Space Programming Models. ACM, 2014.

[2] Filippone, Salvatore, and Michele Colajanni. "PSBLAS: A library for parallel linear algebra computation on sparse matrices." ACM Transactions on Mathematical Software (TOMS) 26.4 (2000): 527-550.

[3] DAmbra, Pasqua, Daniela Di Serafino, and Salvatore Filippone. "MLD2P4: a package of parallel algebraic multilevel domain decomposition preconditioners in Fortran 95." ACM Transactions on Mathematical Software (TOMS) 37.3 (2010): 30.

[4] Numrich, Robert W., John Reid, and Kieun Kim. "Writing a multigrid solver using co-array fortran." International Workshop on Applied Parallel Computing. Springer Berlin Heidelberg, 1998.

[5] Garain, Sudip, Dinshaw S. Balsara, and John Reid. "Comparing Coarray Fortran (CAF) with MPI for several structured mesh PDE applications." Journal of Computational Physics 297 (2015): 237-253.