

## A Subspace-Projected Approximate Matrix Method for Systems of Linear Equations

Jan Brandts\* and Ricardo R. da Silva

University of Amsterdam, Faculty of Science, Korteweg-de Vries Institute for  
Mathematics, Science Park 904, Amsterdam, Netherlands.

Received 7 February 2013; Accepted (in revised version) 28 May 2013

Available online 31 May 2013

---

**Abstract.** Given two  $n \times n$  matrices  $A$  and  $A_0$  and a sequence of subspaces  $\{0\} = \mathcal{V}_0 \subset \dots \subset \mathcal{V}_n = \mathbb{R}^n$  with  $\dim(\mathcal{V}_k) = k$ , the  $k$ -th *subspace-projected approximated matrix*  $A_k$  is defined as  $A_k = A + \Pi_k(A_0 - A)\Pi_k$ , where  $\Pi_k$  is the orthogonal projection on  $\mathcal{V}_k^\perp$ . Consequently,  $A_k v = Av$  and  $v^* A_k = v^* A$  for all  $v \in \mathcal{V}_k$ . Thus  $(A_k)_{k \geq 0}^n$  is a sequence of matrices that gradually changes from  $A_0$  into  $A_n = A$ . In principle, the definition of  $\mathcal{V}_{k+1}$  may depend on properties of  $A_k$ , which can be exploited to try to force  $A_{k+1}$  to be closer to  $A$  in some specific sense. By choosing  $A_0$  as a simple approximation of  $A$ , this turns the subspace-approximated matrices into interesting preconditioners for linear algebra problems involving  $A$ . In the context of eigenvalue problems, they appeared in this role in Shepard *et al.* (2001), resulting in their *Subspace Projected Approximate Matrix method*. In this article, we investigate their use in solving linear systems of equations  $Ax = b$ . In particular, we seek conditions under which the solutions  $x_k$  of the approximate systems  $A_k x_k = b$  are computable at low computational cost, so the efficiency of the corresponding method is competitive with existing methods such as the Conjugate Gradient and the Minimal Residual methods. We also consider how well the sequence  $(x_k)_{k \geq 0}$  approximates  $x$ , by performing some illustrative numerical tests.

**AMS subject classifications:** 65F10, 65F08

**Key words:** Linear system, Galerkin method, subspace projected approximate matrix, minimal residual method.

---

### 1. Introduction

Subspace iterative methods for linear algebra problems are based on the repeated application of two consecutive ideologically separate steps — viz. the generic *selection* of a suitable approximation from the space, followed by increasing its dimension in an *expansion* of the subspace. In this section, we recall two well-known selection mechanisms in the context of linear systems — viz. the *Ritz-Galerkin* and *Minimal Residual* procedures. We then suggest an alternative.

---

\*Corresponding author. *Email addresses:* J.H.Brandts@uva.nl (J. Brandts), R.J.ReisdaSilva@uva.nl (R.R. da Silva)

### 1.1. Selecting approximations from a subspace $\mathcal{V}$

In the context of solving a linear system  $Ax = b$  and a given a subspace  $\mathcal{V} \subset \mathbb{C}^n$ , two well known and successful selection mechanisms are the *Ritz-Galerkin* approximation

$$\text{Ritz-Galerkin: Find } \hat{x} \in \mathcal{V} \text{ such that } b - A\hat{x} \perp \mathcal{V}, \quad (1.1)$$

and the *Minimal Residual* approximation

$$\text{Minimal Residual: Find } \hat{x} \in \mathcal{V} \text{ for which } \|b - A\hat{x}\| \text{ is minimal.} \quad (1.2)$$

On choosing a matrix  $V$  with columns spanning  $\mathcal{V}$  and denoting the transpose by the superscript  $t$ , the Ritz-Galerkin approximation can be computed by solving

$$\hat{x} = Vy \quad \text{where} \quad V^t(b - AVy) = 0, \quad (1.3)$$

whereas the Minimal Residual approximation can be computed by solving

$$\hat{x} = Vy \quad \text{where} \quad (AV)^t(b - AVy) = 0, \quad (1.4)$$

because the minimum in (1.2) is realised by the  $\hat{x} \in \mathcal{V}$  for which  $A\hat{x}$  equals the orthogonal projection of  $b$  on  $A\mathcal{V}$ . Although neither approximation  $\hat{x}$  depends on the actual choice of the basis  $V$  for  $\mathcal{V}$ , the basis is of interest for efficient implementation of the method. For example, if the matrix  $A$  is symmetric and positive definite, in the Ritz-Galerkin approach the basis can be chosen to be  $A$ -orthonormal — i.e. such that  $V^tAV = I$  and consequently  $\hat{x} = Vy = VV^tb$ . If the space  $\mathcal{V}$  is then *expanded* by appending another  $A$ -orthonormal basis vector  $v$  to  $V$ , the new approximation differs only by a simple update  $vv^tb$  from the previous one. However, this elegant outcome should not deter us from considering other options for the basis for  $\mathcal{V}$ .

**Remark 1.1.** If the spaces  $(\mathcal{V}_k)_{k=0}^n$  form a sequence of Krylov subspaces, the Ritz-Galerkin approach leads to the *Conjugate Gradient method* [9] when  $A$  is symmetric and positive definite, and the *Full Orthogonalization method* for general non-symmetric  $A$ . The Minimal Residual approach leads to the *MinRES* [12] method if  $A$  is symmetric, and to the *GMRES* [16] method for non-symmetric  $A$ . The Ritz-Galerkin approach is also used in *coarse grid corrections* within the *MultiGrid* method. Both the Ritz-Galerkin and the Minimal Residual approach are also used in finite element methods to approximately solve partial differential equations.

### 1.2. Selecting an approximation associated with a subspace $\mathcal{V}$

The Ritz-Galerkin and the Minimal Residual approach define approximations  $v$  from the space  $\mathcal{V}$ , in the sense that  $\hat{x} \in \mathcal{V}$ . If the space has dimension  $k \ll n$ , not more than  $k$  matrix-vector products (MVPs) with  $A$  are needed to compute these approximations. Since the number of MVPs with  $A$  is often a good indication of the cost of a subspace iterative

method, in this article we investigate alternative selection methods that need at most  $k$  MVPs with  $A$ .

Returning to the case that  $A$  is symmetric and positive definite and the Ritz-Galerkin method is applied, we observe the following. On writing  $V_\perp$  for any matrix that spans the  $A$ -orthogonal complement of  $\mathcal{V}$ , we have that

$$A^{-1} = VV^t + V_\perp V_\perp^t. \quad (1.5)$$

This suggests that the method is unable to include any information about  $b$  in the approximation process other than  $VV^t b$ , and the cost of forming  $V_\perp V_\perp^t b$  is of the same order of magnitude as inverting  $A$ . On the other hand, we may choose  $V$  such that  $V^t V = I$ . Then instead of the elegant expression  $\hat{x} = VV^t b$  we get  $\hat{x} = V(V^t A V)V^t b$ , which involves solving a  $k \times k$  linear system. However, in return we do have  $V_\perp V_\perp^t = I - VV^t$ , which unlike (1.5) might provide an opportunity to somehow use the information from the term  $V_\perp V_\perp^t b$  in the approximation process.

Indeed, rather than an  $A$ -orthonormal and an  $I$ -orthonormal basis, there may be a range of feasible choices. Generally, if  $H$  is a symmetric and positive definite approximation of  $A$  with inverse  $H^{-1}$  that is explicitly available or inexpensive to apply, the relation  $H^{-1} = VV^t + V_\perp V_\perp^t$  that holds if  $(V|V_\perp)$  represents an  $H$ -orthonormal basis, which may be used to define alternative approximations of  $x$  as investigated in this paper. Since such approximations do not necessarily lie *in* the space  $\mathcal{V}$ , we will call them approximations *associated with*  $\mathcal{V}$ . As with approximations *from*  $\mathcal{V}$ , their MVPs with  $A$  equals  $k$ , and  $k$  linear systems with  $H$  need to be solved. Since  $H$  may be used as a preconditioner of the original linear system  $Ax = b$ , the fairest comparison with the Ritz-Galerkin and Minimal Residual methods that select approximation *from*  $\mathcal{V}$  is with their *preconditioned* versions.

It emerges that the notion of an approximation associated with a space  $\mathcal{V}$  leads to a connection with the *Subspace Projected Approximate Matrix* method [17], or SPAM for short. In this subspace iterative method for eigenvalue problems, a sequence  $(A_k)_{k \geq 0}$  is defined based on an initial approximation  $A_0$  of  $A$ . The matrices  $A_k$ , called *subspace-projected approximate matrices*, are increasingly better approximations of  $A$ . Apart from their use in the SPAM method, at least theoretically they can be used to define approximations of the solution  $x$  of  $Ax = b$ , for instance by the relation  $A_k x_k = b$ . Indeed, since each matrix  $A_k$  is *associated with* a subspace  $V_k$ , this renders  $x_k$  as an approximation *associated with* this space rather than *from* this space. The main concern is that  $x_k$  should be computable efficiently, and if possible by using at most  $k$  MVPs with  $A$ .

### 1.3. Outline

In Section 2, we recall the definition and use of the Subspace Projected Approximate matrices [17], leading to their SPAM eigenvalue method — cf. also [4, 8, 11, 14, 20]. In Section 3, the selection mechanism based on the subspace-projected approximate matrices is developed and converted into an algorithm. Section 4 discusses the selection in combination with an arbitrary expansion and with a Krylov subspace expansion of the sequence of subspaces, and then some numerical tests. Our concluding remarks are in Section 5.

## 2. The SPAM method for Eigenvalue Problems

We briefly recall the Subspace Projected Approximate Matrix method (SPAM) from Ref. [17]. This is a so-called *inner-outer iteration method* to compute approximations of eigenvalues of a Hermitian matrix  $A$  via an approximation  $A_0$  of  $A$  in a particular way.

### 2.1. Subspace Projected Approximate Matrices

Given a sequence of subspaces  $\{0\} = \mathcal{V}_0 \subset \dots \subset \mathcal{V}_n = \mathbb{R}^n$  with  $\dim(\mathcal{V}_k) = k$ , let us write  $V_k$  for the  $n \times k$  orthogonal matrix where the first  $j$  columns span  $\mathcal{V}_j$  for each  $j \in \{1, \dots, k\}$ , and write  $\Pi_k = I - V_k V_k^t$  for the orthogonal projection on  $\mathcal{V}_k^\perp$ .

**Definition 2.1** ([17]). Let  $A$  be an  $n \times n$  matrix, and  $A_0$  an approximation of  $A$ . Then for each  $k \in \{0, \dots, n\}$  the matrix

$$A_k = A + \Pi_k(A_0 - A)\Pi_k \tag{2.1}$$

is called the *kth subspace-projected approximated matrix* of  $A$  based on  $A_0$ .

This definition is consistent for  $k = 0$  and  $A_n = A$ . Moreover,

$$A_k V_k = A V_k \quad \text{and} \quad V_k^t A_k = V_k^t A, \quad \text{and in particular} \quad M = V_k^t A_k V_k = V_k^t A V_k. \tag{2.2}$$

In other words, with respect to the orthonormal basis of  $\mathbb{R}^n$  given by the columns of  $V_n$  the matrices  $A$  and  $A_k$  take the block form

$$V_n^t A V_n = \begin{bmatrix} M & G^t \\ R & S \end{bmatrix} =: \hat{A} \quad \text{and} \quad V_n^t A_k V_n = \begin{bmatrix} M & G^t \\ R & \hat{S} \end{bmatrix} =: \hat{A}_k, \tag{2.3}$$

where  $S$  and  $\hat{S}$  are of size  $(n - k) \times (n - k)$  and  $A_k$  is a rank-2 update of  $A_{k-1}$ . Indeed,

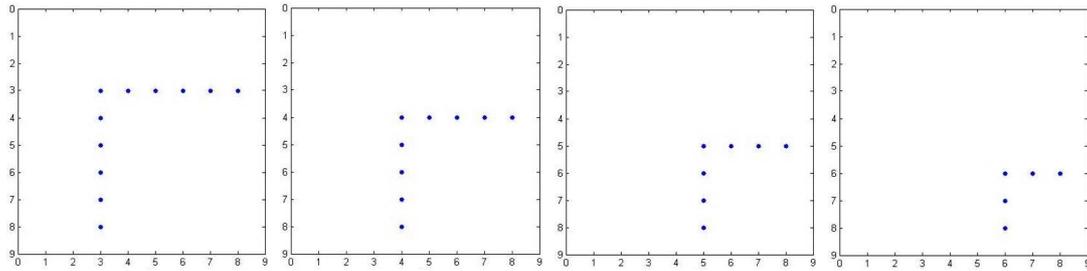


Figure 1: Sparsity pattern of the rank-2 updates for increasing values of  $k$ .

with respect to the basis  $V_n$  in (2.3), the matrices  $\hat{A}_k$  and  $\hat{A}_{k-1}$  can easily be seen to differ only at the  $2(n - k) + 1$  matrix entries depicted in Fig. 1, so  $\hat{A}_0$  gradually changes into  $\hat{A}$  for increasing values of  $k$ . On transforming back to the standard basis of  $\mathbb{R}^n$ , one can verify that the following explicit updating formula is valid.

**Proposition 2.1.** *Let  $n \geq k \geq 1$ , and write  $V_k = (V_{k-1}|v)$ . Then  $A_k$  is the following rank-2 update of  $A_{k-1}$ :*

$$A_k = A_{k-1} + (\Pi_k w)v^t + v(\Pi_{k-1}u)^t, \quad (2.4)$$

where  $w = (A - A_0)v$  and  $u = (A - A_0)^t v$ .

*Proof.* Substituting  $\Pi_k = \Pi_{k-1} - vv^t$  into the definition of  $A_k$  in Eq. (2.1) and using the notations  $w = (A - A_0)v$  and  $u = (A - A_0)^t v$ , the statement is obtained after some rearrangement.  $\square$

**Remark 2.1.** When  $A$  and  $A_0$  are symmetric, the vectors  $u$  and  $w$  in Proposition 2.1 coincide and the rank-2 update is also symmetric. For general matrices, two matrix-vector products are needed to compute the update, one with  $A - A_0$  and one with  $A^t - A_0^t$ .

In Ref. [17], there is a dependence between the space  $\mathcal{V}_k$  and the matrices  $A_k$  other than that described above — viz. the next space  $\mathcal{V}_{k+1}$  is defined as  $\mathcal{V}_k \oplus \langle v \rangle$ , where  $v$  is a properly chosen eigenvector of  $A_k$ . We explain this in the next subsection, and by doing so introduce the *Subspace Projected Approximate Matrix method* to compute approximations to the eigenvalues of  $A$ .

## 2.2. Subspace-projected approximate matrices in eigenvalue computations

The subspace-projected approximate matrices  $(A_k)_{k \geq 0}$  are used to compute approximations of eigenvalues of an Hermitian matrix  $A$  [17]. Let  $\tau \in \mathbb{R}$  be the target — i.e. the eigenvalue of  $A$  closest to  $\tau$  we seek. Let  $\mathcal{V}_k \in \mathbb{R}^n$  be a given subspace. The matrix  $V_k$  with columns forming an orthonormal basis for  $\mathcal{V}_k$  is employed in the usual way to compute the *Ritz values of  $A$  in  $\mathcal{V}_k$* , which are the eigenvalues  $\mu_k \leq \mu_{k-1} \leq \dots \leq \mu_2 \leq \mu_1$  of the  $k \times k$  matrix  $M = V_k^* A V_k$ . The vectors  $u_j = V_k z_j$ , where  $\{z_1, \dots, z_k\}$  is an orthonormal basis for  $\mathbb{C}^k$  consisting of eigenvectors of  $M$  belonging to the corresponding  $\mu_j$ , are the *Ritz vectors of  $A$  in  $\mathcal{V}$* . The vectors  $r_j = Au_j - u_j \mu_j \perp \mathcal{V}_k$  associated with the respective *Ritz pairs*  $(\mu_j, u_j)$  are called the *residuals*. The Ritz pairs can be viewed as approximate eigenpairs of  $A$  — cf. for example [1, 10, 13, 18]. The way in which the next subspace  $\mathcal{V}_{k+1}$  is defined distinguishes the subspace methods for eigenvalue problems from each other. Unlike methods such as the Lanczos [6], Jacobi-Davidson [19], and Riccati [2] methods, in the SPAM method we have  $\mathcal{V}_k \oplus \langle v \rangle$  where  $v$  is (an approximation of) a *target eigenvector* of the subspace-projected approximate matrix  $A_k$  — i.e. an eigenvector corresponding to an eigenvalue of  $A_k$  closest to  $\tau$ . It is notable that after computing the Ritz pairs and their residuals corresponding to the space  $\mathcal{V}_k$ , the matrix  $A_k$  is available without performing any additional matrix-vector products with  $A$ . Since  $A_k$  resembles  $A$  more closely as  $k$  increases, the hope is that expanding  $\mathcal{V}_k$  with an approximate eigenvector of  $A_k$  enriches the search space and so leads to a Ritz pair in  $\mathcal{V}_{k+1}$  that is a better approximation of the target eigenvalue than the Ritz values computed in  $\mathcal{V}_k$ . With  $\mathcal{V}_{k+1}$  defined in terms of  $\mathcal{V}_k$ , it remains to define  $\mathcal{V}_1$ , which is (an approximation of) the target eigenvector of the initial approximating matrix  $A_0$ .

**Remark 2.2.** An inner iteration is needed to compute an approximation of the target eigenvector of  $A_k$ . The philosophy of Ref. [17] is that, by choosing  $A_0$  such that its action is much cheaper to apply than the action of  $A$  itself, the inner iteration will be less expensive than the Jacobi-Davidson method [19] for instance. Previously, we presented a mathematical and numerical comparison of SPAM with certain preconditioned versions of the Jacobi-Davidson and Lanczos methods. This analysis shows that the methods coincide for certain choices, but that there do exist circumstances in which SPAM performs slightly better than the Jacobi-Davidson method — cf. Ref. [3] for relevant details.

### 3. Definition of a SPAM Method for Linear Systems

We now investigate if subspace-projected approximate matrices can play a role in solving linear systems of equations  $Ax = b$ , for instance via the solutions  $x_k$  of the systems  $A_k x_k = b$ . As a reference point, we first recall the Conjugate Gradient method [9], and present the method in such a way that a correspondence with subspace-projected approximate matrices becomes evident. This leads to a definition of a SPAM-based method to approximate the solutions of linear systems.

#### 3.1. The $A$ -inner product and the Conjugate Gradient method

Let  $A$  be a symmetric positive definite  $n \times n$  matrix, and write  $\langle \cdot, \cdot \rangle_A$  for the so-called  $A$ -inner product defined by

$$\langle y, z \rangle_A = y^t A z. \quad (3.1)$$

Given a sequence of subspaces  $\{0\} = \mathcal{V}_0 \subset \dots \subset \mathcal{V}_n = \mathbb{R}^n$  with  $\dim(\mathcal{V}_k) = k$ , let us write  $V_k$  for the  $n \times k$  matrix where the first  $j$  columns  $v_1, \dots, v_j$  form an  $A$ -orthonormal basis for  $\mathcal{V}_j$  for each  $j \in \{1, \dots, k\}$ . Then  $V_n^t A V_n = I$  and

$$A^{-1} = V_n V_n^t = \sum_{j=1}^n v_j v_j^t, \quad \text{and thus} \quad P_k(y) = \sum_{j=1}^k v_j \langle y, v_j \rangle_A = \sum_{j=1}^k v_j v_j^t A y \quad (3.2)$$

is the  $A$ -orthogonal projection of  $y \in \mathbb{R}^n$  on  $\mathcal{V}_k$ . Therefore if  $Ax = b$ , the  $A$ -orthogonal projection of  $x$  on  $\mathcal{V}_k$  can be computed from  $b$  and  $V_k$  as

$$x_k := P_k(x) = \sum_{j=1}^k v_j v_j^t A x = V_k V_k^t b. \quad (3.3)$$

This shows that an iterative method can be set up that produces a sequence  $(x_k)_{k \geq 0}$  of  $x$  that converges monotonically to  $x$  in the norm associated with the  $A$ -inner product (3.1). One iteration encompasses  $A$ -orthonormalization of a new basis vector, and the computed update of the current approximation as

$$x_{k+1} = x_k + v_{k+1} v_{k+1}^t b. \quad (3.4)$$

If each subspace  $\mathcal{V}_k$  equals the  $k$ -th Krylov subspace  $\mathcal{K}^k(A, b)$  defined by

$$\mathcal{K}^k(A, b) = \text{span}\{b, Ab, \dots, A^{k-1}b\}, \quad (3.5)$$

the resulting method is mathematically equivalent to the Conjugate Gradient (CG) method [9]. In the next section, we will interpret CG as a particular instance of a subspace-projected approximate matrix method for linear systems.

### 3.2. Towards a SPAM-type linear system solver

Let the  $k$ -th search space  $\mathcal{V}_k$  be given. Assume that  $Y = (V|V_\perp)$  is nonsingular, that the columns of  $V$  span  $\mathcal{V}_k$ , and that  $k \ll n$ . Then  $Ax = b$  if and only if  $Y^tAYy = Y^tb$  with  $x = Yy$ , or

$$(V|V_\perp)^t \left( A(V|V_\perp) \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - b \right) = 0, \quad (3.6)$$

with

$$x = x_1 + x_2 = Vy_1 + V_\perp y_2 = (V|V_\perp) \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}. \quad (3.7)$$

This system can be written in block form as

$$\begin{bmatrix} V^tAV & V^tAV_\perp \\ V_\perp^tAV & V_\perp^tAV_\perp \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} V^tb \\ V_\perp^tb \end{pmatrix}. \quad (3.8)$$

Rewriting the original linear system  $Ax = b$  like this is in principle the same as applying *two-sided preconditioning* [7], as discussed in some detail below — cf. 3.2.3.

**Remark 3.1.** If the columns of  $Y$  form an  $A$ -orthonormal basis, then  $Y^tAY = I$  and the system (3.8) reduces to the situation described in the previous subsection. Note that  $V_\perp V_\perp^t = A^{-1} - VV^t$  and thus  $V_\perp V_\perp^t$  is unavailable in practice, so only  $x_1$  can be computed.

Motivated by the subspace-projected approximate matrices and their role in the SPAM method for eigenvalues, we now proceed to investigate alternative choices for  $Y = (V|V_\perp)$ , such that an approximation of  $x$  can be computed at relatively low cost and somehow involve the other blocks in the matrix — or possibly approximations of those blocks. We assume that  $V$ ,  $AV$  and  $V^tAV$  have all been computed explicitly.

**Remark 3.2.** It is important to realise that we do not change the *column span* of the matrix  $V$ , but consider different choices for the basis and apply different approximation steps. The alternatives that we propose therefore concern the way an approximation based on the *given spaces*  $\mathcal{V}_k$  is computed, and later we also investigate the way that the spaces are to be *expanded*.

### 3.2.1. Identifying and approximating the bottlenecks

The problems with finding low cost approximations of the solution of (3.8) become evident when we apply block elimination to (3.8) in order to separate the computation of  $y_1$  and  $y_2$ , for arbitrary non-singular  $Y = (V|V_\perp)$ . This results in

$$\left( V^t AV - V^t AV_\perp (V_\perp^t AV_\perp)^{-1} V_\perp^t AV \right) y_1 = V^t b - V^t AV_\perp (V_\perp^t AV_\perp)^{-1} V_\perp^t b, \quad (3.9)$$

$$V_\perp^t AV_\perp y_2 = V_\perp^t (b - AV y_1). \quad (3.10)$$

Three potential computational hazards can now easily be identified:

- (1) the formation and inversion of  $V_\perp^t AV_\perp$ , or equivalently, solving systems with  $V_\perp^t AV_\perp$ ;
- (2) the computation of  $V_\perp V_\perp^t z$  for a given  $z$ ;
- (3) the computation of  $V_\perp y_2$ , after  $y_2$  has been computed.

A possible way to tackle these problems is as follows. Let  $H$  be a symmetric positive definite approximation of  $A$  with known inverse  $H^{-1}$ , or such that solving systems is inexpensive. Assuming that the columns of  $Y$  form an  $H$ -orthonormal basis, we then have that

$$V_\perp V_\perp^t = H^{-1} - VV^t. \quad (3.11)$$

The second item above has then been accounted for without any loss of exactness. To also meet the first item, we apply an approximation step, by replacing  $A$  in each occurrence of the expression  $V_\perp^t AV_\perp$  with  $H$ . Since  $V_\perp$  is  $H$ -orthonormal we have that  $V_\perp^t H V_\perp = I$ , so given (3.11) we proceed to approximate (3.9)-(3.10) by

$$\left( V^t AV - V^t A (H^{-1} - VV^t) AV \right) \hat{y}_1 = V^t b - V^t A (H^{-1} - VV^t) b, \quad (3.12)$$

$$V_\perp \hat{y}_2 = (H^{-1} - VV^t) (b - AV \hat{y}_1), \quad (3.13)$$

where we have denoted the approximations of  $y_1$  and  $y_2$  by  $\hat{y}_1$  and  $\hat{y}_2$ . This results in an approximation  $\hat{x}$  of  $x$  through

$$\hat{x} = \hat{x}_1 + \hat{x}_2 = V \hat{y}_1 + V_\perp \hat{y}_2. \quad (3.14)$$

It is notable that in (3.13) we have left-multiplied the second equation by  $V_\perp$  in order to get a computable expression for  $V_\perp \hat{y}_2$ , which at the same time also solves the third item in the above list of potential computational hazards. It is easily seen that (3.12)-(3.13) is equivalent with the approximate linear system

$$\begin{bmatrix} V^t AV & V^t AV_\perp \\ V_\perp^t AV & I \end{bmatrix} \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \end{pmatrix} = \begin{pmatrix} V^t b \\ V_\perp^t b \end{pmatrix} \quad \text{with } Y = (V|V_\perp) \text{ and } Y^t H Y = I, \quad (3.15)$$

and that if  $H = A$  this system reduces to the original system (3.8).

The system matrix of the approximate linear system (3.15) is obviously related to the subspace-projected approximated matrices from Definition 2.1, and the block form of these

matrices in (2.3). The difference is only that  $(V|V_\perp)$  is orthogonal with respect to the  $H$ -inner product, instead of the standard inner product. Thus the system matrix in (3.15) is *identical* to the subspace-projected approximate matrix from Definition 2.1 (with approximation  $A_0 = I$  and subspace  $\mathcal{V}$ ) only if  $H = I$ . However, although  $H = I$  is the most efficient choice, it does not seem to be very attractive from a approximation theoretical viewpoint cf. 3.2.3. However, let us first comment on the computational cost for solving the system.

### 3.2.2. Inventory of the computational cost

A quick glance at (3.12) and (3.13) shows that, because  $V_\perp$  has now completely disappeared from the formulation (apart from implicitly in the term  $V_\perp \hat{y}_2$  in (3.13)), the numerical effort required to compute  $\hat{x}$  consists of (1)  $k$  matrix-vector products with  $A$ ; (2) certain inner products between  $n$ -vectors; (3) the solution of  $k + 1$  linear systems with  $H$ ; and (4) the solution of the  $k \times k$  linear system for  $\hat{y}_1$ . If we introduce the notation

$$Z = H^{-1}W \quad \text{where} \quad W = AV, \quad \text{and} \quad M = V^t W = W^t V, \quad (3.16)$$

then (3.12) and (3.13) may be rewritten in the more compact form

$$(M + M^2 - W^t Z) \hat{y}_1 = (MV^t + V^t - Z^t)b, \quad (3.17)$$

$$\hat{x}_2 = (H^{-1} - VV^t)b - (Z - VM)\hat{y}_1. \quad (3.18)$$

To get a more precise account of the cost in computing the approximation  $\hat{x}$  in (3.14), we investigate the effect of appending another column  $v$  to the  $n \times k$  matrix  $V$ , which leads us to a new system to solve. In order to set up this new system, we compute

$$w = Av, \quad z = H^{-1}w, \quad f = W^t z, \quad \beta = w^t z, \quad u = V^t w, \quad \gamma = v^t w, \quad h = Mu + \gamma u. \quad (3.19)$$

This is enough to append to  $W$  and  $Z$  their new columns, to make available the matrix

$$(W|w)^t(Z|z) = \begin{bmatrix} W^t Z & f \\ f^t & \beta \end{bmatrix} \quad (3.20)$$

from the left-hand side of (3.17) while re-using  $W^t Z$ , and to update  $M$  and  $M^2$  into  $M_+$  and  $M_+^2$  via

$$M_+ = \begin{bmatrix} M & u \\ u^t & \gamma \end{bmatrix} \quad \text{and} \quad M_+^2 = \begin{bmatrix} M^2 + uu^t & h \\ h^t & \gamma^2 + u^t u \end{bmatrix}. \quad (3.21)$$

The terms  $V^t b$  and  $Z^t b$  in the right-hand side of Eq. (3.17) are trivial to update. Moreover, since

$$M_+(V|v)^t b = \begin{bmatrix} M & u \\ u^t & \gamma \end{bmatrix} \begin{pmatrix} V^t b \\ v^t b \end{pmatrix} = \begin{pmatrix} MV^t b + uv^t b \\ u^t V^t b + \gamma v^t b \end{pmatrix}, \quad (3.22)$$

the already computed quantities  $MV^t b$  and  $V^t b$  can be re-used here too, leaving only small updates to perform. However, updating the term  $VM\hat{y}_1$  into  $(V|v)M_+\hat{y}_1$  in the right-hand side of Eq. (3.18) is a different matter, because unlike  $b$  the vector  $\hat{y}_1$  changes in every iteration. On the other hand, the vector  $H^{-1}b$  in the right-hand side of Eq. (3.18) needs to be computed only once.

In conclusion, in order to compute the approximation  $\hat{x}$  from the expanded system of size  $(k+1) \times (k+1)$ , re-using the quantities computed during the previous iteration as much as possible, we need:

- one MVP with  $A$ ,
- one system solve with  $H$ , or one MVP with  $H^{-1}$ ,
- $2k+3$  inner products between  $n$ -vectors,  $2k+2$  of which are in (3.19), and  $v^t b$ ,
- one system solve with  $M + M^2 - W^t Z$ ,

together with updates of  $n$ -vectors and some manipulations with  $k$ -vectors and matrices. We may also need to explicitly  $H$ -orthonormalise  $v$  to  $V$ .

**Remark 3.3.** At the start of the iteration, when  $V$  can be interpreted as a  $0 \times n$  matrix, the initial approximation resulting from (3.17)-(3.18) in a natural way is  $\hat{x} = H^{-1}b$ . Of course, the above considerations clearly show how the method can be turned into an algorithm.

**Remark 3.4.** Instead of solving the system (3.15) exactly as described above, one could also *approximate* its solution. (This is of course necessary in the SPAM method for eigenvalue problems.) One could choose to do so, but it makes sense to consider this option only if the *exact* solution results in a competitive method.

### 3.2.3. Relation with two-sided preconditioning

In the approach outlined above, we have assumed the availability of a preconditioner  $H$  of  $A$ . This preconditioner could, alternatively and at least in theory, also be applied to the original system first. In order for the preconditioned system to be symmetric and positive definite, *two-sided preconditioning* [7] is preferred. This means we decompose  $H$  as  $H = N^t N$  from some matrix  $N$ , and solve the preconditioned system

$$N^{-t} A N^{-1} z = N^{-t} b \quad \text{where} \quad x = N^{-1} z. \quad (3.23)$$

However, since  $H^{-1} = Y Y^t$  and thus  $H = Y^{-t} Y^{-1}$ , solving the system (3.6) — or equivalently, (3.8) — can be interpreted as a particular instance of using  $H$  as a two-sided preconditioner, in the sense that the decomposition  $H = N^t N$  is not unique. Indeed, if  $Q$  is orthogonal then  $H = (Q N)^t Q N$  is such a decomposition. Each  $Q$  leads to a different preconditioned matrix

$$Q N^{-t} A N^{-1} Q^t, \quad (3.24)$$

which are all spectrally equivalent and hence have the same condition number.

**Remark 3.5.** Two-sided preconditioning has two different effects, the best known being a change in the *condition number* of the resulting system independent of  $Q$ . The second effect is an orthogonal similarity transformation which in subspace iterative methods, one tries to manipulate such that the relevant information of the transformed matrix is in its leading  $k \times k$  principal submatrix.

Now, let  $H = N^t N$  be an arbitrary factorization of  $H$ . We can apply the approximation method of 3.2.1 to approximate the solution  $z$  of the preconditioned system in (3.23). Since the preconditioner has already been applied to the linear system, it makes sense to apply the method with the identity matrix  $I$  instead of  $H$ , and then compute the corresponding approximation of  $x$ . Replacing the matrix  $N^{-t} A N^{-1}$  in the block  $V_{\perp}^t N^{-t} A N^{-1} V_{\perp}$  by  $I$  also seems attractive — and indeed, after preconditioning the identity is arguably the best available approximation of the preconditioned system. The matrix  $Y = (V|V_{\perp})$  is then orthogonal with respect to the standard inner product, and the resulting system matrix is a true *subspace-projected approximate matrix* in the sense of Definition 2.1.

#### 4. Numerical Tests with Two Types of Subspace Expansion

We have introduced a SPAM-type selection method that, with the start vector  $x_0 = H^{-1}b$ , computes for each  $k \in \{1, \dots, n\}$  an approximation  $x_k$  associated with the space  $\mathcal{V}_k$ , although generally not from  $\mathcal{V}_k$ . We will refer to the new method as LinSPAM, and assume that  $A$  is a preconditioned matrix. Thus  $H = I$ , the basis in  $(V|V_{\perp})$  is orthogonal, and LinSPAM solves the subspace-projected approximate system

$$\begin{bmatrix} M & R^t \\ R & I \end{bmatrix} \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \quad (4.1)$$

where  $M = V^t A V$ ,  $R = V_{\perp}^t A V$ ,  $b_1 = V^t b$  and  $b_2 = V_{\perp}^t b$ , and sets  $\hat{x} = \hat{x}_1 + \hat{x}_2 = V \hat{y}_1 + V_{\perp} \hat{y}_2$ . In comparison, the Ritz-Galerkin and Minimal Residual approach respectively solve

$$M \tilde{y}_1 = b_1 \quad \text{and} \quad \begin{bmatrix} M \\ R \end{bmatrix} \begin{pmatrix} \tilde{y}_1 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \quad (4.2)$$

and set  $\tilde{x} = V \tilde{y}_1$  and  $\tilde{\tilde{x}} = V \tilde{\tilde{y}}_1$ , which clearly shows the similarities and distinctions between the methods. Although  $R$  is in principle not available, it is notable that the *block residual*

$$\hat{R} = V_{\perp} R = (I - V V^t) A V = A V - V M \quad (4.3)$$

is available, and hence also  $R^t R = \hat{R}^t \hat{R}$ . This is of interest, since (3.12)-(3.13) with the choice  $H = I$  becomes

$$(M - R^t R) \hat{y}_1 = b_1 - \hat{R}^t b, \quad (4.4)$$

$$\hat{x}_1 = b - V b_1 - \hat{R} \hat{y}_1. \quad (4.5)$$

We now offer choices for the sequence of subspaces  $\mathcal{V}_1 \subset \dots \subset \mathcal{V}_n$ , and compare LinSPAM with the Ritz-Galerkin and Minimal Residual approximations in some numerical tests.

#### 4.1. Non-structured subspaces with low approximation quality

For general subspaces, the matrix  $M$  in (3.17) is full, and the costs for LinSPAM are comparable to the costs of applying the Ritz-Galerkin or Minimal Residual selection. Further, the residual blocks  $R = V_{\perp}AV$  and  $R^t = VAV_{\perp}$  in (4.1) will be full. Since LinSPAM includes them in the approximation process, this may provide substantial differences with the Ritz-Galerkin approach (that uses neither  $R$  nor  $R^t$ ) and the Minimal Residual approach (that uses  $R$  only), as is evident in (4.2).

Motivated by the Kaczmarz method [5], we investigate the choice  $V = I$  such that  $W$  simply equals the first  $k$  columns of  $A$ . Especially for sparse matrices, this leads to an efficient selection — but as one can expect from the trivial expansion, it also leads to a slowly converging method. Indeed, in an iterative method the choice  $V = I$  does not make much sense, but does isolate the quality from the approximating space  $\mathcal{V}$  from the quality of the selection mechanism that we wish to study.

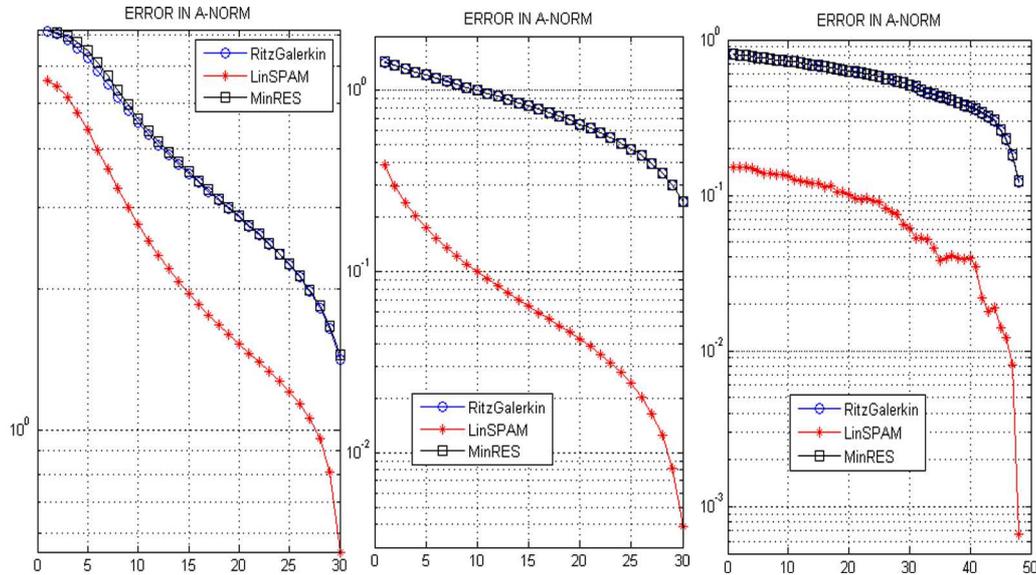


Figure 2: Ritz-Galerkin versus Minimal Residuals versus LinSPAM:  $A$ -norm of the error. Left: reaction-diffusion problem. Middle: banded matrix. Right: random symmetric matrix with eigenvalues  $51, \dots, 100$ . The Minimal Residual and Ritz-Galerkin graphs almost coincide.

We compared the LinSPAM approach with Ritz-Galerkin and Minimal Residuals for three simple linear systems. Their convergence histories are given in Fig. 2, where the errors in the approximations measured in the  $A$ -norm are displayed. As shown in the left picture, the linear system resulted from a finite difference discretization of a reaction-diffusion problem with 32 degrees of freedom. The system matrix from the middle picture is also of size  $32 \times 32$ , and is a banded matrix with 11 nonzero diagonals. In the right picture, we took for  $A$  a random symmetric  $50 \times 50$  matrix with eigenvalues  $51, \dots, 100$ . In all cases, the preconditioner  $A_0$  was the diagonal of  $A$ , and applied in advance.

**Remark 4.1.** The preconditioned matrices are all relatively well-conditioned, corresponding to our assumption that the identity is a reasonable approximation of the block  $V_{\perp}^t A V_{\perp}$ . Due to the well-conditioning, minimising the  $A$ -norm of the error or the  $A^2$ -norm of the error (the standard norm of the residual) does not make much difference, so the Ritz-Galerkin and Minimal Residuals approaches behave almost the same.

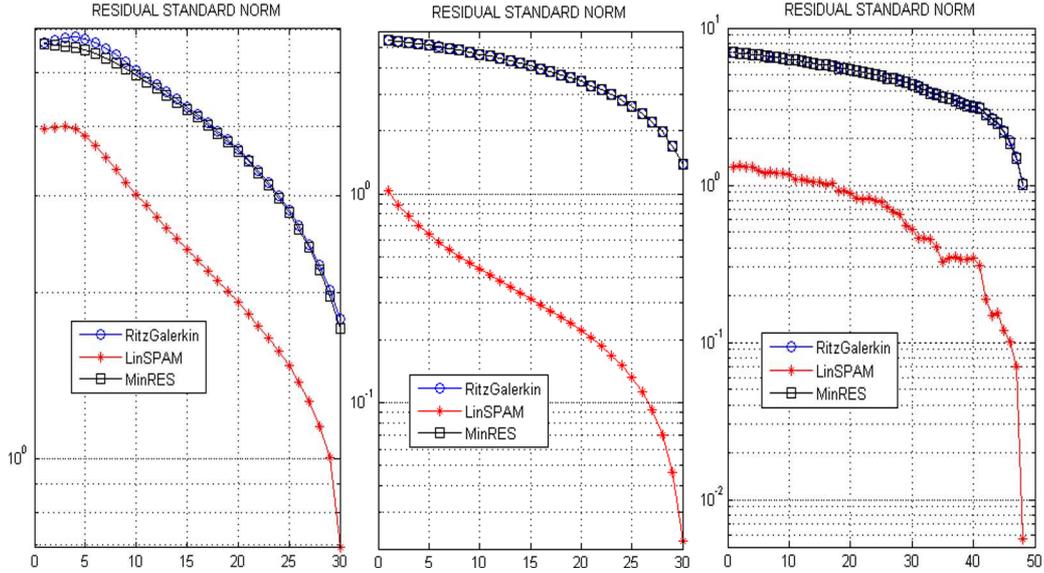


Figure 3: Ritz-Galerkin versus Minimal Residuals versus LinSPAM: residual norms. Left: reaction-diffusion problem. Middle: banded matrix. Right: random symmetric matrix with eigenvalues  $51, \dots, 100$ . The Minimal Residual and Ritz-Galerkin graphs almost coincide

In Fig. 3 we plot the standard norms of the residuals for the same numerical experiments. Due to the well-conditioning of the problems, the differences with the  $A$ -norm are again small, although here MinRES is always better than Ritz-Galerkin, whereas in the previous experiment as shown in Fig. 2 the converse was true.

With the choice  $V = I$ , it is notable that the LinSPAM approach corresponds to replacing the trailing principle submatrix of  $A$  of size  $(n - k) \times (n - k)$  by the identity, and solving the resulting system. As can be seen from the convergence graphs, this is always better than solving a system which involves only the  $k \times k$  leading principal submatrix  $M$ , as is the case in the Ritz-Galerkin approach. The Minimal Residual approach, which also takes into account the block below  $M$ , seems to do better if the standard norm of the residuals is of interest.

## 4.2. Krylov subspace expansion

At first sight, the LinSPAM approach seems more expensive than the preconditioned CG (PCG) method. However, the efficiency of the PCG derives largely from the use of Krylov subspaces, and LinSPAM can also benefit from Krylov subspaces. Indeed, let  $\mathcal{V}_k = \mathcal{K}^k(A, b)$

for the presumed preconditioned system  $Ax = b$ , such that  $V$  has orthonormal columns. Then  $M$  in (4.4) is tridiagonal, and  $R$  is zero apart from its top right entry. Consequently,  $\hat{R}$  is zero apart from its right column, and  $\hat{R}^t \hat{R} = R^t R$  is zero apart from its bottom right entry. Similarly, the vector  $\hat{R}^t b$  in the right hand side in (4.4) is just a multiple of  $e_k$ . Thus the LinSPAM system for  $\hat{x}_1$  is only a minor perturbation of the corresponding PCG system, and can be computed with the same efficiency. Moreover, the residual of the PCG approximation is orthogonal to  $\mathcal{V}$ , so we may expect the right-hand side of (4.5) (and thus also  $\hat{x}_2$ ) to be small. All in all, using the same Krylov subspaces as PCG we expect that the new approach will not differ very much from the PCG method, but at the same time any difference may perhaps favour our new approach. This is confirmed in numerical experiments.

We compared the LinSPAM selection using Krylov subspaces with the PCG method. Since the residuals of MinRES [12] — i.e. the Minimal Residual approach in the context of Krylov subspace methods — are a known function of the residuals of CG when using Krylov subspaces, we refrain from a comparison with MinRES. Thus in all experiments we simply took the diagonal of  $A$  as an approximation  $A_0$ .

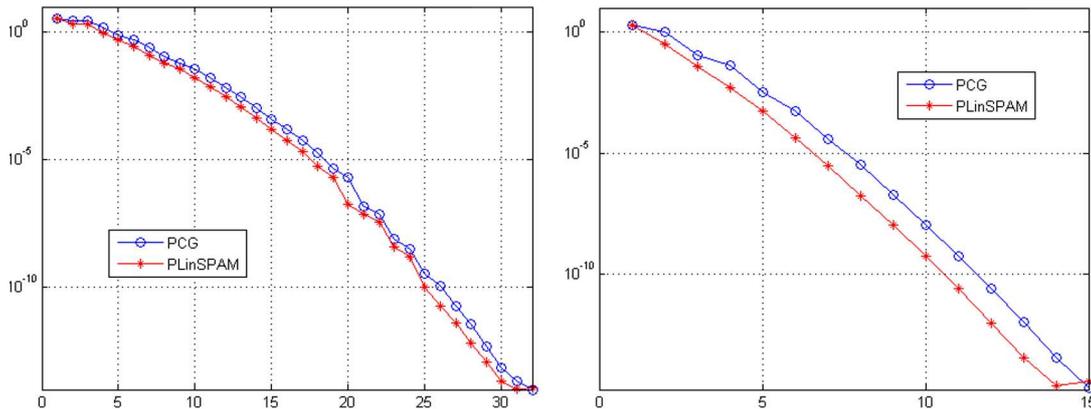


Figure 4: (Preconditioned) Conjugate Gradients versus (Preconditioned) LinSPAM:  $A$ -norm of the error. Left: reaction-diffusion problem. Right: banded matrix.

The first linear system is again the discretised reaction-diffusion problem. The system matrix from the second test is the banded matrix with 11 nonzero diagonals. The convergence histories are given in Fig. 4, where the errors in the approximations measured in the  $A$ -norm are displayed. Clearly, LinSPAM provides slightly better approximations than the PCG method.

In the next experiment depicted in Fig. 5, we took the  $729 \times 729$  matrix *nos7* from *Matrix Market* resulting from the finite difference approximation of a diffusion equation with variable diffusivity in a 3D unit cube with Dirichlet boundary conditions. On the left, we again plot the  $A$ -norm of the respective errors against the iteration number, and in the right picture the standard norm of the residuals corresponding to those approximations is plotted. As expected, the LinSPAM behaviour is similar to that of the PCG method.

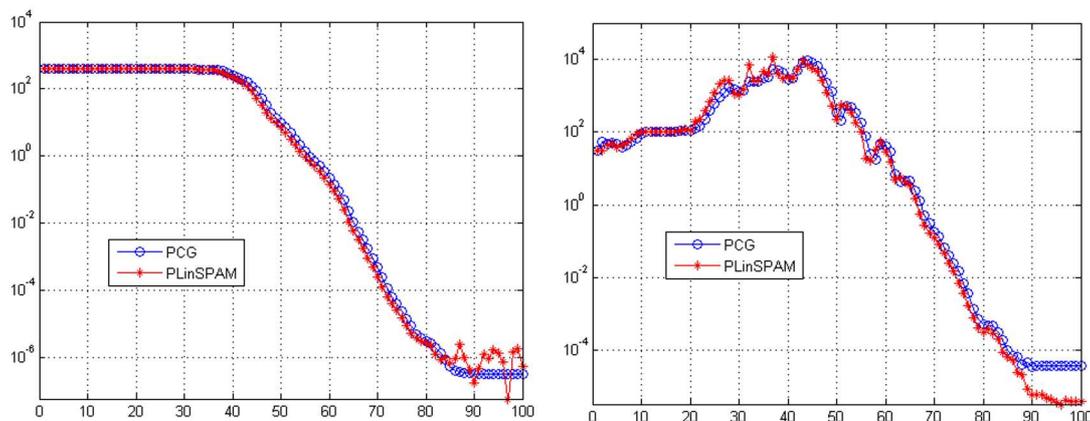


Figure 5: Matrix Market test matrix *nos7* of size  $729 \times 729$ . Left:  $A$ -norm of the error. Right: standard norm of the residual.

We conclude that applying LinSPAM using Krylov subspaces seems to have some advantage over the PCG method at the cost of a small number of additional computations, but without involving another MVP with the system matrix  $A$ . It seems that the gain is about one iteration at most, which does makes sense in the following way. In the PCG method, the off-diagonal elements at positions  $(k, k + 1)$  and  $(k + 1, k)$  of the  $(k + 1) \times (k + 1)$  tridiagonal matrix at the next iteration are already available at iteration  $k$ . However, they are not used in the PCG approximation, whereas in LinSPAM they *are* used together with an approximation of the entry at position  $(k + 1, k + 1)$ . This may result in a fairly good approximation of the PCG iterate  $x_{k+1}$ .

### 4.3. Immediate restart: a one-step LinSPAM method

In general, subspace methods have many favourable properties that result from the expansion of the subspaces, such as monotonicity and finite termination. Each of the methods has its own *restarted version* — i.e. after a certain number of iterations, the current residual  $r = b - A\hat{x}$  is taken as the right-hand side of a new linear system  $Au = r$ , and the method is re-applied to this system. The approximation  $\hat{u}$  of  $u$  in addition to the approximation  $\hat{x}$  of  $x$  computed *before* the restart leads to a new approximation  $\hat{x} + \hat{u}$  of the system  $Ax = b$ .

LinSPAM can also be restarted. Since the above experiments suggest that the gain with LinSPAM in comparison to Ritz-Galerkin and Minimal Residual selection is already present at the first iteration, and does not increase very much in the course of the process, we investigate the *one step version* of LinSPAM, and call it LinSPAM(1) for short. In each step, LinSPAM(1) solves a system with matrix

$$\begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & 1 & & \\ & & & I \end{bmatrix}, \quad (4.6)$$

where  $\alpha_1$  and  $\beta_1$  are the exact numbers that would also appear in the tridiagonalisation of the matrix  $A$  as performed within the PCG method — cf. also (4.1).

**Remark 4.2.** In practice, Krylov subspace methods such as PCG and MinRES for linear systems are never restarted, even though they are highly numerically unstable. Loss of orthogonality or conjugacy does not affect the convergence of these methods, but it may delay the convergence. The comparison below is thus merely an academic exercise to get some insight, to try to determine the optimal one-step method.

We repeated the experiments of Subsection 4.1 to compare LinSPAM(1) with MinRES(1), the version of MinRES that restarts after each iteration — cf. Fig. 6. Although

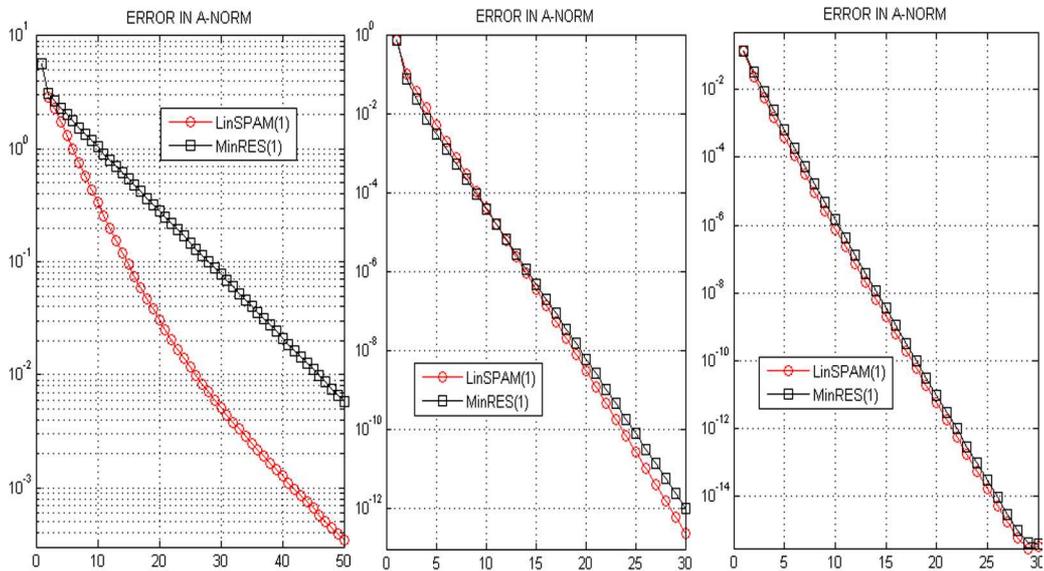


Figure 6: LinSPAM(1) versus MinRES(1):  $A$ -norm of the error. Left: reaction-diffusion problem. Middle: banded matrix. Right: random symmetric matrix with eigenvalues  $51, \dots, 100$ .

LinSPAM(1) was better than MinRES(1) in the first experiment, there was no substantial improvement in the second and third experiments. This also seemed to be so in other numerical experiments, where it was found that MinRES(1) stagnates and LinSPAM(1) explodes (because it has no minimization property), or both methods converge in a similar manner.

#### 4.4. A note on the non-symmetric case

If  $A$  is not symmetric, it is generally difficult to define symmetric positive definite approximations  $H$  of  $A$ . On the other hand, if we again assume that  $A$  has been preconditioned as far as possible, then considering the identity  $I$  to be such an approximation is not entirely without merit. However, the drawback in comparison with Ritz-Galerkin and

Minimal Residual selection is that (4.4) changes into

$$(M - G^t R)\hat{y}_1 = b_1 - \hat{R}^t b, \quad \text{where} \quad G^t R = \hat{G}^t \hat{R} \quad (4.7)$$

with  $\hat{R}$  as in (4.4) and  $\hat{G} = V_\perp G = A^t V - V M^t$  the block *left residual* with respect to  $V$ , where  $G = V^t A V_\perp$  is as in (2.3). This block is not automatically available, and to compute it  $k$  MVPs with  $A^t$  are needed. If the spaces  $\mathcal{V}_k$  are a sequence of Krylov subspaces, the situation becomes much better. Then  $R$  is still zero apart from its top right entry, because  $(V|V_\perp)^t A (V|V_\perp)$  is an upper Hessenberg matrix, so to compute the product  $G^t R$  only the first column of  $G^t$  is needed — hence instead of  $V^t A$ , only  $V^t a_1$  is needed, where  $a_1$  is the first column of  $A$ . On the other hand, the first column of  $G^t$  contains the top  $k$  of the  $k + 1$  coefficients to be computed in the next step of the process of transforming  $A$  to upper Hessenberg form. As such and contrary to the symmetric case, computing them in the  $k$ -th iteration of LinSPAM is unfair, if one regards LinSPAM as an enhancement of GMRES [16] or FOM [15].

**Remark 4.3.** Computational experiments do show the same marginal gain of LinSPAM in comparison with GMRES as witnessed for LinSPAM in comparison with CG, but for the reason just mentioned we refrain from presenting the results here explicitly.

## 5. Concluding Remarks

LinSPAM selection is an attempt to improve upon both Ritz-Galerkin and Minimal Residual selection, and seems especially attractive for (4.1) and (4.2). Moreover, LinSPAM selection has a clear connection with the Subspace Projected Approximate Matrix method for eigenvalue computation [17]. Experiments with rather randomly selected subspaces show the potential of the LinSPAM selection. Using Krylov subspaces, the difference with Ritz-Galerkin and Minimal Residuals is relatively small, but additional costs are very low. Since the gain seems to be immediate, the one-step LinSPAM version LinSPAM(1) may be competitive with other one-step methods. The LinSPAM approach can also be applied to non-symmetric systems if they have been preconditioned and the approximation of  $V_\perp^t A V_\perp$  by the identity makes sense, and in the Krylov subspace context it uses information that GMRES would compute one iteration later.

## Acknowledgments

The authors gratefully acknowledge the valuable remarks of two referees that led to this revised version of our original paper.

## References

- [1] C. Beattie (1998). Harmonic Ritz and Lehmann bounds. *Electron. Trans. Numer. Anal.*, 7:18–39.

- [2] J.H. Brandts (2003). The Riccati Method for eigenvalues and invariant subspaces of matrices with inexpensive action. *Linear Algebra Appl.*, 358:335–365.
- [3] J.H. Brandts and R.R. da Silva (2011). On the subspace projected approximate matrix method. arXiv:1103.1779v1 [math.NA] (23 pages)
- [4] W. Chen and B. Poirier (2006). Parallel implementation of efficient preconditioned linear solver for grid-based applications in chemical physics. II *J. Comput. Phys.*, 219:198–209.
- [5] S. Kaczmarz (1993). Approximate solution of systems of linear equations. Translated from the German. *Internat. J. Control* 57, 6:1269–1271.
- [6] C. Lanczos (1950). An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Research Nat. Bur. Standards*, 45(4):255–282.
- [7] G.H. Golub and C.F. van Loan (1996). *Matrix Computations* (third edition). The John Hopkins University Press, Baltimore and London.
- [8] W. Györfy, P. Seidler and O. Christiansen (2009). Solving the eigenvalue equations of correlated vibrational structure methods: Preconditioning and targeting strategies. *J. Chem. Phys.*, 131:024108 (16 pages)
- [9] M.R. Hestenes and E. Stiefel (1954). Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand.*, 49:409–436.
- [10] Z. Jia and G.W. Stewart (2001). An analysis of the Rayleigh-Ritz method for approximating eigenspaces. *Math. Comp.*, 70(234):637–647.
- [11] D. M. Medvedev, S. K. Gray, A. F. Wagner, M. Minkoff and R. Shepard (2005). Advanced software for the calculation of thermochemistry, kinetics, and dynamics. *J. Physics: Conference Series*, 16:247.
- [12] C.C. Paige and M.S. Saunders (1975). Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 2:617–629.
- [13] B.N. Parlett (1998). *The Symmetric Eigenvalue Problem*. SIAM Classics in Applied Mathematics 20, Philadelphia.
- [14] F. Ribeiro, C. Lung and C. Leforestier (2005). A Jacobi-Wilson description coupled to a block-Davidson algorithm: An efficient scheme to calculate highly excited vibrational levels. *J. Chem. Phys.*, 123:054106 (10 pages)
- [15] Y. Saad (1981). Krylov subspace method for solving large unsymmetric linear systems. *Math. Comp.*, 37:105–126.
- [16] Y. Saad and M.H. Schultz (1986). GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869.
- [17] R. Shepard, A.F. Wagner, J.L. Tilson and M. Minkoff (2001). The subspace-projected approximate matrix (SPAM) modification of the Davidson method. *J. Comput. Phys.*, 172(2):472–514.
- [18] G.L.G. Sleijpen, J. van den Eshof and P. Smit (2003). Optimal a priori error bounds for the Rayleigh-Ritz method. *Math. Comp.*, 72(242):677–684.
- [19] G.L.G. Sleijpen and H.A. van der Vorst (1996). Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Applic.*, 17:401–425.
- [20] Y. Zhou, R. Shepard and M. Minkoff (2005). Computing eigenvalue bounds for iterative subspace matrix methods. *Comput. Phys. Commun.*, 167:90–102.