Karlsruhe Institute of Technology

Mixed Precision Error Correction Methods for Linear Systems: Convergence Analysis based on Krylov Subspace Methods

Hartwig Anzt
Vincent Heuveline
Björn Rocker

No. 2010-02

Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)

KIT – University of the State of Baden-Wuerttemberg and
National Research Center of the Helmholtz Association

www.emcl.kit.edu

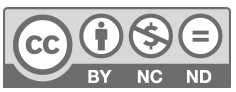
Preprint Series of the Engineering Mathematics and Computing Lab (EMCL)
ISSN 2191–0693
No. 2010-02

Impressum

Karlsruhe Institute of Technology (KIT)
Engineering Mathematics and Computing Lab (EMCL)

Fritz-Erler-Str. 23, building 01.86
76133 Karlsruhe
Germany

KIT – University of the State of Baden Wuerttemberg and
National Laboratory of the Helmholtz Association



Mixed Precision Error Correction Methods for Linear Systems: Convergence Analysis based on Krylov Subspace Methods

Hartwig Anzt, Björn Rocker and Vincent Heuveline

Karlsruhe Institute of Technology (KIT)
Institute for Applied and Numerical Mathematics 4
Fritz-Erler-Str. 23
76133 Karlsruhe, Germany

hartwig.anzt@kit.edu, bjoern.rocker@kit.edu,
vincent.heuveline@kit.edu

Abstract. The convergence analysis of Krylov subspace solvers usually provides an estimation for the computational cost. Exact knowledge about the convergence theory of error correction methods using different floating point precision formats would enable to determine a priori whether the implementation of a mixed precision error correction solver using a certain Krylov subspace method as error correction solver outperforms the plain solver in high precision.
This paper reveals characteristics of mixed precision error correction methods using Krylov subspace methods as inner solver.

1 Introduction

In computational science, the acceleration of linear solvers is of high interest. Currently coprocessor technologies like GPUs offer outstanding single precision performance. To exploit this computation power without sacrificing the accuracy of the result which is often needed in double precision, numerical algorithms have to be designed that use different precision formats.

Especially the idea of using a lower precision than working precision within the error correction solver of an error correction method has turned out to improve the computational cost of the solving process for many linear problems without sacrificing the accuracy of the final result [1], [2], [3] and [4]. In many of these papers, this approach is referred to as "mixed precision iterative refinement method". Although this notation is widespread, we do think that the name "error correction solver" emphasizes the error-correcting character of the algorithm. For this reason, we will use both terms in our paper, but usually the latter one. Although the free choice of the error correction solver type offers a large variety of error correction methods, this work is focused on Krylov subspace methods, since they are used for many problems.

The combination of a given outer stopping criterion for the error correction method and a chosen inner stopping criterion for the error correction solver has strong influence on the characteristics of the solver. A small quotient between outer and inner stopping criterion leads to a high number of inner iterations performed by the error correction solver and a low number of outer iterations performed by the error correction method. A large quotient leads to a low number of inner iterations but a higher number of outer iterations, and therefore to a higher number of restarts of the inner solver.

To optimize this trade-off, exact knowledge about the characteristics of both the solver and the linear system is necessary. Still, a theoretical analysis is difficult, since the convergence analysis of the error correction solver is affected when using different precision formats within the method.

This paper presents results of numerical analysis concerning error correction methods based on Krylov subspace solvers. First the general mathematical background of error correction methods is drafted, then the mixed precision approach is introduced and analyzed with respect to the theoretical convergence rate. A conclusion and prospects to future work complete the paper.

2 Mathematical Background

2.1 Error Correction Methods

The motivation for the error correction method can be obtained from Newton's method. Here f is a given function and x_i is the solution in the $i-th$ step:

$$x_{i+1} = x_i - (\nabla f(x_i))^{-1} f(x_i). \tag{1}$$

This method can be applied to the function $f(x) = b - Ax$ with $\nabla f(x) = A$, where $Ax = b$ is the linear system that should be solved.

By defining the residual $r_i := b - Ax_i$, one obtains

$$\begin{aligned} x_{i+1} &= x_i - (\nabla f(x_i))^{-1} f(x_i) \\ &= x_i + A^{-1}(b - Ax_i) \\ &= x_i + A^{-1} r_i. \end{aligned}$$

Denoting the solution update with $c_i := A^{-1} r_i$ and using an initial guess x_0 as starting value, an iterative algorithm can be defined, where any linear solver can be used as error correction solver.

1: initial guess as starting vector: x_0
2: compute initial residual: $r_0 = b - Ax_0$
3: **while** ($\| Ax_i - b \|_2 > \varepsilon \| r_0 \|$) **do**
4: $\quad r_i = b - Ax_i$
5: $\quad$ solve error correction equation: $Ac_i = r_i$
6: $\quad$ update solution: $x_{i+1} = x_i + c_i$
7: **end while**

Algorithm 1: Error Correction Method

In each iteration, the inner correction solver searches for a c_i such that $Ac_i = r_i$ and the solution approximation is updated by $x_{i+1} = x_i + c_i$.

2.2 Error Correction Solver

Due to the fact that the error correction method makes no demands on the inner solver, any linear solver can be chosen. Still, especially the Krylov subspace methods have turned out to be an adequate choice for many cases. These provide an approximation of the residual error iteratively in every computation loop, which can efficiently be used to control the stopping criterion of the error correction solver.

2.3 Convergence Analysis of Error Correction Methods

If we denote the residual in the ith step as

$$r_i = b - Ax_i$$

we can analyze the improvement associated with one iteration loop of the error correction method.

Applying a solver to the error correction equation $Ac_i = r_i$ which generates a solution approximation with a relative residual error of at most $\varepsilon_{inner} \parallel r_i \parallel$, we get an error correction term c_i, fulfilling

$$r_i - Ac_i = d_i,$$

where d_i is the residual of the correction solver with the property

$$\parallel d_i \parallel \leq \varepsilon_{inner} \parallel r_i \parallel .$$

In the case of using a Krylov subspace method as inner solver, the threshold $\varepsilon_{inner} \parallel r_i \parallel$ can be chosen as residual stopping criterion.

Updating the solution $x_{i+1} = x_i + c_i$, we can obtain the new residual error term

$$\begin{aligned} \parallel r_{i+1} \parallel &= \parallel b - Ax_{i+1} \parallel \\ &= \parallel b - A(x_i + c_i) \parallel \\ &= \parallel \underbrace{b - Ax_i}_{=r_i} \underbrace{-Ac_i}_{=d_i - r_i} \parallel \\ &= \parallel d_i \parallel \leq \varepsilon_{inner} \parallel r_i \parallel . \end{aligned}$$

Hence, the accuracy improvements obtained by performing one iteration loop equal the accuracy of the residual stopping criterion of the error correction solver. Using this fact, we can prove by induction, that after i iteration loops, the residual r_i fulfills

$$\parallel r_i \parallel \leq \varepsilon_{inner}^{i} \parallel r_0 \parallel . \tag{2}$$

If we are interested in the number i of iterations that is necessary to get the residual error term r_i below a certain threshold

$$\| r_i \| \leq \varepsilon \| r_0 \|$$

we use the properties of the logarithm and estimate

$$\begin{aligned} \| r_i \| &\leq \varepsilon \| r_0 \| \\ \varepsilon_{inner}^i \| r_0 \| &\leq \varepsilon \| r_0 \| \\ \varepsilon_{inner}^i &\leq \varepsilon \\ i &\geq \frac{\log \varepsilon}{\log \varepsilon_{inner}}. \end{aligned}$$

Since i has to be an integer, we use the Gaussian ceiling function and obtain

$$i = \left\lceil \frac{\log(\varepsilon)}{\log(\varepsilon_{inner})} \right\rceil \tag{3}$$

for the number of outer iterations that is necessary to guarantee an accuracy of $\| r_i \| \leq \varepsilon \| r_0 \|$.

3 Mixed Precision Error Correction Solvers

3.1 Mixed Precision Approach

The underlying idea of mixed precision error correction methods is to use different precision formats within the algorithm of the error correction method, updating the solution approximation in high precision, but computing the error correction term in lower precision. This approach was also suggested by [1], [2], [3] and [4].

Using the mixed precision approach to the error correction method, we have to be aware of the fact that the residual error bound of the error correction solver may not exceed the accuracy of the lower precision format. Furthermore, each error correction produced by the inner solver in lower precision cannot exceed the data range of the lower precision format. This means that the smallest possible error correction is the smallest number ϵ_{low}, that can be represented in the lower precision. Thus, the accuracy of the final solution cannot exceed ϵ_{low} either. This can become a problem when working with very small numbers, because then the solution correction terms can not be denoted in low precision, but in most cases, the problem can be avoided by converting the original values to a lower order of magnitude. If the final accuracy does not exceed the smallest number that can be represented in the lower precision format, the mixed precision error correction method gives exactly the same solution approximation as if the solver was performed in the high precision format.

When comparing the algorithm of an error correction solver using a certain Krylov subspace solver as error correction solver to the plain solver, we realize,

that the error correction method has more computations to execute due to the additional residual computation, solution updates and typecasts.

The goal is to analyze in which cases the mixed precision error correction method outperforms the plain solver in high precision. Obviously this is the case if the additional operations (denoted with K) are overcompensated by the cheaper execution of the iterative solver in low precision. Using an explicit residual computation the computational costs of K is in the magnitude of the matrix-vector multiplication. In case of an iterative update for the residual, the complexity is even lower.

3.2 Convergence Analysis of Mixed Precision Approaches

When discussing the convergence of the error correction methods in section 2.3, we derived a model for the number of outer iterations that are necessary to obtain a residual error below a certain residual threshold $\varepsilon \parallel r_0 \parallel_2$. Having a relative residual stopping criterion ε_{inner} of the Krylov subspace solver used as error correction solver, we need to perform (3)

$$i = \left\lceil \frac{\log(\varepsilon)}{\log(\varepsilon_{inner})} \right\rceil$$

iterations to obtain an approximation x_i which fulfills

$$\parallel r_i \parallel_2 = \parallel b - Ax_i \parallel_2 \leq \varepsilon \parallel b - Ax_0 \parallel_2 = \varepsilon \parallel r_0 \parallel_2 .$$

If we use the error correction technique in mixed precision, we have to modify this convergence analysis due to the floating point arithmetic. In fact, two phenomena may occur that require additional outer iterations.

1. Independently of the type of the inner error correction solver, the low precision format representations of the matrix A and the residual r_i contain representation errors due to the floating point arithmetic. These rounding errors imply that the error correction solver performs the solving process to a perturbed system $(A+\delta A)c_i = r_i + \delta r_i$. Due to this fact, the solution update c_i gives less improvement to the outer solution than expected. Hence, the convergence analysis of the error correction method has to be modified when using different precision formats. To compensate the smaller improvements to the outer solution, we have to perform additional outer iterations.
2. When using a Krylov subspace method as inner correction solver, the residual is computed iteratively within the solving process. As floating point formats have limited accuracy, the iteratively computed residuals may differ from the explicit residuals due to rounding errors. This can lead to an early breakdown of the error correction solver. As in this case the improvement to the outer residual error is smaller than expected, the convergence analysis for error correction methods using Krylov subspace solvers as error correction solvers has to be modified furthermore. It may happen, that additional outer iterations are necessary to compensate the early breakdowns of the error correction solver.

We denote the total number of additional outer iterations, induced by the rounding errors and the early breakdowns when using Krylov subspace methods for the inner solver, with g, and obtain

$$\left\lceil \frac{\log \varepsilon}{\log \varepsilon_{inner}} \right\rceil + g \tag{4}$$

for the total number of outer iterations. It should be mentioned, that in fact g does not only depend on the type of the error correction solver, but also on the used floating point formats, the conversion and the properties of the linear problem including the matrix structure.

In order to be able to compare a mixed precision error correction solver to a plain high precision solver, we derive a model serving as an upper bound for the computational cost. We denote the complexity of a Krylov subspace solver generating a solution approximation with the relative residual error ε as $C_{solver}(\varepsilon)$. We can obtain this complexity estimation from the convergence analysis of the Krylov subspace solvers [5]. Using this notation, the complexity of an error correction method using a correction solver with relative residual error ε_{inner} can be displayed as

$$\begin{aligned} &C_{mixed}(\varepsilon) = \\ &\left(\left\lceil \frac{\log(\varepsilon)}{\log(\varepsilon_{inner})} \right\rceil + g \right) \cdot \left(C_{solver}(\varepsilon_{inner}) \cdot s + K \right), \end{aligned} \tag{5}$$

where $s \leq 1$ denotes the speedup gained by performing computations in the low precision format (eventually parallel on the low precision device) instead of the high precision format. We denote the quotient between the mixed precision error correction approach to a certain solver and the plain solver in high precision with $f_{solver} = \frac{C_{mixed}(\varepsilon)}{C_{solver}(\varepsilon)}$, and obtain

$$f_{solver} = \frac{\left(\left\lceil \frac{\log(\varepsilon)}{\log(\varepsilon_{inner})} \right\rceil + g \right) \cdot \left(C_{solver}(\varepsilon_{inner}) \cdot s + K \right)}{C_{solver}(\varepsilon)} \tag{6}$$

Analyzing this fraction, we can state the following propositions:

1. If $f_{solver} < 1$, the mixed precision error correction approach to a certain solver performs faster than the plain precision solver. This superiority of the mixed precision approach will particularly occur, if the speedup gained by performing the inner solver in a lower precision format (e.g. on a accelerator) overcompensates the additional computations, typecasts and the eventually needed transmissions in the mixed precision error correction method.
2. The inverse $\frac{1}{f_{solver}}$ could be interpreted as *speedup factor* obtained by the implementation of the mixed precision refinement method with a certain error correction solver. Although this notation does not conform with the classical definition of the speedup concerning the quotient of a sequentially and a parallelly executed algorithm, we can construe $\frac{1}{f_{solver}}$ as measure for the acceleration triggered by the use of the mixed precision approach (and the eventually hybrid system).

3. The iteration loops of Krylov subspace solvers are usually dominated by a matrix-vector multiplication. Hence, using a Krylov subspace method as error correction solver, the factor f_{solver} is independent of the problem size for large dimension. This can also be observed in numerical experiments (see [6]).

Exact knowledge of all parameters would enable to determine a priori whether the mixed precision refinement method using a certain error correction solver outperforms the plain solver. The computational cost of a Krylov subspace solver depends on the dimension and the condition number of the linear system [5].

While the problem size can easily be determined, an approximation of the condition number of a certain linear system can be obtained by performing a certain number of iterations of the plain Krylov subspace solver, and analyzing the residual error improvement.

The only factor that poses problems is g, the number of additional outer iterations necessary to correct the rounding errors generated by the use of a lower precision format for the inner solver. As long as we do not have an estimation of g for a certain problem, we are not able to determine a priori, which solver performs faster.

To resolve this problem, an implementation of an intelligent solver suite could use the idea to determine a posteriori an approximation of g, and then choose the optimal solver. To get an a posteriori approximation of g, the solver executes the first iteration loop of the inner solver and then compares the improvement of the residual error with the expected improvement. Through the difference, an estimation for the number of additional outer iterations can be obtained, that then enables to determine the factor f_{solver} and choose the optimal version of the solver.

4 Conclusions and Future Work

This paper shows results of numerical analysis concerning the convergence theory of mixed precision error correction methods. These results contribute to the possibility to control the usage of different precision formats within a error correction solver.

A problem still requiring a more satisfactory solution is to determine the exact dependency of the number of additional outer iterations on the characteristics of the linear system, the solver type, the inner and outer stopping criterion, and the used floating point precision formats. Further work in this field is necessary to enable an estimation depending on these parameters.

Technologies like FPGAs and application-specific designed processors offer a free choice of floating point formats. Controlling the usage of these precision formats within error correction solvers is necessary for optimizing the performance.

References

1. Marc Baboulin, Alfredo Buttari, Jack J. Dongarra, Julie Langou, Julien Langou, Piotr Luszcek, Jakub Kurzak, and Stanimire Tomov. Accelerating scientific computations with mixed precision algorithms. 2008.
2. Alfredo Buttari, Jack J. Dongarra, Julie Langou, Julien Langou, Piotr Luszcek, and Jakub Kurzak. Mixed precision iterative refinement techniques for the solution of dense linear systems. 2007.
3. D. Göddeke, R. Strzodka, and S. Turek. Performance and accuracy of hardware–oriented native–, emulated– and mixed–precision solvers in FEM simulations. 2007.
4. D. Göddeke and R. Strzodka. Performance and accuracy of hardware–oriented native–, emulated– and mixed–precision solvers in FEM simulations (part 2: Double precision gpus). Technical report, Fakultät für Mathematik, TU Dortmund, 2008.
5. Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.
6. H. Anzt, Rocker B., and Heuveline V. An error correction solver for linear systems: Evaluation of mixed precision implementations. In *EMCL Preprint Series*, 2010.

recent issues

No. 2010-01 Hartwig Anzt, Vincent Heuveline, Björn Rocker: An Error Correction Solver for Linear Systems: Evaluation of Mixed Precision Implementations

No. 2009-02 Rainer Buchty, Vincent Heuveline, Wolfgang Karl, Jan-Philipp Weiß: A Survey on Hardware-aware and Heterogeneous Computing on Multicore Processors and Accelerators

No. 2009-01 Vincent Heuveline, Björn Rocker, Staffan Ronnas: Numerical Simulation on the SiCortex Supercomputer Platform: a Preliminary Evaluation