The backward stable variants of GMRES in variable accuracy

Emmanuel Agullo, Olivier Coulaud, Luc Giraud, Martina Iannacito, Gilles Marait, Nick Schenkels

The backward stable variants of GMRES in variable accuracy

Emmanuel Agullo, Olivier Coulaud, Luc Giraud, Martina Iannacito, Gilles Marait, Nick Schenkels

RESEARCH REPORT
N° 9483
September 2022
Project-Team Concace

ISSN 0249-6399 ISRN INRIA/RR--9483--FR+ENG

The backward stable variants of GMRES in variable accuracy

Emmanuel Agullo*, Olivier Coulaud*, Luc Giraud*, Martina Iannacito*, Gilles Marait*, Nick Schenkels*

Project-Team Concace

Research Report n° 9483 — September 2022 — 73 pages

Abstract: In the context where the representation of the data is decoupled from the arithmetic used to process them, we investigate the backward stability of two backward-stable implementations of the GMRES method, namely the so-called Modified Gram-Schmidt (MGS) and the Householder variants. Considering data may be compressed to alleviate the memory footprint, we are interested in the situation where the leading part of the rounding error is related to the data representation. When the data representation of vectors introduces componentwise perturbations, we show that the existing backward stability analyses of MGS-GMRES [27] and Householder-GMRES [15] still apply. We illustrate this backward stability property in a practical context where an agnostic lossy compressor is employed and enables the reduction of the memory requirement to store the orthonormal Arnoldi basis or the Householder reflectors. Although technical arguments of the theoretical backward stability proofs do not readily apply to the situation where only the normwise relative perturbations of the vector storage can be controlled, we show experimentally that the backward stability is maintained; that is, the attainable normwise backward error is of the same order as the normwise perturbations induced by the data storage. We illustrate it with numerical experiments in two practical different contexts. The first one corresponds to the use of an agnostic compressor where vector compression is controlled normwise. The second one arises in the solution of tensor linear systems, where low-rank tensor approximations based on Tensor-Train [26] is considered to tackle the curse of dimensionality.

Key-words: GMRES, backward stability, variable accuracy, mixed precision, Modified Gram-Schmidt, Householder reflection, lossy compression, tensor, tensor-train

* Inria, Inria centre at the University of Bordeaux

RESEARCH CENTRE
BORDEAUX – SUD-OUEST

200 avenue de la Vieille Tour
33405 Talence Cedex

Les variantes rétro-stables de GMRES en précision variable

Résumé : Dans le contexte où la représentation des données est découplée de l'arithmétique utilisée pour les traiter, nous étudions la stabilité inverse des deux implémentations stables de la méthode GMRES, à savoir la variante dite Modified Gram-Schmidt (MGS) et la variante Householder. Considérant que les données peuvent être compressées pour réduire l'empreinte mémoire, nous nous intéressons à la situation où la partie principale de l'erreur d'arrondi est liée à la représentation des données. Lorsque la représentation des données des vecteurs introduit des perturbations par composantes, les analyses de stabilité inverse existantes de MGS-GMRES [27] et Householder-GMRES [15] restent applicables. Nous illustrons cette propriété de stabilité dans un contexte pratique pratique où un compresseur agnostique à perte est utilisé et permet de réduire la mémoire nécessaire pour stocker la base orthonormale d'Arnoldi ou les réflecteurs de Householder. Bien que les arguments techniques des preuves théoriques de de stabilité inverse ne s'appliquent pas facilement à la situation où seules les perturbations relatives en norme sont utilisées, nous montrons expérimentalement que la stabilité inverse est maintenue ; c'est-à-dire que l'erreur inverse atteignable est du même ordre que les perturbations normalisées induites par le stockage des données. Nous rapportons des expériences numériques dans deux contextes pratiques différents. Le premier correspond à l'utilisation d'un compresseur agnostique. Le deuxième se présente dans la résolution de systèmes linéaires tensoriels, définis sur un produit tensoriel d'espaces linéaires, où les approximations tensorielles à faible rang basées sur Tensor-Train [26] est envisagée pour lutter contre la malédiction de la dimensionnalité.

Mots-clés : GMRES, stabilité inverse, précision variable, précision mitxe, Gram-Schmidt Modifié, réflexion d'Householder, compression avec perte, tenseur, train de tenseur

Contents

1 Introduction

We consider the solution of a linear system

$$Ax = b \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ is a non singular matrix, $b \in \mathbb{R}^n$ is the right-hand side and $x \in \mathbb{R}^n$ is the sought solution, via the two following variants of the GMRES method: the original method [32] based on Modified Gram-Schmidt (MGS) and denoted MGS-GMRES, and its Householder variant [35], denoted H-GMRES. The normwise backward stability of both these algorithms in IEEE arithmetic has been proved [15, 27] under the classical IEEE model assumption where the unit rounding error of floating point operations and the data storage are both bounded by the same unit roundoff u, which depends on the selected working arithmetic. In this work, we are interested in the attainable accuracy in terms of normwise backward error of the solver when the leading part of the rounding error is related to the selected data representation. This is motivated by situations where storing all the data in classical IEEE floating point format is not affordable and part of the data has to be compressed. From a numerical point of view, this introduces relative perturbations on the data that will later on be involved in further computations. When the data representation introduces relative componentwise perturbations, the existing backward stability analyses for MGS-GMRES [15] and H-GMRES [27] still apply. We illustrate this property in a practical context, using an agnostic lossy compressor where the compression parameter is set to ensure a prescribed targeted accuracy. We compress either the Arnoldi basis in MGS-GMRES or the Householder vectors in H-GMRES. Although the technical details of the theoretical backward stability results do not readily apply to the situation where only the normwise relative perturbations of the data storage can be controlled, we show experimentally that a similar numerical property is maintained. In particular, we consider two practical case studies. The first one is based on lossy compression where the compressor is controlled normwise. The second one is based on low-rank tensor approximation, employed to tackle the curse dimensionality.

As exhibited by the recent and exhaustive surveys by Abdelfattah et al. and Higham and Mary, mixed precision arithmetic is yet again a very active research topic [1, 22]. Higham defines the *precision* as "the accuracy with which the basic operations +,-,*,/ are performed" [21]. Higham and Mary define a mixed precision algorithm as one that "uses two or more precisions chosen from a small number of available precisions, which are typically half, single and double precision, provided in hardware, and quadruple precision, provided in software" [22] and indeed most numerical linear algebra work on mixed precision arithmetic does follow these lines [1, 22]. The present study borrows a different path. First, data is not stored with the same accuracy as the precision. In view of this, it would be qualified as an algorithm "decoupling formats for data storage and processing" by Higham and Mary [22, Section 8.7] to which they associate [2, 5, 20]. However, as stated by Higham and Mary, the work of Antz et al. [5], as well as that of Grutzmacher et al. [20], "focus on storing the data at a lower *precision* than that at which the computations are performed." Indeed, Antz et al. state that they "maintain IEEE fp64 in all arithmetic operations, but employ a more compact lower precision IEEE format (fp32 or fp16) for the memory operations" [5]. On the contrary, the present work aims at *decoupling the data storage from hardware constraints at all.* We previously investigated this opportunity in a FGMRES context in [2], which is, to the best of our knowledge, the sole other study following a similar methodological path. The key idea is to compress vectors all at once, into which particular scalars are therefore not constrained by any memory alignment. In particular, they do not need to stick with an available (hardware) precision such as fp32 or fp16. Scalars do not live in "two or more precisions chosen from a small number of available precisions" [21] but on a continuum of possible accuracy. We call such as scheme *variable accuracy data storage*, or, for short, *variable*

accuracy.

Among the mixed precision arithmetic literature, GMRES has been the focus of many studies [1, 22]. A first class of approaches relies on iterative refinement, a technique introduced by Wilkinson in 1963 [39, p. 121 et seq.]. We refer to [22, sections 6 and 8.1] for an exhaustive review of these techniques in a mixed precision arithmetic context. In these studies, when it is employed, GMRES is the inner solver of an inner-outer scheme, and the goal is to achieve the desired accuracy for the outer scheme. On the contrary, the present work tackles the backward stability of GMRES itself. We may though mention that two recent papers in this class of iterative refinement algorithms may be further related to our study by the fact that the backward stability of GMRES is also a central argument [4, 8]. Carson and Higham indeed propose iterative refinement algorithms that combine GMRES with LU factorization in three precisions [8]. Amestoy et al. extend it in five precisions [4]. In both studies, the authors show, under reasonable assumptions, that the proposed algorithms are backward stable with respect to the best used arithmetic. A second class of approach, reviewed in [22, Section 8.3], consists in performing the matrix-vector product with a decreasing accuracy, i.e., accepting larger and larger perturbations in the linear operator. Originally introduced experimentally in [7] and later theoretically analyzed in [17, 33, 34], it is referred to as Inexact Krylov. In that context, the inexactness is modeled by perturbations applied to A in the Arnoldi step. The inexactness was later extended to the inner product calculation in a MGS-GMRES context in [19] where it is modeled as a perturbation only on the scalar computed by the inner product. On the other hand, in the present work, we can model our variable accuracy scheme as perturbations on any vectors of size n computed by the algorithm, which is the key we rely on to compress the Arnoldi basis or the Householder reflectors. There is nonetheless a similarity between our approach and the above Inexact GMRES studies. Indeed, in both Inexact GMRES and our proposed variable accuracy scheme, a prescribed backward error can be eventually obtained and the inexactness (or variable accuracy in our case) depends on this user prescribed accuracy. Finally, another approach also close to ours is by Aliaga et al. [3]. It indeed also fits in the algorithms "decoupling formats for data storage and processing" [22, Section 8.7]. The proposed abstraction at the vector level could also enable a variable accuracy scheme but the study only considers storage in a given (lower) hardware precision, following the methodology from [5, 20]. The performance of their approach is assessed on high performance GPUs.

The rest of the paper is organized as follows. First, in Section 2 we recall the main ideas of both considered GMRES variants as well as the main related backward stability results. Section 3 is devoted to the context where length n vectors are stored with componentwise perturbations bounded by a prescribed value δ. We first report on numerical experiments where fake perturbations are artificially introduced on all operations involving vectors of size n to illustrate that the theoretical results from [15, 27] still hold. Next, we consider a practical example, where only the Arnoldi basis vectors or vectors defining the Householder reflectors are stored in a compressed mode using the agnostic compressor SZ [13]. Section 4 is dedicated to the normwise perturbation counterpart of the previous section. In that section, two practical situations are considered. The first one is similar to the componentwise framework where the vectors basis are stored in a compressed format, but with the SZ compressor controlled normwise. The second one is related to the solution of linear systems defined in tensor product of vector spaces, where the tensor calculation is performed on low-rank tensors using the Tensor Train (TT) format [26]. In that context, the so-called TT-rounding operations applied to the low-rank tensor introduced normwise perturbations. Finally, some concluding remarks are given in Section 5.

2 Mathematical background

Starting from an initial guess x_0, GMRES [32] constructs a series of approximations x_k in Krylov subspaces of increasing dimension k and with decreasing residual norm over these nested spaces. More specifically:

$$x_k = \underset{x \in x_0 + \mathcal{K}(A, r_0, k)}{\operatorname{argmin}} \|b - Ax\|,$$

with

$$\mathcal{K}(A, r_0, k) = span\{r_0, Ar_0, \ldots, A^{k-1} r_0\}$$

the k-dimensional Krylov subspace spanned by A and r_0. In practice, a basis $V_k = (v_1, \ldots, v_k) \in \mathbb{R}^{n \times k}$ with orthonormal columns and an upper Hessenberg matrix $\bar{H}_k \in \mathbb{R}^{(k+1) \times k}$ are iteratively constructed using the Arnoldi procedure such that $span\{V_k\} = \mathcal{K}(A, r_0, k)$ and

$$AV_k = V_{k+1} \bar{H}_k, \qquad \text{with} \qquad V_{k+1}^T V_{k+1} = I_{k+1}.$$

This is often referred to as the Arnoldi equality. The minimum residual norm solution $x_k = x_0 + V_k y_k$ is defined with

$$y_k = \underset{y \in \mathbb{R}^k}{\operatorname{argmin}} \left\| \beta e_1 - \bar{H}_k y \right\|,$$

where $\beta = \|b\|$ and $e_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^{k+1}$ so that in exact arithmetic $\|\beta e_1 - \bar{H}_k y_k\| = \|b - Ax_k\|$.

Another matrix equality can be formed based on the Arnoldi relation that reads

$$(v_1, AV_k) = V_{k+1} R_{k+1}, \tag{2}$$

where R_{k+1} is a $(k+1) \times (k+1)$ upper triangular matrix that writes

$$R_{k+1} = \left(e_1, \bar{H}_k\right).$$

Equation (2) shows that the Arnoldi algorithm actually computes a QR factorization of (v_1, AV_k). Such a factorization could also be computed using Householder transformations, this is the core idea of the H-GMRES variant proposed by Walker [35] and depicted in Algorithm 2. The original GMRES by Saad and Schultz [32] is based on a more classical Arnoldi algorithm that uses Modified Gram-Schmidt procedure to orthogonalize the Krylov basis. This algorithm is often referred to as MGS-GMRES and is depicted in Algorithm 1.

The backward error analysis, popularized by J.H. Wilkinson who contributed significantly to its development [36, 37], gives powerful stropping criteria for iterative solvers. In particular, Rigal and Gache [29] showed that the approximate solution x_k at iteration k can be interpreted as the exact solution of a perturbed linear system where the relative norms of the perturbations can be easily computed. We denote $\eta_{A,b}(x_k)$ this normwise backward error for linear systems

$$\begin{aligned} \eta_{A,b}(x_k) &= \min_{\Delta A, \Delta b} \{\tau > 0 : \|\Delta A\| \leq \tau \|A\|, \ \|\Delta b\| \leq \tau \|b\| \\ &\qquad \text{and } (A + \Delta A) x_k = b + \Delta b\} \\ &= \frac{\|Ax_k - b\|}{\|A\| \|x_k\| + \|b\|}. \end{aligned} \tag{3}$$

Based on the basic IEEE model $fl(a \operatorname{op} b) = (a \operatorname{op} b)(1 + \epsilon)$, with $\operatorname{op} \in \{+, -, \times, \div\}$, $|\epsilon| < u$ and u the unit roundoff of the working precision, many theoretical results exist in the literature. In particular, it is known [38, p. 152-161] that the Householder QR factorization of a set of

Algorithm 1 $x = \text{MGS-GMRES}(A,\, x_0,\, b,\, \varepsilon)$

input: A, x_0, b, ε.
$r_0 = b - Ax_0$, $\beta = \|r_0\|$ and $v_1 = r_0/\beta$
for $k = 1, \ldots$ **do**
  $w_k = av_k$
  for $i = 1, \ldots, k$ **do**
    $\bar{h}_{i,k} = v_i^T w_k$
    $w_k = w_k - \bar{h}_{i,k} v_i$
  end for
  $\bar{h}_{k+1,k} = \|w_k\|$
  $v_{k+1} = w_k / \bar{h}_{k+1,k}$
  $y_k = \underset{y \in \mathbb{R}^k}{\operatorname{argmin}} \left\| \beta e_1 - \bar{H}_k y \right\|$
  $x_k = x_0 + v_k y_k$
  if $((\eta_{A,b}(x_k) < \varepsilon)$ **then**
    break
  end if
end for
return: $x = x_k$

Algorithm 2 $x = \text{H-GMRES}(A,\, x_0,\, b,\, \varepsilon)$

input: A, x_0, b, ε.
$r_0 = b - Ax_0$, $\beta = \|r_0\|$
compute u_1 to define the Householder reflector $P_1 = I - 2u_1 u_1^T$ such that $p_1 r_0 = \beta e_1$
for $k = 1, \ldots$ **do**
  $v_k = \prod_{j=k}^{1} P_j e_k$
  $w = \prod_{j=1}^{k} P_j A v_k$
  compute u_{k+1} to define the Householder reflector $P_{k+1} = I - 2u_{k+1}u_{k+1}^T$ s.t. $P_{k+1} w(k+1 : n) = \|w(k+1:n)\| \, e_{k+1}(k+1:n)$ and $p_{k+1} w(1:k) = w(1:k)$
  $\bar{H}_k(:, k) = (P_{k+1} w)(1 : k+1)$
  $y_k = \underset{y \in \mathbb{R}^k}{\operatorname{argmin}} \left\| \beta e_1 - \bar{H}_k y \right\|$
  % compute the current iterate
  $z = 0$
  for $j = k, \ldots, 1$ **do**
    $z = P_j (y_k(j) e_j + z)$
  end for
  $x_k = x_0 + z$
  if $((\eta_{A,b}(x_k) < \varepsilon)$ **then**
    break
  end if
end for
return: $x = x_k$

n-dimensional vectors $S = (s_1, ..., s_m)$ generates an orthonormal basis $\tilde{Q}$ with orthonormality quality

$$\tilde{Q}^T\tilde{Q} = I + E \text{ with } \|E\|_2 \approx u, \tag{4}$$

while MGS produces [6] a $\tilde{Q}$ factor such that

$$\tilde{Q}^T\tilde{Q} = I + E \text{ with } \|E\|_2 \approx \kappa(S)\, u, \tag{5}$$

where $\kappa(S)$ is the 2-norm condition number of the matrix S.

Thanks to the backward stability of Householder QR given by (4), the backward stability of H-GMRES was established in [15, Theorem 4.1 and Corollary 4.2] assuming that the matrix A is not close to singularity.

Theorem 2.1 *Under technical assumptions [15, Corollary 4.2], it is shown that the normwise backward error at the last iterate x_n of H-GMRES is such that $\eta_{A,b}(x_n) = \mathcal{O}(u)$, where u is the unit roundoff error of the working precision.*

In MGS-GMRES, it is known that the Arnoldi basis progressively departs from orthogonality as indicated by (5). However, the following result has been shown in [28]

$$\eta_{A,b}(x_k) \cdot \|I - \tilde{V}_k^T\tilde{V}_k\|_F = \mathcal{O}(u), \tag{6}$$

where the columns of $\tilde{V}_k \in \mathbb{R}^{n\times k}$, computed by MGS-Arnoldi, form a basis for the Krylov space $\mathcal{K}_k(A, b)$. This result implies that it is impossible to have a significant loss of orthogonality as long as the normwise relative backward error is very small. Later, the backward stability of MGS-GMRES was proved in [27], that reads:

Theorem 2.2 ([27]) *Assuming that A is not close to singularity, that is,*

$$\sigma_{min}(A) \gg n^2||A||u,$$

where u is the unit roundoff error of the working precision; let $k \in \mathbb{N}$ be the first integer such that

$$\kappa(\tilde{V}_{k+1}) > \frac{4}{3}, \tag{7}$$

then x_k satisfies

$$\eta_{A,b}(x_k) = \mathcal{O}(u).$$

We refer to [27] for the details of the tedious and technical proof and to [25, p. 227] for a short exposure. We also note that for technical reason, the proof is established using the Frobenius norm of A to define $\eta_{A,b}$ so that the normwise perturbations on A are measured in Frobenius norm and not in Euclidean norm. In practice, the backward stability is observed using the 2-norm and we only consider it in the sequel.

For the numerical experiments, the norm of the linear operator is estimated using a randomized SVD [24, 30]. In addition, for all numerical experiments right preconditioned GMRES is considered to enable a fast convergence, so that GMRES does effectively solve

$$AMz = b$$

where M is a preconditioner operator. For calculation performed in regular matrix case, we consider a preconditioner based on an $ILU(t)$ [31] factorization. In the low-rank tensor case, we use an approximation of the inverse of a nearby problem for which a polynomial approximation can be computed [14]. In the preconditioned context, the backward error $\eta_{AM,b}(z_k)$ is the one that GMRES can drive down to $\mathcal{O}(u)$.

3 Convergence with componentwise perturbations

3.1 δ-componentwise data storage

We are interested in the numerical behaviour of GMRES where the processsed data may be numerically altered for two reasons. The first possible source is the employed finite precision arithmetic discussed above. We still denote u the associated unit roundoff error. In this work, we further assume that the data may be compressed leading to another possible source of inaccuracy due to the corresponding compressed data representation. More precisely, we denote δ-storage() the function that enables to store a given data in a format, referred to as δ-componentwise data format, that induces an unit roundoff δ; that is:

$$\delta\text{-storage}(a) = a(1+\xi) \text{ with } |\xi| \leq \delta.$$

When a basic calculation is performed and the result stored in a δ-representation we have

$$fl_\delta(a \operatorname{op} b) = \delta\text{-storage}(fl(a \operatorname{op} b)) = (a \operatorname{op} b)(1+\epsilon)(1+\xi),$$

with $|\epsilon| \leq u$ and $|\xi| \leq \delta$. Neglecting the second order term $\epsilon\xi$, we obtain

$$fl_\delta(a \operatorname{op} b) = (a \operatorname{op} b)(1+\epsilon+\xi),$$

with $|\epsilon+\xi| \leq u+\delta$, so that all the theoretical results, presented in the previous section, that were established with the same unit roundoff u for the basic operations and the data representation still hold, but with unit roundoff $u+\delta$. In can be noted that in the particular case $u \ll \delta$, it further reduces to $fl_\delta(a \operatorname{op} b) = (a \operatorname{op} b)(1+\xi)$, with $|\xi| \leq \delta$ and those same results then hold with unit roundoff δ. For a matter of readability, we will make this particular assumption (results hold with unit roundoff δ) in rest of this section, but they can be more generally interpreted without this assumption, in which case a $u+\delta$ shall be considered instead.

We numerically investigate the behavior of the MGS-GMRES and H-GMRES where the storage function δ-storage() is used to have a δ-representation of all the vectors of size n. Regular IEEE calculation is used and data associated with small dimension matrices and vectors are stored in regular IEEE format. That are all the data involved in the least squares problem solution.

In Figure 1, we plot the convergence history of the normwise backward error for both MGS-GMRES and H-GMRES for various values of δ for a test matrix from the Florida test collection [11]. The dashed vertical blue line indicates the iteration where $\kappa(\tilde{V}_k) > \frac{4}{3}$ for MGS-GMRES. It can be seen that the backward stability property does still hold for δ roundoff unit due to data representation. Although this is not predicted by any theoretical argument, the convergence of the two GMRES variants is identical up to the proximity of δ. On this example, the convergence of MGS-GMRES and H-GMRES perfectly overlap. Furthermore the relation given by (6) between the loss of orthogonality and the backward error for MGS-GMRES, represented by the dashed green curve, does also hold for δ unit roundoff.

To illustrate that both data representation and finite precision calculation play a role in the attainable accuracy, we present numerical experiments in Figure 2 where the calculation are performed either in fp32 (left plots) or fp64 (right plots). On the first row, where the componentwise perturbation is $\delta = 10^{-4}$, which is much larger than u_{32} and u_{64}, the various GMRES implementations do exhibit the same convergence and identical attainable accuracy, i.e., $\mathcal{O}(\delta)$. On the second row, the fp32 implementation has an attainable accuracy that is $\mathcal{O}(u_{32})$ because the δ componentwise perturbation is hidden by the fp32 format. The fp64 calculation reveals the attainable accuracy already observed in Figure 1, that is $\mathcal{O}(\delta)$, since $\delta \gg u_{64}$. It can be seen that the loss of orthogonality of the MGS-Arnoldi basis (which is characterized by a

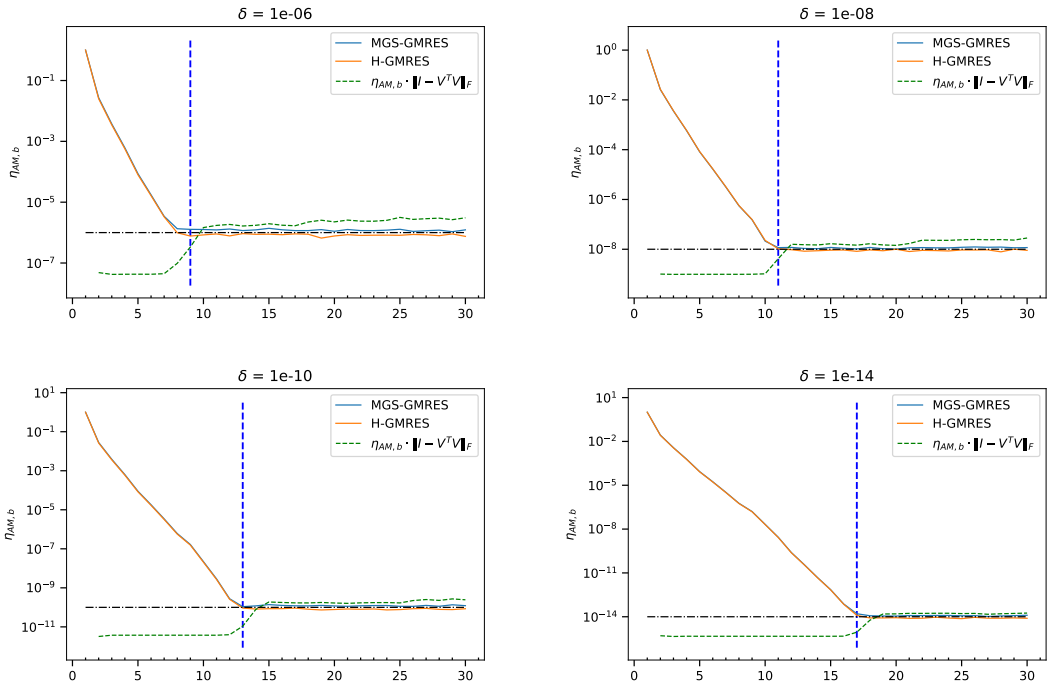

Figure 1: Convergence history of $\eta_{AM,b}$ for gre-115 with ILU(10^{-1}) using δ-data componentwise representation. The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

condition number that deviates from one and becomes larger that $4/3$ after the iteration marked by the vertical dashed blue line) is also affected by δ-representation of the basis vectors. More results on other test examples are depicted in Appendix A.

3.2 Solution techniques using SZ compressed format

In this section we present a first practical application of the GMRES algorithm in variable accuracy. We consider an implementation where the Arnoldi basis in MGS-GMRES or the reflector vectors in H-GMRES are compressed to alleviate their memory footprint. More precisely, in MGS-GMRES we compress w_k in line 10 before normalizing it to define v_{k+1}. In H-GMRES, we compress the vectors u_k that defines the Householder reflectors P_k in lines 3 and 7 of Algorithm 2. For those experiments, we used the SZ [13] lossy compressor, an agnostic compressor that does not attempt to exploit underlying numerical properties of the vectors but operates on their binary representation. For the experiments in this section, we use the capability of SZ to ensure a prescribed componentwise relative error between the original data $z \in \mathbb{R}^n$ and the decompressed data $\tilde{z} \in \mathbb{R}^n$, that is,

$$\max_{i=1,\ldots,n} \frac{|z(i) - \tilde{z}(i)|}{|z(i)|} \leq \delta,$$

with $z(i)$ the ith component of z.

Similarly to the previous section, we display the convergence histories of MGS-GMRES and H-GMRES in Figure 3, for a few values of the compression control parameter δ. Although the

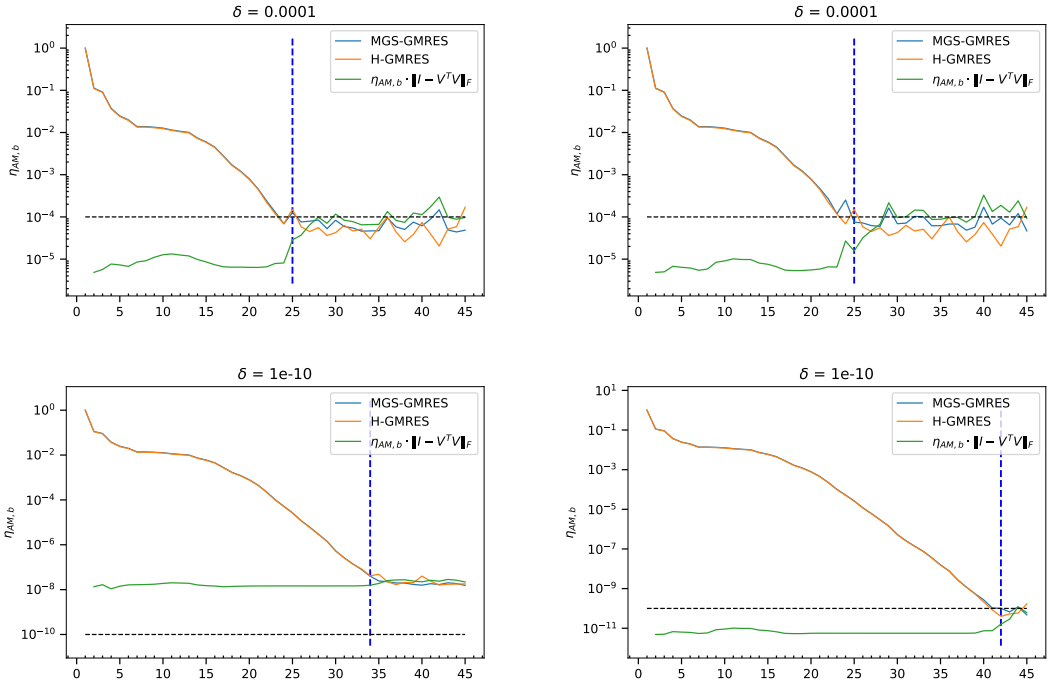

Figure 2: Convergence history of $\eta_{AM,b}$ for gre-343 with ILU($2 \cdot 10^{-1}$) using δ-data componentwise representation with fp32 calculation (left) and fp64 calculation (right). The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

δ componentwise perturbations only occur when storing the Arnoldi basis or reflector vectors, the general trend of the convergence histories of $\eta_{AM,b}$ remains qualitatively the same. The attainable accuracy is slightly – but not significantly – better. The memory saving enabled by SZ is reported in Table 1. A general expected rule of thumb is: "the larger δ, the larger the memory gain". For some matrices, e.g., gre_343, more than 20% of memory saving is observed for a 10^{-14} accuracy. It illustrates the possible significant benefit of these GMRES implementations based on a compressor like SZ for the solution of large problems.

4 Convergence with normwise perturbations

In many contexts, having a control on the relative componentwise error on the vectors is not feasible and only a normwise monitoring is possible. Although the existing backward stability analyses of both considered GMRES variants do not readily apply, because some of the technical arguments of the proof are not longer valid, we show through numerical experiments that the property still holds in practice. In Section 4.1, we report on experiments where the results of all calculations performed on length n vector is artificially perturbed by a relative normwise perturbation. Next we consider two practical computational contexts where such normwise perturbations are encountered in some steps of the algorithms.

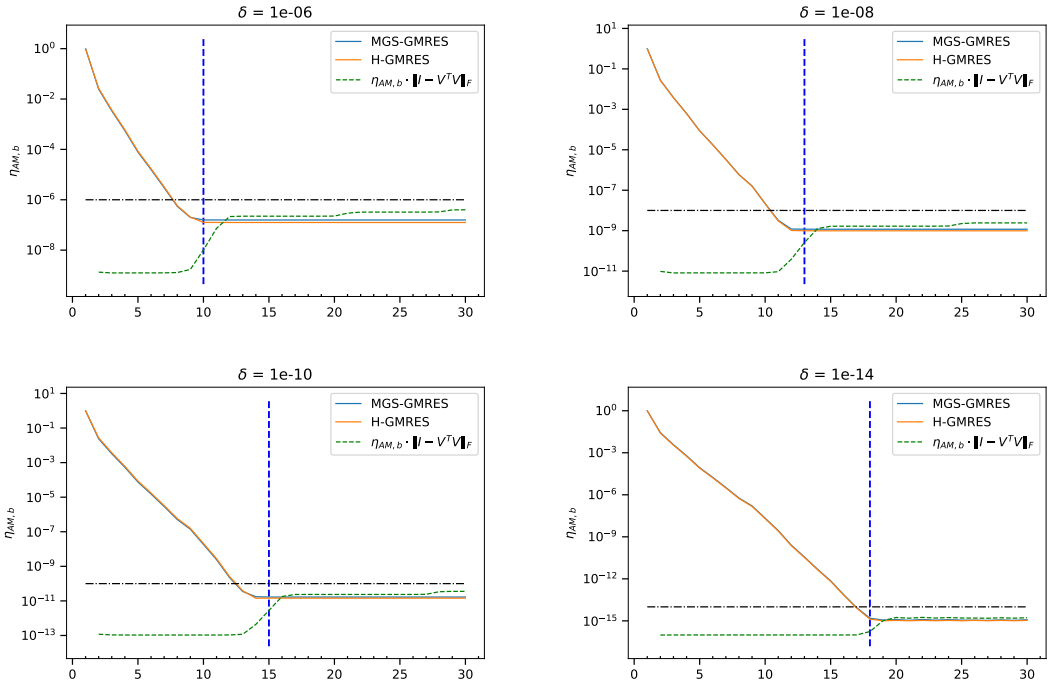

Figure 3: Convergence history of $\eta_{AM,b}$ for gre-115 with ILU(10^{-1}) using SZ-componentwise storage for the Arnoldi basis and Householder reflectors. The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

	δ					
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	10^{-14}
gre_115 with ILU(10^{-1})						
MGS	9.0	25.1	25.7	16.3	6.6	3.9
Householder	39.6	30.2	30.4	22.2	15.4	12.4
gre_185 with ILU(10^{-1})						
MGS	3.9	32.8	25.6	16.2	5.4	0.0
Householder	22.3	38.3	31.4	23.4	14.2	6.9
gre_343 with ILU(10^{-1})						
MGS	22.3	50.8	46.7	38.5	30.1	21.9
Householder	24.7	52.2	47.7	39.9	31.7	24.1
arc130 with ILU($8 \cdot 10^{-4}$)						
MGS	15.3	15.2	12.8	17.5	14.3	6.9
Householder	38.4	5.8	25.5	21.5	18.2	9.4
e05r0000 with ILU(10^{-2})						
MGS	0.1	36.2	29.5	20.2	9.4	0.2
Householder	6.7	37.0	29.7	20.8	10.4	2.6
e05r0400 with ILU(10^{-2})						
MGS	1.1	36.0	27.8	18.7	7.9	0.0
Householder	5.4	37.6	29.4	20.8	10.6	3.4
cavity03 with ILU(10^{-2})						
MGS	2.4	39.7	31.2	22.0	10.9	0.6
Householder	14.1	40.5	32.3	23.0	12.4	4.2
pde225 with ILU($3 \cdot 10^{-1}$)						
MGS	1.6	34.3	28.3	19.1	8.2	0.1
Householder	9.6	37.9	31.8	23.3	13.4	5.4

Table 1: Percentage of memory saving using SZ-componentwise representation

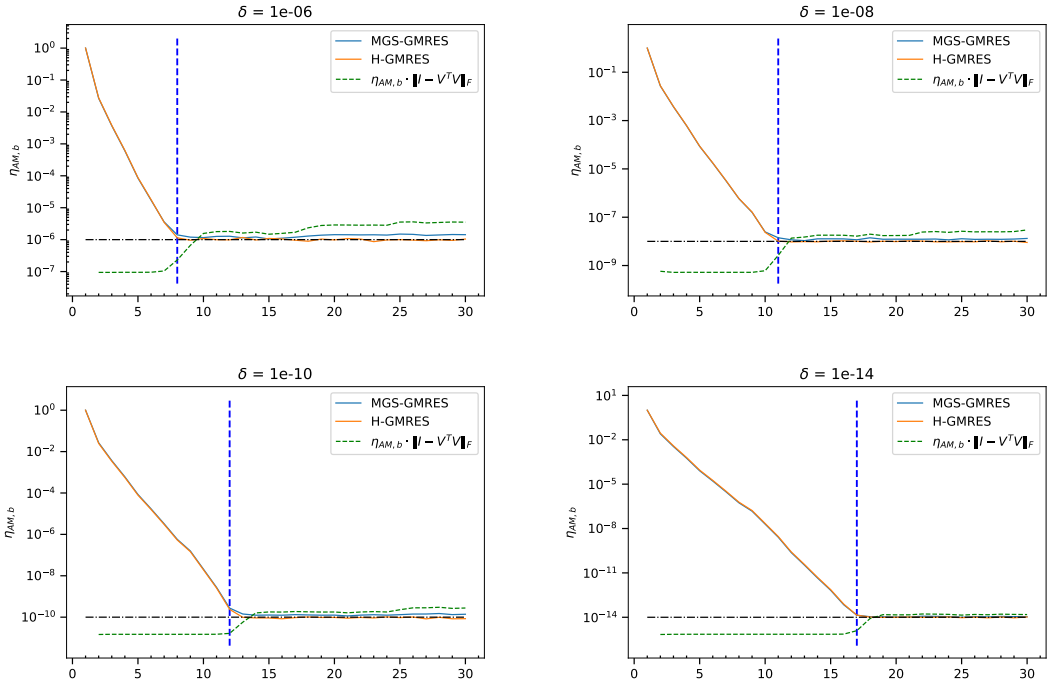

Figure 4: Convergence history of $\eta_{AM,b}$ for gre-115 with ILU(10^{-1}) using δ-data normwise representation. The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

4.1 δ-normwise data storage

In this section we consider the situation where all the length n vectors involved in the algorithms are stored in a normwise δ-representation. That is, any vector $z \in \mathbb{R}^n$ is replaced by $\bar{z}$ so that

$$\frac{\|z - \bar{z}\|}{\|z\|} \leq \delta. \tag{8}$$

We display the convergence history of the normwise backward error for both considered GMRES variants, as well as the product of the backward error times the loss of orthogonality for MGS-GMRES, in Figure 4. It can be seen that the trends are very similar to those that can be observed in Figures 1. The same observations can be made: both the attainable backward error accuracy and the product $\eta_{AM,b}\|I - V^T V\|_F$ reach values close to δ.

4.2 Practical applications

4.2.1 Solution techniques using SZ compressed format

We report the convergence histories of MGS-GMRES and H-GMRES for a few values of the normwise compression control parameter δ in Figure 5, . These results are the normwise counterparts of those displayed in Figure 3 for componentwise compression. It can be observed that the general trends are similar and that the attainable value of $\eta_{AM,b}$ always becomes close to

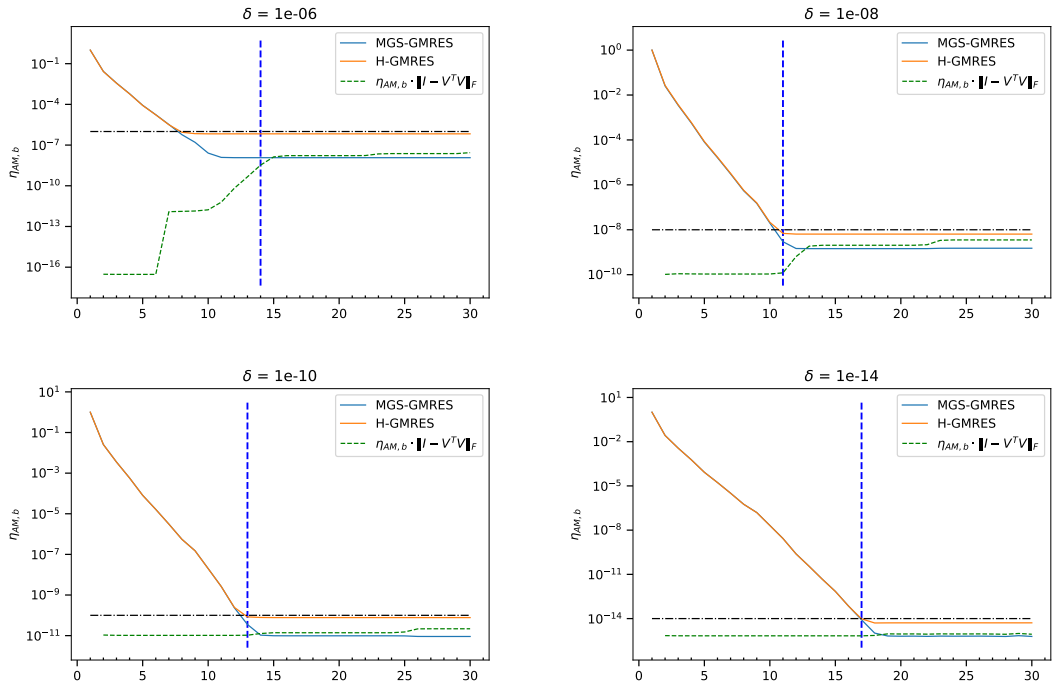

Figure 5: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU(10^{-2}) using SZ-normwise representation of the Arnoldi basis and Householder reflectors. The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

δ. Because ensuring normwise relative error is easier for the SZ-compressor than ensuring componentwise relative error, the memory saving gains reported in Table 2 are larger than those in Table 1.

4.2.2 Solution of tensor linear systems using low-rank tensor format

In this section, we consider a tensor linear system that arises from a discretization on a Cartesian grid of the sum of the product of 1-d operators. Storing the full tensor that represents the operator may not be affordable. A possible alternative to eliminate this memory bottleneck is to work on low-rank approximations of the tensors, which induces perturbations on the data that will eventually be processed by the numerical algorithm.

Let x be an element of the tensor space $\mathbb{R}^{n_1 \times \cdots \times n_d}$ with order d and dimension n_k for mode k for every $k \in \{1, \ldots, d\}$. Since storing the entire tensor $x \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ has a memory cost of $\mathcal{O}(n^d)$ with $n = \max_{i \in \{1,\ldots,d\}}\{n_i\}$, different compression techniques were proposed over the years to reduce the memory consumption [12, 18, 23, 26]. In the present study, we adopt the Tensor Train (TT) format [26]. The key idea of TT is to express a tensor of order d as the contraction of d tensors of order 3 and size (r_{i-1}, n_i, r_i) with $r_{i-1}, r_i \in \mathbb{N}$, where the contraction is essentially the generalization to tensors of the matrix-vector product.

If $r = \max_{i \in \{1,\ldots,d\}}\{r_i\}$, storing a tensor in TT-format requires $\mathcal{O}(dnr^2)$ units of memory. In this case, the memory footprint growths linearly with the tensor order d. However to have a

	δ					
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	10^{-14}
gre_115 with ILU(10^{-1})						
MGS	5.7	8.8	32.9	31.3	23.6	14.3
Householder	50.4	43.3	43.5	35.9	29.0	21.9
gre_185 with ILU(10^{-1})						
MGS	7.2	25.6	37.9	27.7	18.6	7.5
Householder	38.4	45.6	46.6	37.6	29.7	20.6
gre_343 with ILU(10^{-1})						
MGS	22.3	42.0	51.8	46.9	40.3	30.5
Householder	24.5	51.4	58.3	49.5	44.0	35.5
arc130 with ILU($8 \cdot 10^{-4}$)						
MGS	74.6	63.8	51.3	40.4	23.7	21.3
Householder	75.3	70.3	60.9	34.1	14.9	11.5
e05r0000 with ILU(10^{-2})						
MGS	5.6	8.4	40.7	32.3	21.2	11.3
Householder	4.4	40.1	44.7	34.9	24.5	15.7
e05r0400 with ILU(10^{-2})						
MGS	0.1	16.1	39.5	29.4	18.5	9.1
Householder	19.7	42.2	45.5	35.5	25.1	16.5
cavity03						
MGS	0.7	25.6	40.7	29.9	19.1	9.8
Householder	12.5	45.9	47.0	36.9	26.6	18.0
pde225 with ILU($3 \cdot 10^{-1}$)						
MGS	10.8	32.1	37.7	29.4	19.2	8.0
Householder	19.8	43.8	46.5	37.3	29.0	19.2

Table 2: Percentage of memory saving using SZ-normwise representation

significant benefit in the use of this formalism, the value r has to stay bounded and small. A first drawback of the TT-format appears with the addition of two tensors in TT-format. Indeed given two tensors in TT-format x and y with k-th TT-rank r_k and s_k respectively, then the k-th TT-rank of $x + y$ is at most $r_k + s_k$, see [16]. To reduce the TT-ranks and thus re-compress the tensor, a rounding algorithm has been proposed in [26]. Given a tensor in TT-format x and an accuracy δ, the TT-rounding algorithm provides a tensor in TT-format $\tilde{x}$ that is at a relative distance δ, that is

$$\frac{||x - \tilde{x}||}{||x||} \leq \delta$$

which is the same bound as (8). The TT-formalism is also suitable for representing a linear operator among tensor product of spaces. Let $\mathcal{A} : \mathbb{R}^{n_1 \times \cdots \times n_d} \to \mathbb{R}^{n_1 \times \cdots \times n_d}$ be a multilinear operator over the tensor space $\mathbb{R}^{n_1 \times \cdots \times n_d}$, then the tensor $A \in \mathbb{R}^{(n_1 \times n_1) \times \cdots \times (n_d \times n_d)}$ represents $\mathcal{A}$ with respect to the canonical basis of $\mathbb{R}^{n_1 \times \cdots \times n_d}$. If the tensor representing a multilinear operator acting on the linear space $\mathbb{R}^{n_1 \times \ldots n_d}$ is expressed in TT-format, it is usually referred as TT-matrix, while the elements in TT-format of $\mathbb{R}^{n_1 \times \cdots \times n_d}$ are usually called TT-vectors. Henceforth we follow this formalism.

We only consider MGS-GMRES in the numerical experiments for this section, since the Householder reflection algorithm does not straightforwardly extends to the TT-format. To prevent the growth of the memory consumption, the rounding operation is applied systematically before normalizing and storing the Arnoldi basis vectors in line 10 of Algorithm 10. In addition, we also apply some extra rounding during the orthogonalization and after the TT-matrix and TT-vector product (and preconditioner application) when required to prevent running out of memory. For this purpose, we use an estimate of the memory requirement based on the TT-rank and the dimension. When the estimate becomes too large, we compress the computed TT-vector to reduce its TT-rank and enable the algorithm to complete.

We consider the solution of a 3D convection-diffusion problem with a rotating velocity field with boundary conditions defined in (9)

$$\begin{cases} -\Delta u + 2y(1-x^2)\dfrac{\partial u}{\partial x} - 2x(1-y^2)\dfrac{\partial u}{\partial y} = 0 \quad \text{in} \quad \Omega = [-1,1]^3, \\ u_{\{y=1\}} = 1 \qquad \text{and} \qquad u_{\partial\Omega \setminus \{y=1\}} = 0. \end{cases} \tag{9}$$

This equation is discretized on a 3D Cartesian grid, which results in a linear system that can be written in a low-rank tensor form expressing the 3D structure of the PDE and of the mesh. We do not give the details of the formulation and derivation of the linear system in the tensor format, but rather refer the reader to [9, 14] for a detailed exposure. Additional features of the TT-format can be exploited in the GMRES context, we refer to [9] for more details.

The convergence history of MGS-GMRES in TT-format is displayed in Figure 6 for different values of δ. Similarly to classical matrix computation context, the first observation is that the attainable backward error accuracy remains $\mathcal{O}(\delta)$. For some reasons, that we have not explored fully yet, the loss of accuracy remains very low essentially until the attainable accuracy is reached. This induces two main differences with respect to what we have seen so far. The first one is that the product $\eta_{AM,b} \cdot \|I - V_k^T V_k\|$ is not $\mathcal{O}(\delta)$ as long as the attainable accuracy is reached. The second one, still related to the orthogonality quality of the TT-basis, is that the convergence is attained much before $\kappa(V_k) > 4/3$; especially for large values of δ.

Regarding the memory saving, that is the main motivation to solve a linear system in TT-format, we report in Table 3 the percentage of saved memory compared to a situation where full tensor would have been used. It can be seen that the saving is significant and remain close to 40% for a 10^{-12} accuracy.

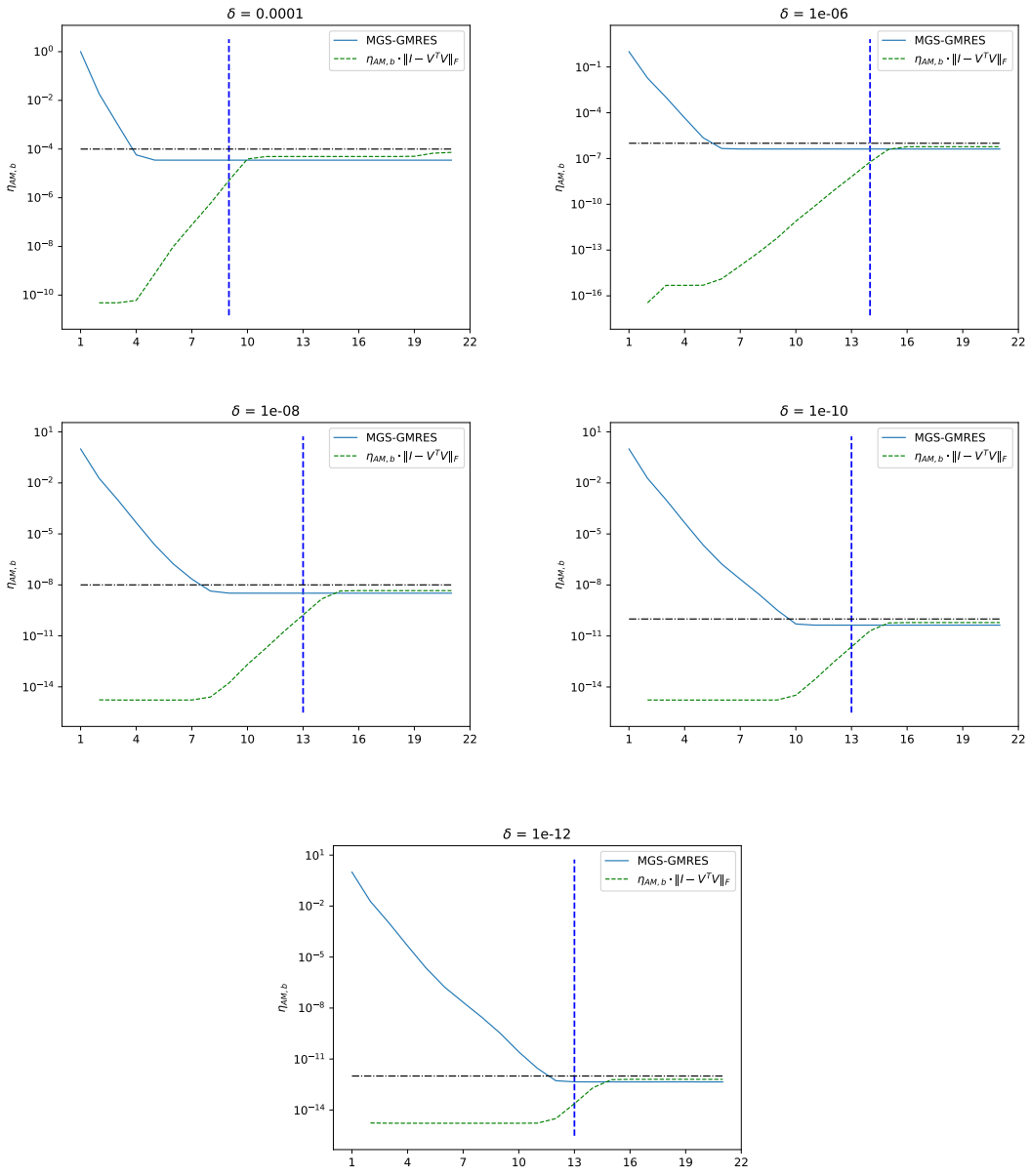

Figure 6: Convergence history of $\eta_{AM,b}$ for the solution of a 3D convection diffusion problem using TT-calculation. The horizontal dashed black line indicates δ. The vertical dash blue line represents the first iteration where $\kappa(V_k) > 4/3$ in MGS-GMRES.

5 Concluding remarks

Most literature [1, 22] considering the design of mixed precision arithmetic algorithms aims at exploiting the opportunities provided by modern hardware to compute with various levels

δ				
10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
88.7	74.4	57.5	45.9	39.7

Table 3: Percentage of memory saving using TT format to store the Arnoldi basis in MGS-GMRES

of accuracy (such as fp64, fp32, fp16 and bfloat16) and subsequently assumes that the data is represented and stored accordingly in memory. In this work, we fully decoupled the data representation from the hardware constraints, possibly doing so in a representation manner agnostic from the numerical algorithm using a generic data compressor. In the context of the backward stable variants of GMRES, we showed that backward stability still holds with accuracy being the maximum between the unit roundoff of the data representation and the one of the processing arithmetic. We illustrated the possible benefits of this result by efficiently storing in compressed format the orthonormal Arnoldi basis in MGS-GMRES or the Householder reflector vectors in H-GMRES.

The results [1] have been numerically assessed with the injection of artificial perturbations, with a generic compressor (SZ) and using TT-format with δ-rounding compression for the solution of tensor linear systems. The conclusion is that the method is effectively robust, not only in the componentwise case for which the existing backward stability analyses [15, 27] did smoothly apply, but also in the normwise case where these analyses do not readily apply. The extension of the theoretical analyses to the normwise case remains to be investigated.

In the TT-format context, we only proposed a numerical assessment for MGS-GMRES. Indeed, the Householder reflection algorithm does not straightforwardly extends to the TT-format. We are finalizing a proposal for such an extension of the Householder reflection algorithm [10]. It might be a basis to devise a H-GMRES algorithm in TT-format.

References

[1] A. Abdelfattah, H. Anzt, E. G. Boman, E. Carson, T. Cojean, J. Dongarra, A. Fox, M. Gates, N. J. Higham, X. S. Li, J. Loe, P. Luszczek, S. Pranesh, S. Rajamanickam, T. Ribizel, B. F. Smith, K. Swirydowicz, S. Thomas, S. Tomov, Y. M. Tsai, and U. M. Yang. "A survey of numerical linear algebra methods utilizing mixed-precision arithmetic". In: *The International Journal of High Performance Computing Applications* 35.4 (2021), pp. 344–369. DOI: 10.1177/10943420211003313.

[2] E. Agullo, F. Cappello, S. Di, L. Giraud, X. Liang, and N. Schenkels. *Exploring variable accuracy storage through lossy compression techniques in numerical linear algebra: a first application to flexible GMRES*. Research Report RR-9342. Inria Bordeaux Sud-Ouest, May 2020.

[3] J. I. Aliaga, H. Anzt, T. Grützmacher, E. S. Quintana-Ortí, and A. E. Tomás. "Compressed basis GMREaS on high-performance graphics processing units". In: *The International Journal of High Performance Computing Applications* 0.0 (0), p. 10943420221115140. DOI: 10.1177/10943420221115140.

[1] For the sake of conciseness, in the core of the report, we made a selection among the numerous numerical experiments that we have run. The whole set of results is available in the appendices.

[4] P. Amestoy, A. Buttari, N. Higham, J.-Y. L'Excellent, T. Mary, and B. Vieuble. "Five-Precision GMRES-based iterative refinement". working paper or preprint. Apr. 2021.

[5] H. Anzt, G. Flegar, T. Grützmacher, and E. S. Quintana-Ortí. "Toward a modular precision ecosystem for high-performance computing". In: *The International Journal of High Performance Computing Applications* 33.6 (2019), pp. 1069–1078. DOI: 10.1177/10943420198465.

[6] Å. Björck. "Solving linear least squares problems by Gram-Schmidt orthogonalization". In: *BIT Numerical Mathematics* 7.1 (1967), pp. 1–21. DOI: 10.1007/BF01934122.

[7] A. Bouras and V. Frayssé. "Inexact matrix-vector products in Krylov methods for solving linear systems: a relaxation strategy". In: *SIAM Journal on Matrix Analysis and Applications* 26.23 (2005), pp. 660–678. DOI: 10.1137/S0895479801384743.

[8] E. Carson and N. J. Higham. "Accelerating the solution of linear systems by iterative refinement in three precisions". In: *SIAM Journal on Scientific Computing* 40.2 (2018), pp. 817–847. DOI: 10.1137/17M1140819.

[9] O. Coulaud, L. Giraud, and M. Iannacito. *A robust GMRES algorithm in Tensor Train format*. Research Report RR-9384. Inria Bordeaux Sud-Ouest, 2022.

[10] O. Coulaud, L. Giraud, and M. Iannacito. *On some orthogonalization schemes in TT-format*. Research Report In preparation. Inria Bordeaux Sud-Ouest, 2022.

[11] T. A. Davis and Y. Hu. "The university of Florida sparse matrix collection". In: *ACM Transactions on Mathematical Software* 38.1 (Nov. 2011), pp. 1–25. DOI: 10.1145/2049662.2049663.

[12] L. De Lathauwer, B. De Moor, and J. Vandewalle. "A Multilinear Singular Value Decomposition". In: *SIAM Journal on Matrix Analysis and Applications* 21.4 (2000), pp. 1253–1278. DOI: 10.1137/S0895479896305696.

[13] S. Di and F. Cappello. "Fast error-bounded lossy HPC data compression with SZ". In: *2016 IEEE international parallel and distributed processing symposium (IPDPS)*. IEEE. 2016, pp. 730–739. DOI: 0.1109/IPDPS.2016.11.

[14] S. V. Dolgov. "TT-GMRES: solution to a linear system in the structured tensor format". In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 28.2 (2013), pp. 149–172. DOI: 10.1515/rnam-2013-0009.

[15] J. Drkosova, A. Greenbaum, M. Rozloznik, and Z. Strakoš. "Numerical stability of GMRES". In: *BIT Numerical Mathematics* 35.February 1994 (1995), pp. 309–330. DOI: 10.1007/BF01732607.

[16] P. Gelß. "The Tensor-Train Format and Its Applications". PhD thesis. Freien Universitat Berlin, 2017. DOI: 10.17169/refubium-7566.

[17] L. Giraud, S. Gratton, and J. Langou. "Convergence in Backward Error of Relaxed GMRES". In: *SIAM Journal on Scientific Computing* 29.2 (Jan. 2007), pp. 710–728. DOI: 10.1137/040608416.

[18] L. Grasedyck. "Hierarchical Singular Value Decomposition of Tensors". In: *SIAM Journal on Matrix Analysis and Applications* 31.4 (2010), pp. 2029–2054. DOI: 10.1137/090764189.

[19] S. Gratton, E. Simon, D. Titley-Peloquin, and P. L. Toint. "A Note on Inexact Inner Products in GMRES". In: *SIAM Journal on Matrix Analysis and Applications* 43.3 (Aug. 2022), pp. 1406–1422. DOI: 10.1137/20m1320018.

[20] T. Grützmacher, H. Anzt, and E. S. Quintana-Ortí. "Using Ginkgo's memory accessor for improving the accuracy of memory-bound low precision BLAS". In: *Software: Practice and Experience* (2021). DOI: 10.1002/spe.3041.

[21] N. J. Higham. *Accuracy and stability of numerical algorithms*. SIAM, 2002. DOI: 10.1137/1.9780898718027.

[22] N. J. Higham and T. Mary. "Mixed precision algorithms in numerical linear algebra". In: *Acta Numerica* 31 (May 2022), pp. 347–414. DOI: 10.1017/s0962492922000022.

[23] T. G. Kolda and B. W. Bader. "Tensor Decompositions and Applications". In: *SIAM Review* 51.3 (Aug. 2009), pp. 455–500. DOI: 10.1137/07070111x.

[24] Martinsson, P. Gunnar, V. Rokhlin, and M. Tygert. "A randomized algorithm for the decomposition of matrices". In: *Applied and Computational Harmonic Analysis* 30.1 (2011), pp. 47–68. DOI: 10.1016/j.acha.2010.02.003.

[25] G. Meurant and J. D. Tebbens. *Krylov Methods for Nonsymmetric Linear Systems*. Springer International Publishing, 2020. DOI: 10.1007/978-3-030-55251-0.

[26] I. V. Oseledets. "Tensor-Train Decomposition". In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317. DOI: 10.1137/090752286.

[27] C. C. Paige, R. Miroslav, and Z. Strakoš. "Modified Gram–Schmidt (MGS), least squares, and backward stability of MGS-GMRES". In: *SIAM Journal on Matrix Analysis and Applications* 28.1 (2006), pp. 264–284. DOI: 0.1137/050630416.

[28] C. C. Paige and Z. Strakoš. "Residual and Backward Error Bounds in Minimum Residual Krylov Subspace Methods". In: *SIAM Journal on Scientific Computing* 23.6 (Jan. 2002), pp. 1898–1923. DOI: 10.1137/s1064827500381239.

[29] J. L. Rigal and J. Gaches. "On the Compatibility of a Given Solution With the Data of a Linear System". In: *Journal of the ACM* 14.3 (July 1967), pp. 543–548. DOI: 10.1145/321406.321416.

[30] V. Rokhlin, A. Szlam, and M. Tygert. "A Randomized Algorithm for Principal Component Analysis". In: *SIAM Journal on Matrix Analysis and Applications* 31.3 (2010), pp. 1100–1124. ISSN: 0895-4798. DOI: 10.1137/080736417.

[31] Y. Saad. "ILUT: A dual threshold incomplete ILU factorization". In: *Numerical Linear Algebra with Applications* 1 (1994), pp. 387–402. DOI: 10.1002/nla.1680010405.

[32] Y. Saad and M. H. Schultz. "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems". In: *SIAM Journal on Scientific and Statistical Computing* 7.3 (1986), pp. 856–869. DOI: 10.1137/0907058.

[33] V. Simoncini and D. B. Szyld. "Theory of Inexact Krylov Subspace Methods and Applications to Scientific Computing". In: *SIAM Journal Scientific Computing* 25 (2003), pp. 454–477. DOI: 10.1137/S1064827502406415.

[34] J. van den Eshof and G. L. G. Sleijpen. "Inexact Krylov subspace methods for linear systems". In: *SIAM Journal on Matrix Analysis and Applications* 26.1 (2004), pp. 125–153. DOI: 10.1137/S0895479802403459.

[35] H. F. Walker. "Implementation of the GMRES method using Householder transformations". In: *SIAM Journal on Scientific Computing* 9.1 (1988), pp. 152–163. DOI: 0.1137/0909010.

[36] J. H. Wilkinson. "Modern Error Analysis". In: *SIAM Review* 13.4 (Oct. 1971), pp. 548–568. DOI: 10.1137/1013095.

[37] J. H. Wilkinson. *Rounding Errors in Algebraic Processes*. Vol. 32. Notes on Applied Science. Also published by Prentice-Hall, Englewood Cliffs, NJ, USA, 1964, translated into Polish as *Bledy Zaokragleń w Procesach Algebraicznych* by PWW, Warsaw, Poland, 1967 and translated into German as *Rundungsfehler* by Springer-Verlag, Berlin, Germany, 1969. Reprinted by Dover Publications, New York, 1994. London, UK: HMSO, 1963, pp. vi + 161. ISBN: 0-486-67999-3 (Dover). DOI: 10.2307/3614445.

[38] J. H. Wilkinson. *The algebraic eigenvalue problem*. en. Numerical Mathematics and Scientific Computation. Oxford, England: Clarendon Press, Jan. 1988.

[39] J. H. Wilkinson. *Rounding Errors in Algebraic Processes*. PRENTICE-HALL, INC., 1963.

How to read the appendices

For the sake of conciseness, in the core of the report, we made a selection among the numerous numerical experiments that we have run. All these results are available in the appendices. Appendices A to D consider 64-bit arithmetic. More accurately, appendices A and B present further results for componentwise and normwise perturbation, respectively. Appendices C and D present similar results with SZ with componentwise and normwise compression, respectively. Lastly, appendices E and F show the behaviour using 32-bit arithmetic for componentwise and normwise perturbations.

A Results with componentwise perturbations on δ-vectors and 64-bit calculation

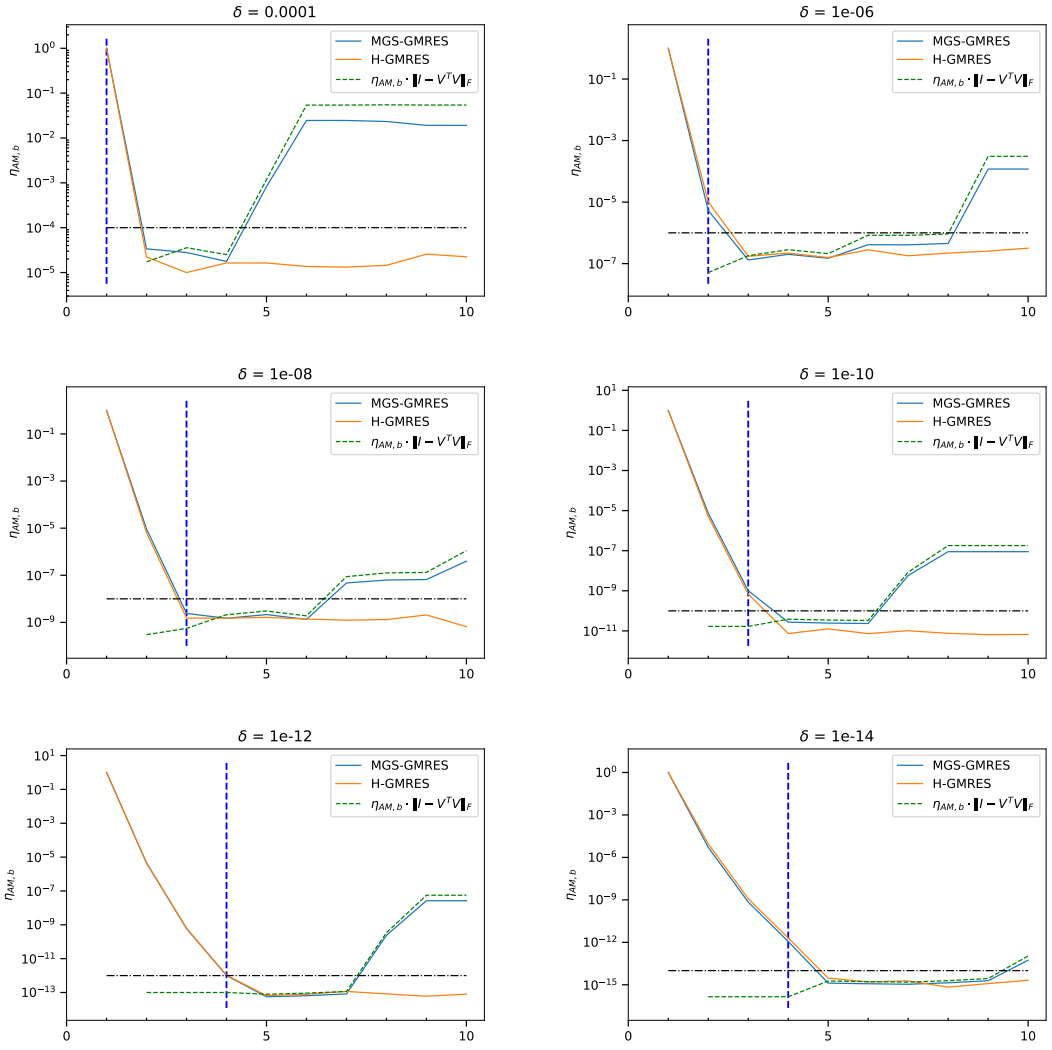

Figure 7: Convergence history of $\eta_{AM,b}$ for arc130 with ILU($8 \cdot 10^{-4}$) using δ-componentwise representation

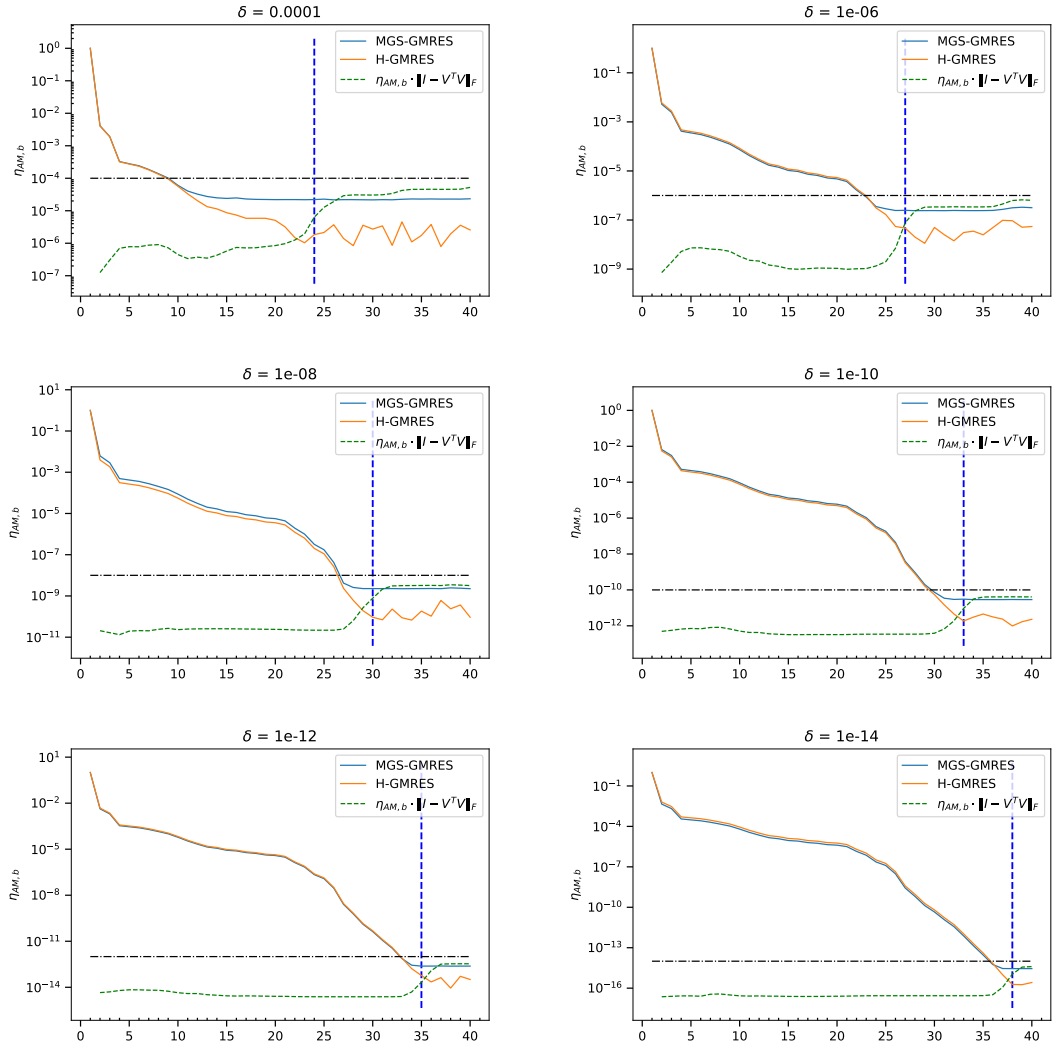

Figure 8: Convergence history of $\eta_{AM,b}$ for cavity03 with ILU(10^{-2}) using δ-componentwise representation

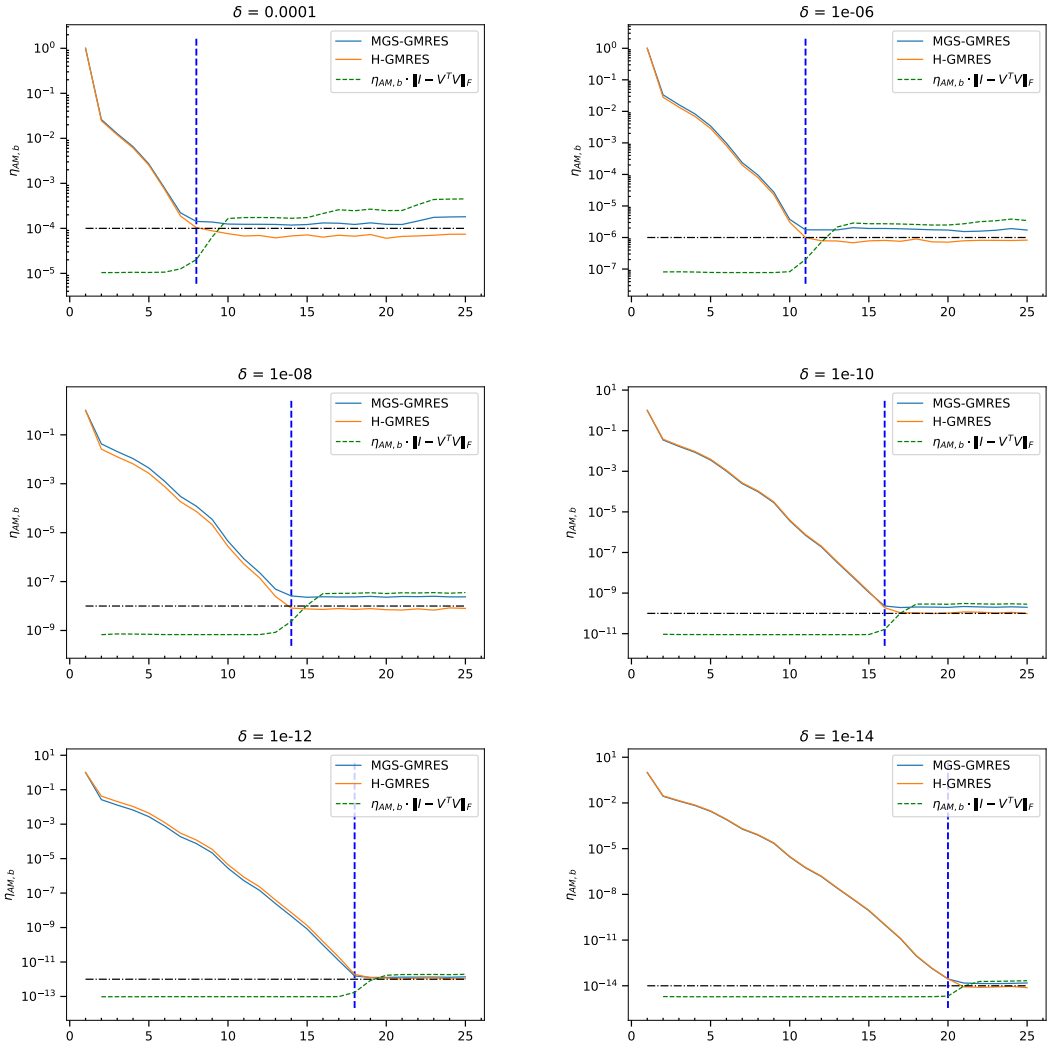

Figure 9: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU(10^{-2}) using δ-componentwise representation

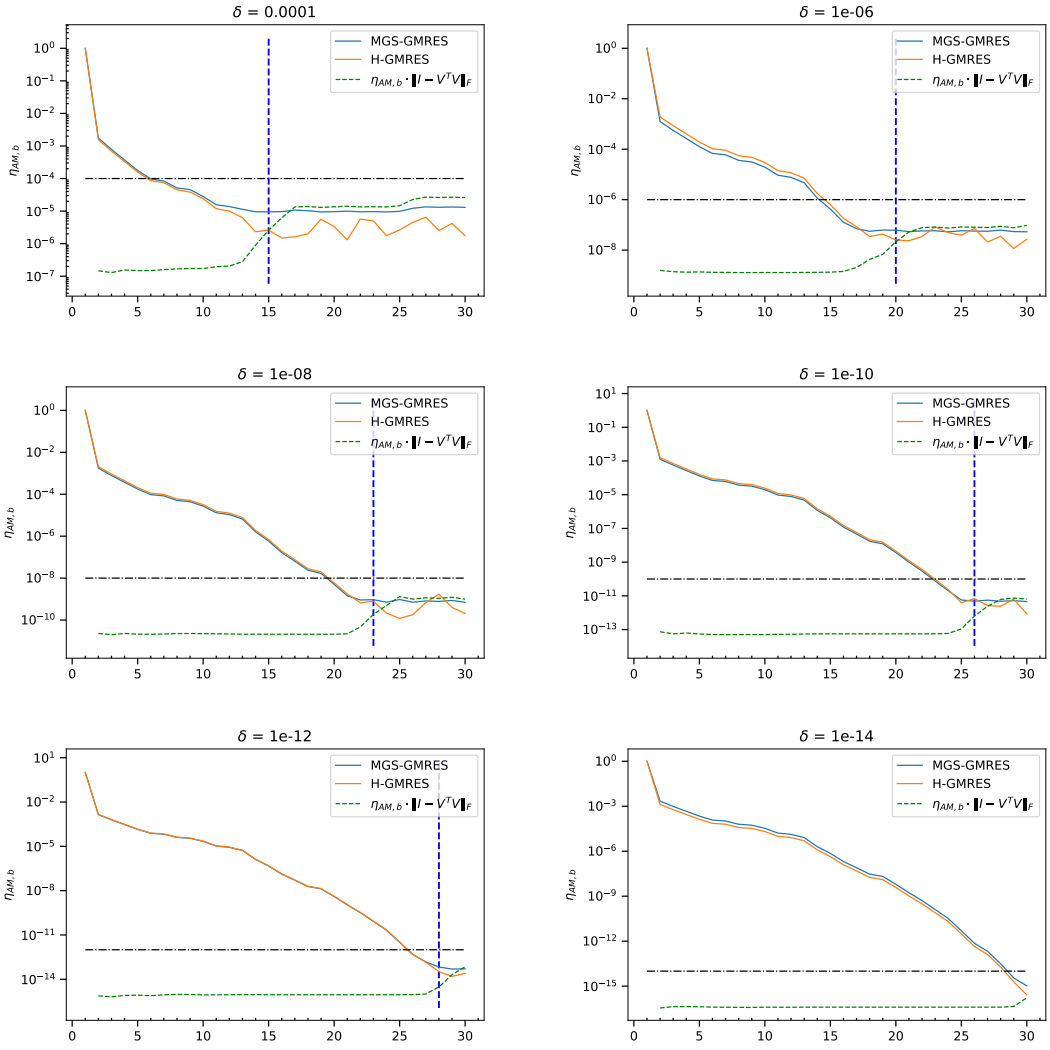

Figure 10: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using δ-componentwise representation

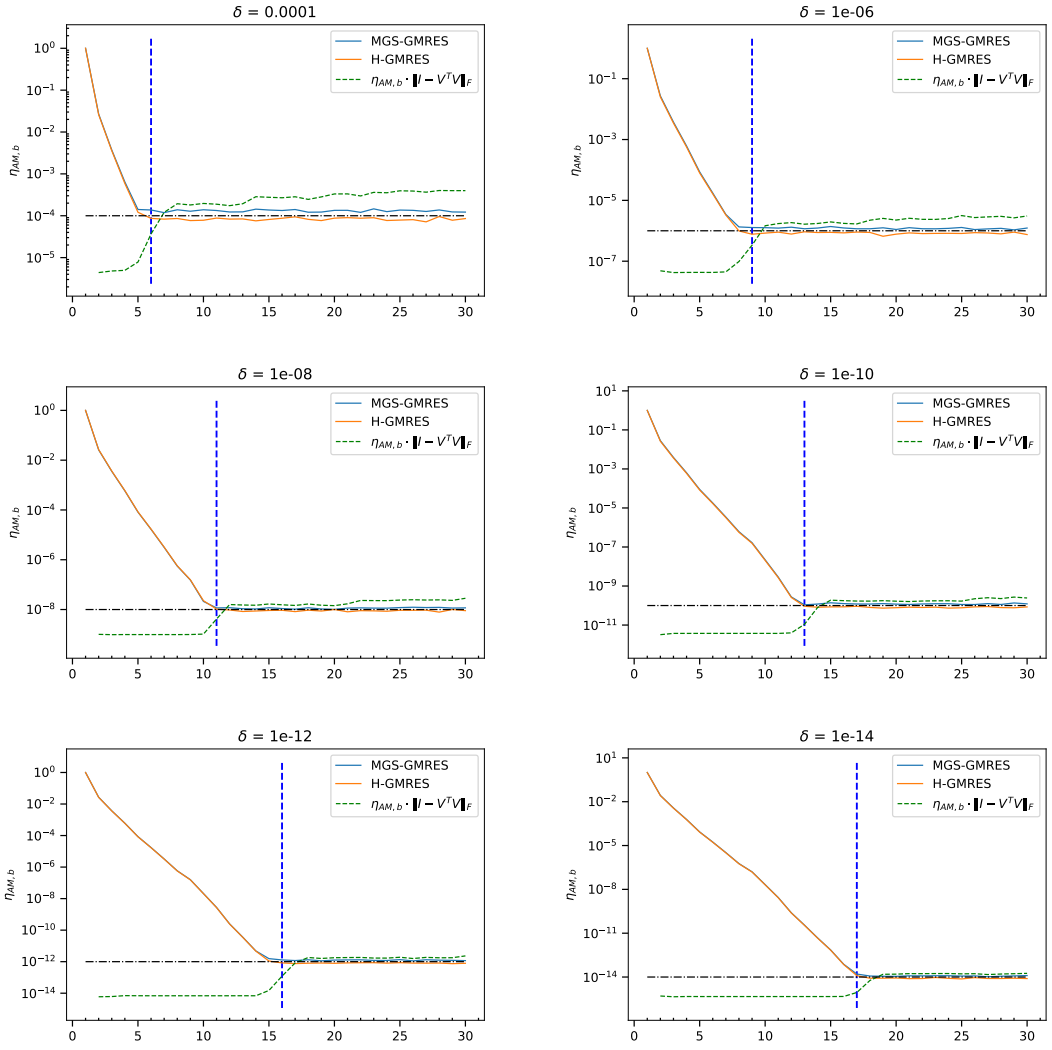

Figure 11: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU(10^{-1}) using δ-componentwise representation

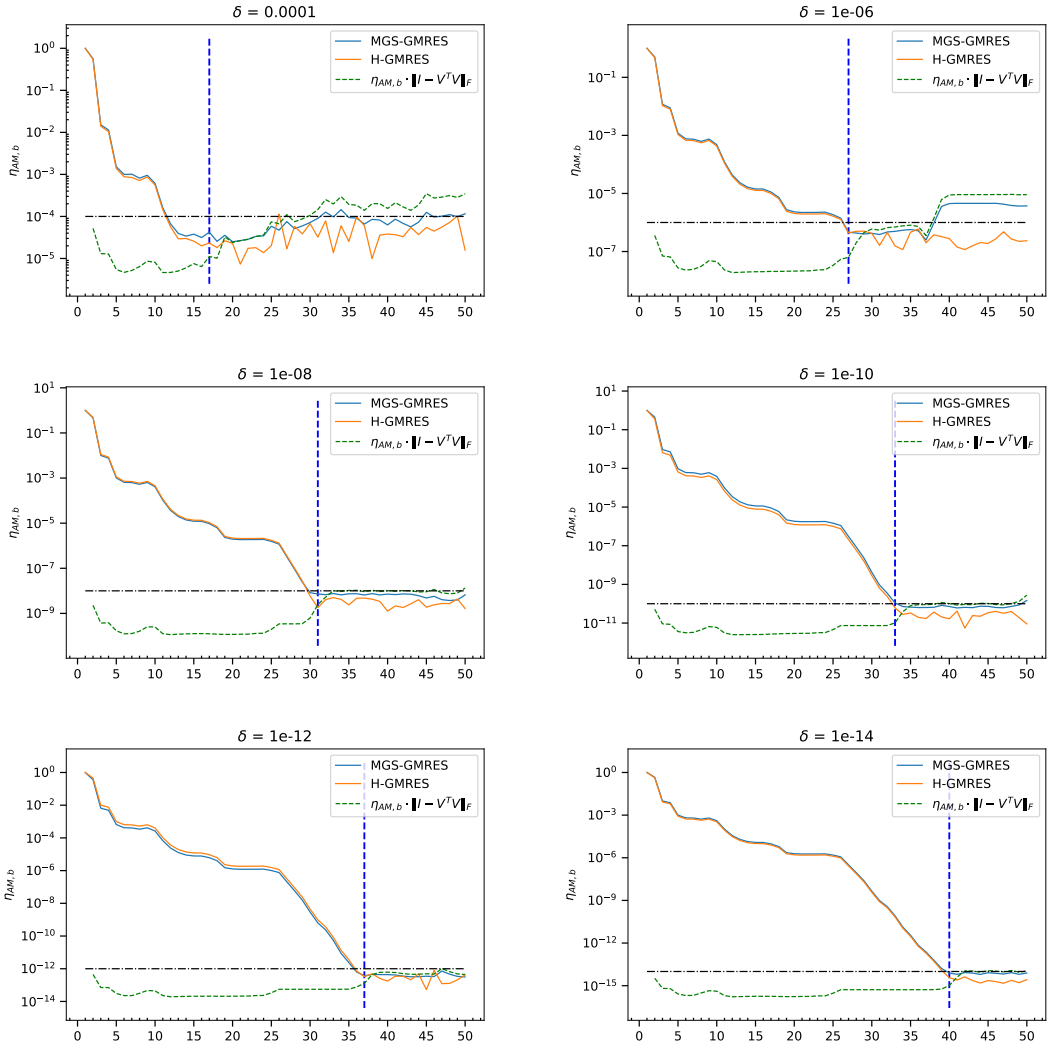

Figure 12: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU(10^{-1}) using δ-componentwise representation

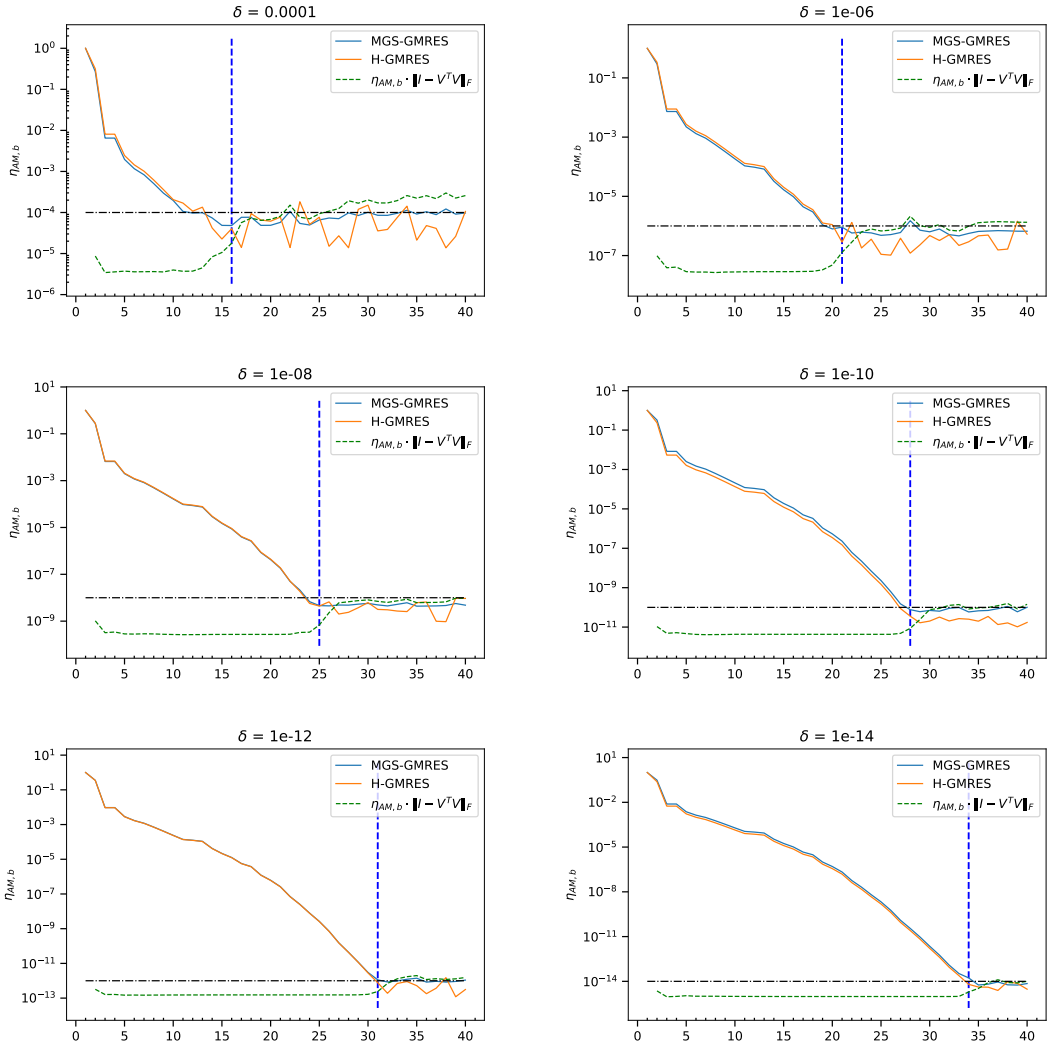

Figure 13: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU(10^{-1}) using δ-componentwise representation

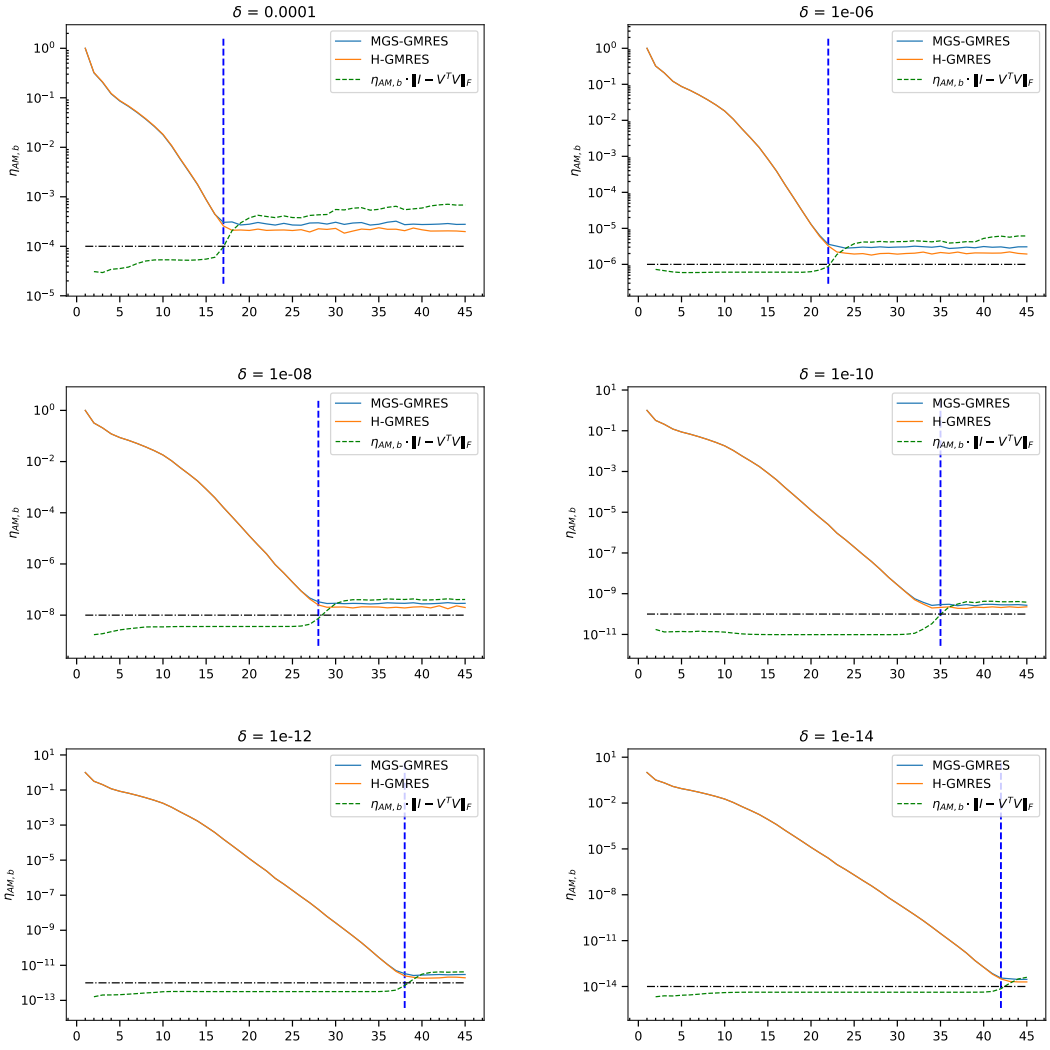

Figure 14: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($3 \cdot 10^{-1}$) using δ-componentwise representation

B Results with normwise perturbations on δ-vectors and 64-bit calculation

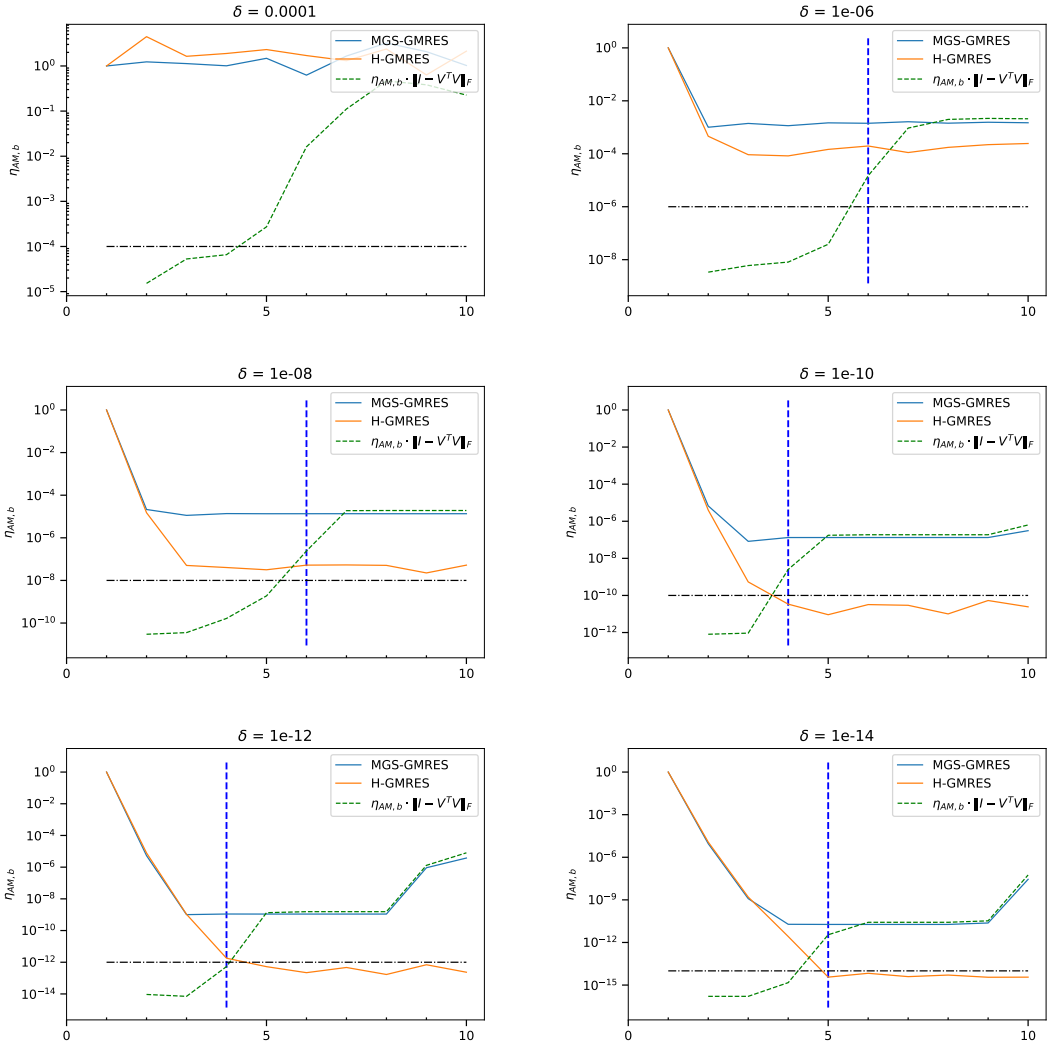

Figure 15: Convergence history of $\eta_{AM,b}$ for arc130 with ILU($8 \cdot 10^{-4}$) using δ-normwise representation

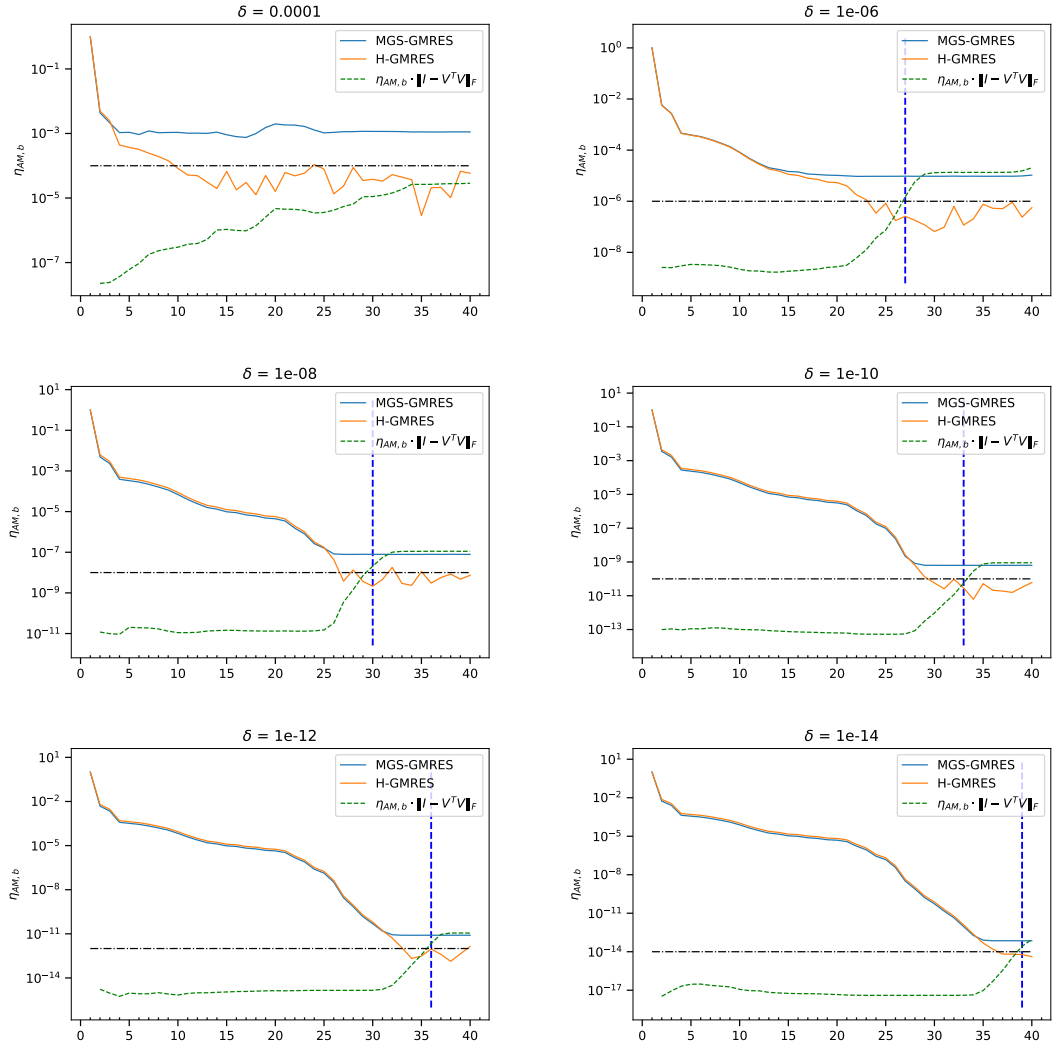

Figure 16: Convergence history of $\eta_{AM,b}$ for cavity03 with ILU(10^{-2}) using δ-normwise representation

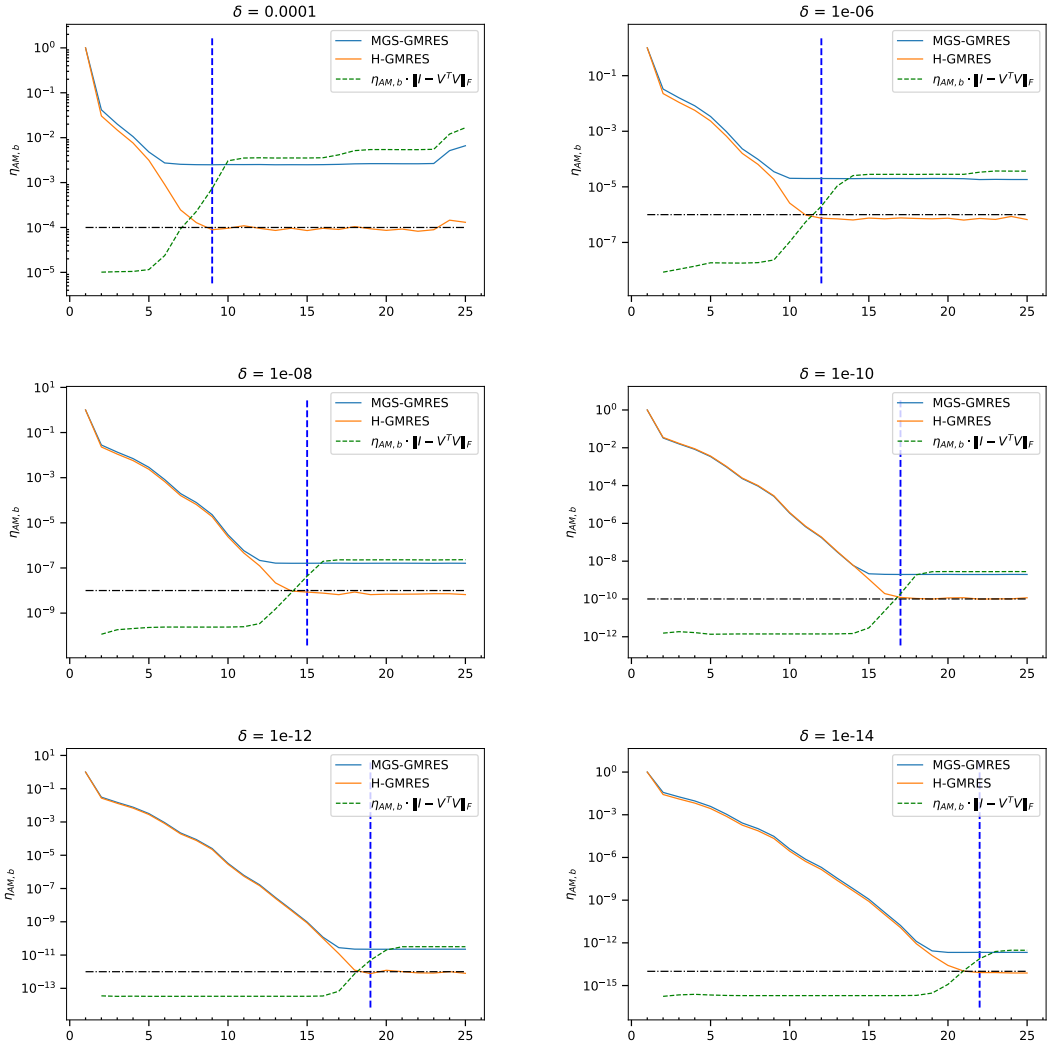

Figure 17: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU(10^{-2}) using δ-normwise representation

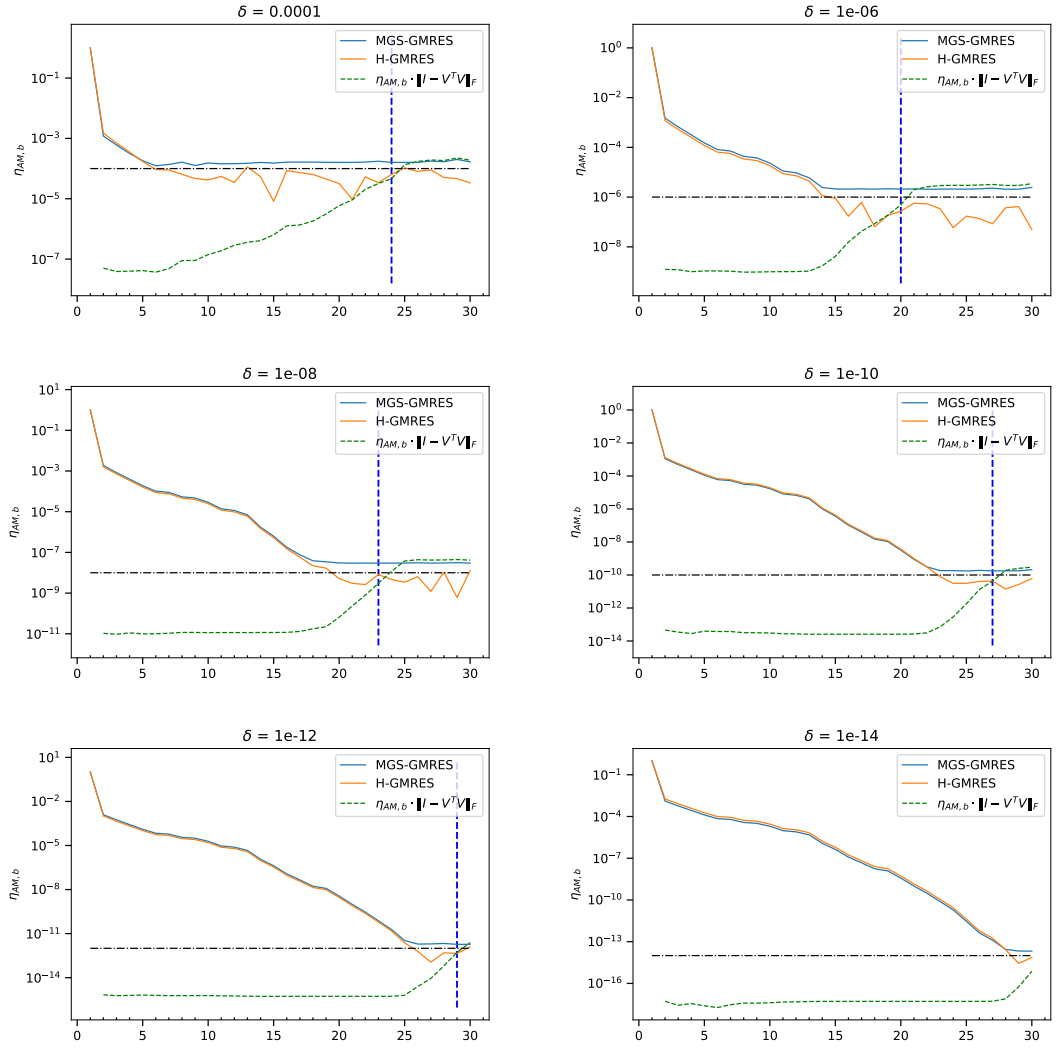

Figure 18: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using δ-normwise representation

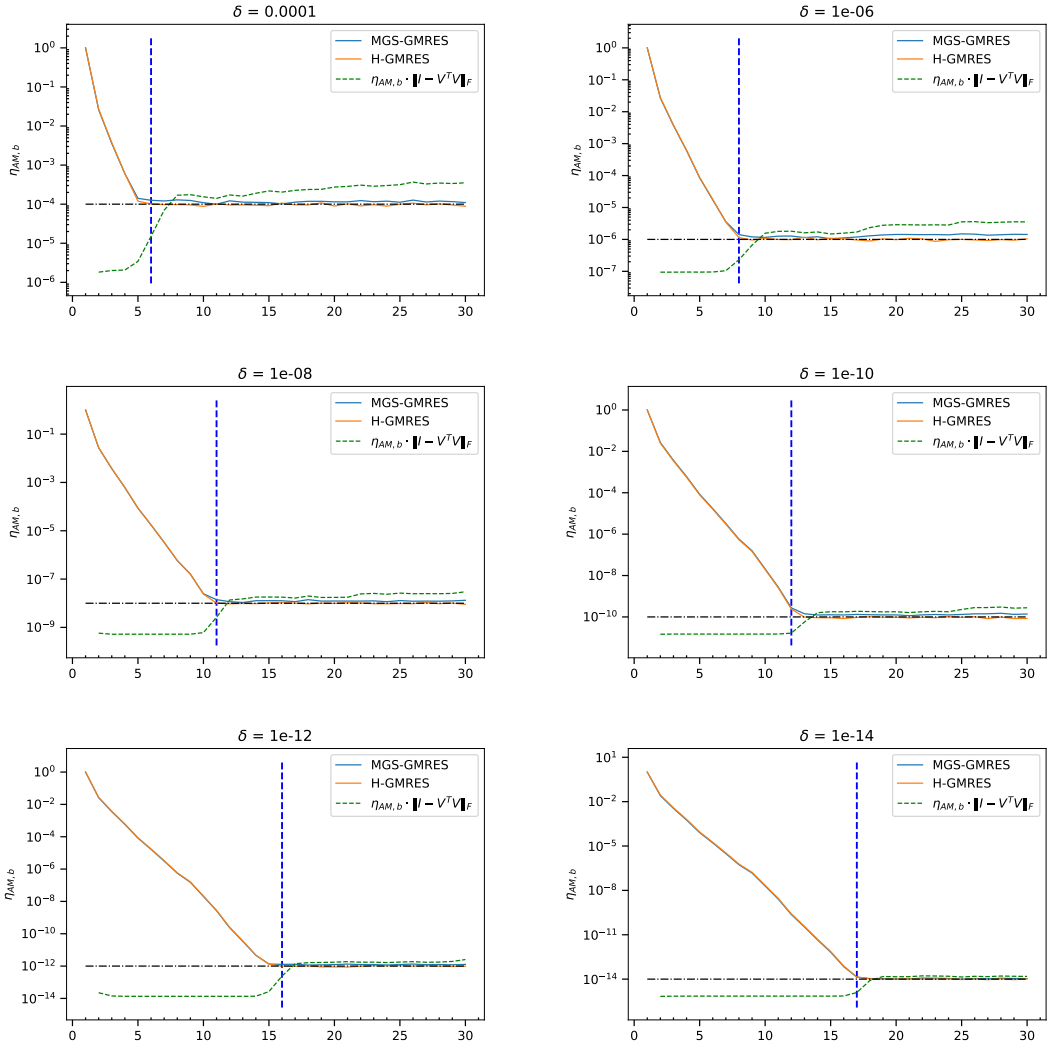

Figure 19: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU(10^{-1}) using δ-normwise representation

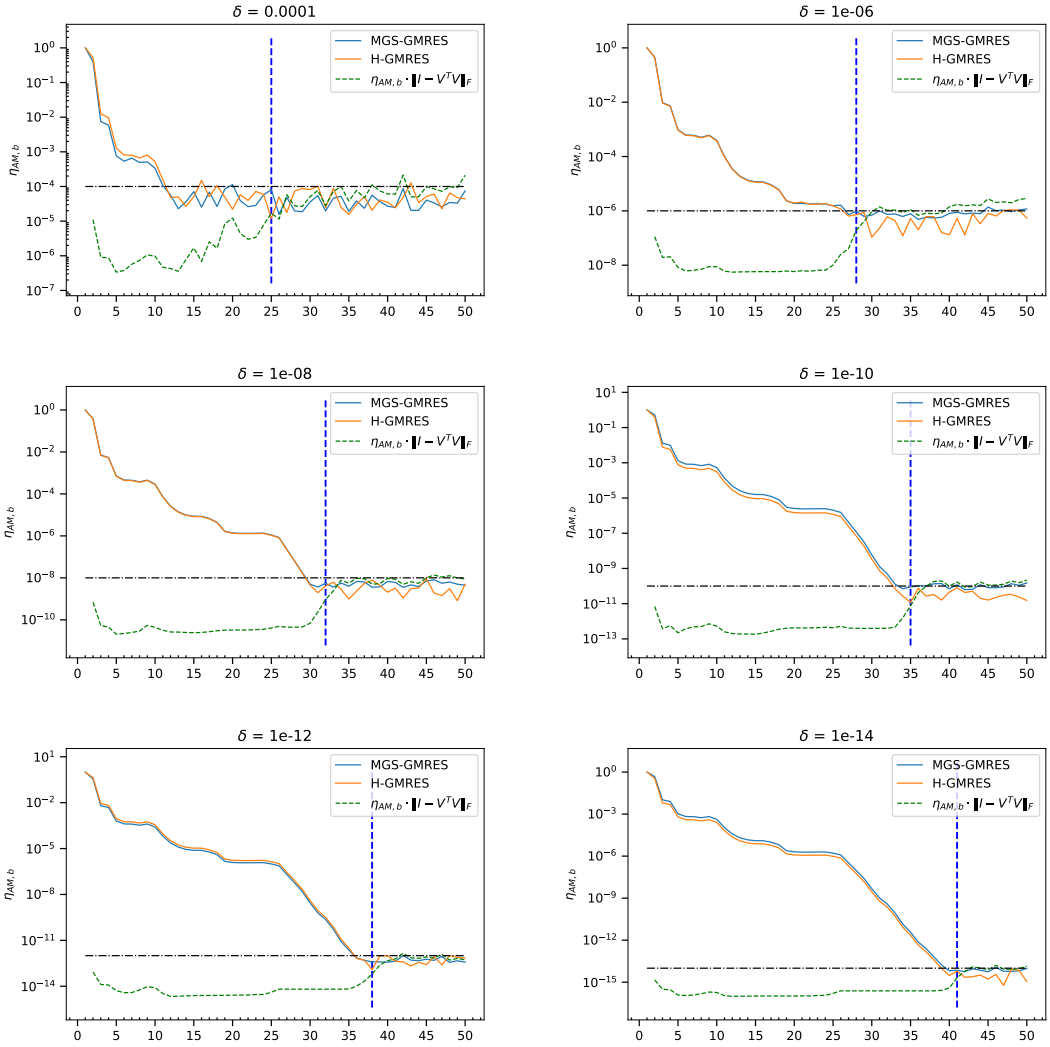

Figure 20: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU(10^{-1}) using δ-normwise representation

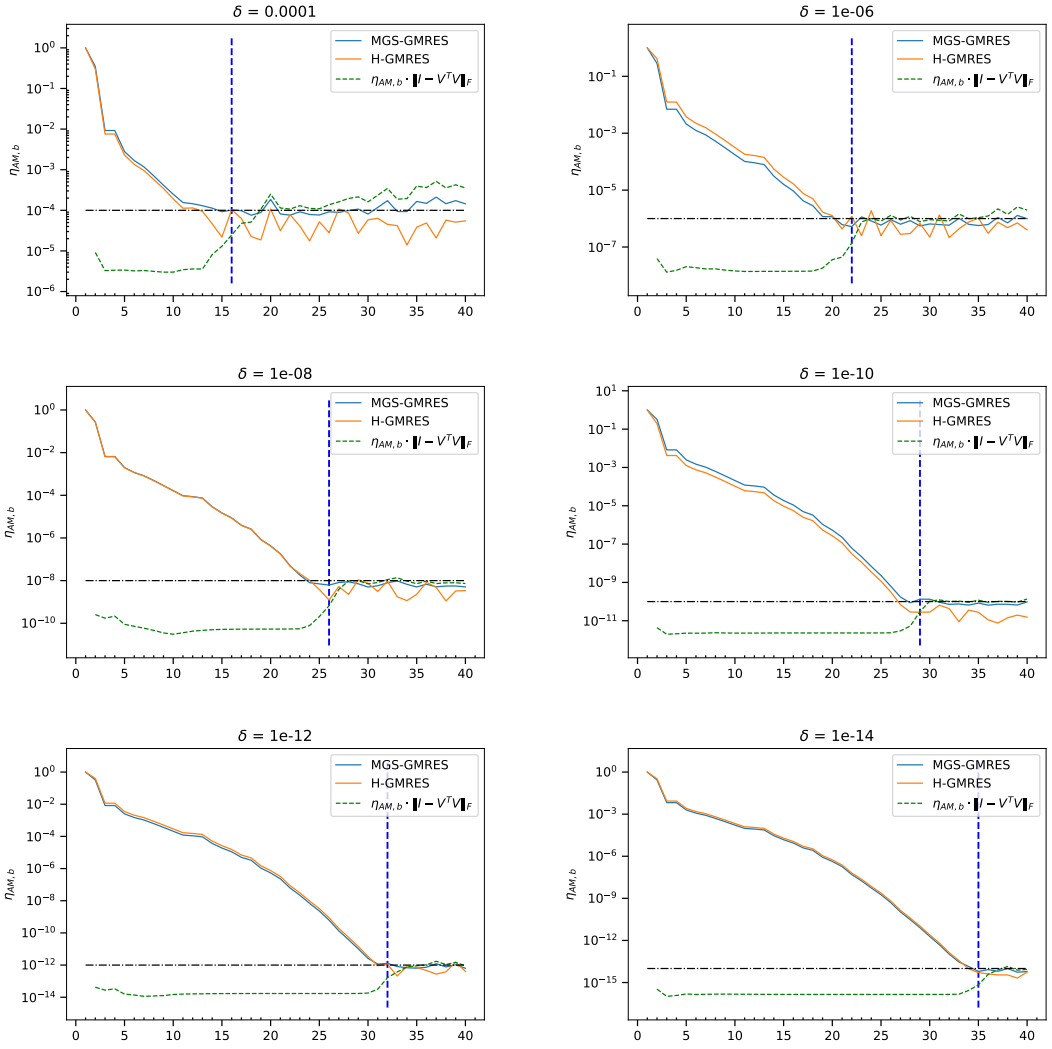

Figure 21: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU(10^{-1}) using δ-normwise representation

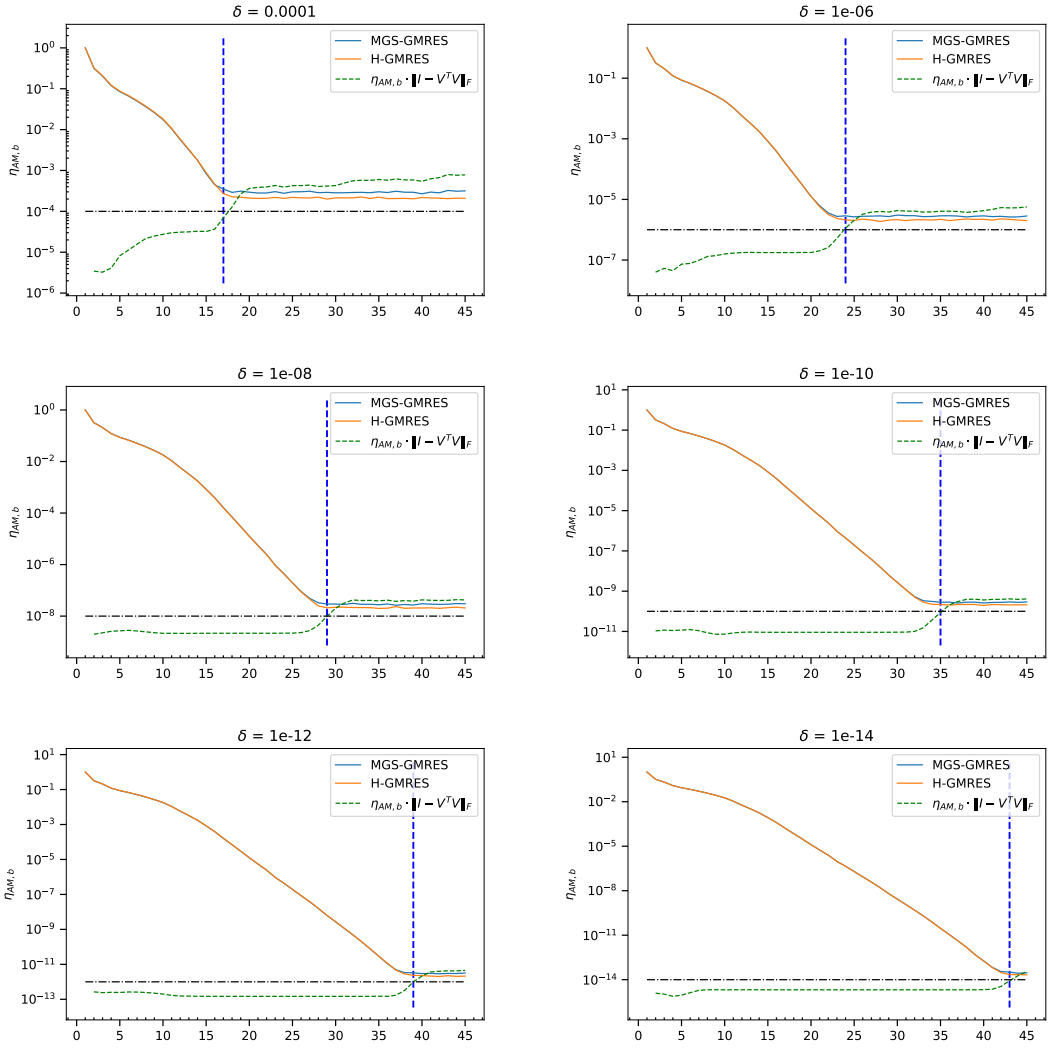

Figure 22: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($3 \cdot 10^{-1}$) using δ-normwise representation

C Results with componentwise SZ compression and 64-bit calculation

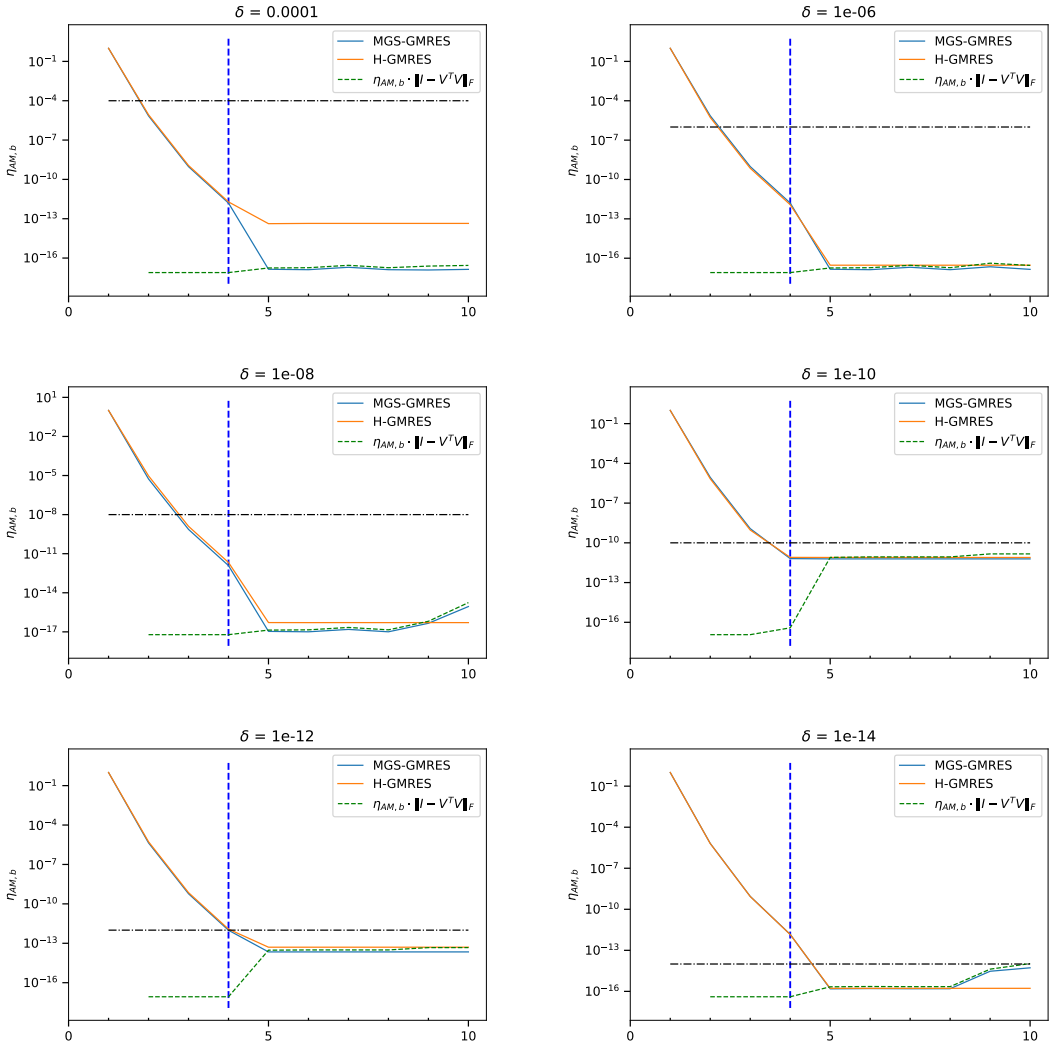

Figure 23: Convergence history of $\eta_{AM,b}$ for arc130 with ILU($8 \cdot 10^{-4}$) using componentwise SZ compression

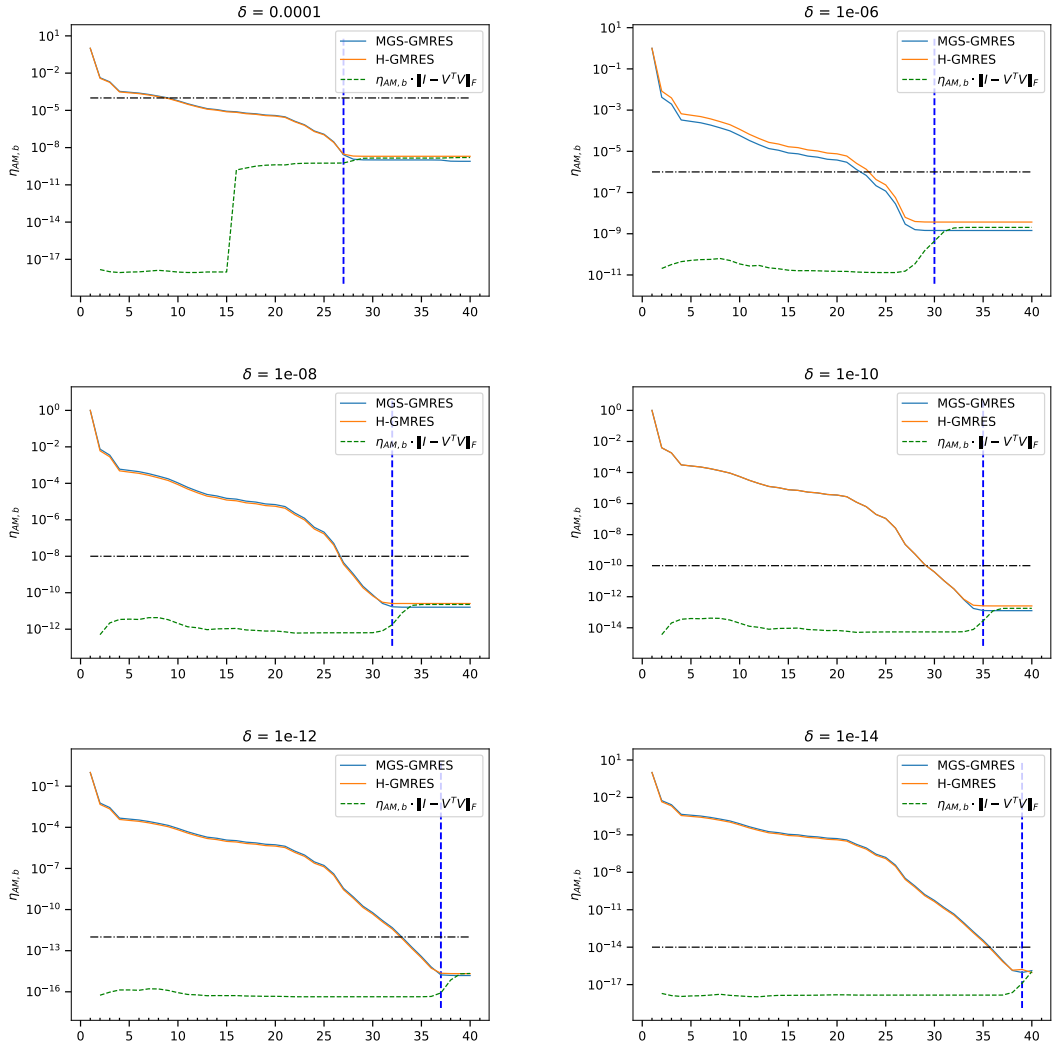

Figure 24: Convergence history of $\eta_{AM,b}$ for cavity03 with ILU(10^{-2}) using componentwise SZ compression

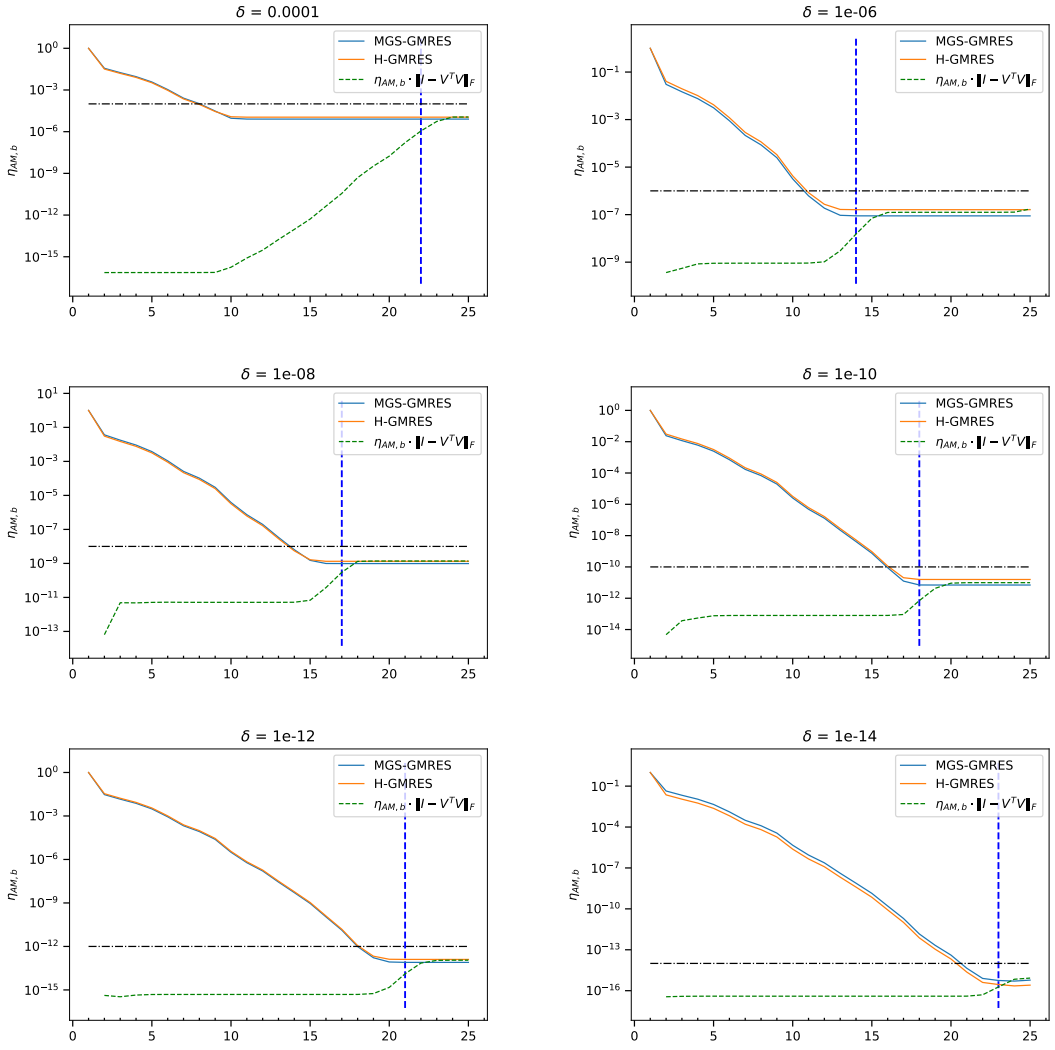

Figure 25: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU(10^{-2}) using componentwise SZ compression

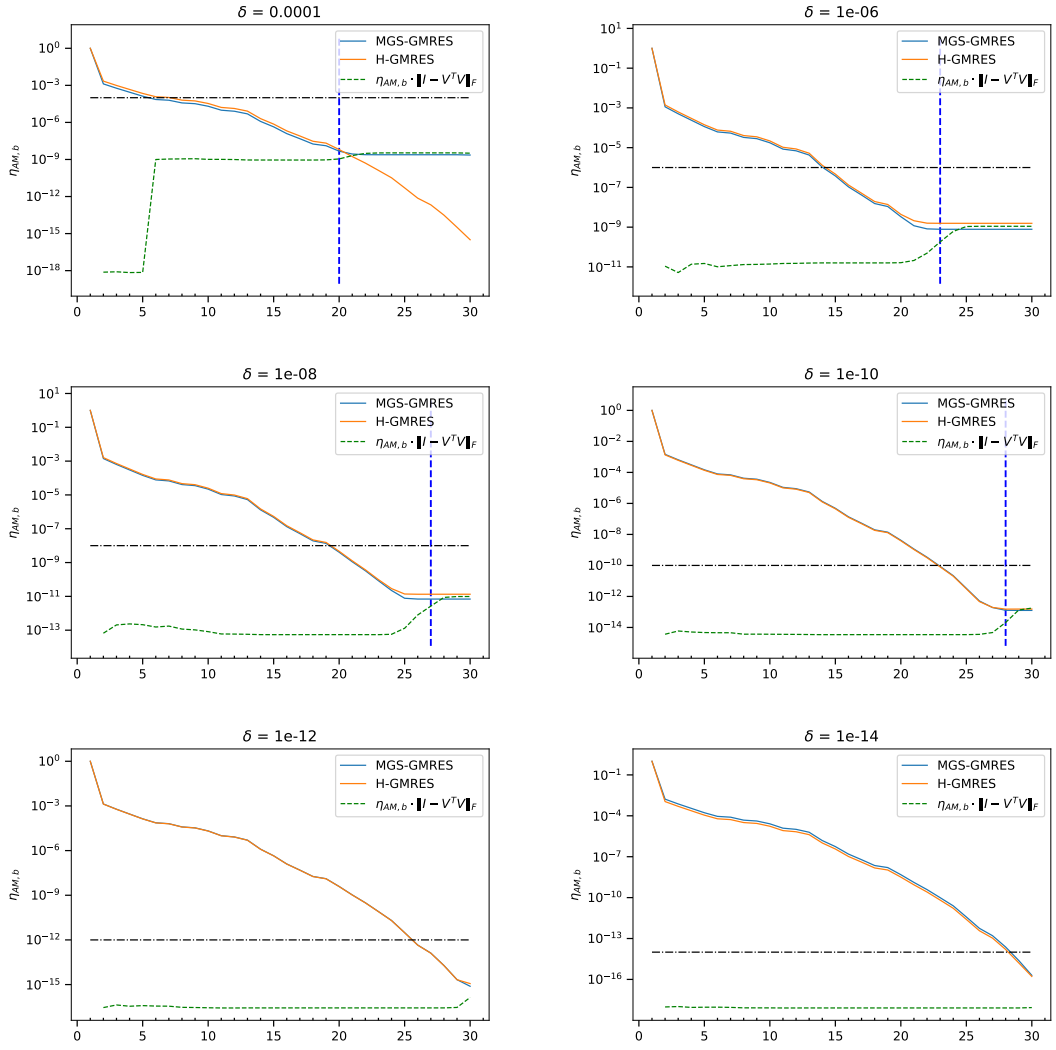

Figure 26: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using componentwise SZ compression

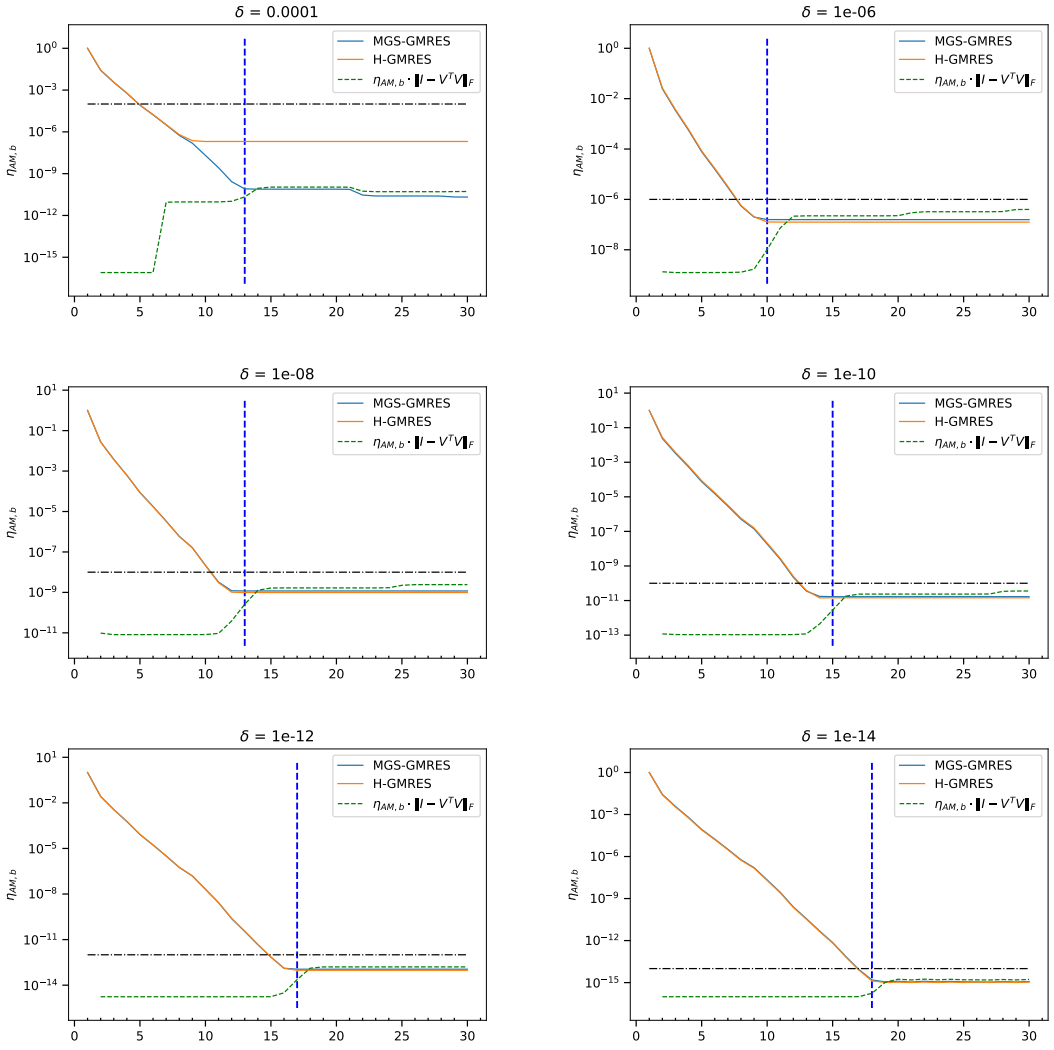

Figure 27: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU(10^{-1}) using componentwise SZ compression

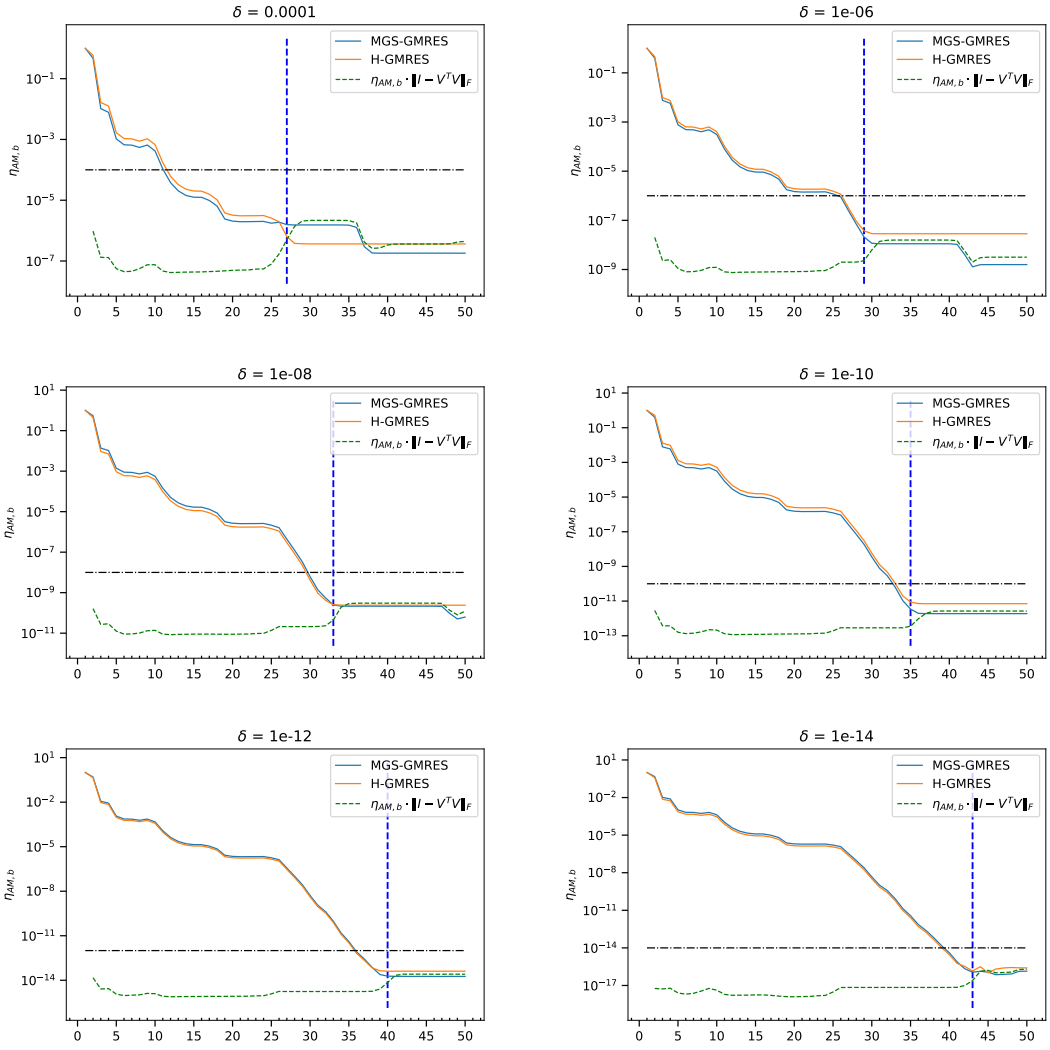

Figure 28: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU(10^{-1}) using componentwise SZ compression

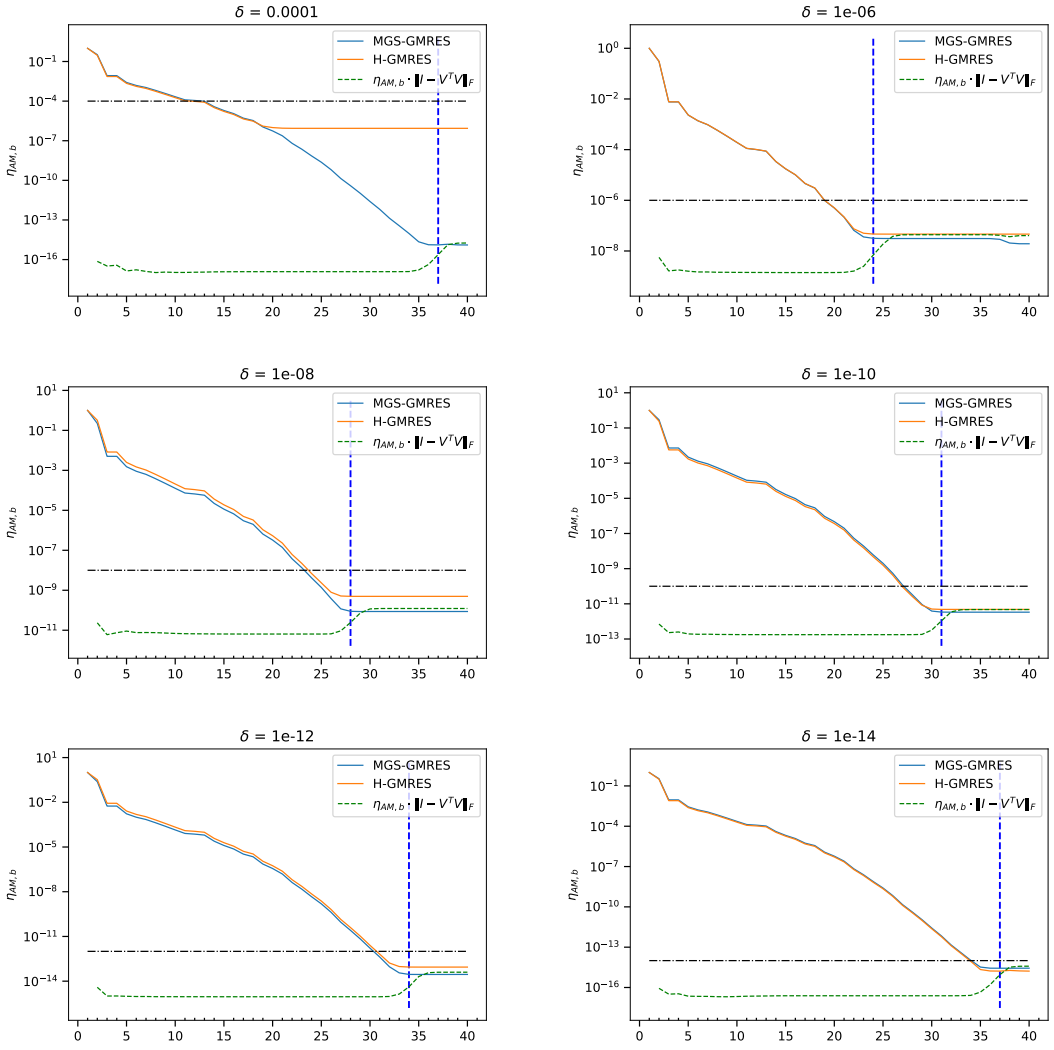

Figure 29: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU(10^{-1}) using componentwise SZ compression

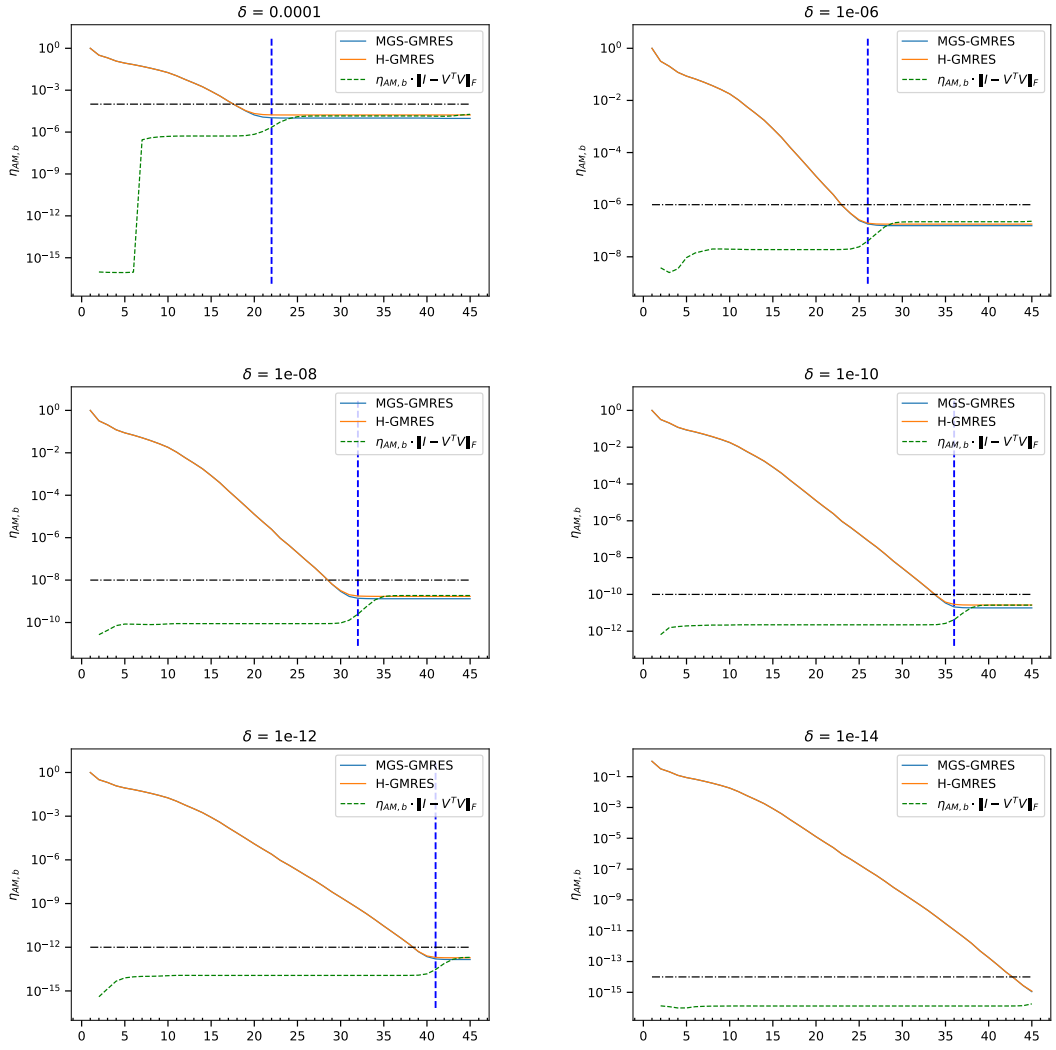

Figure 30: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($3 \cdot 10^{-1}$) using componentwise SZ compression

D Results with normwise SZ compression and 64-bit calculation

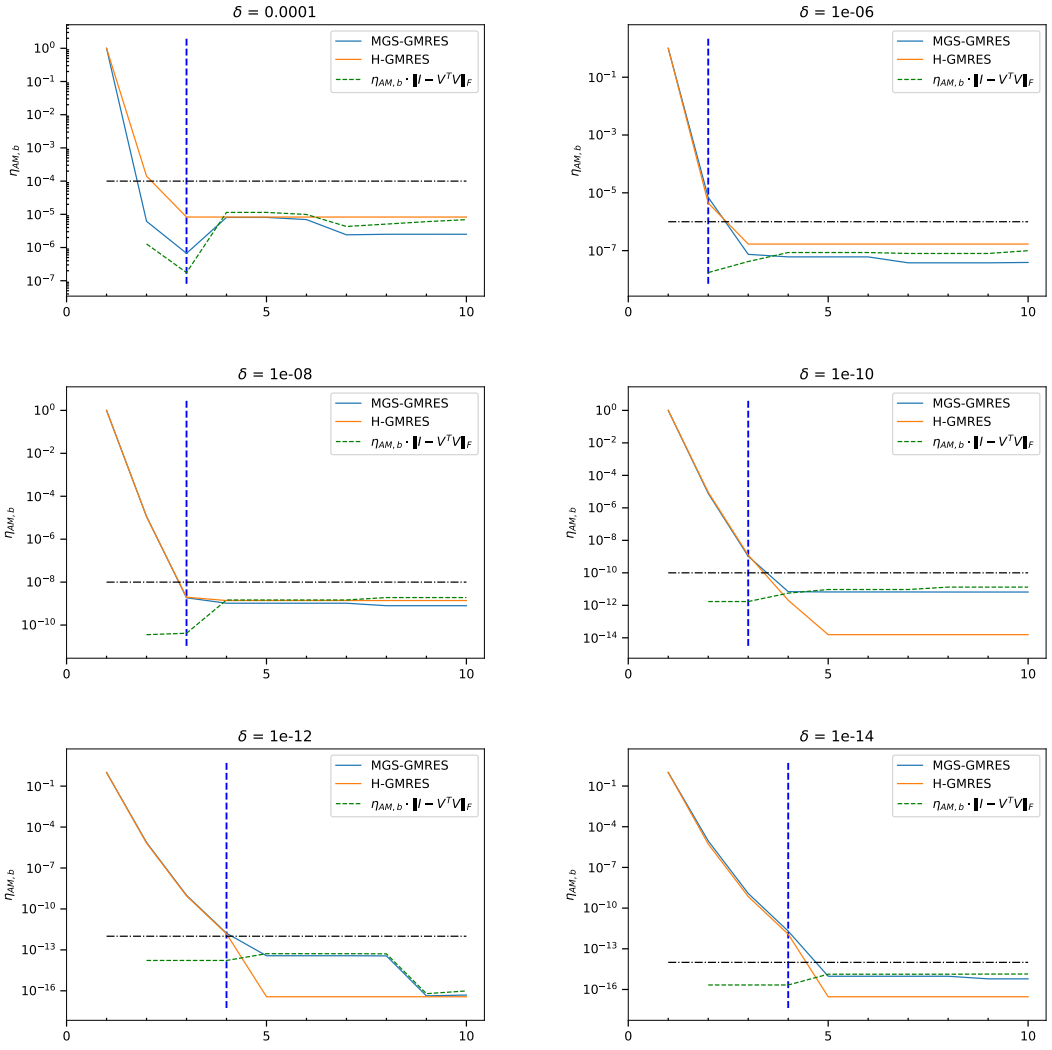

Figure 31: Convergence history of $\eta_{AM,b}$ for arc130 with ILU($8 \cdot 10^{-4}$) using normwise SZ compression

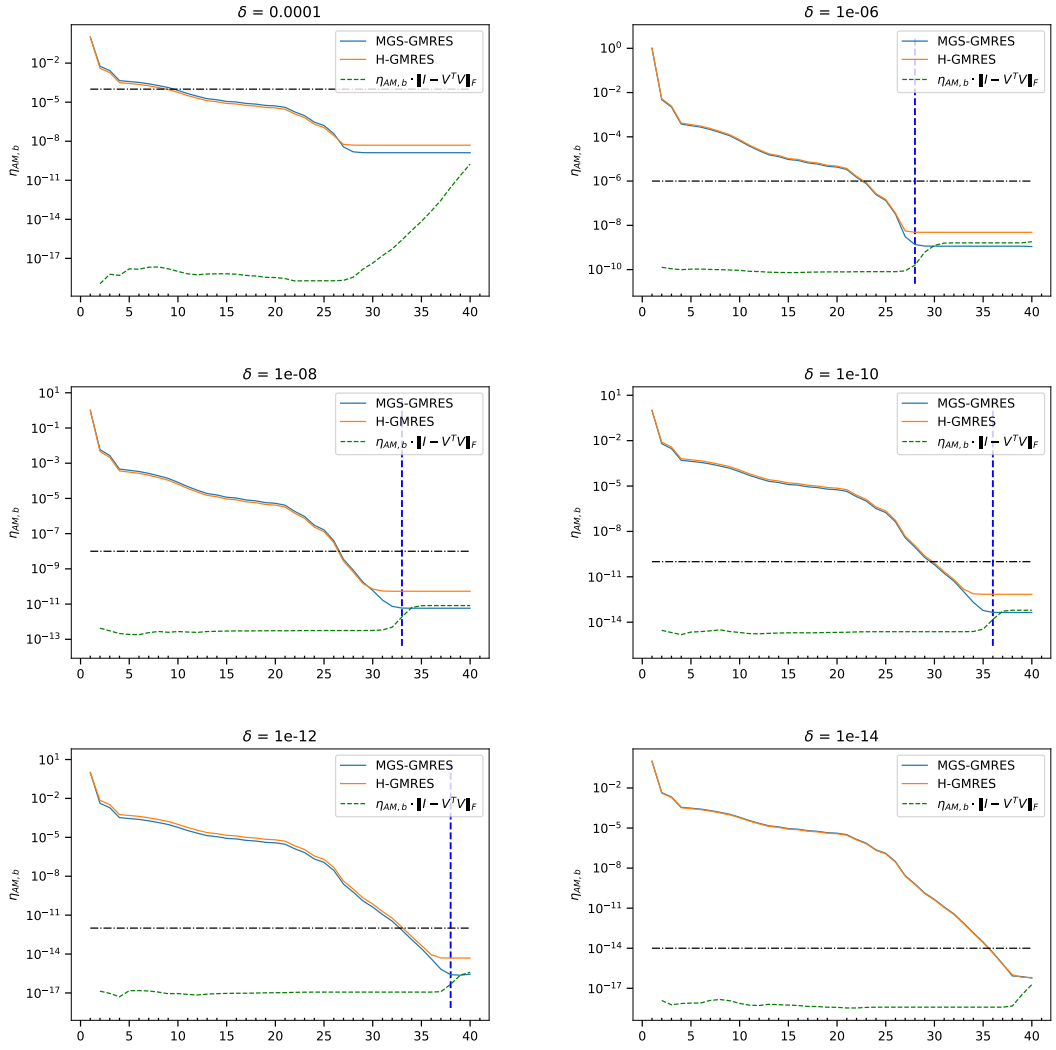

Figure 32: Convergence history of $\eta_{AM,b}$ for cavity03 with ILU(10^{-2}) using normwise SZ compression

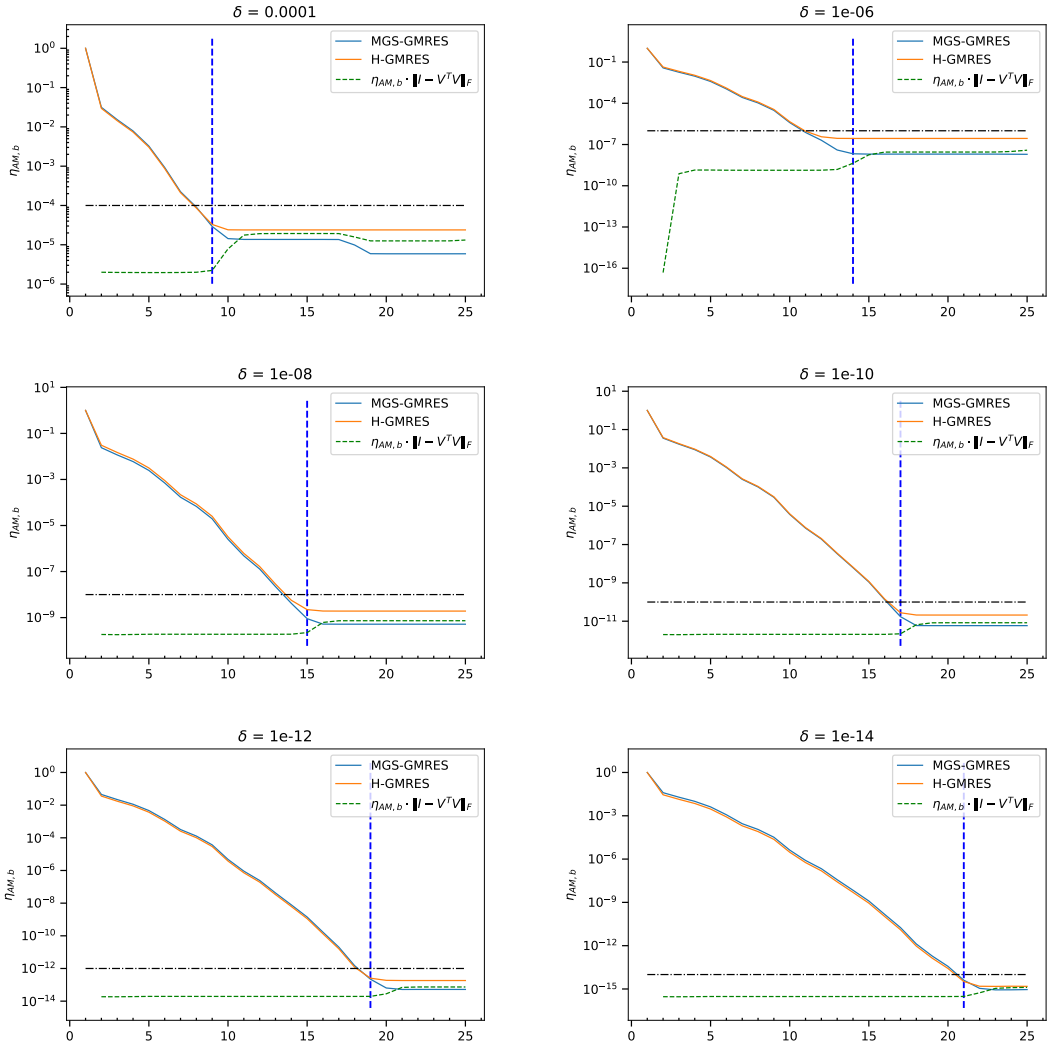

Figure 33: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU(10^{-2}) using normwise SZ compression

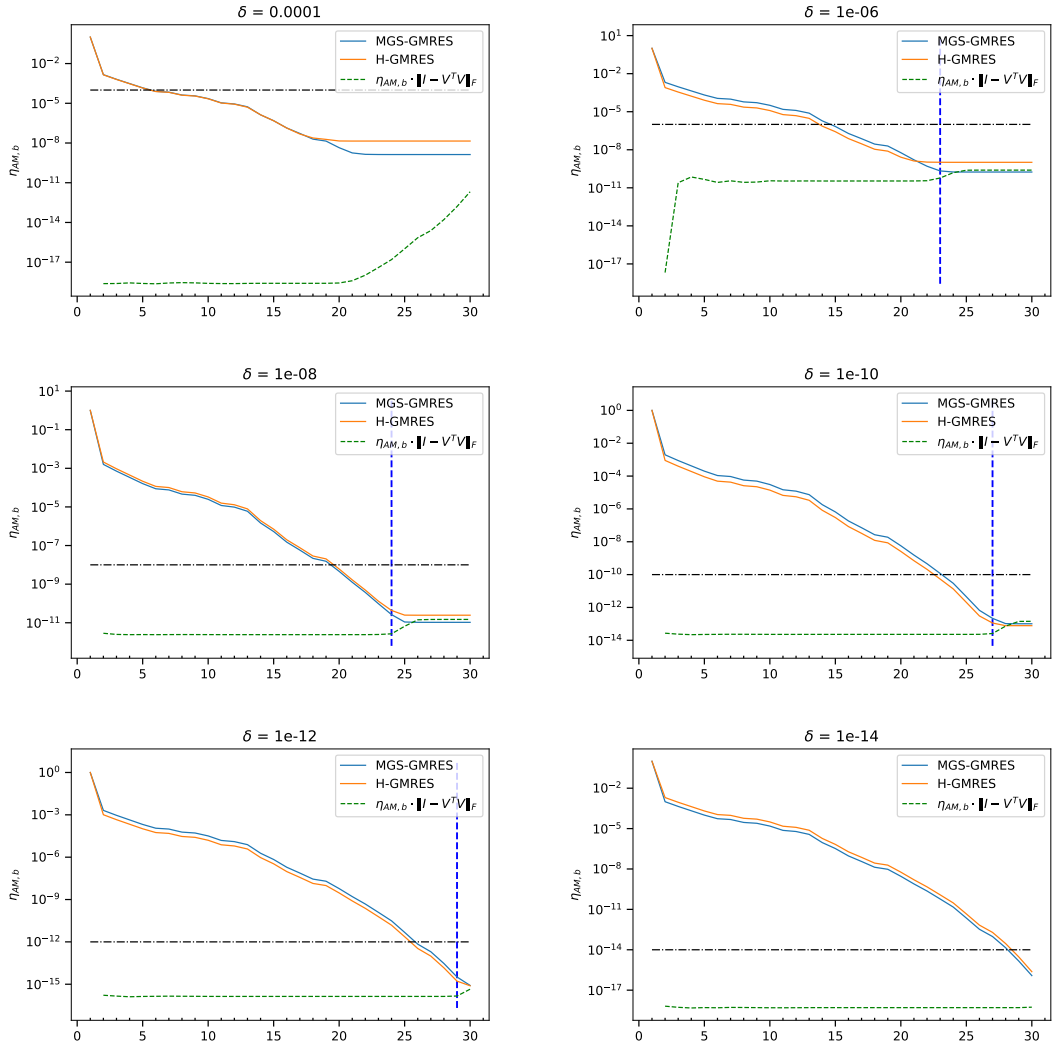

Figure 34: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using normwise SZ compression

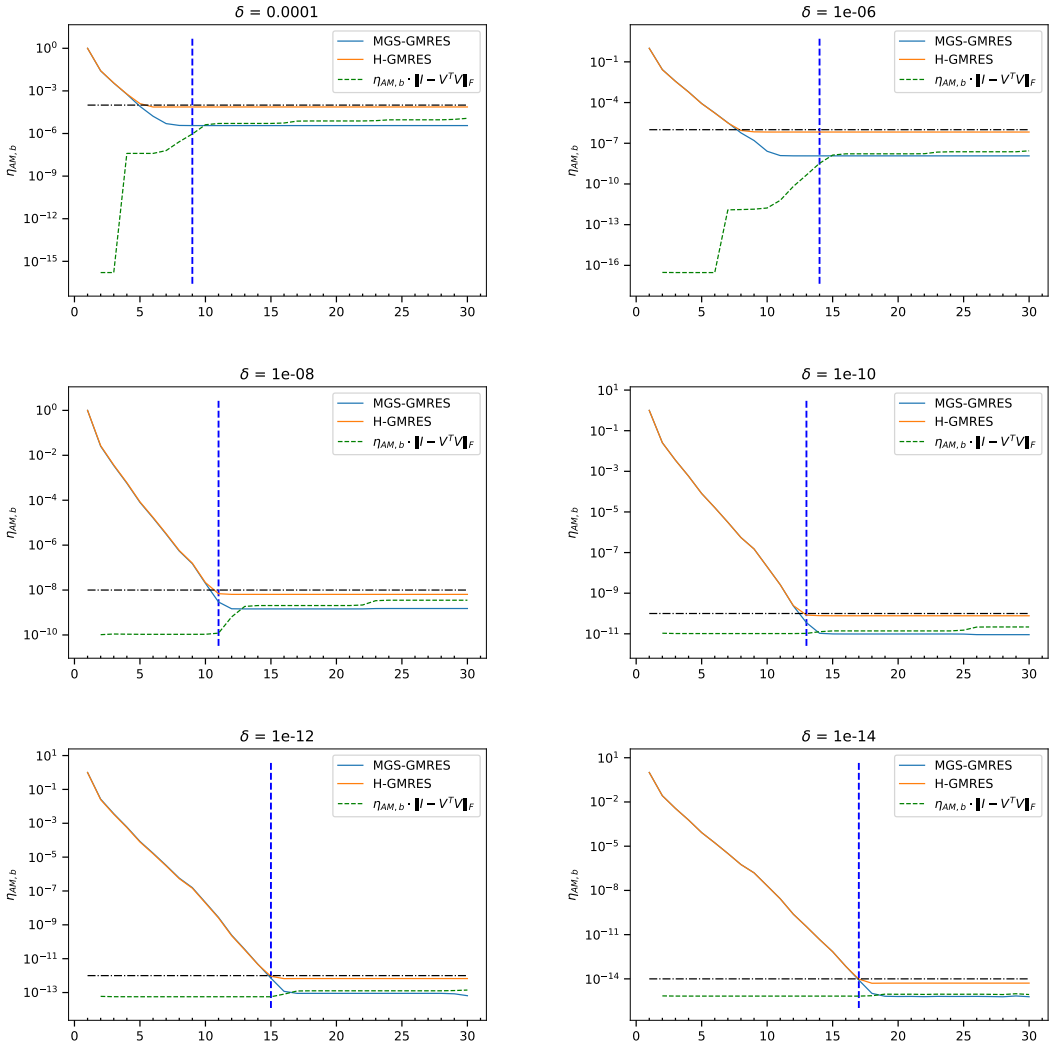

Figure 35: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU(10^{-1}) using normwise SZ compression

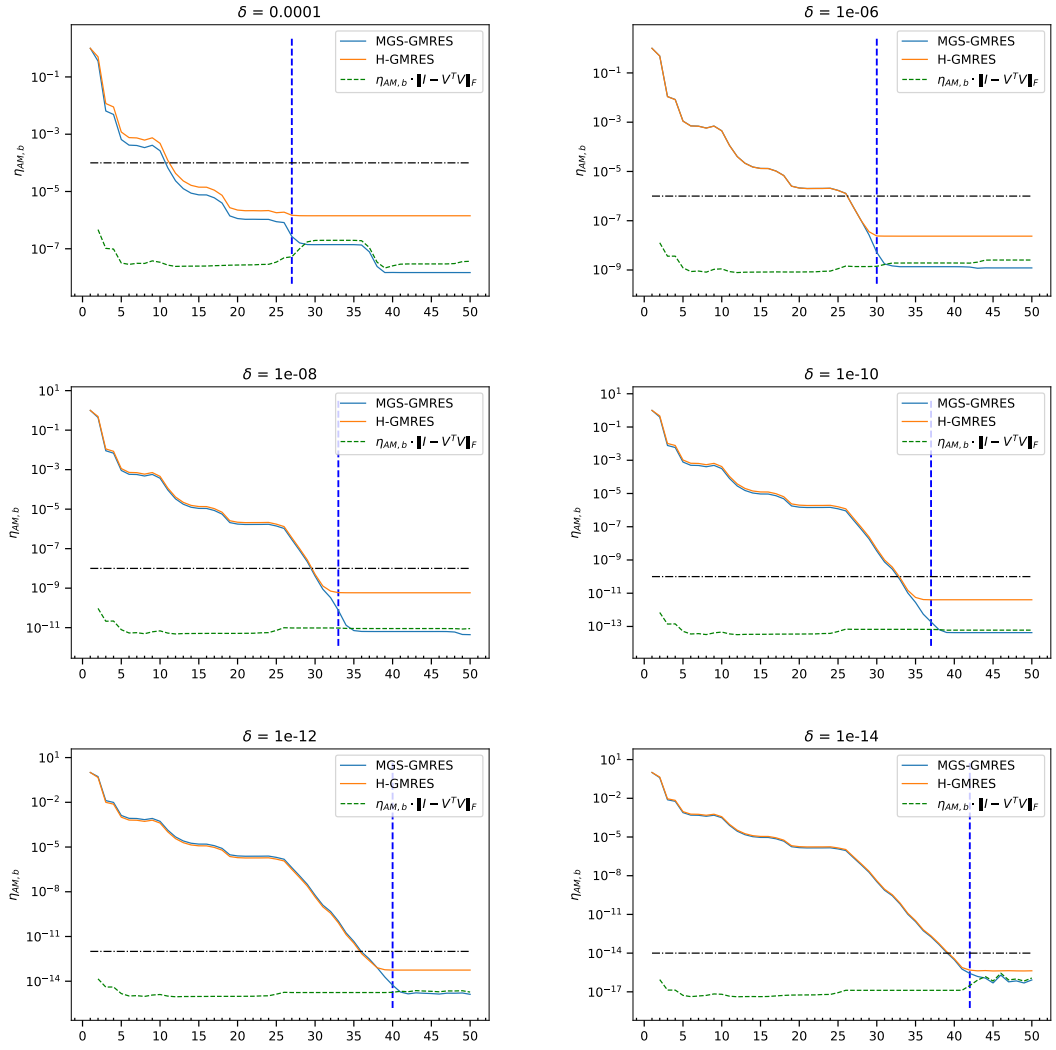

Figure 36: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU(10^{-1}) using normwise SZ compression

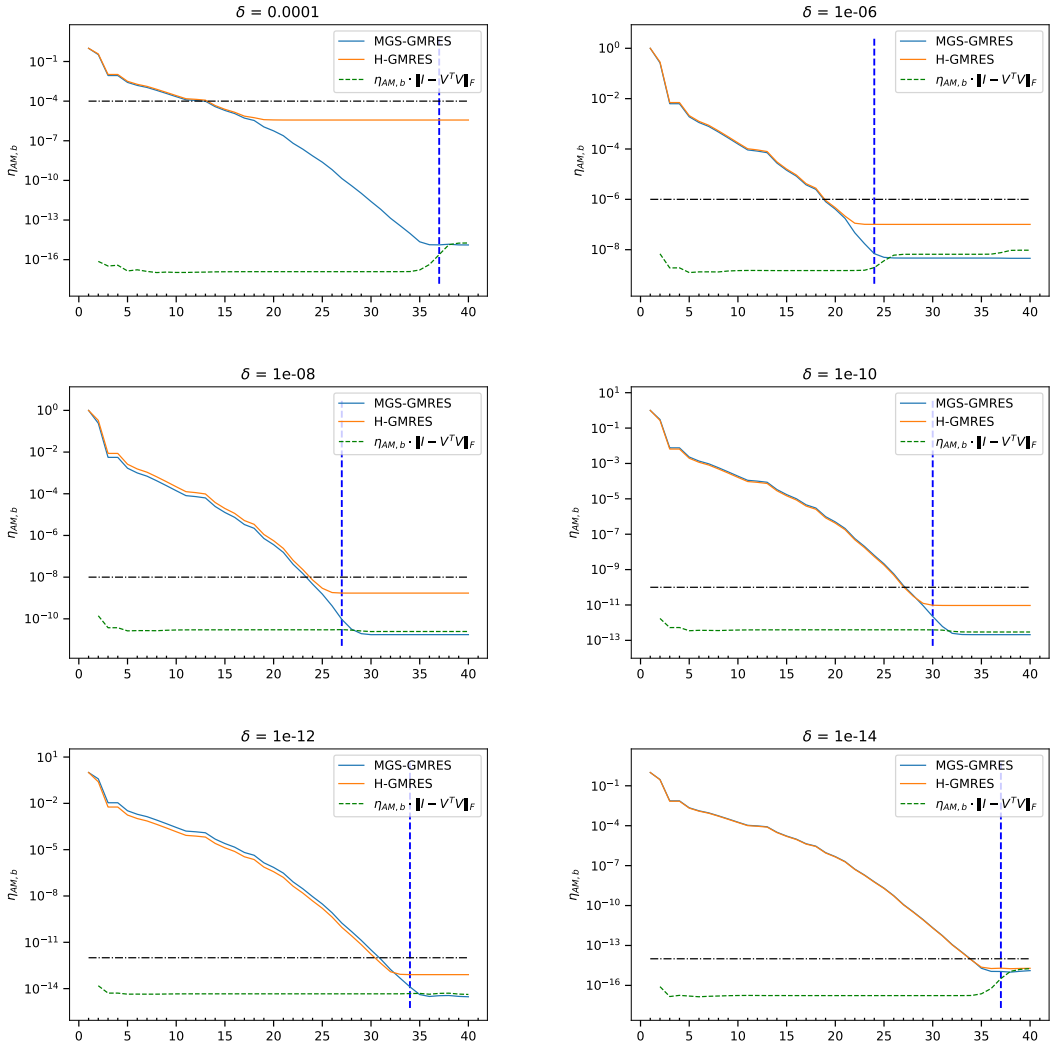

Figure 37: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU(10^{-1}) using normwise SZ compression

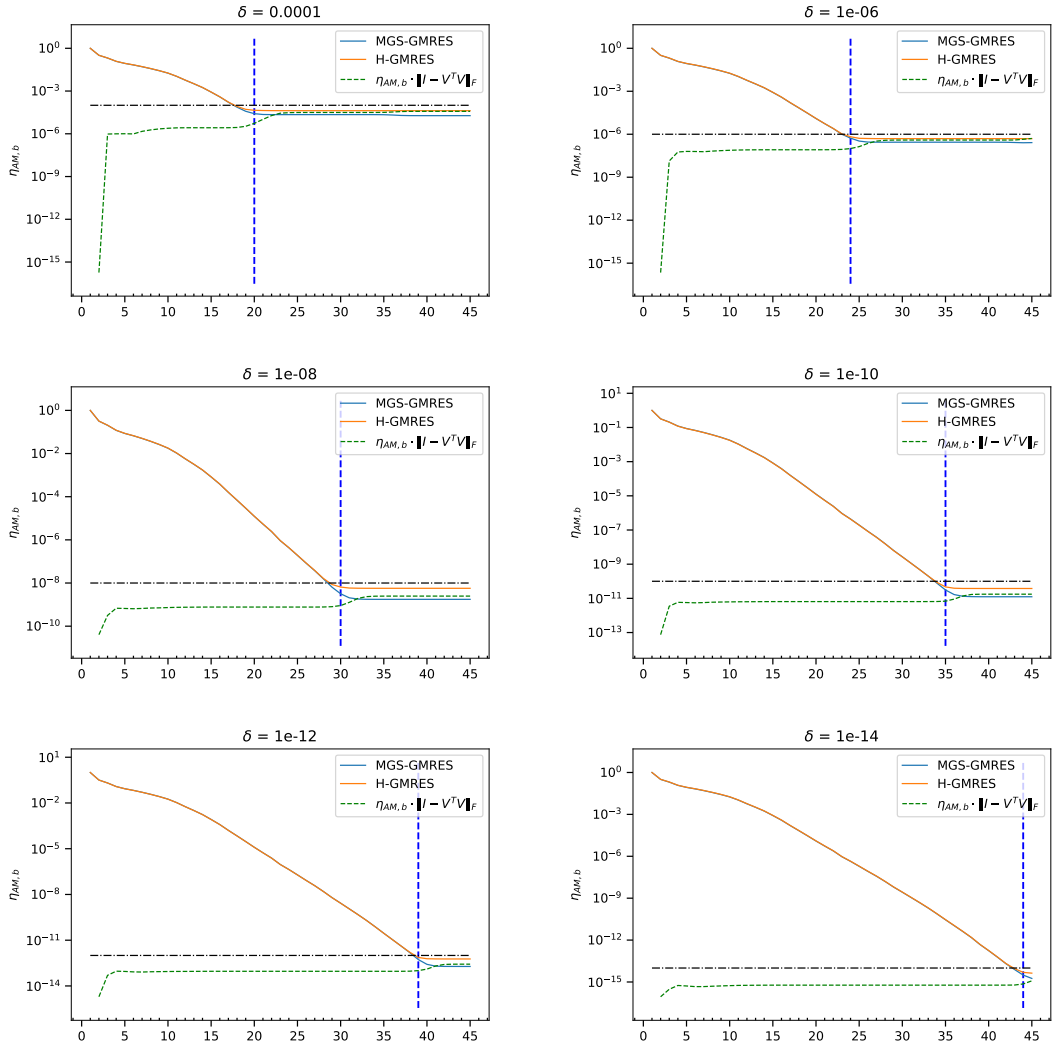

Figure 38: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($3 \cdot 10^{-1}$) using normwise SZ compression

E Results with componentwise perturbations on δ-vectors and 32-bit calculation

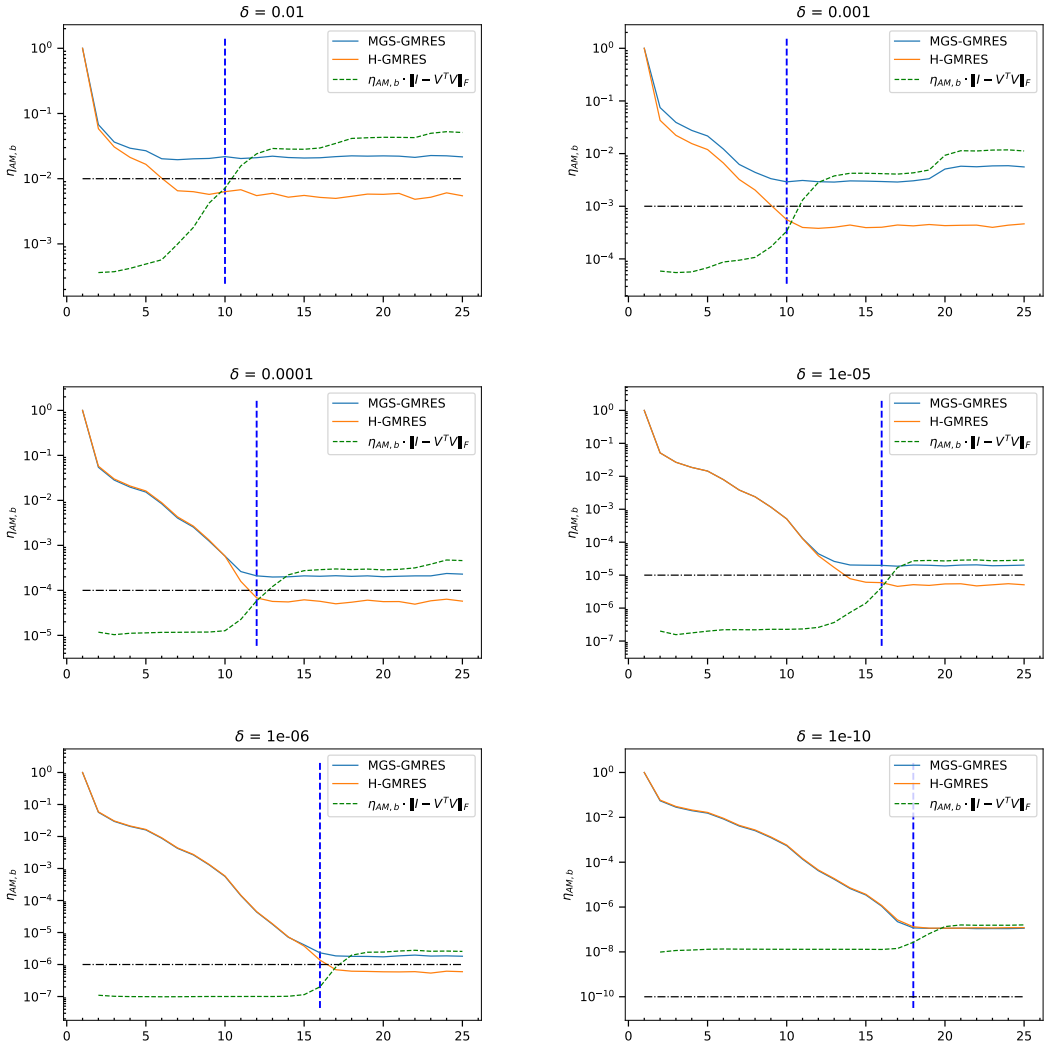

Figure 39: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU($12 \cdot^{-2}$) using δ-componentwise representation

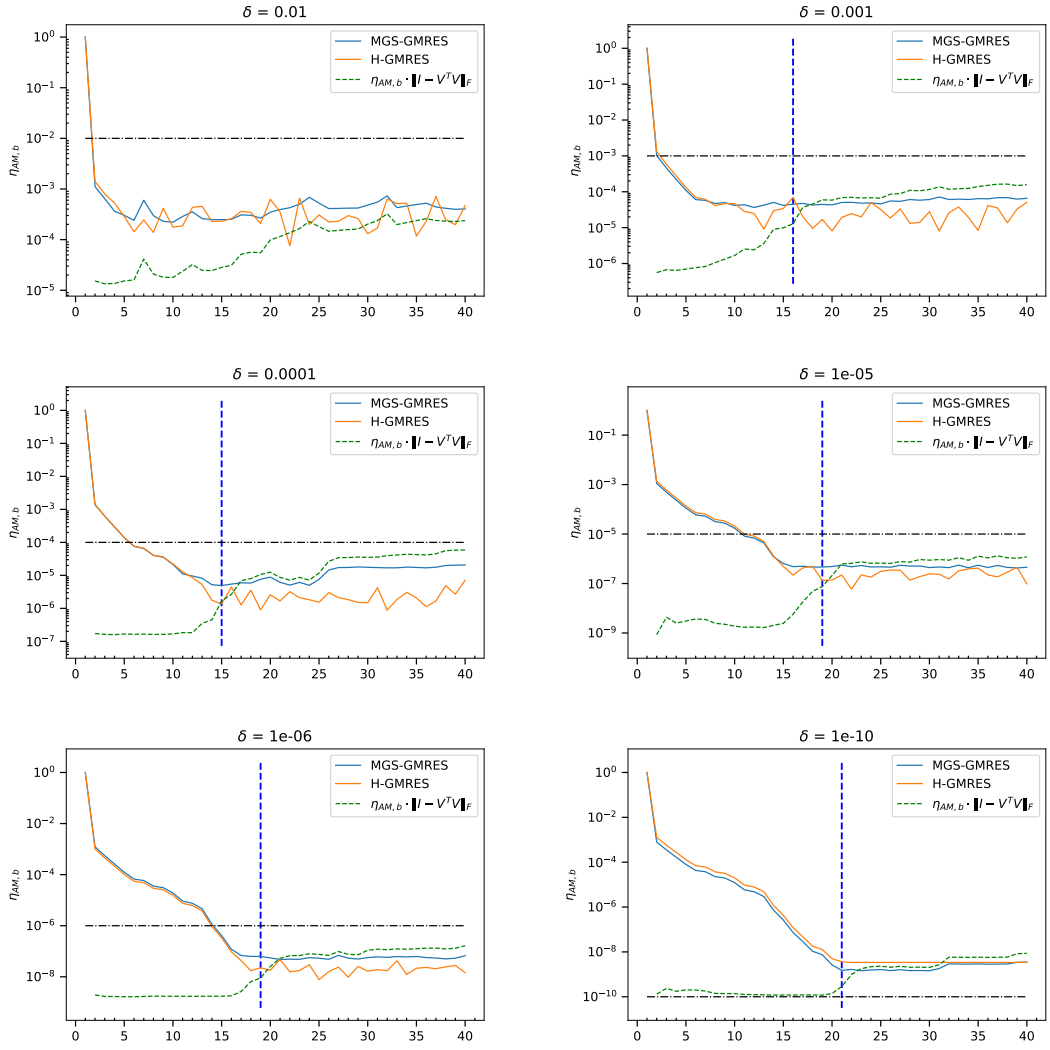

Figure 40: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using δ-componentwise representation

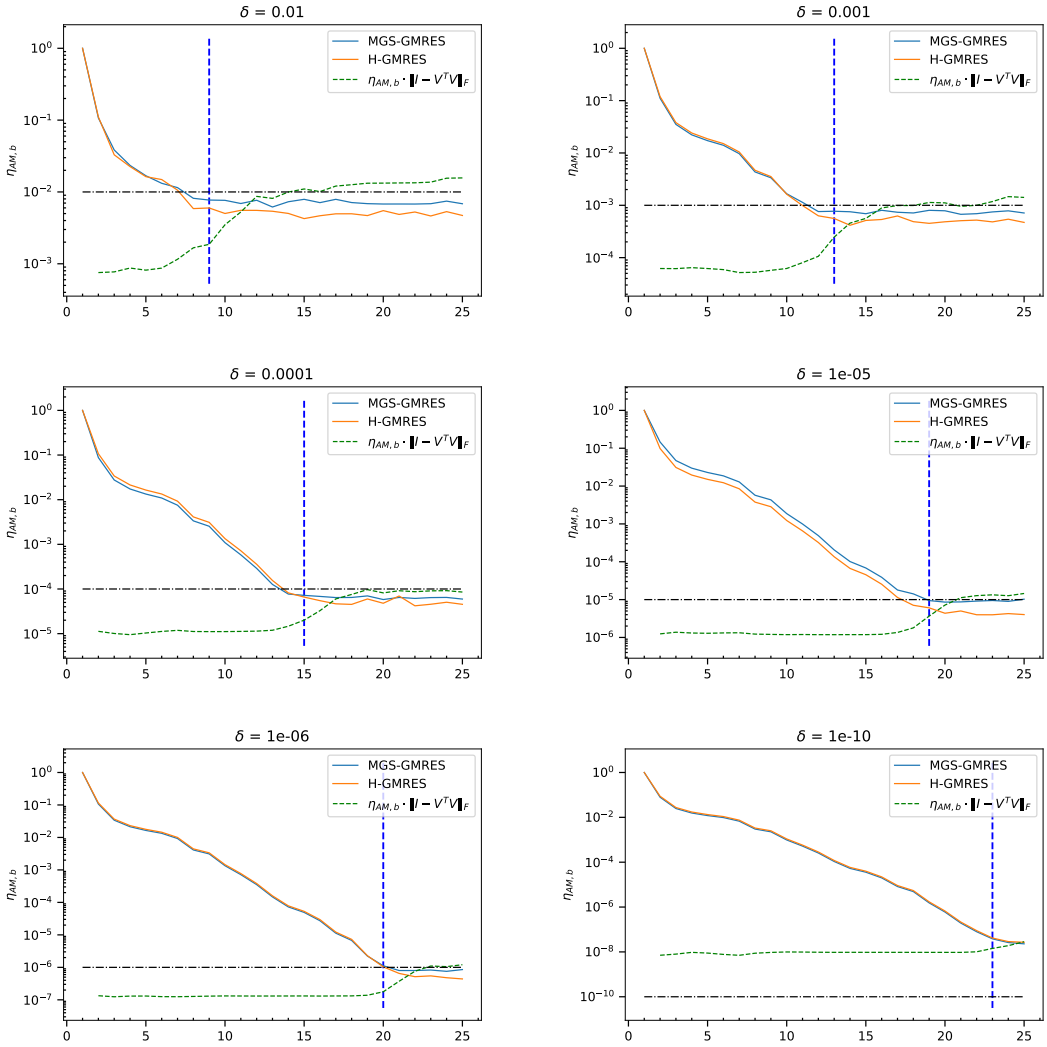

Figure 41: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU($5 \cdot 10^{-1}$) using δ-componentwise representation

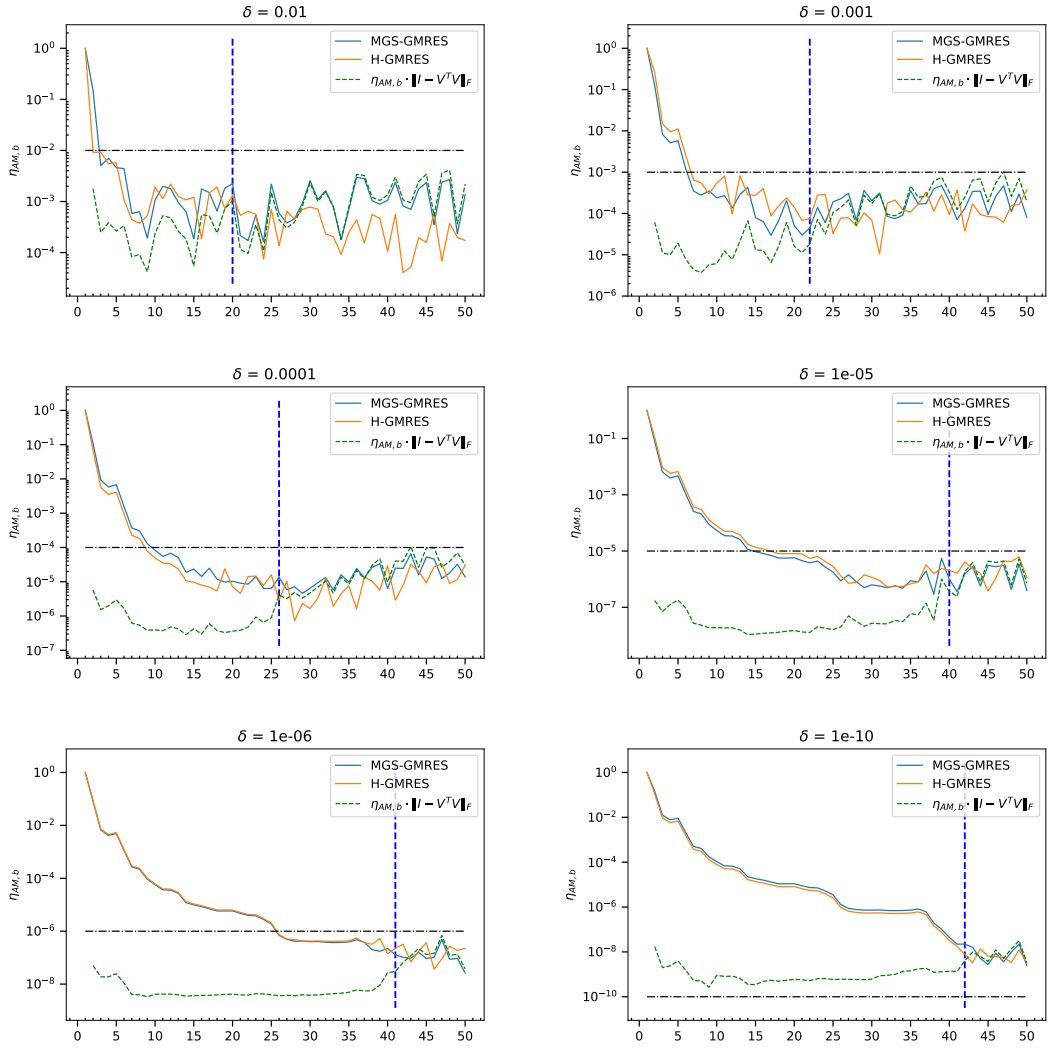

Figure 42: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU($2 \cdot 10^{-1}$) using δ-componentwise representation

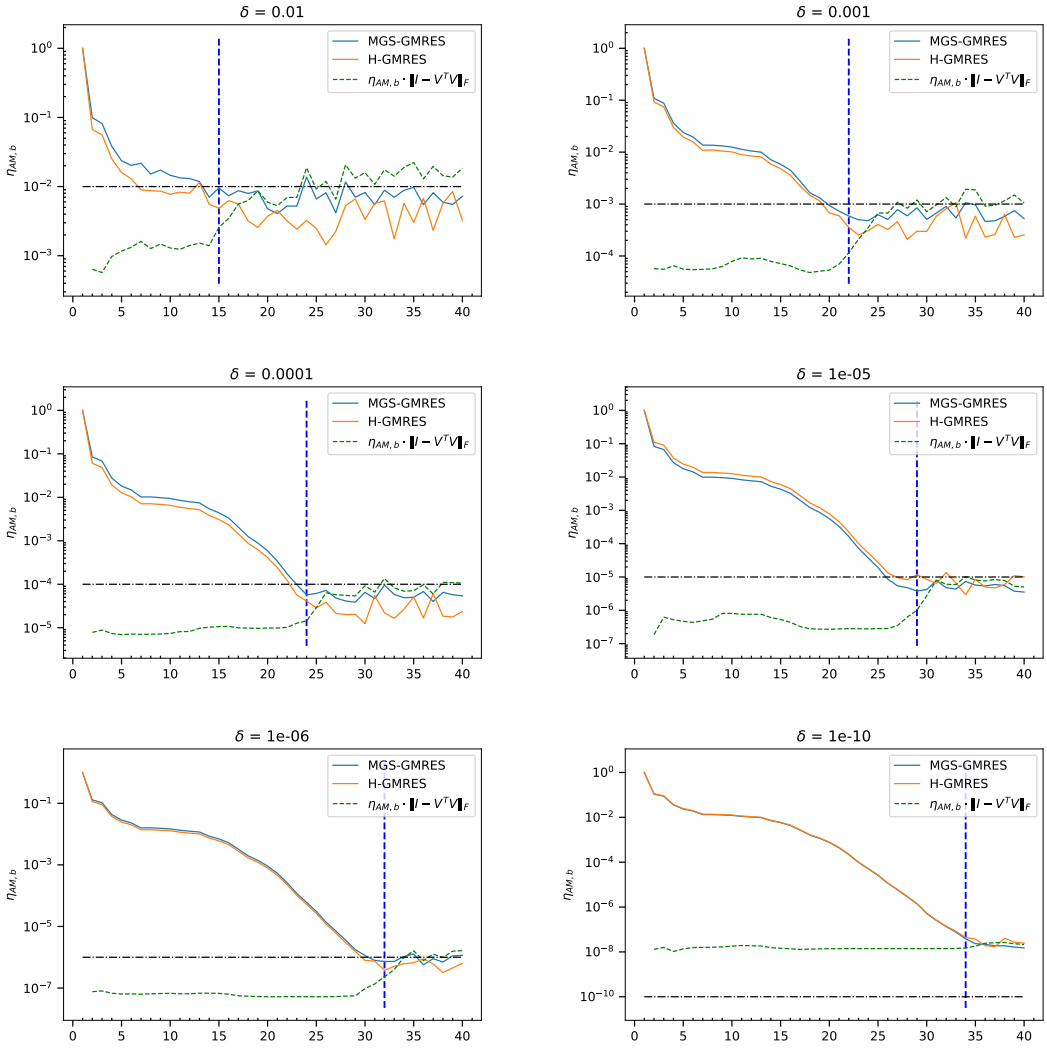

Figure 43: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU($2 \cdot 10^{-1}$) using δ-componentwise representation

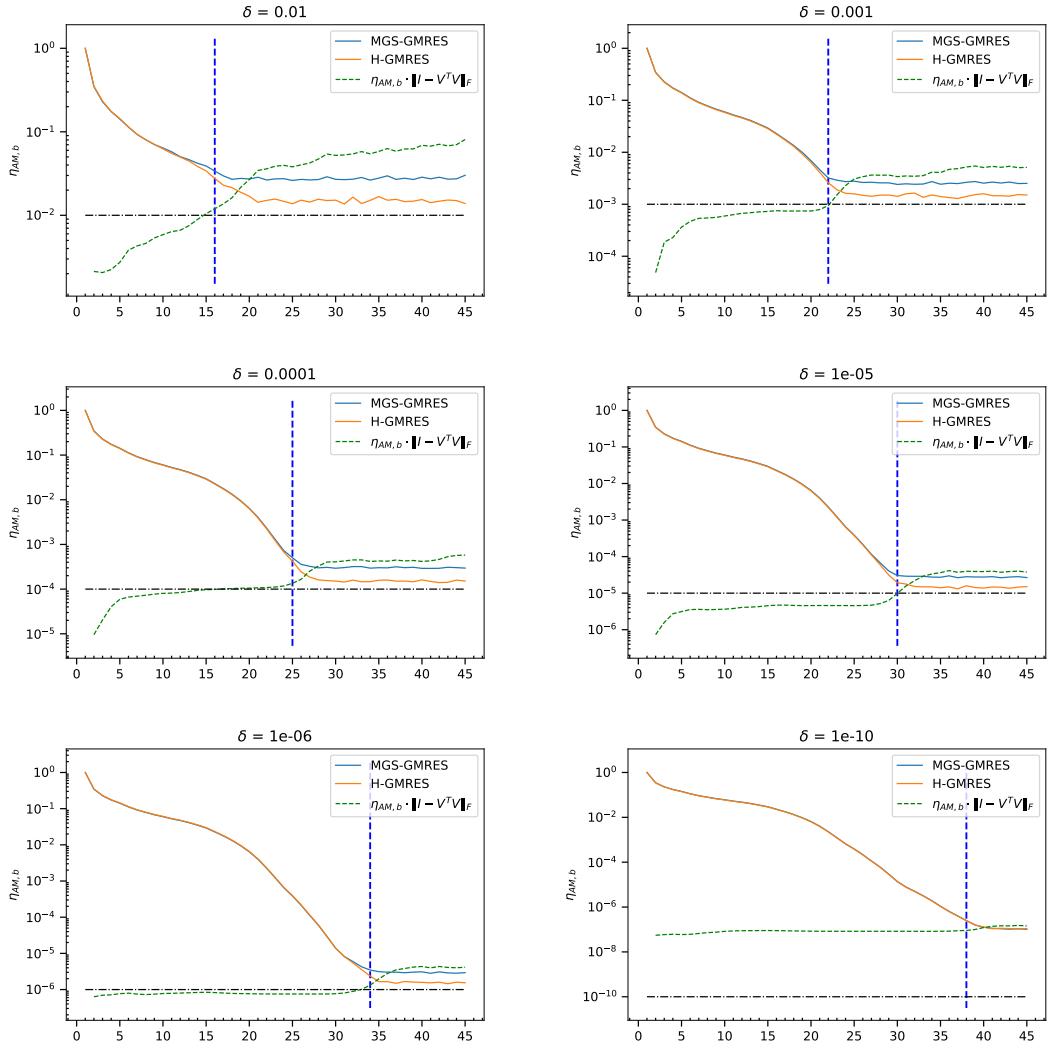

Figure 44: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($5 \cdot 10^{-1}$) using δ-componentwise representation

F Results with normwise perturbations on δ-vectors and 32-bit calculation

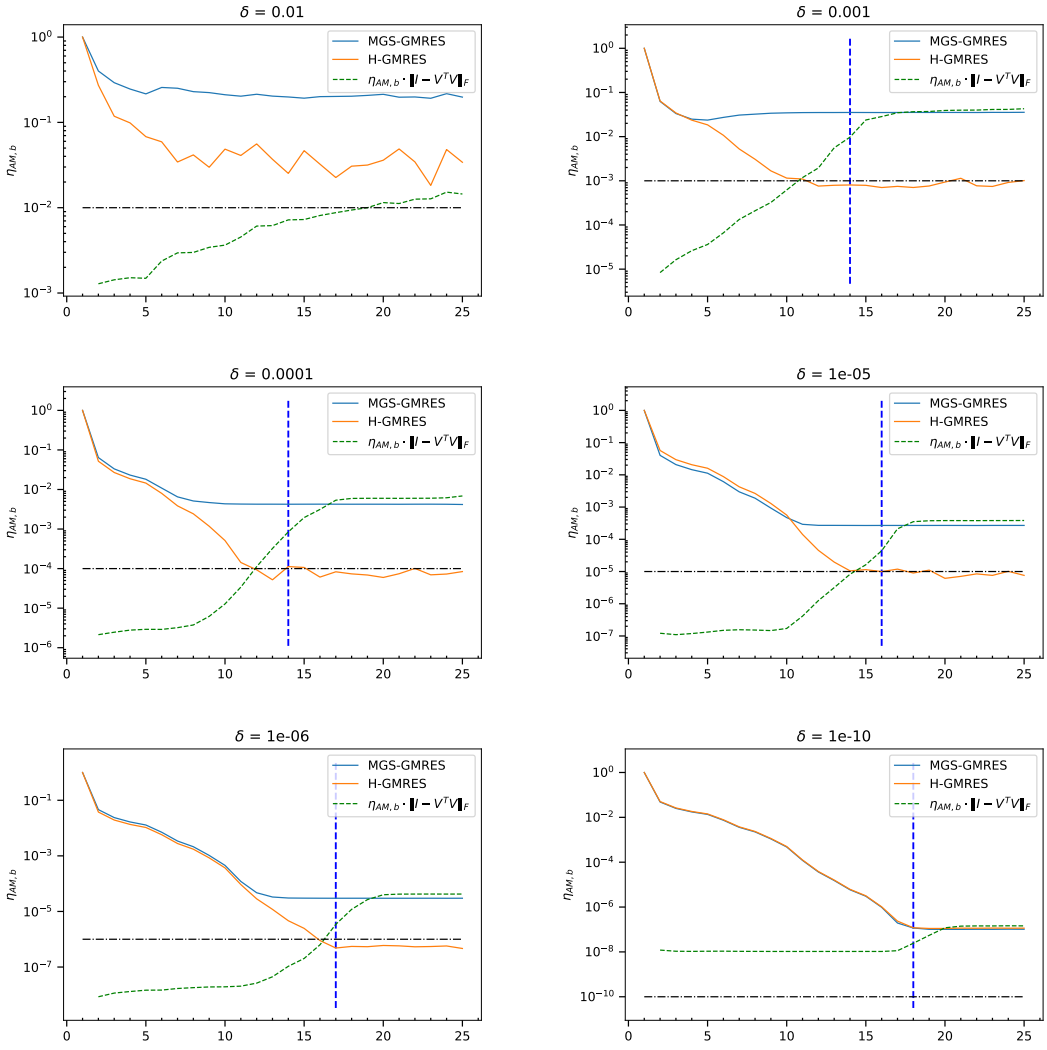

Figure 45: Convergence history of $\eta_{AM,b}$ for e05r0000 with ILU($2 \cdot 10^{-2}$) using δ-normwise representation

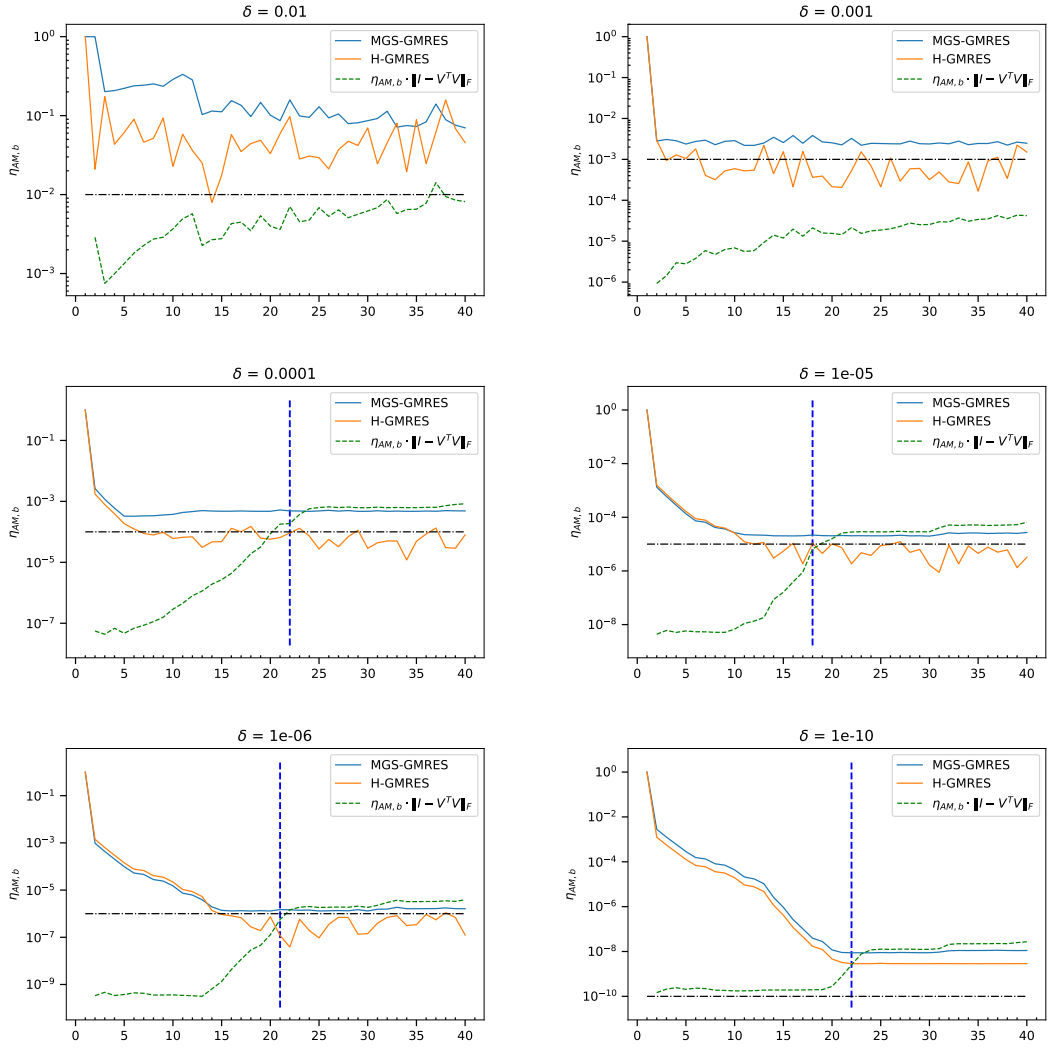

Figure 46: Convergence history of $\eta_{AM,b}$ for e05r0400 with ILU(10^{-2}) using δ-normwise representation

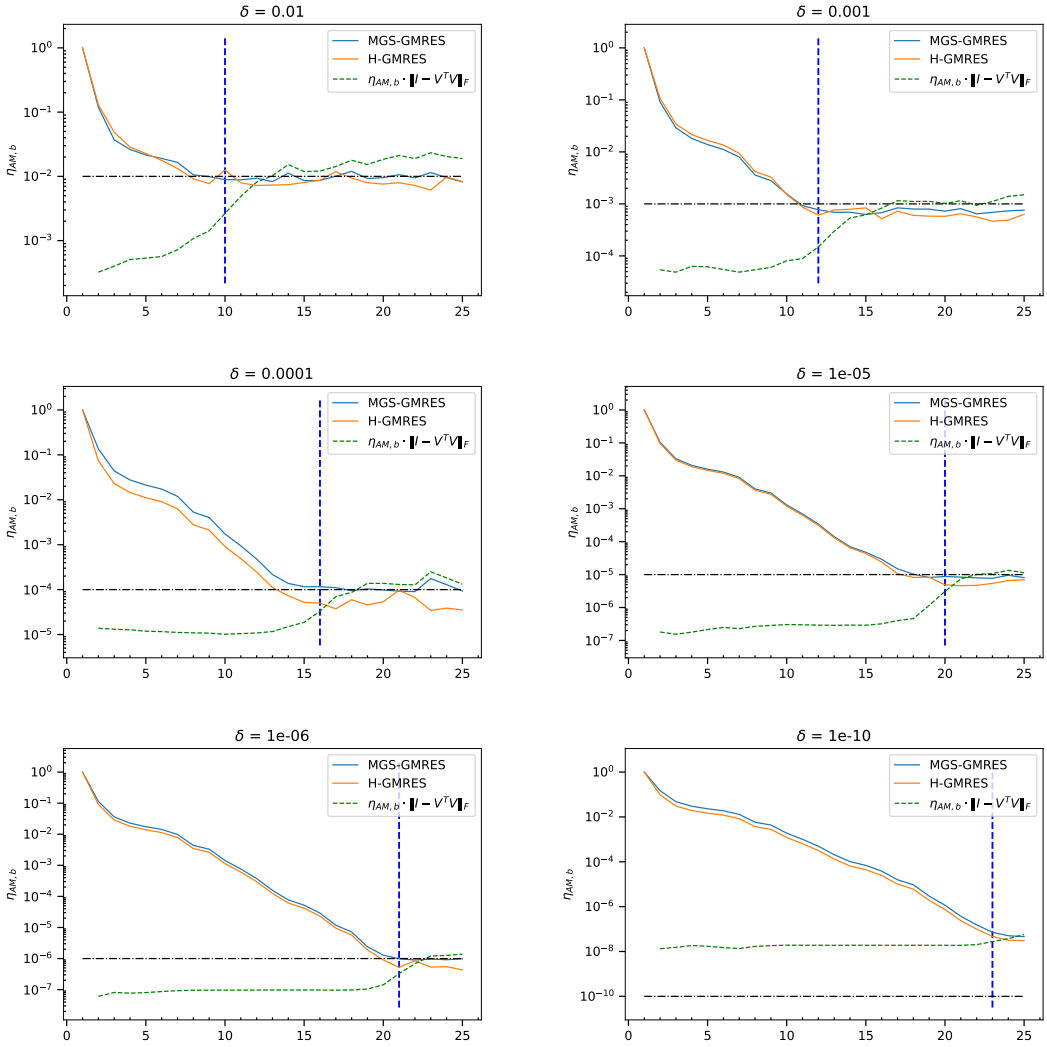

Figure 47: Convergence history of $\eta_{AM,b}$ for gre_115 with ILU($5 \cdot 10^{-1}$) using δ-normwise representation

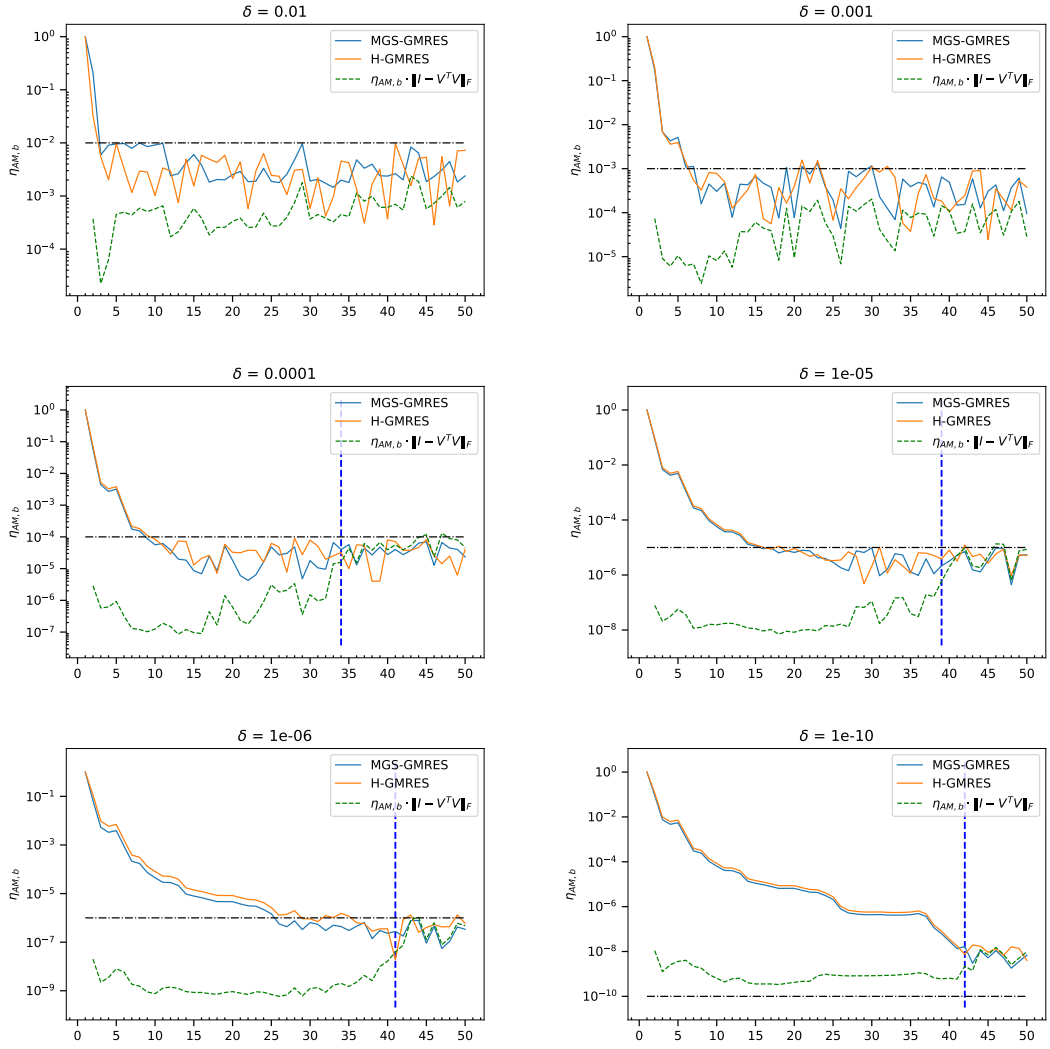

Figure 48: Convergence history of $\eta_{AM,b}$ for gre_185 with ILU($2 \cdot 10^{-1}$) using δ-normwise representation

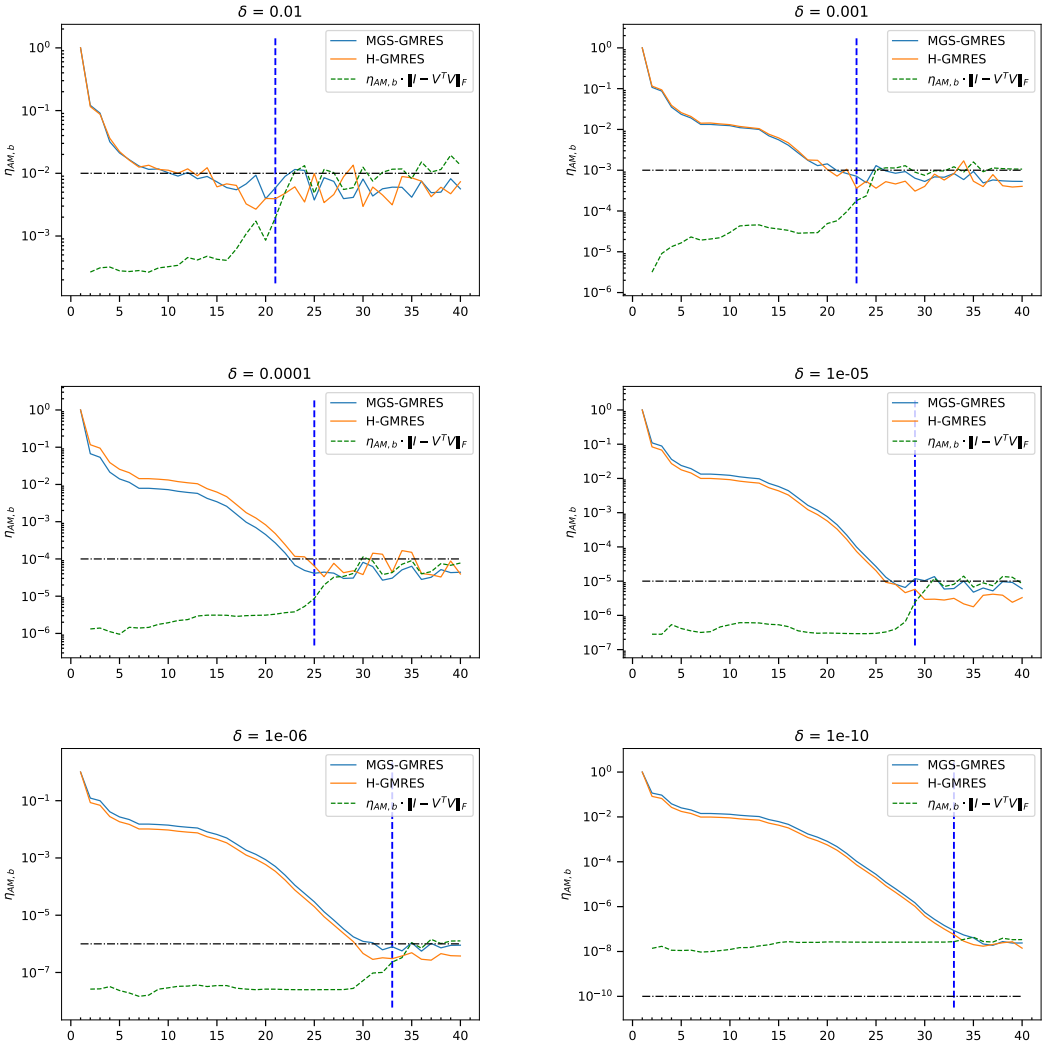

Figure 49: Convergence history of $\eta_{AM,b}$ for gre_343 with ILU($2 \cdot 10^{-1}$) using δ-normwise representation

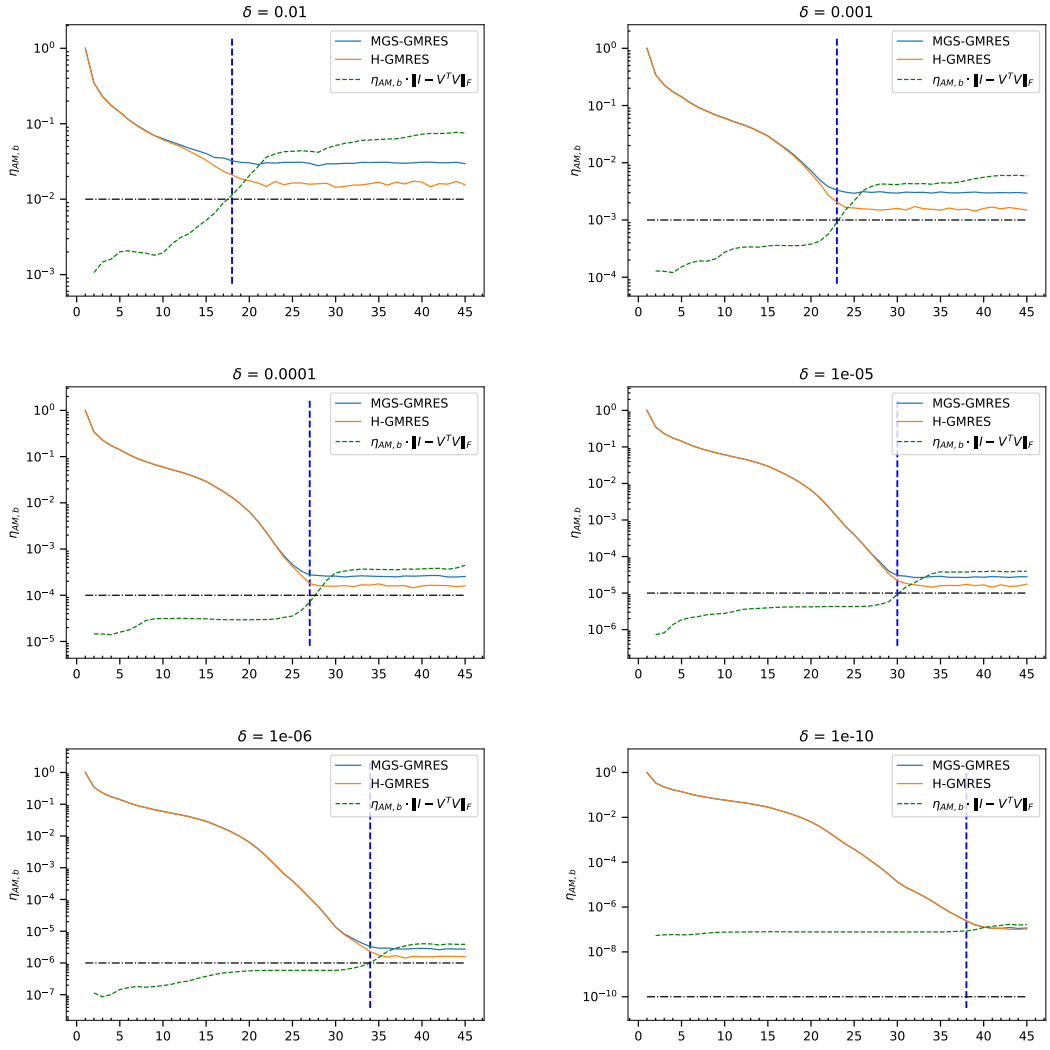

Figure 50: Convergence history of $\eta_{AM,b}$ for pde225 with ILU($5 \cdot 10^{-1}$) using δ-normwise representation

RESEARCH CENTRE
BORDEAUX – SUD-OUEST

200 avenue de la Vieille Tour
33405 Talence Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399