Recent Developments in Iterative Methods for Reducing Synchronization

Qinmeng Zou and Frédéric Magoulès
CentraleSupélec, Université Paris-Saclay
91190 Gif-sur-Yvette, France
frederic.magoules@hotmail.com

Abstract—On modern parallel architectures, the cost of synchronization among processors can often dominate the cost of floating-point computation. Several modifications of the existing methods have been proposed in order to keep the communication cost as low as possible. This paper aims at providing a brief overview of recent advances in parallel iterative methods for solving large-scale problems. We refer the reader to the related references for more details on the derivation, implementation, performance, and analysis of these techniques.

Keywords-communication-avoiding algorithms; s-step iterative methods; pipelined Krylov subspace methods; asynchronous iterations.

I. Introduction

The performance of iterative methods depends on the amount of arithmetic operations and the amount of data movements. The latter depends further on the latency cost and the bandwidth cost. Early researches mainly focused on the arithmetic operations in both the sequential and parallel cases. On modern computer architectures, latency cost is much more significant than bandwidth cost, and bandwidth cost is much more significant than computation cost. The gaps are expected to increase in the future.

Parallelization of iterative methods for solving large-scale problems is constrained by synchronization which leads to processor idle time. These methods consist of the following three basic operations: sparse matrix-vector multiplication (SpMV), dot products, and AXPY (α times x plus y) operations

$$y \leftarrow \alpha x + y.$$

AXPY requires only local operations and thus does not affect parallel efficiency. SpMV often requires communication among neighbors, which depends on the distribution of nonzero values. Dot products require global synchronization before and after a computation. The bottleneck that comes from these operations can be partially overcome by using recent techniques. Roughly speaking, the communication in dot products can be reduced by simultaneously constructing multiple direction vectors or being overlapped with other operations; if there is no dot product operation, then the synchronization in SpMV can be eliminated by only using existing data for the next computation instead of waiting for a complete data transmission.

We show in this paper how synchronization points could be reduced and highlight some recent advances on promising techniques. Section II presents the s-step iterative methods. Section III presents the pipelined Krylov subspace methods. Section IV summarizes recent developments in asynchronous iterations. Section V provides a brief overview of other popular methods. Finally, we draw a conclusion in Section VI.

II. s-Step Iterative Methods

An early paper describing the idea of s-step iterations can be traced to 1950, as quoted in [1], when Birman [2] presented an s-gradient method in a Russian paper. A later paper [3] was published on the method that aims at reducing the number of synchronization operations for the conjugate gradient (CG) method [4], often called Chronopoulos-Gear CG. The thesis written by Hoemmen [5] gives an excellent historical perspective on this topic. We refer the reader to [5] and the references therein for the developments before 2010.

The key feature in s-step iterative methods is to perform $\mathcal{O}(s)$ computation steps of the classical algorithms for each communication step, thus allowing to reduce the number of synchronization points by a factor of s. Krylov subspace methods (KSMs) are often the methods of choice for solving eigenvalue and linear system problems (see, e.g., [6]), which can be viewed as projection techniques. Given two subspace $\mathcal{K}_m$ and $\mathcal{L}_m$ where

$$\mathcal{K}_m(A, v) = \text{span}\left\{v, \, Av, \, \ldots, \, A^{m-1}v\right\},$$

KSMs search solution vectors in $\mathcal{K}_m$ such that residuals are orthogonal to $\mathcal{L}_m$. The latent bottleneck of successive matrix-vector multiplications in KSMs can be relieved by the "matrix powers kernel" as described in [7]. The global synchronization of dot products in Lanczos-based methods can be reduced by using a Gram matrix [8]. In addition, the GMRES method [9] was improved in [5] by using the tall and skinny QR factorization [10].

Hoemmen [5] discussed the s-step GMRES (see also [11]) and s-step CG. Carson et al. [8] discussed the s-step BICG [12] and s-step BICGSTAB [13]. They also addressed the stability issues relating to the basis construction. The term "s-step" can be often replaced by "communication-avoiding (CA)". Although the latter is commonly used in the literature, as mentioned in [14], it is slightly dubious since the communication cost is only partially reduced rather than avoided. Ballard et al. [15] summarized theoretical bounds on communication for techniques used in the s-step algorithms.

In finite precision cases, s-step formulation with monomial basis can lead to stability issues, which have been

discussed in many references [16], [17], [5], [8]. The maximum attainable accuracy of s-step KSMs and the residual replacement strategy were discussed in [18].

More recently, a new variant proposed by Imberti and Erhel [19] is to use an increasing sequence of block sizes in s-step GMRES instead of a fixed size. On the other hand, block coordinate descent (BCD) methods have been successfully used in machine learning. Devarakonda et al. [20] extended the s-step methods to the primal and dual BCD methods for solving regularized least-square problems. The new methods are called CA-BCD and CA-BDCD, respectively, that can, like other s-step iterative methods, reduce the latency cost by a factor of s but increase computation and bandwidth costs.

III. Pipelined Krylov Subspace Methods

The pipelined iterative methods aim at overlapping expensive communication phases with computations. Some approaches for this purpose applied to CG and GMRES appeared in the mid-1990s, see [21], [22]. Ghysels et al. [23], [24] introduced the modern pipelined Krylov subspace methods. It is interesting to note that the term "pipelined" has often been replaced by "communication-hiding" in their camp, just like the alternatives mentioned in the preceding section.

Ghysels et al. [23] proposed the pipelined GMRES method. Ghysels and Vanroose [24] proposed the pipelined CG method. Their work has promoted other promising ideas. For example, Sanan et al. [25] discussed some pipelined variants of flexible Krylov subspace methods. Cools and Vanroose [26] presented a general framework for pipelined methods, from which the pipelined BICGSTAB method was successfully derived.

In finite precision arithmetic, Cools et al. [14] discussed the the effect of local rounding error propagation on the maximal attainable accuracy of the pipelined CG method and compared it with the classical CG and Chronopoulos-Gear CG [3]. In a later paper, Cools [27] gave a similar discussion for the pipelined BICGSTAB method. Carson et al. [28] discussed the stability issues for synchronization-reducing algorithms and presented a methodology for the theoretical analysis of some CG variants.

IV. Asynchronous Iterations

Asynchronous iterations were introduced by Chazan and Miranker [29] in 1969, originally called chaotic iterations. The modern mathematical expression was formally introduced in [30], which can be summarized as follows:

$$x_i^{(n+1)} = \begin{cases} f_i\left(x_1^{(\tau_{i,1,n})}, \ldots, x_p^{(\tau_{i,p,n})}\right), & i \in P_n, \\ x_i^{(n)}, & i \notin P_n, \end{cases}$$

where $x_i^{(n)}$ denotes the ith element of the solution vector, $\tau_{i,j,n}$ denotes the iteration number with retards for each element j in each processor i, which is smaller than n, and $P_n \subset \{1, \ldots, p\}$ is a subset of processors. In addition, some multi-stage models and a number of convergence results have been proposed during the past century, we refer the reader to [31], [32] for more details. The key feature in asynchronous iterations is to eliminate waiting times in communication at the expense of more iterations.

The main focus of recent research is asynchronous domain decomposition methods. Chau et al. [33] used the asynchronous Schwarz method for the solution of obstacle problems. Magoulès et al. [34] investigated the asynchronous optimized Schwarz method and provided some convergence results. Magoulès and Venet [35] gave a discussion about the asynchronous substructuring methods. More recently, Yamazaki et al. [36] gave some numerical experiments for the optimized Schwarz method. Theoretical analysis of this approach for various types of subdomains were considered in [37], [38]. On the other hand, asynchronous multisplitting methods were studied in [39], which have been applied to the fluid-structure interaction problem in a recent paper [40].

For the time domain decomposition methods, the derivation of asynchronous waveform relaxation can be found in [31]. Magoulès et al. [41], [42] proposed the asynchronous variant of Parareal algorithm (see also [43], [44]). Another asynchronous time-parallel method based on Laplace transform can be found in [45].

From a computational point of view, the implementation of asynchronous iterative methods requires more than a straightforward update of synchronous versions. Magoulès and Gbikpi-Benissan [46], [47] developed an MPI-based communication library for both synchronous and asynchronous iterative computing. However, the issue of asynchronous convergence detection must be tackled for such kind of libraries. Early work can be found in [48] based on the snapshot algorithm. Magoulès and Gbikpi-Benissan [49] continued this work and proposed several promising variants. Bahi et al. [50] introduced another approach in which the detection process is superimposed onto the asynchronous iterations. Numerical experiments of asynchronous iterative methods for GPU were conducted and described in [51], [52].

V. Other Popular Methods

It is clear that other techniques exist for reducing synchronization in parallel architectures, which have not been reviewed in the previous sections. For example, cyclic gradient iterative methods (see, e.g., [53]) are intrinsically adapted for parallel computing, which can reduce both computation and communication costs. Some techniques (see, e.g. [54]) called improved Krylov methods revolve around the overlap of communication and computation. McInnes et al. [55] proposed hierarchical and nested Krylov subspace methods. Grigori et al. [56] proposed enlarged Krylov subspace methods which can be viewed as a special case of augmented Krylov subspace methods (see, e.g, [57]).

VI. Conclusion

There is much still to understand about synchronization-reducing methods. For example, development of efficient preconditioners for parallel algorithms is still an open question [58]. Theoretical analysis of basis in s-step algorithms requires more work [5]. The loss of orthogonality

in finite precision is also an issue to be tackled [28]. For this reason, we hope that continued contributions could be made on this rapidly growing field in the future.

References

[1] G. E. Forsythe, "On the asymptotic directions of the s-dimensional optimum gradient method," *Numer. Math.*, vol. 11, no. 1, pp. 57–76, 1968.

[2] M. S. Birman, "Some estimates for the method of steepest descent," *Uspekhi Mat. Nauk*, vol. 5(3), no. 37, pp. 152–155, 1950, (in Russian).

[3] A. T. Chronopoulos and C. W. Gear, "s-step iterative methods for symmetric linear systems," *J. Comput. Appl. Math.*, vol. 25, no. 2, pp. 153–168, 1989.

[4] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Natl. Bur. Stand.*, vol. 49, no. 6, pp. 409–436, 1952.

[5] M. Hoemmen, "Communication-avoiding Krylov subspace methods," Ph.D. dissertation, UC Berkeley, 2010.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Johns Hopkins University Press, 2013.

[7] J. W. Demmel, M. Hoemmen, M. Mohiyuddin, and K. Yelick, "Avoiding communication in sparse matrix computations," in *Proceedings of the 22nd International Parallel and Distributed Processing Symposium*. Miami, FL, USA: IEEE, 2008, pp. 1–12.

[8] E. C. Carson, N. S. Knight, and J. W. Demmel, "Avoiding communication in nonsymmetric Lanczos-based Krylov subspace methods," *SIAM J. Sci. Comput.*, vol. 35, no. 5, pp. S42–S61, 2013.

[9] Y. Saad and M. H. Schultz, "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, no. 3, pp. 856–869, 1986.

[10] J. W. Demmel, L. Grigori, M. Hoemmen, and J. Langou, "Communication-optimal parallel and sequential QR and LU factorizations," *SIAM J. Sci. Comput.*, vol. 34, no. 1, pp. A206–A239, 2012.

[11] M. Mohiyuddin, M. Hoemmen, J. W. Demmel, and K. Yelick, "Minimizing communication in sparse matrix solvers," in *Proceedings of the 22nd Conference on High Performance Computing, Networking, Storage and Analysis*. Portland, OR, USA: ACM, 2009, pp. 1–12.

[12] R. Fletcher, "Conjugate gradient methods for indefinite systems," in *Numerical Analysis*. Springer, 1976, pp. 73–89.

[13] H. A. van der Vorst, "Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 13, no. 2, pp. 631–644, 1992.

[14] S. Cools, E. F. Yetkin, E. Agullo, L. Giraud, and W. Vanroose, "Analyzing the effect of local rounding error propagation on the maximal attainable accuracy of the pipelined conjugate gradient method," *SIAM J. Matrix Anal. Appl.*, vol. 39, no. 1, pp. 426–450, 2018.

[15] G. Ballard, E. C. Carson, J. W. Demmel, M. Hoemmen, N. S. Knight, and O. Schwartz, "Communication lower bounds and optimal algorithms for numerical linear algebra," *Acta Numer.*, vol. 23, pp. 1–155, 2014.

[16] W. D. Joubert and G. F. Carey, "Parallelizable restarted iterative methods for nonsymmetric linear systems. Part I: Theory," *Int. J. Comput. Math.*, vol. 44, no. 1-4, pp. 243–267, 1992.

[17] Z. Bai, D. Hu, and L. Reichel, "A Newton basis GMRES implementation," *IMA J. Numer. Anal.*, vol. 14, no. 4, pp. 563–581, 1994.

[18] E. C. Carson, N. Knight, and J. W. Demmel, "A residual replacement strategy for improving the maximum attainable accuracy of s-step Krylov subspace methods," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 1, pp. 22–43, 2014.

[19] D. Imberti and J. Erhel, "Varying the s in your s-step GMRES," *Electron. Trans. Numer. Anal.*, vol. 47, pp. 206–230, 2017.

[20] A. Devarakonda, K. Fountoulakis, J. W. Demmel, and M. W. Mahoney, "Avoiding communication in primal and dual block coordinate descent methods," *SIAM J. Sci. Comput.*, vol. 41, no. 1, pp. C1–C27, 2019.

[21] J. W. Demmel, M. T. Heath, and H. A. van der Vorst, "Parallel numerical linear algebra," *Acta Numer.*, vol. 2, pp. 111–197, 1993.

[22] E. de Sturler and H. A. van der Vorst, "Reducing the effect of global communication in GMRES(m) and CG on parallel distributed memory computers," *Appl. Numer. Math.*, vol. 18, no. 4, pp. 441–459, 1995.

[23] P. Ghysels, T. J. Ashby, K. Meerbergen, and W. Vanroose, "Hiding global communication latency in the GMRES algorithm on massively parallel machines," *SIAM J. Sci. Comput.*, vol. 35, no. 1, pp. C48–C71, 2013.

[24] P. Ghysels and W. Vanroose, "Hiding global synchronization latency in the preconditioned conjugate gradient algorithm," *Parallel Comput.*, vol. 40, no. 7, pp. 224–238, 2014.

[25] P. Sanan, S. M. Schnepp, and D. A. May, "Pipelined, flexible Krylov subspace methods," *SIAM J. Sci. Comput.*, vol. 38, no. 5, pp. C441–C470, 2016.

[26] S. Cools and W. Vanroose, "The communication-hiding pipelined BiCGstab method for the parallel solution of large unsymmetric linear systems," *Parallel Comput.*, vol. 65, pp. 1–20, 2017.

[27] S. Cools, "Analyzing and improving maximal attainable accuracy in the communication hiding pipelined BiCGStab method," *Parallel Comput.*, vol. 86, pp. 16–35, 2019.

[28] E. C. Carson, M. Rozložník, Z. Strakoš, P. Tichý, , and M. Tůma, "The numerical stability analysis of pipelined conjugate gradient methods: Historical context and methodology," *SIAM J. Sci. Comput.*, vol. 40, no. 5, pp. A3549–A3580, 2018.

[29] D. Chazan and W. L. Miranker, "Chaotic relaxation," *Linear Algebra Appl.*, vol. 2, no. 2, pp. 199–222, 1969.

[30] G. M. Baudet, "Asynchronous iterative methods for multiprocessors," *J. ACM*, vol. 25, no. 2, pp. 226–244, 1978.

[31] A. Frommer and D. B. Szyld, "On asynchronous iterations," *J. Comput. Appl. Math.*, vol. 123, no. 1-2, pp. 201–216, 2000.

[32] J. M. Bahi, S. Contassot-Vivier, and R. Couturier, *Parallel Iterative Algorithms: From Sequential to Grid Computing.* CRC Press, 2007.

[33] M. Chau, T. Garcia, and P. Spiteri, "Asynchronous Schwarz methods applied to constrained mechanical structures in grid environment," *Adv. Eng. Softw.*, vol. 74, pp. 1–15, 2014.

[34] F. Magoulès, D. B. Szyld, and C. Venet, "Asynchronous optimized Schwarz methods with and without overlap," *Numer. Math.*, vol. 137, no. 1, pp. 199–227, 2017.

[35] F. Magoulès and C. Venet, "Asynchronous iterative substructuring methods," *Math. Comput. Simul.*, vol. 145, pp. 34–49, 2018.

[36] I. Yamazaki, E. Chow, A. Bouteiller, and J. J. Dongarra, "Performance of asynchronous optimized Schwarz with one-sided communication," *Parallel Comput.*, vol. 86, pp. 66–81, 2019.

[37] J. C. Garay, F. Magoulès, and D. B. Szyld, "Convergence of asynchronous optimized Schwarz methods in the plane," in *Domain Decomposition Methods in Science and Engineering XXIV*, ser. Lecture Notes in Computer Science and Engineering. Springer, 2018, pp. 333–341.

[38] J. C. Garay, F. Magoulès, and D. B. Szyld, "Optimized Schwarz method for Poisson's equation in rectangular domains," in *Domain Decomposition Methods in Science and Engineering XXIV*, ser. Lecture Notes in Computer Science and Engineering. Springer, 2018, pp. 533–541.

[39] J. M. Bahi, J.-C. Miellou, and K. Rhofir, "Asynchronous multisplitting methods for nonlinear fixed point problems," *Numer. Algorithms*, vol. 15, no. 3, pp. 315–345, 1997.

[40] V. Partimbene, T. Garcia, P. Spiteri, P. Marthon, and L. Ratsifandrihana, "Asynchronous multi-splitting method for linear and pseudo-linear problems," *Adv. Eng. Softw.*, vol. 133, pp. 76–95, 2019.

[41] F. Magoulès, G. Gbikpi-Benissan, and Q. Zou, "Asynchronous iterations of Parareal algorithm for option pricing models," *Mathematics*, vol. 6, no. 4, pp. 1–18, 2018.

[42] F. Magoulès and G. Gbikpi-Benissan, "Asynchronous Parareal time discretization for partial differential equations," *SIAM J. Sci. Comput.*, vol. 40, no. 6, pp. C704–C725, 2018.

[43] Q. Zou, G. Gbikpi-Benissan, and F. Magoulès, "Asynchronous Parareal algorithm applied to European option pricing," in *Proceedings of the 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science.* Anyang, China: IEEE, 2017, pp. 37–40.

[44] Q. Zou, G. Gbikpi-Benissan, and F. Magoulès, "Asynchronous communications library for the parallel-in-time solution of Black-Scholes equation," in *Proceedings of the 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science.* Anyang, China: IEEE, 2017, pp. 45–48.

[45] F. Magoulès and Q. Zou, "Asynchronous time-parallel method based on Laplace transform," *preprint available at arXiv:1909.01473*, 2019.

[46] F. Magoulès and G. Gbikpi-Benissan, "JACK: An asynchronous communication kernel library for iterative algorithms," *J. Supercomput.*, vol. 73, no. 8, pp. 3468–3487, 2017.

[47] F. Magoulès and G. Gbikpi-Benissan, "JACK2: An MPI-based communication library with non-blocking synchronization for asynchronous iterations," *Adv. Eng. Softw.*, vol. 119, pp. 116–133, 2018.

[48] S. A. Savari and D. P. Bertsekas, "Finite termination of asynchronous iterative algorithms," *Parallel Comput.*, vol. 22, no. 1, pp. 39–56, 1996.

[49] F. Magoulès and G. Gbikpi-Benissan, "Distributed convergence detection based on global residual error under asynchronous iterations," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 4, pp. 819–829, 2018.

[50] J. M. Bahi, S. Contassot-Vivier, R. Couturier, and F. Vernier, "A decentralized convergence detection algorithm for asynchronous parallel iterative algorithms," *IEEE Trans. Parallel Distrib. Syst.*, vol. 16, no. 1, pp. 4–13, 2005.

[51] H. Anzt, S. Tomov, J. J. Dongarra, and V. Heuveline, "A block-asynchronous relaxation method for graphics processing units," *J. Parallel Distrib. Comput.*, vol. 73, no. 12, pp. 1613–1626, 2013.

[52] E. Chow, H. Anzt, and J. J. Dongarra, "Asynchronous iterative algorithm for computing incomplete factorizations on GPUs," in *Proceedings of the 30th International Conference, ISC High Performance 2015.* Springer, 2015, pp. 1–16.

[53] Q. Zou and F. Magoulès, "Reducing the effect of global synchronization in delayed gradient methods for symmetric linear systems," *Adv. Eng. Softw.*, submitted.

[54] T.-X. Gu, X.-Y. Zuo, L.-T. Zhang, W.-Q. Zhang, and Z.-Q. Sheng, "An improved bi-conjugate residual algorithm suitable for distributed parallel computing," *Appl. Math. Comput.*, vol. 186, no. 2, pp. 1243–1253, 2007.

[55] L. C. McInnes, B. F. Smith, H. Zhang, and R. T. Mills, "Hierarchical Krylov and nested Krylov methods for extreme-scale computing," *Parallel Comput.*, vol. 40, no. 1, pp. 17–31, 2014.

[56] L. Grigori, S. Moufawad, and F. Nataf, "Enlarged Krylov subspace conjugate gradient methods for reducing communication," *SIAM J. Matrix Anal. Appl.*, vol. 37, no. 2, pp. 744–773, 2016.

[57] V. Simoncini and D. B. Szyld, "Recent computational developments in Krylov subspace methods for linear systems," *Numer. Linear Algebra Appl.*, vol. 14, no. 1, pp. 1–59, 2007.

[58] E. C. Carson, "Communication-avoiding Krylov subspace methods in theory and practice," Ph.D. dissertation, UC Berkeley, 2015.