

# Rational interpolation: Modified rational Arnoldi algorithm and Arnoldi-like equations

Michalis Frangos and Imad M. Jaimoukha

**Abstract**—Krylov projection methods are used for model reduction of large scale systems. An algorithm which belongs to the class of Krylov subspace methods is the Arnoldi algorithm. The standard version of this algorithm tends to create reduced order models that poorly approximate low frequency dynamics. The rational Arnoldi algorithm produces reduced models that approximate dynamics at different interpolation points. This paper tackles the issue of developing a computationally efficient model reduction procedure based on a modified rational Arnoldi algorithm. Moment matching properties are established and a breakdown analysis for the algorithm is provided. A set of Arnoldi-like equations for the algorithm is also derived.

## I. INTRODUCTION

Consider a linear time-invariant single-input single-output system described by the state-space equations

$$E\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) \quad (1)$$

where  $x(t) \in \mathbb{C}^n$  denotes the state vector and  $u(t)$  and  $y(t)$  the scalar input and output signals, respectively. The system matrices  $A$ ,  $E \in \mathbb{C}^{n \times n}$  are assumed to be large and sparse, and  $B, C' \in \mathbb{C}^n$ . These assumptions are met by large scale models in many applications. The transfer function for the system in (1) is denoted as

$$G(s) = C(sE - A)^{-1}B \stackrel{s}{=} \left[ \begin{array}{c|c} E & A \\ \hline * & C \end{array} \middle| \begin{array}{c} B \\ 0 \end{array} \right].$$

To simplify subsequent analysis and design based on the large  $n$ th order model in (1), the model reduction problem seeks an approximate  $m$ th order model of the form

$$E_m \dot{x}_m(t) = A_m x_m(t) + B_m u(t), \quad y_m(t) = C_m x_m(t)$$

where  $x_m(t) \in \mathbb{C}^m$ ,  $E_m, A_m \in \mathbb{C}^{m \times m}$ ,  $B_m, C_m' \in \mathbb{C}^m$  and  $m < n$ . The associated lower order transfer function is denoted by

$$G_m(s) = C_m(sE_m - A_m)^{-1}B_m \stackrel{s}{=} \left[ \begin{array}{c|c} E_m & A_m \\ \hline * & C_m \end{array} \middle| \begin{array}{c} B_m \\ 0 \end{array} \right].$$

Krylov projection methods, and in particular the Arnoldi and Lanczos algorithms [2], [4], [16], [17], [26], exploit the sparsity of the large scale model and have been extensively used for model reduction of large scale systems; see [1], [3], [5], [26] and the references therein. In these approaches,  $G_m(s)$  is computed such that it matches the moments of  $G(s)$ , that is the value of  $G(s)$  and its derivatives, at certain

interpolation points. In the standard approaches, the moments are matched around a single interpolation point usually at  $\infty$ , (e.g. [1], [8], [11]). Rational Arnoldi and Lanczos algorithms [9], [12], [22]–[24] have been developed which produce reduced models that match the moments of  $G(s)$  at different interpolation points. In [6] we have shown how the approximation offered by the rational Arnoldi and Lanczos techniques can be greatly improved by a proper selection of the interpolation points. Briefly the issues discussed in [6] are as follow.

- Simple Arnoldi and Lanczos-like equations as in the standard Arnoldi and Lanczos form have been derived in the rational case.
- Error analysis for the rational case has been derived comparable to the standard algorithms. Residual error expressions were derived which are important for a better choice of the interpolation points.

Another existing problem for rational Krylov methods is summarized below

- When calculating a new basis vector of the Krylov subspaces, many current algorithms (see for example [9], [14]) require knowledge of whether the current interpolation frequency is equal to any of the previous ones. This necessitates either a priori knowledge of all the interpolation points, which is not feasible in an adaptive update of these interpolation points, or storage of the previous bases and interpolation points, which is awkward and could be expensive. In [22]–[24], a modified Arnoldi algorithm in the context of eigenvalue analysis was suggested, however, no moment matching properties were established and no breakdown analysis provided. The authors in [11], [19], [20] have applied this alternative implementation for model reduction and they provide moment matching properties. In this paper an extended version of the modified Arnoldi algorithm and a detailed analysis are presented in detail.

*Outline and Contribution:* Section II gives a review of approximation techniques by moment matching. The standard Arnoldi and the rational Arnoldi algorithms for moment matching are also described. The contribution of this paper is in the development of a general modified algorithm for rational Arnoldi, first suggested in [23], and the extension of the existing theory for the moment matching properties of this general algorithm. In our previous work [6] we have emphasized the importance on deriving simple Arnoldi-like equations for the rational Krylov algorithms. We also establish the Arnoldi-like equations that describe the general

This work was not supported by any organization  
M. Frangos and I. M. Jaimoukha are with the Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Road, London, SW7 2AZ, UK  
michalis.frangos99@imperial.ac.uk,  
i.jaimoukha@imperial.ac.uk

modified algorithm developed and in addition we present a breakdown analysis. These subjects are described in detail in section III. Finally section IV gives our conclusions.

## II. KRYLOV BASED METHODS FOR MODEL REDUCTION

### A. Moment matching problem

To simplify the presentation of our results, we only consider the case when  $E = I_n$ , where  $I_n$  is the identity matrix of size  $n$  and we will write

$$G(s) = C(sI_n - A)^{-1}B \doteq \left[ \begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right].$$

The system in (1) can be expanded by Taylor series around an interpolation point  $s_0 \in \mathbb{C}$  as

$$G(s) = g_0 + g_1(s-s_0) + g_2(s-s_0)^2 + \dots$$

where the Taylor coefficients  $g_i$  are known as the moments of the system around  $s_0$  and are related to the transfer function of the system and its derivatives evaluated at  $s_0$ . The approximation problem by moment matching is to find a lower order system  $G_m(s)$  with transfer function expanded as

$$G_m(s) = \hat{g}_0 + \hat{g}_1(s-s_0) + \hat{g}_2(s-s_0)^2 + \dots$$

such that  $g_i = \hat{g}_i$ , for  $i = 0, 1, \dots, m$ .

In the case where  $s_0 = \infty$  the moments are called Markov parameters and are given by  $g_i = CA^iB$ . The moments around a finite interpolation point  $s_0 \in \mathbb{C}$  are known as shifted moments and they are defined as  $g_i = C(s_0I_n - A)^{-i}B$ .

A more general definition of approximation by moment matching is related to rational interpolation. By rational interpolation we mean that the reduced order system matches the moments of the original system at multiple interpolation points.

Let  $V_m, W_m \in \mathbb{C}^{n \times m}$ . By projecting the states of the high order system with the projector

$$P_m = V_m(W_m'V_m)^{-1}W_m', \quad (2)$$

assuming that  $W_m'V_m$  is nonsingular, a reduced order model is obtained as:

$$G_m(s) \doteq \left[ \begin{array}{c|cc} E_m & A_m & B_m \\ \hline * & C_m & 0 \end{array} \right] := \left[ \begin{array}{c|cc} W_m'V_m & W_m'AV_m & W_m'B \\ \hline * & CV_m & 0 \end{array} \right]. \quad (3)$$

A careful selection of  $V_m$  and  $W_m$  as the bases of certain Krylov subspaces results in moment matching (e.g. [1], [7], [8], [11], [20]). For  $A \in \mathbb{C}^{n \times n}$ ,  $B \in \mathbb{C}^n$ ,  $s \in \mathbb{C}$  and integer  $m > 0$  a Krylov subspace  $\mathcal{K}_m(A, B, s)$  is defined as

$$\mathcal{K}_m(A, B, s) := \text{colsp} \left[ B, (sI_n - A)^{-1}B, \dots, (sI_n - A)^{-(m-1)}B \right], \\ \text{if } s \neq \infty \text{ and} \\ \mathcal{K}_m(A, B, s) := \text{colsp} [B, AB, \dots, A^{m-1}B], \quad \text{if } s = \infty$$

where  $\text{colsp}$  denotes column span. If  $m = 0$ ,  $\mathcal{K}_m(A, B, s)$  is defined to be the empty set.

Due to space limitation we present the results in the case where  $W_m = V_m$ . The results for the two-sided Arnoldi case can be derived in a similar way and are therefore omitted.

### Algorithm 1 Basic Arnoldi algorithm

- 
- 1: Inputs: system data  $A, B, C$ , reduced order  $m$
  - 2: and tolerance  $\varepsilon > 0$
  - 3: Initialise:  $V_j = [ ]$ ,  $\hat{v}_1 = B$
  - 4: **if**  $\{\|\hat{v}_1\| < \varepsilon\}$ , **Stop, end**
  - 5:  $v_1 = \hat{v}_1/\|\hat{v}_1\|$
  - 6: **for**  $j = 1 \rightarrow m$
  - 7:  $V_{j+1} = [V_j \ v_j]$
  - 8:  $\hat{v}_{j+1} = AV_j - V_jV_j'AV_j$ ,
  - 9: **if**  $\{\|\hat{v}_{j+1}\| < \varepsilon\}$ , **Stop, end**
  - 10:  $v_{j+1} = \hat{v}_{j+1}/\|\hat{v}_{j+1}\|$ ,
  - 11: **end**
  - 12: Outputs:  $V_{m+1} = \{v_1, \dots, v_{m+1}\}$
- 

### B. Standard Arnoldi method

An iterative method for model reduction based on Krylov projections is the Arnoldi process [2]. The basic (one-sided) Arnoldi process, given in Algorithm 1, constructs the base  $V_m = [v_1, \dots, v_m] \in \mathbb{C}^{n \times m}$  for the Krylov subspace  $\mathcal{K}_m(A, B, \infty)$  such that it is unitary, i.e.,  $V_m'V_m = I_m$ . We assume that no breakdown (or near breakdown) occurs in Algorithm 1 so that  $\|\hat{v}_j\| > \varepsilon$  where  $\varepsilon$  is a tolerance level, for  $j = 1, \dots, m$ . See [25] for more details on the breakdown of the Arnoldi algorithm.

The following equations,

$$AV_m = V_mA_m + v_{m+1}Cv_m, \quad B = V_mB_m$$

referred to as the Arnoldi equations hold [15], where

$$\left[ \begin{array}{cc} A_m & B_m \\ Cv_m & * \end{array} \right] := \left[ \begin{array}{cc} V_m'AV_m & V_m'B \\ v_{m+1}'AV_m & * \end{array} \right], \quad (4)$$

and  $*$  denotes a term of no interest.

The base  $V_m$  is constructed such that  $\mathcal{K}_m(A, B, \infty) \subseteq \text{colsp}(V_m)$  and so the reduced order system  $G_m(s)$ , defined in (3), matches the first  $m$  markov parameters of  $G(s)$ .

### C. The rational Arnoldi method

The rational Arnoldi procedure [18], [22]–[24] is an algorithm for constructing unitary bases of the union of Krylov subspaces. Let  $V_m \in \mathbb{C}^{n \times m}$  be the base of such subspace and let  $P_m$  be a projector defined as  $P_m = V_mV_m'$ . Applying this projector on the system in (1) a reduced order system is obtained with a transfer function as in (3), with  $W_m = V_m$ . The next result shows that a proper selection of Krylov subspaces will result in reduced order systems that matches the moments of the system at given interpolation points.

**THEOREM 2.1:** Let  $\mathbf{S} = \{s_1, s_2, \dots, s_K\} \subset \mathbb{C}$  be a set of distinct interpolation points, with multiplicities  $m_{s_1}, m_{s_2}, \dots, m_{s_K}$ . Suppose that  $V_m \in \mathbb{C}^{n \times m}$  satisfies

$$\text{colsp}(V_m) \supseteq \mathcal{K}_{m_{s_1}}(A, B, s_1) \cup \dots \cup \mathcal{K}_{m_{s_K}}(A, B, s_K)$$

where  $\sum_{k=1}^K m_{s_k} = m$ . Then assuming that  $(sI_n - A)^{-1}$  exists for all  $s \in \mathbf{S}$ , then  $G_m(s)$  matches the first  $m_{s_i}$  moments of  $G(s)$  at  $s_k$ .

*Proof:* The proof can be found in [11] ■

**Algorithm 2** Rational Arnoldi algorithm

---

```

1: Inputs:  $\mathbf{S} = \{s_1, \dots, s_K\}$ ,  $A, B, C, \varepsilon, m_{s_k}$ 
2: Initialize:  $V_j = [\ ]$ ,  $j = 0$ 
3: for  $k = 1 \rightarrow K$ 
4:   if  $\{s_k = \infty\}$ ,  $\hat{v}_{j+1} = B$  else
5:      $\hat{v}_{j+1} = (s_k I_n - A)^{-1} B$ 
6:   end
7:   if  $\|\hat{v}_{j+1}\| < \varepsilon$ , Stop end
8:    $v_{j+1} = \frac{\hat{v}_{j+1}}{\|\hat{v}_{j+1}\|}$ ,  $V_{j+1} = [V_j \ v_{j+1}]$ 
9:    $j = j + 1$ 
10:  for  $i = 1 \rightarrow m_{s_k} - 1$ 
11:    if  $\{s_k = \infty\}$ ,  $\hat{v}_{j+1} = Av_j$  else
12:       $\hat{v}_{j+1} = (s_k I_n - A)^{-1} v_j$ 
13:    end
14:     $\hat{v}_{j+1} = \hat{v}_{j+1} - V_j V_j' \hat{v}_{j+1}$ 
15:    if  $\|\hat{v}_{j+1}\| < \varepsilon$ , Stop end
16:     $v_{j+1} = \hat{v}_{j+1} / \|\hat{v}_{j+1}\|$ ,  $V_{j+1} = [V_j \ v_{j+1}]$ 
17:     $j = j + 1$ 
18:  end
19: end
20: Outputs :  $V_m = \{v_1, \dots, v_m\}$ 

```

---

One simple version of the rational Arnoldi algorithm is given in Algorithm 2.

*D. Arnoldi-like equations in the rational case*

A set of equations known as the Arnoldi or Lanczos equations are satisfied on every iteration of the standard versions of Arnoldi and Lanczos algorithms. These equations are useful for error analysis, for deriving residual error expressions and stopping criteria for the algorithms, for perturbation analysis and restarts [6], [13], [15], [21], [27]. The authors in [9]–[11], [14], [24] derive equations that describe the rational algorithms; however these are not in the standard Arnoldi and Lanczos form. In [6] we have shown how to derive with minimum additional effort, Arnoldi and Lanczos-like equations in the standard form. In the case of Arnoldi method this is described in Lemma 2.1.

LEMMA 2.1: Let all variables be as defined in Theorem 2.1 and assume that  $V_m' V_m = I_m$ . Let  $m_\infty$  be the multiplicity of  $\infty$  in  $\mathbf{S}$  ( $m_\infty = 0$  if  $\infty \notin \mathbf{S}$ ). Suppose  $v_{m+1} \in \mathbb{C}^n$  is defined such that, with  $V_{m+1} := [V_m \ v_{m+1}]$ ,

$$\text{colsp}([V_m \ A^{m_\infty} B]) = \text{colsp}(V_{m+1})$$

$$V_{m+1}' V_{m+1} = I_{m+1}.$$

Define

$$\begin{bmatrix} A_m & B_m \\ C_{V_m} & b_m \end{bmatrix} := \begin{bmatrix} V_m' A V_m & V_m' B \\ v_{m+1}' A V_m & v_{m+1}' B \end{bmatrix}.$$

Then

$$A V_m = V_m A_m + v_{m+1} C_{V_m} \quad (5)$$

$$B = V_m B_m + v_{m+1} b_m \quad (6)$$

Furthermore,  $b_m = 0$  if  $\infty \in \mathbf{S}$ .

*Proof:* The proof can be found in [6] ■

In the following section we present a modified rational Arnoldi algorithm and its analysis. This algorithm allows for an efficient implementation of Lemma 2.1 and the derivation of the Arnoldi equations.

## III. MODIFIED RATIONAL ARNOLDI ALGORITHM

*A. Description of the Algorithm*

The moment matching problem in model reduction reduces in computing the bases of Krylov subspaces as defined in Theorem 2.1. The work in [9], [14] developed iterative algorithms for computing the base of the above Krylov subspace. However these algorithms require storage of previous residuals which is expensive. In this paper we present a practical implementation of the rational Arnoldi algorithm based on the algorithm presented in [22]–[24]. At each iteration, moments are matched without keeping track of the multiplicities of the interpolation points already computed and the algorithm does not require storage of extra vectors. The algorithms presented in [22]–[24] are used for eigenvalue computation and model reduction but no moment matching properties were established. The authors in [11], [19], [20] have applied a simplified version of this alternative implementation for model reduction and they provide moment matching properties.

In [11], [19], [20] on the  $j$ th iteration a new column is obtained in the direction of  $(s_k I_n - A)^{-1} v_j$ . The algorithm developed in [22]–[24] is more general in the sense that on the  $j$ th iteration the new column is obtained in the direction of  $(s_k I_n - A)^{-1} \tilde{v}_j$ , where  $\tilde{v}_j$  is not just the last column computed in  $V_j$  but it can be a linear combination of any columns in  $V_j$ . Also in both implementations  $s_k$  is considered to be a new finite interpolation point not at infinity.

In this section we develop an algorithm for model reduction in the more general case of [22]–[24] with the incremental improvement of including infinity as a possible interpolation point selection. The proof that moment matching is actually achieved with this method is given and the modified algorithm for the Arnoldi case is presented in Algorithm 3. The following result is key for subsequent analysis of Algorithm 3. The main difficulty in the proofs is including infinity as a possible interpolation point.

**Algorithm 3** Modified Rational Arnoldi

---

```

1: Inputs:  $A, B, C, \mathbf{S}, S_1 \dots S_j$  and  $e_j$  defined in Lemma 3.1;  $\varepsilon > 0$ . Remark:  $\mathbf{S}$  can contain a number of equal  $s_i$  in any order.
2: Initialise:  $V_j = [\ ]$ ,  $\hat{v}_1 = S_1 B$ 
3: if  $\|\hat{v}_1\| < \varepsilon$ , Stop, end
4:  $v_1 = \hat{v}_1 / \|\hat{v}_1\|$ ,  $V_1 = v_1$ 
5: for  $\{j = 1 \rightarrow m - 1\}$ 
6:    $\hat{v}_{j+1} = S_{j+1} V_j e_j$ ,  $\hat{v}_{j+1} = \hat{v}_{j+1} - V_j V_j' \hat{v}_{j+1}$ 
7:   if  $\{\|\hat{v}_{j+1}\| < \varepsilon\}$ , Stop, end
8:    $v_{j+1} = \hat{v}_{j+1} / \|\hat{v}_{j+1}\|$ ,  $V_{j+1} = [V_j \ v_{j+1}]$ 
9: end
10: Output :  $V_m$ 

```

---

LEMMA 3.1: Let  $A \in \mathbb{C}^{n \times n}$  and  $B \in \mathbb{C}^{n \times 1}$  and let  $m < n$ . Let  $\mathbf{S} = \{s_1, \dots, s_m\} \subset \mathbb{C}$  be given and assume that  $s_i$  is not an eigenvalue of  $A$  for all  $s_i \in \mathbf{S}$ . Define

$$S_1 = \begin{cases} I_n, & \text{if } s_1 = \infty \\ (s_1 I_n - A)^{-1}, & \text{if } s_1 \neq \infty \end{cases}$$

and for  $j = 2, \dots, m$

$$S_j = \begin{cases} A, & \text{if } s_j = \infty \\ (s_j I_n - A)^{-1}, & \text{if } s_j \neq \infty \end{cases},$$

For  $j = 1, \dots, m-1$  let  $0 \neq e_j \in \mathbb{C}^{j \times 1}$  be given. Let  $v_1, \dots, v_m \in \mathbb{C}^{n \times 1}$  be as generated by Algorithm 3 and let  $V_j = [v_1 \ \dots \ v_j]$  so that

$$\begin{bmatrix} S_1 B & S_2 V_1 e_1 & \dots & S_m V_{m-1} e_{m-1} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ 0 & r_{22} & \dots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & r_{mm} \end{bmatrix} \begin{bmatrix} v_1 & v_2 & \dots & v_m \end{bmatrix} \quad (7)$$

with  $V_m$  unitary and  $r_{ji} = 0$  for all  $j > i$ . Assume that  $r_{jj} \neq 0$  for  $j = 1, \dots, m$ . Then, for  $j = 1, \dots, m-1$ ,

$$\text{colsp}(S_{j+1} V_j) \subset \text{colsp}(V_{j+1})$$

*Proof:* Since  $e_1 \neq 0$  and

$$S_2 V_1 = [v_1 \ v_2] \begin{bmatrix} r_{12} \\ r_{22} \end{bmatrix} e_1^{-1}$$

by construction, the result is true for  $j = 1$ . Assume that the result is true for  $j-1$ , with  $j > 1$ , so that

$$S_j V_{j-1} = V_j R_{j-1} \quad (8)$$

for some  $R_{j-1} \in \mathbb{C}^{j \times (j-1)}$ . Define

$$E_{j-1} = \begin{bmatrix} I_{j-1} \\ 0_{1,j-1} \end{bmatrix}, \quad r_j = \begin{bmatrix} r_{1,j+1} \\ \vdots \\ r_{j,j+1} \end{bmatrix}.$$

The proof is divided into four parts, depending on the values of  $s_j$  and  $s_{j+1}$ .

- 1)  $s_j \neq \infty, s_{j+1} \neq \infty$ : Here,  $S_j = S_{j+1} + (s_j - s_{j+1})S_{j+1}S_j$ . Substituting in (8),

$$\begin{aligned} S_{j+1} V_{j-1} + (s_j - s_{j+1}) S_{j+1} S_j V_{j-1} &= V_j R_{j-1} \\ \Rightarrow S_{j+1} V_{j-1} + (s_j - s_{j+1}) S_{j+1} V_j R_{j-1} &= V_j R_{j-1} \\ \Rightarrow S_{j+1} V_j (E_{j-1} + (s_j - s_{j+1}) R_{j-1}) &= V_j R_{j-1}. \end{aligned}$$

Combining this with

$$S_{j+1} V_j e_j = V_{j+1} \begin{bmatrix} r_j \\ r_{j+1,j+1} \end{bmatrix} \quad (9)$$

from (7), we get

$$S_{j+1} V_j \begin{bmatrix} L = [L_1 \ e_j] \\ (E_{j-1} + (s_j - s_{j+1}) R_{j-1}) \ e_j \end{bmatrix} = V_{j+1} \begin{bmatrix} R \\ R_{j-1} \ r_j \\ 0 \ r_{j+1,j+1} \end{bmatrix} \quad (10)$$

Next, we show that  $L$  is nonsingular. Suppose on the contrary that

$$\begin{bmatrix} L_1 & e_j \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0, \quad \begin{bmatrix} x \\ y \end{bmatrix} \neq 0. \quad (11)$$

Post-multiplying (10) by  $\begin{bmatrix} x' & y' \end{bmatrix}'$  shows that  $y = 0$  since  $r_{j+1,j+1} \neq 0$ . Thus  $R_{j-1}x = 0$ , since  $V_{j+1}$  is unitary, and  $L_1x = 0$ . This shows that  $x = 0$ . This contradiction shows that  $L$  is nonsingular.

- 2)  $s_j = \infty, s_{j+1} \neq \infty$ : It can be verified that, in this case,  $S_{j+1}S_j = s_{j+1}S_{j+1} - I_n$ . Pre-multiplying (8) by  $S_{j+1}$ , substituting and using (9),

$$S_{j+1} V_j \begin{bmatrix} L \\ (s_{j+1} E_{j-1} - R_{j-1}) \ e_j \end{bmatrix} = V_{j+1} \begin{bmatrix} R \\ E_{j-1} \ r_j \\ 0 \ r_{j+1,j+1} \end{bmatrix} \quad (12)$$

Since  $R$  has full column rank, then  $L$  is nonsingular.

- 3)  $s_j \neq \infty, s_{j+1} = \infty$ : It can be verified that, in this case,  $S_{j+1}S_j = s_j S_j - I_n$ . Pre-multiplying (8) by  $S_{j+1}$ , substituting and using (9),

$$S_{j+1} V_j \begin{bmatrix} L = [L_1 \ e_j] \\ R_{j-1} \ e_j \end{bmatrix} = V_{j+1} \begin{bmatrix} R \\ s_j R_{j-1} - E_{j-1} \ r_j \\ 0 \ r_{j+1,j+1} \end{bmatrix} \quad (13)$$

Next, we show that  $L$  is nonsingular. Suppose on the contrary that (11) is satisfied. Post-multiplying (13) by  $\begin{bmatrix} x' & y' \end{bmatrix}'$  shows that  $y = 0$  since  $r_{j+1,j+1} \neq 0$ . Thus  $R_{j-1}x = 0$ , since  $V_{j+1}$  is unitary, and  $R_{j-1}x = 0$ . This shows that  $x = 0$ . This contradiction shows that  $L$  is nonsingular.

- 4)  $s_j = \infty = s_{j+1}$ : Here,  $S_{j+1} = S_j = A$ . In this case, it is straightforward to show that

$$S_{j+1} V_j \begin{bmatrix} L = [L_1 \ e_j] \\ E_{j-1} \ e_j \end{bmatrix} = V_{j+1} \begin{bmatrix} R \\ R_{j-1} \ r_j \\ 0 \ r_{j+1,j+1} \end{bmatrix} \quad (14)$$

Next, we show that  $L$  is nonsingular. Suppose on the contrary that (11) is satisfied. Post-multiplying (14) by  $\begin{bmatrix} x' & y' \end{bmatrix}'$  shows that  $y = 0$  since  $r_{j+1,j+1} \neq 0$ . Thus  $L_1x = 0$ , which shows that  $x = 0$ . This contradiction shows that  $L$  is nonsingular.

In all four cases, we have  $S_{j+1} V_j = V_{j+1} R_j$  with  $R_j = R L^{-1}$ , which proves the induction step and establishes the lemma.  $\blacksquare$

## B. Moment Matching Properties

The next corollary establishes a moment matching property for Algorithm 3.

COROLLARY 3.1: Let all variables be as defined in Lemma 3.1. Suppose that  $q \in \mathbf{S}$  has multiplicity  $p$ . Then

$$\mathcal{X}_p(A, B, q) \subseteq \text{colsp}(V_m)$$

*Proof:* Suppose that  $q = s_{i_1} = \dots = s_{i_p}$ , where  $1 \leq i_1 < \dots < i_p \leq m$  so that  $S_{i_1} = \dots = S_{i_p}$ . Now, it follows from Lemma 3.1 that, for  $j = 2, \dots, m$  and for  $k = 1, \dots, j-1$ , we have

$$S_j V_k = V_j R_{j,k} \quad (15)$$

for some  $R_{j,k} \in \mathbb{C}^{j \times k}$ . Also, we have

$$S_1 B = V_1 r_{11} e_1^{-1} \quad (16)$$

from (7).

1) Suppose that  $q = \infty$ . A little reflection will show that

$$K_p(A, B, q) = \text{colsp} \left( \begin{bmatrix} B & S_{i_2} B & \cdots & (S_{i_p} \cdots S_{i_2} B) \end{bmatrix} \right)$$

There are two possibilities:

a)  $i_1 = 1$ : Here  $S_{i_1} = I_n$ , and so (16) implies that

$$B = V_{i_1} R_{i_1,0} \quad (17)$$

where  $R_{i_1,0} = r_{11} e_1^{-1}$ .

b)  $i_1 > 1$ : Here  $s_1 \neq \infty$ , and so pre-multiplying (16) by  $(s_1 I - A)$  and using (15) gives

$$\begin{aligned} B &= s_1 V_1 r_{11} e_1^{-1} - S_{i_1} V_1 r_{11} e_1^{-1} \\ &= s_1 V_1 r_{11} e_1^{-1} - V_{i_1} R_{i_1,1} r_{11} e_1^{-1} \end{aligned}$$

which has the form (17) for some  $R_{i_1,0} \in \mathbb{C}^{i_1 \times 1}$ .

In either case, (17) is satisfied. An iteration using (15) and (17) now gives

$$\begin{aligned} B &= V_{i_1} R_{i_1,0} \in \text{colsp}(V_m) \\ S_{i_2} B &= S_{i_2} V_{i_1} R_{i_1,0} = V_{i_2} R_{i_2,i_1} R_{i_1,0} \in \text{colsp}(V_m) \\ &\vdots \\ S_{i_p} \cdots S_{i_2} B &= V_{i_p} R_{i_p,i_{p-1}} \cdots R_{i_2,i_1} R_{i_1,0} \in \text{colsp}(V_m) \end{aligned}$$

which proves the corollary.

2) Suppose that  $q \neq \infty$ . Here

$$\begin{aligned} \mathcal{K}_p(A, B, q) &= \\ &\text{colsp} \left( \begin{bmatrix} S_{i_1} B & (S_{i_2} S_{i_1} B) & \cdots & (S_{i_p} \cdots S_{i_1} B) \end{bmatrix} \right) \end{aligned}$$

There are three possibilities:

a)  $i_1 = 1$ : Here we simply write (16) as

$$S_{i_1} B = V_{i_1} R_{i_1,0} \quad (18)$$

where  $R_{i_1,0} = r_{11} e_1^{-1}$ .

b)  $i_1 > 1$ ,  $s_1 \neq \infty$ : Here,  $S_{i_1} = S_1 + (s_{i_1} - s_1) S_{i_1} S_1$ . Using (15) and (16),

$$\begin{aligned} S_{i_1} B &= S_1 B + (s_{i_1} - s_1) S_{i_1} S_1 B \\ &= V_1 R_{1,0} + (s_{i_1} - s_1) V_{i_1} R_{i_1,1} R_{1,0} \end{aligned}$$

which has the form (18) for some  $R_{i_1,0} \in \mathbb{C}^{i_1 \times 1}$ .

c)  $i_1 > 1$ ,  $s_1 = \infty$ : Here  $S_{i_1} S_1 = S_{i_1}$ . Thus  $S_{i_1} B = S_{i_1} S_1 B = S_{i_1} V_1 r_{11} e_1^{-1} = V_{i_1} R_{i_1,1} r_{11} e_1^{-1}$  which has the form (18) with  $R_{i_1,0} = R_{i_1,1} r_{11} e_1^{-1}$ .

Thus in all three cases, (18) is satisfied. Now an iteration using (15) and (18) gives

$$\begin{aligned} S_{i_1} B &= V_{i_1} R_{i_1,0} \\ S_{i_2} S_{i_1} B &= S_{i_2} V_{i_1} R_{i_1,0} = V_{i_2} R_{i_2,i_1} R_{i_1,0} \\ &\vdots \\ S_{i_p} \cdots S_{i_1} B &= V_{i_p} R_{i_p,i_{p-1}} \cdots R_{i_2,i_1} R_{i_1,0}. \end{aligned}$$

This completes the proof.  $\blacksquare$

### C. Arnoldi-like equations

The second corollary derives Arnoldi-like equations for Algorithm 3.

COROLLARY 3.2: Let all variables be as given in Lemma 3.1. Suppose  $v_{m+1} \in \mathbb{C}^n$  is defined such that, with  $V_{m+1} := \begin{bmatrix} V_m & v_{m+1} \end{bmatrix}$ ,

$$\begin{aligned} \text{colsp} \left( \begin{bmatrix} V_m & AV_m e_m \end{bmatrix} \right) &= \text{colsp}(V_{m+1}), \\ V_{m+1}' V_{m+1} &= I_{m+1} \end{aligned}$$

where  $0 \neq e_m \in \mathbb{C}^{m \times 1}$ . Define

$$\begin{bmatrix} A_m & B_m \\ C_{V_m} & b_m \end{bmatrix} := \begin{bmatrix} V_m' AV_m & V_m' B \\ v_{m+1}' AV_m & v_{m+1}' B \end{bmatrix}.$$

Then the following Arnoldi-like equations are satisfied

$$AV_m = V_m A_m + v_{m+1} C_{V_m} \quad (19)$$

$$B = V_m B_m + v_{m+1} b_m \quad (20)$$

Furthermore,  $b_m = 0$  if  $\infty \in \mathbf{S}$ .

*Proof:* It is easy to see that the construction of  $v_{m+1}$  is equivalent to running Algorithm 3 for one extra interpolation point at  $s_{m+1} = \infty$ , so that  $S_{m+1} = A$ . Thus a direct application of Lemma 3.1 gives (19).

To prove (20) we proceed as follows. If  $s_1 = \infty$ , so that  $S_1 = I_n$ , (7) gives  $B = V_1 r_{11} e_1^{-1}$  and the result is proved. If  $s_1 \neq \infty$ , so that  $S_1 = (s_1 I_n - A)^{-1}$ , then (7) gives  $B = (s_1 I_n - A) V_1 r_{11} e_1^{-1}$  and so  $B \in \text{colsp}(V_{m+1})$  from (19), which proves (20).

To prove the final part, suppose that  $\infty \in \mathbf{S}$ . Then Lemma 3.1 implies that  $K_p(A, B, \infty) \subseteq \text{colsp}(V_m)$  for some  $p \geq 1$ . In particular, this implies that  $B \in \text{colsp}(V_m)$  and so  $b_m = v_{m+1}' B = 0$  and the result is proved.  $\blacksquare$

As mentioned before the importance of the derivation of the Arnoldi equations in the rational case in the standard form is that they are very useful for error analysis, for deriving residual error expressions, stopping criteria and also for best choice of the interpolation points [6].

### D. Breakdown analysis

An important issue for Algorithm 3 is its possible breakdown in line 8 (ignoring the trivial case  $S_1 B = 0$ ). This is related to the choice of the updating vectors  $e_j$  and corresponds to

$$\hat{v}_{j+1} = \overbrace{(S_{j+1} V_j - V_j (V_j' S_{j+1} V_j))}^{Z_j} e_j = 0$$

in exact arithmetic. There are two possible sources for this breakdown.

1) The first occurs when  $Z_j = 0$ . In this case the algorithm breaks down for all choices of  $e_j$ . Suppose that  $Z_j = 0$ , or

$$S_{j+1} V_j = V_j (V_j' S_{j+1} V_j).$$

This is equivalent to  $V_j$  being an invariant subspace of  $S_{j+1}$  and, since  $S_{j+1}$  is either equal to  $A$  (if  $s_{j+1} = \infty$ ) or  $(s_{j+1} I_n - A)^{-1}$  (if  $s_{j+1} \neq \infty$ ), it follows that  $Z_j = 0$

if and only if  $V_j$  is an invariant subspace of  $A$ , that is, if and only if  $AV_j = V_jA_j$  for some  $A_j \in \mathbb{C}^{j \times j}$ . Furthermore, since  $S_1B = v_1r_{11}$  and since  $S_1$  is either equal to  $I_n$  (if  $s_1 = \infty$ ) or  $(s_1I_n - A)^{-1}$  (if  $s_1 \neq \infty$ ), it follows that  $Z_j = 0$  implies that either  $B = v_1r_{11}$  or  $B = s_1v_1r_{11} - Av_1r_{11}$ . It follows that, in both cases,  $Z_j = 0$  implies that  $B = V_jB_j$  for some  $B_j \in \mathbb{C}^{j \times 1}$ . Thus  $Z_j = 0$  implies that the projection of  $(A, B)$  onto  $\text{colsp}(V_j)$  is exact, i.e.,  $(sI_n - A)^{-1}B = V_j(sI_j - A_j)^{-1}B_j$  for all  $s \in \mathbb{C}$ . This will therefore be referred to as a lucky breakdown.

- 2) The second occurs when  $Z_j \neq 0$  but  $Z_j e_j = 0$ , that is, the source of the breakdown is an inappropriate choice of  $e_j$ . Two proposed choices are  $e_j = \begin{bmatrix} 1 & 0_{1 \times (j-1)} \end{bmatrix}'$  and  $e_j = \begin{bmatrix} 0_{1 \times (j-1)} & 1 \end{bmatrix}'$ . It is clear that the first choice will cause breakdown if  $s_{j+1} = s_i$  for  $1 < i < j+1$ . What is less clear is that the second choice may also cause a breakdown as the following argument shows. Suppose  $A$  and  $B$  are such that  $B'AB = 0$  and  $S = \{\infty, \infty, 0\}$ . It is easy to find examples for which  $Z_3 \neq 0$ , although the last column of  $Z_3$  is zero, e.g.,  $A = \text{diag}(1, 2, 3, -6)$  and  $B = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}'$ . Thus for Algorithm 3, a breakdown, which is not a lucky breakdown, may occur because of the choice of  $e_j$ . Note however, that if  $s_i = \infty$  for all  $i$ , that is, in the standard Arnoldi algorithm, then this second choice of  $e_j$  guarantees that  $Z_j e_j = 0$  if and only if  $Z_j = 0$ , so that only lucky breakdowns may occur.

There are various ways of choosing  $e_j$ . We can choose  $e_j$  at random, checking that  $\hat{v}_{j+1} \neq 0$ . A more systematic, though also more expensive, way is to choose  $e_j$  as the singular vector of  $Z_j$  corresponding to the largest singular value. This will guarantee that breakdown will always be a lucky breakdown. Also, these choices maximize  $\|\hat{v}_{j+1}\|$  (subject to  $\|e_j\| \leq 1$ ), which will tend to improve the numerical properties of the algorithm, since we divide by  $\|\hat{v}_{j+1}\|$  in Algorithm 3.

Remark: Given a real system  $G(s)$  it is desirable to obtain the reduced order system with real matrices  $A_m, B_m, C_m, D_m$ . The work presented here is in complex arithmetic since real arithmetic implementation is beyond the scope of this paper. More details of real arithmetic implementation can be found in [22], [26].

#### IV. CONCLUSIONS

In this paper we have given an analysis of a modified rational Arnoldi algorithm first suggested in [23]. We have established the moment matching properties of the algorithm and given a breakdown analysis. We have also derived a set of Arnoldi-like equations for the algorithm. As shown in [6] in the context of the standard rational Arnoldi algorithm, these equations are very useful for error analysis, choice of interpolation points and other applications.

#### REFERENCES

[1] A.C. Antoulas, D.C. Sorensen, and S. Gugercin. A survey of model reduction methods for large-scale systems. *Contemp. Math*, 280:193–219, 2001.

[2] W.E Arnoldi. The principle of minimized iterations in the solution of matrix eigenvalue problem. *Quart. Appl. Math.*, 9:17–29, 1951.

[3] D.L. Boley. Krylov space methods on state-space control models. Technical report TR92-18, University of Minnesota, 1992.

[4] D.L. Boley and G. Golub. The nonsymmetric Lanczos algorithm and controllability. *Syst. Contr. Lett.*, 16:97–105, 1991.

[5] A. Bultheel and M. Van Barel. Padé techniques for model reduction in linear system theory: a survey. *J. Comput. Appl. Math*, 14:401–438, March 1986.

[6] M. Frangos and I.M. Jaimoukha. Adaptive rational Krylov algorithms for model reduction. In *Proc. European Control Conference*, pages 4179–4186, July 2007.

[7] R. W. Freund. Model reduction methods based on krylov subspaces. *Acta Numerica*, 12:267319, 2003.

[8] K. Gallivan, E. Grimme, and P. Van Dooren. Padé approximation of large-scale dynamic systems with Lanczos methods. In *Proc. IEEE Conference on Decision and Control*, 1994.

[9] K. Gallivan, E. Grimme, and P. Van Dooren. A rational Lanczos algorithm for model reduction. *Numer. Algorithms*, 12:33–64, 1996.

[10] K. Gallivan, A. Vandendorpe, and P. Van Dooren. Sylvester equations and projection-based model reduction. *J. Comput. Appl. Math*, 162:213–229, 2004.

[11] E. Grimme. *Krylov projection methods for model reduction*. PhD thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1997.

[12] E. Grimme, K. Gallivan, and P. Van Dooren. A rational Lanczos algorithm for model reduction II: Interpolation point selection. Technical report, University of Illinois at Urbana Champaign., 1998.

[13] E.J. Grimme, D.C. Sorensen, and P. Van Dooren. Model reduction of state space systems via an implicitly restarted Lanczos method. *Numer. Algorithms*, 12:1–31, 1995.

[14] H.J.Lee, C.C.Chu, and W.S.Feng. An adaptive-order rational Arnoldi method for model-order reductions of linear time-invariant systems. *Linear Algebra and its Applications*, 415:235–261, 2006.

[15] I.M. Jaimoukha and E.M. Kasenally. Implicitly restarted Krylov subspace methods for stable partial realizations. *SIMAX*, 18(3):633–652, 1997.

[16] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Nat. Bur. Stand*, 45:255–282, 1950.

[17] C. Lanczos. Solution of systems of linear equations by minimized iterations. *J. Res. Nat. Bur. Stand*, 49:33–53, 1952.

[18] K. Henrik A. Olsson. Model order reduction in FEMLAB by dual rational arnoldi. Licentiate thesis, Chalmers University of Technology, 2002.

[19] K. Henrik A. Olsson. *Model order reduction with rational Krylov Methods*. PhD thesis, KTH Computer Science and Communication, Stockholm, Sweden, 2005.

[20] K. Henrik A. Olsson and A. Ruhe. Rational krylov for eigenvalue computation and model order reduction. *BIT Numerical Mathematics*, (46):S99–S111, September 2006.

[21] V. Papakos. *Restarted Lanczos algorithms for model reduction*. PhD thesis, Imperial College of Science, Technology and Medicine, London, UK, Mar 2003.

[22] A. Ruhe. The rational Krylov algorithm for nonsymmetric eigenvalue problems III: complex shifts for real matrices. *BIT*, 34:165–176, November 1994.

[23] A. Ruhe. Rational Krylov algorithms for nonsymmetric eigenvalue problems, II: Matrix pairs. *Linear Algebra and its Applications*, 197–198:283–295, January-February 1994.

[24] A. Ruhe. Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils. *SIAM J. Sci. Comput.*, 19:1535–1551, September 1998.

[25] Y. Saad. Analysis on some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 29:209–228, 1992.

[26] Y. Saad and Henk A. van der Vorst. Iterative solution of linear systems in the 20th century. *J. Comput. Appl. Math*, 66(0):1–33, 2000.

[27] D. Sorensen. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM J. Matrix Anal. Appl*, 13(1):357–385, 1992.