Commun. Comput. Phys.
doi: 10.4208/cicp.091014.060315a

Vol. **18**, No. 5, pp. 1313-1335
November 2015

An Adaptive Combined Preconditioner with Applications in Radiation Diffusion Equations

Xiaoqiang Yue[1], Shi Shu[1,*], Xiaowen Xu[2] and Zhiyang Zhou[1]

[1] *School of Mathematics and Computational Science, Xiangtan University, Hunan 411105, P.R. China.*
[2] *Institute of Applied Physics and Computational Mathematics, Beijing 100088, P.R. China.*

Received 9 October 2014; Accepted (in revised version) 6 March 2015

Abstract. The paper aims to develop an effective preconditioner and conduct the convergence analysis of the corresponding preconditioned GMRES for the solution of discrete problems originating from multi-group radiation diffusion equations. We firstly investigate the performances of the most widely used preconditioners (ILU(k) and AMG) and their combinations (B_{co} and $\tilde{B}_{co}$), and provide drawbacks on their feasibilities. Secondly, we reveal the underlying complementarity of ILU(k) and AMG by analyzing the features suitable for AMG using more detailed measurements on multiscale nature of matrices and the effect of ILU(k) on multiscale nature. Moreover, we present an adaptive combined preconditioner B_{co}^{α} involving an improved ILU(0) along with its convergence constraints. Numerical results demonstrate that B_{co}^{α}-GMRES holds the best robustness and efficiency. At last, we analyze the convergence of GMRES with combined preconditioning which not only provides a persuasive support for our proposed algorithms, but also updates the existing estimation theory on condition numbers of combined preconditioned systems.

AMS subject classifications: 65F10, 65F15, 65N55, 65Z05

Key words: Radiation diffusion equations, combined preconditioner, adaptive preconditioning, convergence analysis.

1 Introduction

Equations of radiation hydrodynamics (RHE), which play a quite important role in numerical simulations of inertial confinement fusion (ICF), contain a set of coupled equations that describe the motion of fluids and radiative transition processes, where the former is described by a multi-material Eulerian hydrodynamic equations while the latter

*Corresponding author. *Email addresses:* yuexq@xtu.edu.cn (X. Yue), shushi@xtu.edu.cn (S. Shu), xwxu@iapcm.ac.cn (X. Xu), peghoty@163.com (Z. Zhou)

 ©2015 Global-Science Press

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

approximatively by multi-group radiation diffusion (MGD) equations [1]. It is a widely-used way to discretize the hydrodynamic equations explicitly and implicitly for MGD equations, which gives rise to large-scale sparse linear systems solved for high resolutions. It should be stressed that the computational costs occupy the predominantly time-consuming part (more than 80% in general), numerical solutions of discretized algebraic systems become the crucial bottleneck in ICF simulations. The preconditioned iterative method is a primary choice to handle them, although its efficiency is affected severely by the preconditioner.

In recent years, numerous preconditioners constructed for radiation diffusion equations have been presented, see [2–10] and references therein for details. These preconditioners mainly fall into several preconditioning algorithms, such as incomplete LU factorization (ILU) [11,12] and algebraic multigrid method (AMG) [13,14], which will be called standalone preconditioners below. However, the actual physical modelings of ICF are becoming more and more meticulous along with the development of high-performance computers and the capability of numerical simulations. The existing standalone preconditioners are difficult or even no more to adapt to numerical simulations of complex physical modelings, which is mainly due to the following two aspects: (i) they are not designed for MGD but for the 3-temperature heat-conduction equations, hard to handle the complex coupling relationships in MGD; (ii) they usually lack inconsistency with respect to the different matrices from nonlinear iterations at each integration time-step, nearly impossible to accurately simulate a problem with up to 10^5 integration time-steps.

More recently, Hu et al. [15] proposed a combined preconditioner B_{co}, which was successfully employed to solve symmetric and positive definite (SPD) linear systems arising from petroleum reservoir simulations. On this basis, we develop an adaptive combined preconditioner B_{co}^{α}, and apply it to MGD equations and further an actual numerical simulation of ICF implosion. We firstly introduce the popular strength measurement ψ and define the distribution measurements ϕ and ϱ on the multiscale nature of an arbitrary matrix, where ϱ is the number of magnitude intervals, and ϕ reflects how close these magnitude intervals are. Secondly, we obtain three adaptability conditions ensuring the feasibility of AMG by an investigation of two-dimensional convection-diffusion equations and classical one-group and twenty-group problems. Meanwhile we provide the fact that ILU(k) can weaken the multiscale strength and concentrate corresponding distributions, which brings to light their underlying complementarity and offers theoretical help to the effectiveness of combined preconditioners. Moreover, we present an improved ILU(k) by adopting a simpler criterion to filtrate tiny matrix elements prior to performing ILU(k) factorization, namely I$\hat{L}\hat{U}$(k) in this study, which is a subtle version roughly equivalent to ILU(k) but with a notable reduction in memory storage and computational cost when disposing of multiscale problems. B_{co}^{α} is subsequently achieved by an additional heuristic remedy on I$\hat{L}\hat{U}$(0)-GMRES. Our numerical results verify the considerable advantage of B_{co}^{α} over others.

In addition, we obtain a qualitative convergence estimation of a nonsymmetric combined preconditioner. Its analysis procedure consists of a transformation into the estima-

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

tion of $\kappa(B_{co}A)$ and a modification on the existing estimation theory of B_{co}, resulting in a new sufficient condition stated in Theorem 6.1. In particular, an extra condition (see Theorem 6.2) is appended to yield the same conclusion as Theorem 2.2 in [15].

An outline of the paper is as follows. MGD equations and the discretization scheme are introduced in Section 2. Section 3 provides the performances of ILU(k), AMG, B_{co} and $\tilde{B}_{co}$. We analyze the multiscale nature of a given matrix and propose B^{α}_{co} in Section 4. Section 5 reports and analyzes the numerical results of an actual ICF implosion to show the advantage of B^{α}_{co}. A qualitative analysis of $\tilde{B}_{co}$-GMRES is presented in Section 6, and some concluding remarks are given in Section 7.

2 Model problem and discretization

2.1 Multi-group radiation diffusion equations

Consider the multi-group radiation diffusion (MGD) equations [1,16]

$$\begin{cases} \frac{\partial E_g}{\partial t} = \nabla\cdot(D_g(E_g)\nabla E_g) + c(\sigma_{Bg}B(T_E)_g - \sigma_{Pg}E_g) + S_g, \ \ g=1,\cdots,G, \\ \rho c_E \frac{\partial T_E}{\partial t} = \nabla\cdot(D_E(T_E)\nabla T_E) - c\sum_g(\sigma_{Bg}B(T_E)_g - \sigma_{Pg}E_g) + \omega_{IE}(T_I - T_E), \\ \rho c_I \frac{\partial T_I}{\partial t} = \nabla\cdot(D_I(T_I)\nabla T_I) - \omega_{IE}(T_I - T_E), \end{cases} \tag{2.1}$$

where g is the group index and G is the number of groups,

- E_g, $D_g(E_g)$: the g-th spectral radiation energy density and radiation diffusion coefficient;
- σ_{Bg}, σ_{Pg}: the g-th scattering coefficient and absorption coefficient of Planck-averaged electron energy;
- $B(T_E)_g$: the g-th electron scattering energy density;
- S_g, c, ρ: the g-th radiation source, speed of light and medium density;
- T_α, $D_\alpha(T_\alpha)$, c_α: the temperature, thermal-conductivity coefficient and specific heat, where $\alpha = E$ stands for electron and $\alpha = I$ for ion;
- ω_{IE}: the coefficient of energy exchange between electron and ion.

The MGD equations (2.1) are a coupled nonlinear system of spectral radiation energy densities, electron and ion, where $D_g(E_g)$, $D_E(T_E)$, $D_I(T_I)$ and $B(T_E)_g$ are all nonlinear functions of energy densities and temperatures. Energy densities, without any exchange themselves, are all coupled with electron through exchanges of energy $c(\sigma_{Bg}B(T_E)_g - \sigma_{Pg}E_g)$, while electron and ion are coupled through $\omega_{IE}(T_I - T_E)$.

In particular, when $G=1$, the radiation temperature T_R can be obtained by

$$E_g = E_R = aT_R^4, \quad a = const.,$$

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

which obviously degrades Eq. (2.1) into classical 3-temperature heat-conduction equations.

2.2 Discrete linear systems

In order to guarantee the numerical stability and larger time-step size in practical applications, fully implicit schemes, such as adaptive backward Eulerian, are frequently used along the time direction, which yields a steady state nonlinear system of Partial Differential Equations (PDEs). We usually adopt the frozen-in coefficients method to linearize their nonlinear terms in numerical simulations, as a consequence, a steady linear PDEs is required to be solved at each nonlinear iteration. In fact, numerous spatial discretization methods available in the literatures, such as finite difference (FD), finite element (FE) and finite volume (FV) methods, can be employed. Since the FV method can locally preserve the conservation of certain physical quantities (mass, energy and so on), it has become one of the most popular numerical methods in many software packages, such as five-point discretization in two-dimensional and seven-point discretization in three-dimensional on a regular (Eulerian) grid [1], nine-point discretization in two-dimensional on a Lagrangian grid with large deformations [2]. No matter which scheme is utilized in the spatial direction, the resulted linear system has the form

$$Au = f, \tag{2.2}$$

whose coefficient matrix $A=(a_{ij})_{N\times N}$ takes the sparse block form

$$\begin{bmatrix} A_1 & & & D_{1E} & \\ & \ddots & & \vdots & \\ & & A_G & D_{GE} & \\ D_{E1} & \cdots & D_{EG} & A_E & D_{EI} \\ & & & D_{IE} & A_I \end{bmatrix},$$

where $N=(G+2)\times n$ and n is the number of mesh cells, the diagonal block $A_\alpha=(a_{ij}^\alpha)_{n\times n}$ ($\alpha=1,\cdots,G,E,I$) reflects the diffusion of the α-th energy density or temperature in the medium, and the nonzero diagonal matrix $D_{\alpha\beta}$ ($\alpha\neq\beta$), one of the so-called coupling terms, expresses the coupling relation between the α-th and the β-th.

Normally, the coupling terms in Eq. (2.2) satisfy

$$D_{EI}=D_{IE},\quad D_{Eg}\neq D_{gE},\quad g=1,\cdots,G,$$

which implies that A is not necessarily symmetric but only positive definite. In such a case, the most popular choice for solving Eq. (2.2) is the preconditioned restarted generalized minimum residual method, denoted by PGMRES(m) [17], which restarts the algorithm once performing m iterations of PGMRES. Despite its performance, not only the efficiency but also the robustness, PGMRES(m) is extremely affected by the variety and quality of the underlying preconditioner.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

In the upcoming analyses and numerical experiments, all the preconditioners, to which we devote in this paper, are used to precondition GMRES method with the Krylov subspace of dimension 30, namely GMRES(30).

3 Performance of two popular preconditioners and their combinations

Nowadays, both ILU and AMG are regarded as the most popular preconditioners in real applications, since the simplicity of ILU, which specifies the sparsity of its factors to approximate the exact one, and the virtue of scalability and applicability of AMG. Below we focus on the numerical performance of these preconditioners for two sets of linear algebraic systems from simulations on capsule implosion. The first set consists of 3 linear systems, denoted by S_1-S_3, arising from the five-point discretization with second-order accuracy of a classical one-group problem on 8000×6, 16000×12 and 32000×24 grids, while the second set includes 6 linear systems, denoted by M_1-M_6, originating from the same discretization of a multi-group problem on 4000×12, 8000×24 and 16000×48 grids, where the number of groups $G=20$.

All our numerical experiments are performed in a 64 bit Fedora 14 platform, in double precision arithmetic on Intel Xeon (W5590) with 24.0 GB RAM, 3.33 GHz, with an -O2 optimization parameter. In the following tables, dashed entries (-) indicate the solutions either diverge or fail to converge after 200 iterations, *It* is the number of iterations until the stopping criterion is reached, T_c represents the CPU time including both Setup and Solve phases, with second as its unit.

Table 1 shows the results of applying GMRES with ILU(k) and AMG preconditioning, referred to as ILU(k)-GMRES and AMG-GMRES respectively, to aforementioned one-group and twenty-group problems, where the tolerance for stopping is 10^{-8}. Unless otherwise stated, we choose the ILU(k) preconditioner from Euclid library [12] and the AMG preconditioner provided in the HYPRE (High Performance Preconditioners) library [18] (developed in the Center for Applied Scientific Computing at Lawrence Livermore National Laboratory), empirically with a strength threshold 0.25, coarse-grid selection by Falgout Coarsening (a hybrid RS/CLJP scheme), one presmoothing and postsmoothing sweep performed by the symmetric hybrid Gauss-Seidel, and other parameters of these two preconditioners chosen from their default sets. Here and below we only list the numerical results of ILU(0) preconditioner due to the considerations of computational efficiency and memory capacity, while the preconditioning behavior of ILU(k) ($k=1,2,3$), with higher time and space complexity, is quite similar to that of ILU(0).

As shown in Table 1, despite the fact that it works well for S_1 and M_1-M_4, ILU(k)-GMRES lacks robustness for S_2 and M_5, or even it requires more than 200 iterations for S_3 and M_6. Meanwhile, the less the stopping criterion, the more linear systems don't converge in 200 iterations. So the effects of ILU(k) preconditioners couldn't be predicted, and sometimes ILU(k)-GMRES may even break down. Besides a marked improvement in

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Table 1: Number of iterations and wall time of ILU(0) and AMG.

	8000×6				16000×12				32000×24			
	Euclid		AMG		Euclid		AMG		Euclid		AMG	
	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c
S_1	3	0.4	11	0.5	3	1.6	11	2.0	4	6.5	11	8.9
S_2	15	0.5	33	1.3	26	2.7	46	8.4	65	18.7	56	42.6
S_3	-	-	14	0.6	-	-	16	3.5	-	-	13	11.0
	4000×12				8000×24				16000×48			
	Euclid		AMG		Euclid		AMG		Euclid		AMG	
	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c
M_1	2	3.3	14	14.2	2	13.8	35	179.6	3	60.6	48	1002.3
M_2	2	3.3	14	16.1	2	14.0	39	212.3	3	61.6	42	972.6
M_3	2	3.3	22	27.8	2	14.1	37	201.4	3	60.1	41	932.1
M_4	2	3.1	48	43.6	2	12.8	51	206.9	3	56.4	52	832.8
M_5	6	3.8	26	22.7	9	19.1	24	97.6	19	138.1	29	482.4
M_6	-	-	27	25.6	-	-	28	120.2	-	-	34	602.9

the convergence property over ILU(k)-GMRES, AMG-GMRES doesn't converge robustly enough, possessing an excessive number of iterations to achieve convergence (56 to S_2 on a 32000×24 grid and 52 to M_4 on 16000×48), and worse performance in terms of CPU time. Hence, an efficient and robust preconditioner is of especially high demand for MGD equations.

More recently, Hu and his co-authors developed a combined preconditioner B_{co} [15], which is proven to be SPD and its key components are a contractive smoother S and a SPD preconditioner B. They state clearly that there is no need to use B_{co} unless neither B nor $\tilde{S}$ works well, where $\tilde{S}:=S+S^T-S^TAS$ and A is the SPD coefficient matrix. Afterwards, B_{co} is employed as a preconditioner of the preconditioned conjugate gradient method in petroleum reservoir simulations, with higher efficiency and better robustness, where S is selected from classical AMG methods and B as incomplete Cholesky factorization with zero level of fill-in. Observing that the original matrix of Eq. (2.2) is generally nonsymmetric, we construct a nonsymmetric combined preconditioner $\tilde{B}_{co}$, whose Setup phase and Preconditioning phase are described below.

Algorithm 1. (Nonsymmetric combined preconditioner $\tilde{B}_{co}$)

Step 1 Setup phase of $\tilde{B}_{co}$.

1.1 (B: ILU(k)) Perform an incomplete LU factorization of A, which produces its factors L and U.

1.2 (S: AMG) Apply the Setup phase of AMG to A, which builds the ingredients required by a hierarchy of levels, coarsest to finest.

Step 2 Preconditioning phase of $\tilde{B}_{co}$, i.e. $w=\tilde{B}_{co}g$, where g is an arbitrarily given vector.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

2.1 (B: ILU(k)) Compute $w_1=(LU)^{-1}g$ and the residual $r=g-Aw_1$.

2.2 (S: AMG) Taking the zero vector as the initial guess, the new iterate w_2 is obtained by invoking V-cycle once with respect to the linear system $Aw_2=r$.

2.3 Compute $w=w_1+w_2$.

Table 2 compares the results of B_{co}-GMRES and $\tilde{B}_{co}$-GMRES. Notice here that they both converge robustly and their convergence rates are essentially the same, indicating that B_{co} and $\tilde{B}_{co}$ both overcome the poor robustnesses of ILU(k) and AMG. But it is worth noting that an extra AMG preconditioning effect of B_{co} makes it more expensive than $\tilde{B}_{co}$ to use, for example, the wall time of M_5 on three different grids are 9.4 s, 38.2 s, 125.7 s and 5.6 s, 23.0 s, 102.7 s for B_{co} and $\tilde{B}_{co}$, respectively.

Table 2: Number of iterations and wall time of B_{co} and $\tilde{B}_{co}$.

	8000×6				16000×12				32000×24			
	B_{co}		$\tilde{B}_{co}$		B_{co}		$\tilde{B}_{co}$		B_{co}		$\tilde{B}_{co}$	
	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c
S_1	1	0.5	2	0.5	2	2.5	2	2.3	2	10.0	2	9.2
S_2	5	0.7	6	0.6	5	3.1	6	2.7	5	12.7	6	11.2
S_3	7	0.8	10	0.7	6	3.3	10	3.2	8	15.4	11	13.8
	4000×12				8000×24				16000×48			
	B_{co}		$\tilde{B}_{co}$		B_{co}		$\tilde{B}_{co}$		B_{co}		$\tilde{B}_{co}$	
	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c	It	T_c
M_1	1	5.1	1	4.7	1	20.9	1	19.1	3	119.1	3	98.2
M_2	1	5.3	1	4.8	1	21.5	1	19.5	2	110.6	2	93.9
M_3	1	5.2	1	4.7	1	21.3	1	19.4	2	109.0	2	94.1
M_4	1	5.4	1	4.8	1	22.1	1	19.5	2	111.6	2	95.0
M_5	6	9.4	3	5.6	6	38.2	3	23.0	3	125.7	3	102.7
M_6	4	7.9	3	5.8	3	29.2	3	23.5	7	191.1	3	103.6

By an investigation in terms of the wall time in Table 1 and Table 2, it is easy to see that $\tilde{B}_{co}$ partially requires less wall time than the standalone preconditioners. In spite of the fact that it is stable for MGD problems, $\tilde{B}_{co}$ won't solve them in an optimal way due to the cost of one step of certain standalone preconditioner is much more efficient than $\tilde{B}_{co}$. Based on these observations, as a premise of maintaining the robustness of $\tilde{B}_{co}$, an adaptive preconditioning technique will be considered in the next section.

4 An adaptive combined preconditioner

Generally speaking, the key ingredient used in constructing an adaptive preconditioner resides the clear distinction between linear systems suitable for the above standalone

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

and combined preconditioners. With this observation, the desired distinction, which we describe below, will be used as the critical strategy to adaptively pick a preconditioner.

4.1 Multiscale nature of matrices and its effect on AMG

Given a nonsingular matrix $B=(b_{ij})_{n\times n}$, along with its index set I_B denoted by $I_B=\{1,\cdots,n\}$, the conventional measurement on the multiscale nature of B is introduced by

$$\psi(B):=\lfloor \lg(\max_{i\in I_B} v_B(i))\rfloor,$$

where $\lfloor\cdot\rfloor$ is the rounding down of its argument, and the real array v_B is defined as

$$v_B(i)=\frac{\max_{k\neq i}|b_{ik}|}{\min_{j\neq i,b_{ij}\neq 0}|b_{ij}|},\quad i\in I_B.$$

It is trivial to see that the measurement ψ primarily represents the strength of the multiscale nature. Therefore, B and its associated linear system are said to be strongly multiscaled if $\psi(B)>\theta_\psi$ for a relatively large integer θ_ψ.

In the ordinary way, the intensity of ψ is utilized as the judgment to determine whether or not AMG is an preferable preconditioner [19]. However, it is sometimes quite unreliable. In fact, we discover that AMG is still perfectly suitable to use when the distribution of elements in v_B satisfies certain constraints, which are described below by the following two measurements we propose.

Taking the definition of $\psi(B)$ into consideration, we can assume that all the values of v_B scatter in $[10^{k_l},10^{k_l+1})$ $(l=1,\cdots,\varrho(B))$, where $\varrho(B)$ is the number of magnitude intervals, and the nonnegative integers obey $k_1<k_2<\cdots<k_{\varrho(B)}=\psi(B)$. In addition, aiming to acquire the absolute distance between these magnitude intervals, we define $\phi(B)=0$ if $\varrho(B)=1$, and $\phi(B)=\sum_{i=2}^{\varrho(B)}(k_i-k_{i-1}-1)$ otherwise. Thus, $\varrho(B)$ and $\phi(B)$ can be viewed as the first and second measurement on the distribution of the multiscale nature. In practice, we also draw a threshold $\theta_p>0$ into the actual calculation of $\varrho(B)$ and $\phi(B)$, with the purpose to omit the disturbance of $[10^{k_l},10^{k_l+1})$ if $n_l/n<\theta_p$, where n_l is the number of elements in v_B scattered in that interval.

In order to exploit the effect of the above measurements ψ, ϱ and ϕ on AMG, We firstly consider a two-dimensional convection-diffusion equation with anisotropy and discontinuity.

Example 1. Consider linear systems resulting from the five-point finite difference approximation (one-order upwind scheme) of

$$-c_x u_{xx}-u_{yy}+u_x=1,\quad \text{in } \Omega=(0,1)^2,$$

with the Dirichlet boundary condition

$$u=0,\quad \text{on } \partial\Omega,$$

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

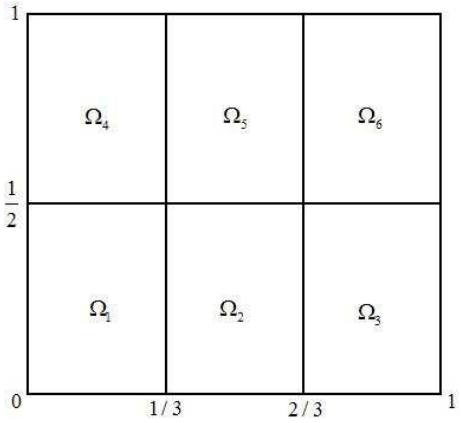


Figure 1: Illustration for the distribution of c_x in 3×2 blocks.

where the anisotropic and discontinuous coefficient $c_x>0$ is distributed over $\Omega=\bigcup_{i=1}^{6}\Omega_i$ as illustrated in Fig. 1.

Here, we distinguish 7 situations, where $c_x^i=c_x|_{\Omega_i}$, $i=1,\cdots,6$.

Case 1. $c_x^1=c_x^2=c_x^3=1.0$, $c_x^4=c_x^5=c_x^6=10^3$;

Case 2. $c_x^i=10^i$, $i=1,\cdots,6$;

Case 3. $c_x^1=c_x^2=1.0$, $c_x^3=c_x^6=10^3$, $c_x^4=c_x^5=10^6$;

Case 4. $c_x^1=c_x^2=c_x^3=1.0$, $c_x^4=c_x^5=c_x^6=10^{14}$;

Case 5. $c_x^1=c_x^2=10^{11}$, $c_x^3=c_x^6=10^{12}$, $c_x^4=c_x^5=10^{14}$;

Case 6. $c_x^1=c_x^2=10^2$, $c_x^3=c_x^6=10^8$, $c_x^4=c_x^5=10^{14}$;

Case 7. $c_x^1=c_x^2=1.0$, $c_x^3=c_x^6=10^7$, $c_x^4=c_x^5=10^{14}$.

The effect of the above 7 cases on the number of iterations of AMG-GMRES by ψ, ϱ and ϕ is indicated in Table 3, where the first column represents the grid partitioning $n_p\times n_p$ by n_p^2, the tolerance for stopping is 10^{-8}, and dashed entries (-) indicate the solutions fail to converge in 200 iterations.

Setting thresholds $\theta_\psi=4$, $\theta_\varrho=3$, $\theta_\phi=3$ and $\theta_p=0.1\%$, we can observe from Table 3 that AMG-GMRES will be stable in 3 situations. The first one is $\psi<\theta_\psi$, i.e. relatively weak multiscale nature (see Case 1); the second is $\psi\geq\theta_\psi$ and $\varrho<\theta_\varrho$, i.e. strongly multi-scaled but few magnitude intervals (shown by Case 4); the third is $\psi\geq\theta_\psi$, $\varrho\geq\theta_\varrho$ and $\phi<\theta_\phi$, i.e. strongly multi-scaled and too many magnitude intervals but closely gathered (presented in Case 2 and Case 5). As a consequence, we obtain by induction three adaptability conditions below of linear systems suitable to solve by AMG-GMRES.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Table 3: Effect of ψ, ϱ and ϕ on AMG.

n_p^2	Case 1		Case 2		Case 3		Case 4		Case 5		Case 6		Case 7	
	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*	ψ,ϱ,ϕ	*It*
128^2	3,3,1	8	6,7,0	12	6,4,3	31	14,2,13	8	14,4,1	1	14,4,10	51	14,3,12	-
256^2	3,3,1	10	6,7,0	14	6,4,3	-	14,2,13	9	14,4,1	2	14,4,10	71	14,3,12	-
512^2	2,2,1	11	5,6,0	18	5,3,3	-	13,2,12	13	13,3,1	2	13,3,10	104	13,3,11	-
1024^2	3,2,2	11	6,6,0	21	6,3,4	-	14,2,13	11	14,3,1	2	14,3,10	-	14,3,12	-

Cond 1. $\psi < \theta_\psi$;

Cond 2. $\psi \geq \theta_\psi$ but $\varrho < \theta_\varrho$;

Cond 3. $\psi \geq \theta_\psi$, $\varrho \geq \theta_\varrho$ but $\phi < \theta_\phi$.

Let us examine the above sufficient conditions with linear systems from one-group and twenty-group problems. Table 4 presents the multiscale measurements ψ, ϱ and ϕ of the original matrix A from S_1-S_3 on 8000×6 grid and M_1-M_6 on 4000×12.

Table 4: ψ, ϱ and ϕ of the original matrix A.

	S_1	S_2	S_3	M_1	M_2	M_3	M_4	M_5	M_6
ψ	24	24	18	23	23	23	32	32	32
ϱ	16	17	16	18	18	18	19	17	16
ϕ	1	3	2	6	6	6	14	16	17

Combining Table 4 and Table 1, we can see that S_1 and S_3 both satisfy **Cond 3** while AMG-GMRES is stable to solve them. By contrast, S_2 and M_1-M_6 violate all the sufficient conditions while AMG-GMRES solve them either not robustly enough or with a rather excessive number of iterations. These observations confirm the validity of **Cond 1-Cond 3**.

4.2 The effect of ILU(k) on multiscale nature

Here we concentrate on the effect of ILU(k) preconditioner performed on the multiscale nature of original matrices. Table 5 lists ψ, ϱ and ϕ of the preconditioned matrix $C := (LU)^{-1}A$ for the same problems used earlier, where L and U are the incomplete factors obtained from ILU(0) factorization of A.

Table 5: ψ, ϱ and ϕ of the preconditioned matrix C.

	S_1	S_2	S_3	M_1	M_2	M_3	M_4	M_5	M_6
ψ	22	21	18	22	22	22	22	23	23
ϱ	6	5	1	8	7	7	7	8	8
ϕ	0	0	0	0	0	0	0	0	0

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

By comparing Table 4 with Table 5, it can be easily seen that the strength measurement ψ of A decreases by performing ILU(k) preconditioning once, as well as a significant improvement on the distribution measurements ϱ and ϕ, especially all the preconditioned matrices satisfy **Cond 3**. Take (S_2, M_4) as an example, values of ψ are reduced from (24, 32) to (21, 22), ϱ from (17, 19) down to (5, 7), and ϕ from (3, 14) down to (0, 0), respectively.

From the above observation, an immediate consequence is that AMG is the very preconditioner to solve these ILU(0) preconditioned systems. Also, note that the underlying thought of the combined preconditioner $\tilde{B}_{co}$ in Algorithm 1 is to perform an extra AMG with respect to the ILU(k) preconditioned system, $\tilde{B}_{co}$ is an effective preconditioner associated with the original system, of which the data in Table 2 are in support.

4.3 An improved algorithm of ILU(k)

Since the coefficient matrix A of MGD equations has roughly strong multiscale nature, and its incomplete LU factorization maintains the primary part of A, a straightforward idea is to filter relatively small entries out of A, and then perform ILU(k) factorization for the filtered matrix $\hat{A}$. The idea derives a new ILU factorization, denoted by I$\hat{L}\hat{U}$(k), which is stated below.

Algorithm 2. (I$\hat{L}\hat{U}$(k) factorization)

Step 1 Generate the filtered matrix $\hat{A}=(\hat{a}_{ij})_{n\times n}$ of $A=(a_{ij})_{n\times n}$ by the relation

$$\hat{a}_{ij}=\begin{cases} a_{ij}, & \text{if } |a_{ij}|>\theta_d\cdot|a_{ii}|, \\ 0, & \text{otherwise,} \end{cases} \qquad i,j=1,\cdots,n,$$

where the drop-tolerance $\theta_d\in[0,1]$ is a given scalar.

Step 2 Perform the ILU(k) factorization for $\hat{A}$ with the resulting factors $\hat{L}$ and $\hat{U}$ as I$\hat{L}\hat{U}$(k) factors of A.

An intuitive claim of I$\hat{L}\hat{U}$(k) is that its preconditioning behavior shouldn't be much worse than that of ILU(k). Table 6 shows the comparison between ILU(0)-GMRES and I$\hat{L}\hat{U}$(0)-GMRES with respect to S_2 and M_5, where the drop-tolerance $\theta_d=10^{-5}$, T_1 and T_2 show the wall time of Setup and Solve phases, and the column labeled *AuxLength* indicates the actual length of auxiliary arrays required in the implementation of ILU(0) or I$\hat{L}\hat{U}$(0).

It is observed from Table 6 that I$\hat{L}\hat{U}$(0) is almost equivalent to ILU(0), with a notable reduction in memory storage (on the average, 50% and 31%, respectively) and computational cost. For S_2 on a 32000×24 grid, I$\hat{L}\hat{U}$(0)-GMRES runs 41 times faster than ILU(0)-GMRES in Setup phase while 10.99s abated from 13.73s in Solve phase. Thus, the above facts confirm our former claim. Moreover, as a consequence of the tactical advantage of Algorithm 2, $\hat{B}_{co}$-GMRES is predicted more efficient than $\tilde{B}_{co}$-GMRES, where $\hat{B}_{co}$ is generated by replacing ILU(k) with I$\hat{L}\hat{U}$(k) in Algorithm 1.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Table 6: Comparison between ILU(0) and I$\hat{L}\hat{U}$(0).

S_2	Euclid				I$\hat{L}\hat{U}$(0)			
	T_1	T_2	*It*	*AuxLength*	T_1	T_2	*It*	*AuxLength*
8000×6	0.37	0.13	15	863,964	0.01	0.10	15	434,875
16000×12	1.46	1.30	26	3,551,928	0.03	1.00	26	1,771,933
32000×24	5.84	14.45	65	14,399,856	0.14	10.99	65	7,174,898
M_5	Euclid				I$\hat{L}\hat{U}$(0)			
	T_1	T_2	*It*	*AuxLength*	T_1	T_2	*It*	*AuxLength*
4000×12	2.69	1.12	6	7,119,472	0.12	0.96	6	4,902,172
8000×24	10.85	7.08	9	28,830,944	0.51	6.33	9	19,926,150
16000×48	44.26	83.74	19	116,029,888	2.11	80.32	19	80,093,475

4.4 Definition and performance of an adaptive combined preconditioner

From the previous preliminary investigations, we can summarize that the sharp distinction between AMG and others is drawn by **Cond 1**-**Cond 3**, but it is difficult to assess in advance an appropriate distinction with respect to ILU(k)-like because of its unpredictability and inconsistency shown in Table 1. A simple and heuristic way to remedy this appeals to the reduction in the residual norm at each step of I$\hat{L}\hat{U}$(0)-GMRES.

Using the above arguments, we can get an adaptive combined preconditioner B_{co}^{α}, which is based on AMG, I$\hat{L}\hat{U}$(0) and $\hat{B}_{co}$. Below is the algorithm of B_{co}^{α} and its PGMRES(m), where σ_1 and σ_2 are respectively given thresholds for $\|r_1\|/\|r_0\|$ and $\|r_k\|/\|r_{k-1}\|$ ($k\geq 2$), and $r_l = f - Au_l$ is the residual vector obtained at step l of Eq. (2.2) by I$\hat{L}\hat{U}$(0)-GMRES.

Algorithm 3. (Adaptive combined preconditioner B_{co}^{α} and its PGMRES(m))

Given an initial guess u_0, stopping criterion ε, thresholds θ_d, θ_ψ, θ_p, θ_ϱ, θ_ϕ, σ_1 and σ_2.

Step 1 If A satisfies **Cond 1** or **Cond 2** or **Cond 3**,

1.1 Set B_{co}^{α} = AMG;

1.2 Solve Eq. (2.2) by B_{co}^{α}-GMRES(m) until convergence.

Step 2 If A doesn't satisfy **Cond 1**-**Cond 3**,

2.1 Set B_{co}^{α} = I$\hat{L}\hat{U}$(0);

2.2 Compute u_1 by performing B_{co}^{α}-GMRES(m) once to Eq. (2.2).

2.3 Compute the adjacent residual reduction $\alpha^1 = \|r_1\|/\|r_0\|$;

2.4 If $\alpha^1 > \sigma_1$,

2.4.1 Set B_{co}^{α} = $\hat{B}_{co}$;

2.4.2 Solve Eq. (2.2) by B_{co}^{α}-GMRES(m) with u_0 as the initial guess until convergence.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

2.5 If $\alpha^1 \leq \sigma_1$ then set $k=1$ and until $\alpha^k > \sigma_2$ or $\alpha < \varepsilon$ Do

2.5.1 Compute u_{k+1} by performing B_{co}^{α}-GMRES(m) once to Eq. (2.2).

2.5.2 $k=k+1$;

2.5.3 Compute the adjacent residual reduction $\alpha^k = \|r_k\|/\|r_{k-1}\|$;

2.5.4 Compute the absolute residual reduction $\alpha = \|r_k\|/\|r_0\|$;

2.6 If $\alpha \geq \varepsilon$,

2.6.1 Set $B_{co}^{\alpha} = \hat{B}_{co}$;

2.6.2 Solve Eq. (2.2) by B_{co}^{α}-GMRES(m) with u_{k-1} as the initial guess until convergence.

Let us illustrate the effectiveness of Algorithm 3 with classical one-group and multi-group problems, and the numerical results are summarized in Table 7, where $u_0 = 0$, $\varepsilon = 10^{-8}$, $\theta_d = 10^{-5}$, $\theta_\psi = 4$, $\theta_p = 0.1\%$, $\theta_\varrho = 3$, $\theta_\phi = 3$, $\sigma_1 = 10^{-4}$ and $\sigma_2 = 0.1$.

Table 7: Number of iterations and wall time of $\tilde{B}_{co}$ and B_{co}^{α}.

	8000×6				16000×12				32000×24			
	$\tilde{B}_{co}$		B_{co}^{α}		$\tilde{B}_{co}$		B_{co}^{α}		$\tilde{B}_{co}$		B_{co}^{α}	
	It	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c
S_1	2	0.5	11	0.5	2	2.3	11	2.0	2	9.2	11	8.9
S_2	6	0.6	6(1)	0.3	6	2.7	6(1)	1.1	6	11.2	6(1)	4.7
S_3	10	0.7	14	0.6	10	3.2	16	3.5	11	13.8	13	11.0
	4000×12				8000×24				16000×48			
	$\tilde{B}_{co}$		B_{co}^{α}		$\tilde{B}_{co}$		B_{co}^{α}		$\tilde{B}_{co}$		B_{co}^{α}	
	It	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c
M_1	1	4.7	2	0.6	1	19.1	2	2.6	3	98.2	3	15.7
M_2	1	4.8	2	0.7	1	19.5	2	2.8	2	93.9	3	16.6
M_3	1	4.7	2	0.7	1	19.4	2	2.8	2	94.1	3	17.2
M_4	1	4.8	2	0.5	1	19.5	2	2.0	2	95.0	3	11.9
M_5	3	5.6	3(2)	2.9	3	23.0	3(3)	12.8	3	102.7	3(3)	60.7
M_6	3	5.8	3(2)	3.0	3	23.5	3(3)	13.2	3	103.6	3(3)	62.3

As expected, the results show that B_{co}^{α} = AMG to S_1 and S_3, B_{co}^{α} = I$\hat{L}\hat{U}(0)$ to M_1-M_4, while iteration numbers of B_{co}^{α} are listed by $s(t)$ for S_2 and M_5-M_6, where s is the actual iteration number of $\hat{B}_{co}$-GMRES to solve, t is the iteration number of I$\hat{L}\hat{U}(0)$-GMRES used for judgement. All such costs of I$\hat{L}\hat{U}(0)$-GMRES are perfectly acceptable to solutions of these entire linear systems since all ts are quite small and the computational complexity required by I$\hat{L}\hat{U}(0)$-GMRES is rather low. From the viewpoint of s, B_{co}^{α}-GMRES is fairly robust, roughly the same as that of $\tilde{B}_{co}$-GMRES. In addition, B_{co}^{α}-GMRES is even more efficient than $\tilde{B}_{co}$-GMRES, runs 1.3, 1.2 and 1.4 times faster on the average for one-group

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

problems on 3 different grids, and 3.6, 3.4 and 3.2 for multi-group problems. Furthermore, compared with AMG-GMRES and ILU(0)-GMRES (See Table 1), B_{co}^{α}-GMRES has a significant advantage in computational cost and robustness.

5 Application in simulations of hydrodynamic instability

We now consider an application of B_{co}^{α} in the hydrodynamic instabilities simulated by LARED-S [1] code, which is a two-dimensional and three-dimensional multi-material Eulerian radiation hydrodynamic code series, developed by Institute of Applied Physics and Computational Mathematics (IAPCM) based on a parallel application framework named J Adaptive Structured Meshes applications INfrastructure (JASMIN) [20] on large-scale parallel computers. LARED-S code exploits the operator splitting principle to manage RHE, firstly explicitly discretize Eulerian hydrodynamic equations and fully-implicit scheme for MGD equations (2.1) at each time step. Since Eq. (2.1) is a system of coupled nonlinear equations, LARED-S code takes advantage of the Picard iterative scheme to linearize their nonlinear terms, and the resulting linear system with the shape of Eq. (2.2) is required to be solved at each nonlinear iteration.

Consider a MGD model in spherical coordinates, whose computational domain $\Omega = [0,2000\mu m]\times[0,\pi/2]$, the g-th radiation source S_g is converted from the g-th radiation temperature which is known and tested in SG-II laser facility [21], the number of groups $G = 20$, radiation diffusion coefficient $D_g(E_g) = c\lambda(E_g)/\kappa_g$, electron and ion thermal-conductivity coefficients $D_\alpha(T_\alpha) = k_\alpha T_\alpha^{5/2}$ ($\alpha = E, I$), where $\lambda(E_g)$ is a nonlinear limiter, κ_g is the Rosseland mean absorption coefficient, k_α ($\alpha = E, I$) are constants and c is the speed of light. We impose zero-flow boundary conditions at all physical boundaries of electron and ion temperatures, as well as the angle directions and spherical center of radiation energy densities, but inflow boundary conditions $F_g = cE_g/4 + F_{-g}$ at the outer radius, where F_{-g} is the inflow which is known. Additionally, the initial velocity of the fluid is zero, the initial radiation, electron and ion temperatures are 3.0×10^{-4} in the whole computational domain, and the initial spectral radiation energy densities are converted from radiation temperatures via Plank interpolation. The energy exchanges between photons of different frequency, electron and ion will take place during the time stepping advance process, diffuse to the spherical center and eventually run up to their diffusion equilibria in the region. In our simulation, LARED-S code runs 0.1 nanoseconds in physical time, and the whole simulation needs 120 time steps on a 16000×24 grid. Due to the need of 5 to 10 Picard iterations at each time step, there are altogether 1020 linear algebraic systems solved by PGMRES method, with the maximal number of iterations 100 and a decrease in residuals by 8 orders of magnitude. Note that ILU(0) can't converge in 100 iterations for certain time steps, which would seriously affect the accuracy of the simulation, thus we only compare the performance of B_{co}^{α} with B_{co} and AMG.

Fig. 2 compares the number of iterations for all of the first-Picard systems, which arise from the first Picard iteration at each time step.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

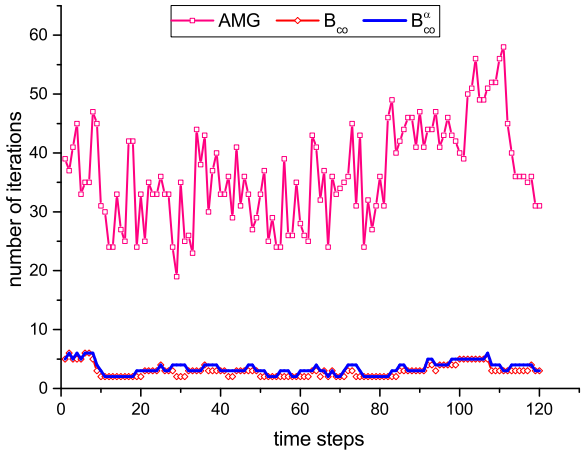


Figure 2: Number of iterations for the first-Picard systems.

Table 8 gives the total number of iterations and CPU time (in hours) for these 120 time steps.

Table 8: Total number of iterations and CPU time (in hours) for 120 time steps.

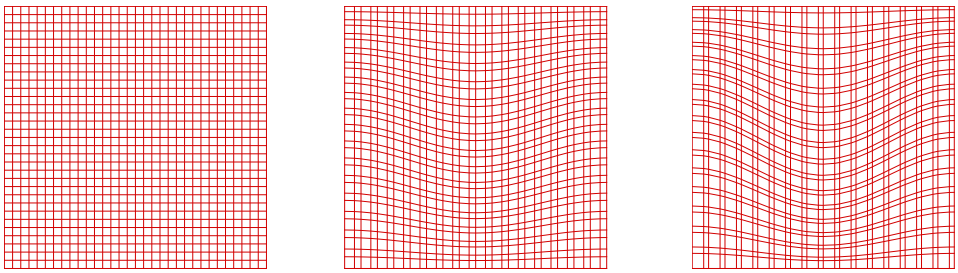
		AMG	B_{co}	B_{co}^{α}
Total number of iterations		43860	3570	3774
CPU Time	Linear Solver's CPU Time	63.0	17.7	11.6
	Total CPU Time	75.0	29.1	22.9

From Fig. 2 and Table 8, we can obtain that both B_{co} and B_{co}^{α} have a reduction up to 12 compared with AMG in the number of iterations, and they are quite robust with respect to the time step. Moreover, B_{co}^{α} runs 5.4 and 1.5 times faster than AMG and B_{co} in Linear Solver's CPU Time, 3.3 and 1.3 times in Total CPU Time. In addition, excluding the Linear Solver's CPU Time, it respectively costs 12.0, 11.4 and 11.3 hours for AMG, B_{co} and B_{co}^{α}, which clarifies B_{co}^{α} accelerates the simulation with a bit higher precision. Hence, it again indicates the obvious advantage of B_{co}^{α} over ILU(0), AMG and B_{co}.

Remark 5.1. (Resistance capability to grid-distortion) A model is further considered to demonstrate the capability of the new preconditioner for the solutions of radiation diffusion problems on distorted meshes (see Fig. 3(b) and 3(c)), which are generated by $x_{ij}=(i+mod(i,2)\theta_1)/n_x$ and $y_{ij}=\gamma_j+\theta_2\gamma_j(1-\gamma_j)\cos(2\pi x_{ij})$ on the unit square domain by $n_x\times n_y$ grid and a linear mapping, where $\gamma_j=(j+mod(j,2)\theta_1)/n_y$. We respectively denote the distorted meshes in Fig. 3 as Q_a, Q_b and Q_c for the convenience of description.

Table 9 displays the performance of B_{co}^{α} in comparison with AMG and B_{co} to classical one-group problems S_2 and S_3, where nine-point spatial discretization method [2] is used. Similar to the other problems, the convergence rates of B_{co}^{α} and B_{co} are approximately

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press



(a) Q_a (b) Q_b (c) Q_c

Figure 3: Illustration for distorted meshes when (a) $\theta_1=\theta_2=0$, (b) $\theta_1=\theta_2=\frac{1}{5}$, (c) $\theta_1=\frac{2}{5}$, $\theta_2=\frac{1}{3}$.

equal, yet an apparent advantage over AMG and B_{co} in terms of CPU time under Q_a. Moreover, it is easy to see that AMG and B_{co} both break down in cases of Q_b and Q_c, but B_{co}^{α} works well for them, which not only manifests the fact that B_{co}^{α} has a higher resistance capability to grid-distortion, but also derives a further declaration that B_{co}^{α} will be feasible and applicable in large-scale realistic Lagrangian hydrodynamic simulations.

Table 9: Number of iterations and wall time of AMG, B_{co} and B_{co}^{α} under distorted meshes.

		8000×6						16000×12					
		AMG		B_{co}		B_{co}^{α}		AMG		B_{co}		B_{co}^{α}	
		It	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c	*It*	T_c
Q_a	S_2	16	0.51	3	0.63	4(1)	0.23	53	7.67	4	4.07	4(1)	1.18
	S_3	4	0.28	2	0.65	4	0.29	4	1.34	3	3.01	4	1.38
Q_b	S_2	51	1.81	4	0.78	3(1)	0.28	-	-	-	-	6(1)	1.76
	S_3	4	0.37	2	0.77	4	0.39	5	2.03	2	3.28	5	2.10
Q_c	S_2	106	3.62	3	0.73	4(1)	0.30	-	-	-	-	4(1)	1.51
	S_3	4	0.35	2	0.74	4	0.35	5	1.97	3	3.58	5	2.03

6 Convergence analysis

In this section, we investigate the convergence analysis of $\tilde{B}_{co}$-GMRES(m) method, that is to say, in what follows is the effectiveness of the nonsymmetric combined preconditioner $\tilde{B}_{co}$ we construct in Algorithm 1. In order to achieve the theoretical estimate, not quantitative but qualitative, it is desirable to make it a transformation, which below we explain in more detail.

Firstly, we assume that the iterative behavior of $\tilde{B}_{co}$ is approximately equivalent to the one obtained by the symmetric combined preconditioner B_{co}, which is manipulated to ensure that the convergence analysis of B_{co}-GMRES(m) method is roughly identical to

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

that of $\tilde{B}_{co}$-GMRES(m) method. This assumption does have certain rationality in radiation diffusion problems whose numerical results are shown in Table 2, Fig. 2 and Table 9 under Q_a.

In addition, using the fact that B_{co} is SPD and recalling the convergence theory of GMRES(m) method [17,22], we can show that the convergence factor η of B_{co}-GMRES(m) method, associated with Euclidean norm, satisfies

$$\eta = \left(1-\frac{\alpha}{\beta}\right)^{\frac{m}{2}}, \tag{6.1}$$

provided that A_1 is an arbitrary positive real matrix, where $\tilde{A}_1 = B_{co}^{\frac{1}{2}} A_1 B_{co}^{\frac{1}{2}}$, $M = (\tilde{A}_1 + \tilde{A}_1^T)/2$, $\alpha = \lambda_{\min}^2(M)$, $\beta = \lambda_{\max}(\tilde{A}_1^T \tilde{A}_1)$.

Furthermore, when A_1 is essentially symmetric, the following estimate can be derived from the above result

$$\beta \approx \lambda_{\max}(M^T M) = \lambda_{\max}^2(M).$$

Substituting this into Eq. (6.1), yields

$$\eta \approx \left(1-\frac{\lambda_{\min}^2(M)}{\lambda_{\max}^2(M)}\right)^{\frac{m}{2}} = \left(1-\frac{1}{\kappa^2(M)}\right)^{\frac{m}{2}} = \left(1-\frac{1}{\kappa^2(B_{co}A)}\right)^{\frac{m}{2}}, \tag{6.2}$$

where the symmetrization $A = (A_1 + A_1^T)/2$ is SPD.

With the help of the preceding transformation, in particular by Eq. (6.2), we can conclude that the convergence analysis of $\tilde{B}_{co}$-GMRES(m) method to the equation $A_1 u = f$ depends approximatively on the scale of $\kappa(B_{co}A)$, if the coefficient matrix A_1 satisfies certain assumptions. And the smaller $\kappa(B_{co}A)$ the faster convergence of $\tilde{B}_{co}$-GMRES(m) method and the fewer iterations.

For simplicity of presentation, we denote the ILU(k) factorization by B, and AMG method by S. Utilizing the definition of B_{co} in [15], the following equality can be easily derived

$$B_{co} = S + S^T - S^T A_1 S + (I - S^T A_1) B (I - A_1 S).$$

Consequently, when A_1 is an essentially symmetric matrix, B_{co} does approximatively satisfy the expression

$$B_{co} \approx \tilde{S} + (I - S^T A) B (I - AS),$$

where $\tilde{S} = S + S^T - S^T AS$.

We are now turning to the estimation of $\kappa(B_{co}A)$ in the case when the operator B is SPD. The particular case of our most interest is that B_{co} has a better preconditioning behavior than both B and $\tilde{S}$, an alternative way to depict the case is

$$\kappa(B_{co}A) < \kappa(BA) \tag{6.3}$$

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

and

$$\kappa(B_{co}A)<\kappa(\tilde{S}A), \tag{6.4}$$

where the quantity $\kappa(C)$ stands for the condition number of an SPD matrix C. Next we seek the conditions under which the above two equalities hold. To do this, an assumption must be made regarding the SPD property of B and the contractionary effect of I-SA in the so-called energy norm.

Assumption 1. Assume without loss of generality that the following min-max eigenvalues of B (with proper scaling) hold

$$0<m_0=\lambda_{\min}(BA)\leq 1<m_1=\lambda_{\max}(BA) \tag{6.5}$$

and S satisfies the relation

$$\rho=\max_{v\in R^n,v\neq 0}\frac{\|(I-SA)v\|_A^2}{\|v\|_A^2}\in(0,1). \tag{6.6}$$

By Assumption 1, the following essential inequality has been proven by Hu et al. [15]

$$\kappa(B_{co}A)\leq\frac{(1-m_1)(1-\rho)+m_1}{(1-m_0)(1-\rho)+m_0}, \tag{6.7}$$

in succession to utilize the assumption stated in Eq. (6.5), yielding the desired expression (6.3).

Under the same assumption, Hu and his co-authors also give a sufficient condition to ensure Eq. (6.4) stated as Theorem 2.2 in [15]

$$\rho>1-\frac{m_0}{m_1-1}. \tag{6.8}$$

However, we would like to emphasize that the above is derived from exploiting the relation

$$\kappa(\tilde{S}A)=\frac{1}{1-\rho}, \tag{6.9}$$

whose invalidity can be demonstrated by Example 2.

Example 2. Consider $A=\mathbf{diag}(1.0,1.0,1.0)$ and $S=\mathbf{diag}(0.2,0.5,0.8)$. Note that S satisfies the assumption stated in Eq. (6.6), yet without the validity of Eq. (6.9).

Proof. Indeed, it is evident via straightforward manipulation to see that

$$\begin{aligned}\rho&=\max_{v\in R^3,v\neq 0}\frac{\|(I-SA)v\|_A^2}{\|v\|_A^2}=\max_{v\in R^3,v\neq 0}\frac{((I-SA)v,(I-SA)v)_A}{(v,v)_A}\\&=\max_{v=(v_1,v_2,v_3)^T\neq 0}\frac{(0.8v_1)^2+(0.5v_2)^2+(0.2v_3)^2}{v_1^2+v_2^2+v_3^2}=0.64<1,\end{aligned}$$

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

which establishes that S satisfies Eq. (6.6).

It remains to prove a contradiction to Eq. (6.9). From the definition of S and $\tilde{S}$, we have

$$\tilde{S}=S+S^T-S^TAS=\mathbf{diag}(0.36,0.75,0.96).$$

Clearly yields

$$\kappa(\tilde{S}A)=\frac{0.96}{0.36}\neq\frac{1}{0.36}=\frac{1}{1-\rho},$$

which completes the proof. □

Apparently, only under Assumption 1, the above analysis shows that the sufficient condition (6.8), given by Hu et al. in [15], isn't adequate to guarantee Eq. (6.4). Therefore, it is necessary to propose a new sufficient condition. We begin with the following two lemmas.

Lemma 6.1. (Theorem 2.2 in [15]) *Suppose that S satisfies the assumption stated in Eq.* (6.6), *then $\tilde{S}$ is such that*

$$(1-\rho)(w,w)_A\leq(\tilde{S}Aw,w)_A\leq(w,w)_A,\quad\forall w\in R^n. \tag{6.10}$$

Proof. See reference [15] for a proof. □

Lemma 6.2. *Suppose that S satisfies the assumption stated in Eq.* (6.6), *then there exists a scalar $\alpha\in[0,\rho]$ such that*

$$\kappa(\tilde{S}A)=\frac{1}{1-\alpha}. \tag{6.11}$$

Proof. It follows from Eq. (6.10) that

$$\lambda_{\max}(\tilde{S}A)\leq 1 \tag{6.12}$$

and

$$\lambda_{\min}(\tilde{S}A)\geq 1-\rho.$$

Taking the SPD property of $\tilde{S}$ and A into consideration, we have

$$(\tilde{S}Aw,w)_A\geq(1-\rho)(w,w)_A,\quad\forall w\in R^n. \tag{6.13}$$

According to Eq. (6.6), there exists $w_0\neq 0$ such that

$$\rho=\frac{\|(I-SA)w_0\|_A^2}{\|w_0\|_A^2}.$$

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Note that $((I-\tilde{S}A)w,w)_A=\|(I-SA)w\|_A^2$, the above equality immediately becomes

$$(\tilde{S}Aw_0,w_0)_A=(1-\rho)(w_0,w_0)_A. \tag{6.14}$$

Combining Eq. (6.13) with Eq. (6.14), yields

$$\lambda_{\min}(\tilde{S}A)=1-\rho.$$

By recalling the definition of the condition number, we have

$$1\leq\kappa(\tilde{S}A)\leq\frac{1}{1-\rho}.$$

Hence, a scalar $\tilde{\alpha}\in[1-\rho,1]$ exists such that

$$\kappa(\tilde{S}A)=\frac{1}{\tilde{\alpha}},$$

then select the scalar $\alpha=1-\tilde{\alpha}\in[0,\rho]$ and complete the proof. □

A consequence of Lemma 6.1 and Lemma 6.2 is the following theorem regarding the new sufficient condition ensuring Eq. (6.4).

Theorem 6.1. *Suppose that B and S satisfy Assumption* 1*, then Eq.* (6.4) *holds if*

$$m_1\rho(1-\alpha)-m_0\rho<\alpha(1-\rho), \tag{6.15}$$

where $\alpha=1-1/\kappa(\tilde{S}A)$.

Proof. The proof of this result is straightforward by eliminating the relation

$$\frac{(1-m_1)(1-\rho)+m_1}{(1-m_0)(1-\rho)+m_0}<\frac{1}{1-\alpha},$$

where the last item comes from Eq. (6.7) and Eq. (6.11). This completes the proof. □

It is worth noting that, the sufficient condition (6.15) would simplify into Eq. (6.8) when $\alpha=\rho$, yielding the conclusion stated in [15]. Unfortunately, the item α does not always achieve its supremum value ρ only under Assumption 1, such as $\alpha=0.625\neq\rho=0.64$ in Example 2, since $\lambda_{\max}(\tilde{S}A)=0.96$ and $\lambda_{\min}(\tilde{S}A)=0.36$. Therefore, the condition (6.15) assuredly differ from Eq. (6.8) with more accuracy.

Moreover, we expect to shrink the result of Theorem 6.1 by appending a certain condition, which is stated in the following theorem with the purpose to ensure Eq. (6.8) as a sufficient condition of Eq. (6.4) when Assumption 1 holds.

Theorem 6.2. *Suppose that B and S satisfy Assumption* 1*, and SA admits 1 as an eigenvalue. Then Eq.* (6.4) *holds when S satisfies Eq.* (6.8)*.*

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Proof. It is obvious from the above qualitative analysis that the theorem will be proved if we can show $\alpha=\rho$ under its conditions. Here is the detailed proof.

If 1 is an eigenvalue of SA, then there exists a nonzero vector $w_0\neq 0$ such that

$$(I-SA)w_0=0.$$

Noting that $((I-\tilde{S}A)w,w)_A=\|(I-SA)w\|_A^2$, we have

$$(\tilde{S}Aw_0,w_0)_A=(w_0,w_0)_A,$$

and derive that

$$\lambda_{\max}(\tilde{S}A)\geq 1,$$

which, by a combination with the relation (6.12), yields

$$\lambda_{\max}(\tilde{S}A)=1.$$

By exploiting the above equality to undergo suitable modifications to the proof of Lemma 6.2, we obtain the desired result $\alpha=\rho$, and thus prove the theorem. □

Thus we can summarize what we have approximated and proved in this section that the condition number of $\tilde{B}_{co}$ preconditioned system is better than that of both B and $\tilde{S}$ preconditioned system when the coefficient matrix A_1, its ILU(0) factorization serves as B and AMG as S all satisfy certain conditions. This guarantees that $\tilde{B}_{co}$-GMRES(m) gains a faster convergence than B-GMRES(m) and $\tilde{S}$-GMRES(m), which, in the theoretical viewpoint, offers great help for validating the numerical results of our proposed preconditioners in previous sections.

7 Conclusion

In this paper, we proposed an adaptive combined preconditioner B_{co}^{α}, employed to solve discrete problems arising from MGD equations and a simulation of hydrodynamic instability during the processes of radiation driven ICF capsule implosion. The approach was motivated by the recent work from Hu et al. It relied on three adaptability conditions suitable to solve by AMG-GMRES, the crucial reduction of ILU(k)-like on multiscale nature of original matrices, heuristic remedy on the reduction in the residual norm of I$\hat{L}\hat{U}$(0)-GMRES, and the underlying complementarity of ILU(k) and AMG. Our numerical results verified the considerable advantage of B_{co}^{α}. In addition, we analyzed qualitatively the convergence estimation of a nonsymmetric combined preconditioner, with a careful correction on the existing estimation theory of B_{co} established by Hu et al. and an immediate deduction into the existing by an extra condition.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

Acknowledgments

Yue is grateful for the financial support by Hunan Provincial Innovation Foundation for Postgraduate (CX2013B255) and Specialized Research Fund for the Doctoral Program of Higher Education of China Grant 20124301110003. Shu is partially supported by NSFC Grants 91130002 and 11571293. Xu is partially supported by NSFC Grant 61370067. The authors would like to thank the editor and two anonymous reviewers for their constructive comments which helped us to improve the quality of this work.

References

[1] W. B. Pei, The construction of simulation algorithms for laser fusion, Commun. Comput. Phys. 2 (2) (2007), pp. 255-270.
[2] S. W. Fu, H. Q. Fu, L. J. Shen, S. K. Huang, and G. N. Chen, A nine point difference scheme and iteration solving method for two dimensional energy equations with three temperatures, Chin. J. Comput. Phys. 15 (4) (1998), pp. 489-497.
[3] T. X. Gu, Z. H. Dai, X. D. Hang, S. W. Fu, and X. P. Liu, Efficient algebraic methods for two-dimensional energy equations with three temperatures, Chin. J. Comput. Phys. 22 (6) (2005), pp. 471-478.
[4] Z. Y. Mo, S. W. Fu, and L. J. Shen, Parallel lagrange numerical simulations for 2-dimension three temperature hydrodynamics, Chin. J. Comput. Phys. 17 (6) (2000), pp. 625-632.
[5] Z. Y. Mo and S. W. Fu, Application of krylov iterative methods in two dimensional three temperatures energy equation, J. Numer. Meth. Comput. Appl. 24 (2) (2003), pp. 133-143.
[6] Z. Y. Mo, L. J. Shen, and G. Wittum, Parallel adaptive multigrid algorithm for 2-D 3-T diffusion equations, Int. J. Comput. Math. 81 (3) (2004), pp. 361-374.
[7] J. P. Wu, X. P. Liu, Z. H. Wang, Z. H. Dai, and X. M. Li, Two preconditioning techniques for two-dimensional three-temperature energy equations, Chin. J. Comput. Phys. 22 (4) (2005), pp. 283-291.
[8] Y. X. Xiao, S. Shu, P. W. Zhang, Z. Y. Mo, and J. C. Xu, A kind of semi-coarsing AMG method for two dimensional energy equations with three temperatures, J. Numer. Meth. Comput. Appl. 24 (4) (2003), pp. 293-303.
[9] X. W. Xu, Z. Y. Mo, and H. B. An, Algebraic two-level iterative method for 2-D 3-T radiation diffusion equations, Chin. J. Comput. Phys. 26 (1) (2009), pp. 1-8.
[10] C. Baldwin, P. N. Brown, R. Falgout, F. Graziani, and J. Jones, Iterative linear solvers in 2D radiation-hydrodynamics code: methods and performance, J. Comput. Phys. 154 (1999), pp. 1-40.
[11] Y. Saad, Iterative Methods for Sparse Linear Systems, SIAM (2003).
[12] D. Hysom and A. Pothen, A scalable parallel algorithm for incomplete factor preconditioning, SIAM J. Sci. Comput. 22 (6) (2001), pp. 2194-2215.
[13] A. Brandt, S. F. McCormick, and J. Ruge, Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations, Institute for Computational Studies (1982).
[14] J. W. Ruge and K. Stüben, Algebraic multigrid, in Multigrid Methods, Front. Appl. Math. 3 (1987), pp. 73-130.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press

[15] X. Z. Hu, S. H. Wu, X. H. Wu, J. C. Xu, C. S. Zhang, S. Q. Zhang, and L. Zikatanov, Combined Preconditioning with Applications in Reservoir Simulation, Multiscale Model. Simul. 11 (2) (2013), pp. 507-521.
[16] X. D. Hang, J. H. Li, and G. W. Yuan, Convergence analysis on splitting iterative solution of multi-group radiation diffusion equations, Chin. J. Comput. Phys. 30 (1) (2013), pp. 111-119.
[17] Y. Saad and M. H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986), pp. 856-869.
[18] V. E. Henson and U. M. Yang, BoomerAMG: a parallel algebraic multigrid solver and preconditioner, Appl. Numer. Math. 41 (2002), pp. 155-177.
[19] X. W. Xu, Research on scalable parallel algebraic multigrid algorithms, Ph.D. thesis (2007), Chinese Academy of Engineering Physics.
[20] Z. Y. Mo, A. Q. Zhang, X. L. Cao, Q. K. Liu, X. W. Xu, H. B. An, W. B. Pei, and S. P. Zhu, JASMIN: a parallel software infrastructure for scientific computing, Front. Comput. Sci. 4 (4) (2010), pp. 480-488.
[21] H. S. Peng, Z. J. Zheng, B. H. Zhang, et al., Direct-drive implosion experiments on the SG-II laser facility, J. Fusion Energ. 19 (1) (2000), pp. 81-85.
[22] H. C. Elman, Iterative methods for large sparse nonsymmetric systems of linear equations, Ph.D. thesis (1982), Yale University.

https://doi.org/10.4208/cicp.091014.060315a Published online by Cambridge University Press