SIAM J. SCI. COMPUT.
Vol. 24, No. 5, pp. 1693–1709

A QUASI-MINIMAL RESIDUAL METHOD FOR SIMULTANEOUS PRIMAL-DUAL SOLUTIONS AND SUPERCONVERGENT FUNCTIONAL ESTIMATES*

JAMES LU† AND DAVID L. DARMOFAL†

Abstract. The adjoint solution has found many uses in computational simulations where the quantities of interest are the functionals of the solution, including design optimization, error estimation, and control. In those applications where both the solution and the adjoint are desired, the conventional approach is to apply iterative methods to solve the primal and dual problems separately. However, we show that there is an advantage associated with iterating the primal and dual problem simultaneously since this enables the construction of iterative methods where both the primal and the dual iterates may be chosen so that they provide functional estimates that are "superconvergent" in that the error converges at twice the order of the optimal global solution error norm. In particular, we show that the structure of the Lanczos process allows for this superconvergence property and propose a modified QMR method which uses the same Lanczos process to simultaneously solve the primal and dual problems. Thus both the primal and the dual systems are solved at essentially the same computational cost as the conventional QMR method applied to the primal problem alone. Numerical experiments show that our proposed method does indeed exhibit superconvergence behavior.

Key words. QMR, Lanczos process, iterative functional estimates, superconvergence

AMS subject classifications. 65F10, 65G99

PII. S1064827501390625

1. Introduction.

We consider the primal linear output,

$$J^{\mathrm{pr}}(\mathbf{x}) = \mathbf{g}^T\mathbf{x}, \tag{1.1}$$

arising from the solution of a large nonsingular linear system,

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{1.2}$$

The equivalent dual statement of this problem expresses the output as

$$J^{\mathrm{du}}(\mathbf{y}) = \mathbf{y}^T\mathbf{b}, \tag{1.3}$$

where $J^{\mathrm{du}}(\mathbf{y}) = J^{\mathrm{pr}}(\mathbf{x})$ when $\mathbf{y}$ is the adjoint vector and is the solution of

$$\mathbf{A}^T\mathbf{y} = \mathbf{g}. \tag{1.4}$$

The adjoint solution has multiple uses in the context of computational simulation. For example, the adjoint can be employed in design optimization to efficiently calculate the gradients of the outputs with respect to the control factors (Jameson [13], Reuther, Jameson, and Alonso [18, 19], Giles and Pierce [12], Elliot and Peraire [6], Anderson and Venkatakrishnan [2]). Furthermore, the adjoint solution can be used to estimate and control errors in functional outputs of computational simulations (Becker and

*Received by the editors June 8, 2001; accepted for publication (in revised form) August 28, 2002; published electronically April 1, 2003.
http://www.siam.org/journals/sisc/24-5/39062.html
†Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 (jameslu@mit.edu, darmofal@mit.edu).

Rannacher [3], Peraire and Patera [16], Pierce and Giles [17], and Venditti and Darmofal [26]). Of particular interest to this work, Pierce and Giles have demonstrated that superconvergent functional estimates can be achieved in a fairly general setting using an adjoint-based recovery method [17]. Motivated by the numerous uses of the adjoint solution, we consider Krylov methods where both the primal and the dual solutions are desired. In particular, we show how the structure of the Lanczos process allows this to be done efficiently and develop a modification of the QMR method [10] in which we have the following:

- The primal and dual problems are solved simultaneously with essentially the same computational work as solving only one of the problems with the original QMR algorithm.
- The error in the functional estimates obtained from either the primal or dual iterates is "superconvergent" in a sense to be defined later.

A widely used Krylov subspace method for solving nonsymmetric linear systems is GMRES [22], which uses the Arnoldi process to find iterate $\mathbf{x}_n$ which minimizes the 2-norm of the residual vector within the nth Krylov subspace generated by the matrix $\mathbf{A}$ and starting vector $\mathbf{v}_1$,

$$\mathcal{K}_n(\mathbf{v}_1, \mathbf{A}) \equiv \text{span}\{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \mathbf{A}^2\mathbf{v}_1, \ldots, \mathbf{A}^{n-1}\mathbf{v}_1\}. \tag{1.5}$$

However, the Arnoldi process cannot be done recursively since all the vectors obtained in the previous iterations need to be stored. Furthermore, the work per iteration increases linearly with the iteration number. Therefore, for solving large systems such as those arising from the discretization of PDEs, the version of GMRES with restarts is used. This has the undesirable property that the residual norm stagnates after each restart.

For larger systems, an attractive alternative is the class of Krylov subspace methods based on the nonsymmetric Lanczos process. Methods from this class include BiCG and QMR [10]. These methods are characterized by recursivity, requiring no storing of prior vectors and taking constant work per iteration. Here, two Krylov subspaces are generated using starting vectors $\mathbf{v}_1$ and $\mathbf{w}_1$:

$$\begin{aligned} \mathcal{K}_n(\mathbf{v}_1, \mathbf{A}) &\equiv \text{span}\{\mathbf{v}_1, \mathbf{A}\mathbf{v}_1, \mathbf{A}^2\mathbf{v}_1, \ldots, \mathbf{A}^{n-1}\mathbf{v}_1\}, \\ \mathcal{K}_n(\mathbf{w}_1, \mathbf{A}^T) &\equiv \text{span}\{\mathbf{w}_1, \mathbf{A}^T\mathbf{w}_1, (\mathbf{A}^T)^2\mathbf{w}_1, \ldots, (\mathbf{A}^T)^{n-1}\mathbf{w}_1\}. \end{aligned} \tag{1.6}$$

The starting vector $\mathbf{w}_1$ in the Lanczos process may be chosen arbitrarily, as long as $\mathbf{w}_1^T\mathbf{v}_1 \neq 0$. In particular, we note that with the choice of $\mathbf{w}_1 = \mathbf{g}$, the Lanczos process solves not only the primal problem (1.2) but also the dual problem (1.4) at essentially no extra cost. This property of the Lanczos process is well known [21], but the dual solution that is solved implicitly has not been utilized in practice. We demonstrate this simultaneous primal-dual approach using the QMR method.

Superconvergence is a concept usually used in contexts such as Galerkin finite element methods to describe convergence of functional estimates [27]. Let us now describe what we mean by superconvergence in the context of iterative methods. For the iterative solution of linear systems of the form (1.2), we have the following global a priori estimate:

$$\|\mathbf{x} - \mathbf{x}_n\| \leq \|\mathbf{A}^{-1}\|\|\mathbf{r}_n^{\text{pr}}\|, \tag{1.7}$$

where $\mathbf{r}_n^{\text{pr}} \equiv \mathbf{b} - \mathbf{A}\mathbf{x}_n$. The above estimate is optimal in the sense that no higher exponent on the residual norm is possible. Therefore, in general we expect the solution

error and arbitrary output functionals to converge at a rate no higher than that of the residual. Under the special circumstances that certain functionals of the approximate solution converge at orders of the residual norm higher than that given by the global estimate (1.7), those quantities are said to be superconverging. Note that this phenomenon is distinct from superlinear convergence [25, 4], which describes the acceleration of residual convergence rate as the iteration proceeds, or superconvergence as used in [5] to loosely describe the faster residual convergence when certain projected solutions are used as initial guess. In particular, we define an iterative method to be superconvergent if the superconvergence phenomenon can be demonstrated for arbitrary linear systems and linear functionals. A goal of this paper is to illustrate how the underlying structure of the nonsymmetric Lanczos process allows one to obtain iterates giving superconvergent functional estimates. To do this, we need to describe the Lanczos process.

We compare the proposed superconvergent variant of QMR with the conventional QMR method in a number of numerical experiments. It is demonstrated that while both the primal and the dual residual norm converge at rates similar to that of QMR applied to the respective problems separately, the proposed approach provides primal and dual iterates both of which give superconvergent estimates for the linear functional. This translates to linear functional estimates obtained with either the primal or the dual iterates that can be much better than those provided by conventional QMR. Alternatively, the computational work required to produce primal and dual approximations of a fixed functional output accuracy can be reduced compared to conventional approaches.

2. Superconvergent estimates of linear functionals. Most Krylov subspace methods choose iterates to satisfy certain orthogonality conditions or, equivalently, to minimize certain norms. For example, the 2-norm of the residual vector is minimized in the case of the GMRES method [22] and the 2-norm of the error vector in the case of the generalized minimal error method (GMERR) [28]. In cases where the principal quantity of interest is a linear functional $J^{\mathrm{pr}}(\mathbf{x})$ rather than the solution itself, those methods may not give optimal results within the Krylov subspace. Alternatively, one could minimize the error in the linear objective function which may be expressed as a linear functional of the primal residual through the adjoint

$$\begin{aligned}\Delta J_n^{\mathrm{pr}} &\equiv J^{\mathrm{pr}}(\mathbf{x}) - J^{\mathrm{pr}}(\mathbf{x}_n)\\ &= \mathbf{g}^T(\mathbf{x}-\mathbf{x}_n)\\ &= \mathbf{y}^T\mathbf{A}(\mathbf{x}-\mathbf{x}_n)\\ &= \mathbf{y}^T\mathbf{r}_n^{\mathrm{pr}}.\end{aligned}$$

Unfortunately, the adjoint is not easily obtained since the dual problem is just as difficult to solve as the primal problem. Let us now suppose that estimates of the adjoint are obtained in parallel with the primal problem through some means. Denote the nth iterative estimate of the primal problem (1.2) by $\mathbf{x}_n$, and that of the dual problem (1.4) by $\mathbf{y}_n$. Then, it may be shown that the true value of the functional may be expanded as

$$\begin{aligned}J^{\mathrm{pr}}(\mathbf{x}) &= \mathbf{g}^T\mathbf{x}\\ &= \mathbf{g}^T\mathbf{x}_n + \mathbf{y}_n^T\mathbf{r}_n^{\mathrm{pr}} + (\mathbf{y}-\mathbf{y}_n)^T\mathbf{A}(\mathbf{x}-\mathbf{x}_n).\end{aligned}$$

Then, rewriting the above using the dual residual $\mathbf{r}_n^{\mathrm{du}} \equiv \mathbf{g} - \mathbf{A}^T\mathbf{y}_n$, we obtain

$$J^{\mathrm{pr}}(\mathbf{x}) = \mathbf{g}^T\mathbf{x}_n + \mathbf{y}_n^T\mathbf{r}_n^{\mathrm{pr}} + (\mathbf{r}_n^{\mathrm{du}})^T\mathbf{A}^{-1}\mathbf{r}_n^{\mathrm{pr}}. \tag{2.1}$$

The smallest singular value of $\mathbf{A}$, denoted by $\sigma_{\min}$, satisfies the following:

$$\sigma_{\min} = \min_{\mathbf{z},\mathbf{z}\neq\mathbf{0}} \frac{\|\mathbf{A}\mathbf{z}\|}{\|\mathbf{z}\|}. \tag{2.2}$$

Then, we have the following bound on the last term of (2.1), as is also discussed in [14]:

$$|(\mathbf{r}_n^{\mathrm{du}})^T \mathbf{A}^{-1} \mathbf{r}_n^{\mathrm{pr}}| \leq \frac{\|\mathbf{r}_n^{\mathrm{pr}}\| \|\mathbf{r}_n^{\mathrm{du}}\|}{\sigma_{\min}}. \tag{2.3}$$

Using the above, the error in the functional estimate provided by the nth primal iterate $\mathbf{x}_n$ is bounded by

$$|J^{\mathrm{pr}}(\mathbf{x}) - \mathbf{g}^T \mathbf{x}_n| \leq |\mathbf{y}_n^T \mathbf{r}_n^{\mathrm{pr}}| + \frac{\|\mathbf{r}_n^{\mathrm{pr}}\| \|\mathbf{r}_n^{\mathrm{du}}\|}{\sigma_{\min}}. \tag{2.4}$$

Our approach to obtaining superconvergent functional estimates is to seek the primal iterates within the respective Krylov subspaces such that the term linear in the primal residual, $\mathbf{y}_n^T \mathbf{r}_n^{\mathrm{pr}}$, contributes little to the functional error. That is, we will seek to minimize, in some sense, the primal residual weighted by the adjoint approximation, while ensuring that the primal residual norm goes to zero. If the first term on the right-hand side of (2.4) may be successfully removed, we see that the functional estimates evaluated using the primal iterates would scale as

$$|J^{\mathrm{pr}}(\mathbf{x}) - \mathbf{g}^T \mathbf{x}_n| \sim \frac{\|\mathbf{r}_n^{\mathrm{pr}}\| \|\mathbf{r}_n^{\mathrm{du}}\|}{\sigma_{\min}}. \tag{2.5}$$

That is, this method would have the potential of producing primal iterates which give functional estimates converging at the rate of the product of the primal and the dual residual norms. Therefore, if it can be assumed that the convergence of dual residual norms corresponding to the given adjoint approximations may be bounded above by some power of the primal residual norm, superconvergence is attained. This gives the ingredients for the construction of the class of Krylov methods which have the superconvergence property.

Our basis for the modification will be the QMR method [10], which is robust and exhibits satisfactory residual norm convergence behavior. We will attempt to remove the first order dependence of the functional error on the primal residual norm through a choice of certain parameters present in the norm-minimization. We note that the Lanczos process treats the primal and dual problem symmetrically and the primal and dual matrices have the same eigenvalue distribution. Therefore, if the problem is relatively well-conditioned (perhaps by the use of effective preconditioners) so that eigenvalue convergence bounds are descriptive, the dual convergence is the same as that of the primal and we expect superconvergence of the primal functional estimates at twice the order of the primal residual convergence rate.

It is to be noted that the above observations made regarding superconvergent iterative estimates also apply to the dual problem. We have the result analogous to (2.4),

$$|J^{\mathrm{du}}(\mathbf{y}) - \mathbf{y}_n^T \mathbf{b}| \leq |\mathbf{x}_n^T \mathbf{r}_n^{\mathrm{du}}| + \frac{\|\mathbf{r}_n^{\mathrm{pr}}\| \|\mathbf{r}_n^{\mathrm{du}}\|}{\sigma_{\min}}. \tag{2.6}$$

So, here the approach is to choose the dual residual such that

$$\mathbf{x}_n^T \mathbf{r}_n^{\mathrm{du}} \approx 0. \tag{2.7}$$

Our proposed method incorporates the strategy towards superconvergence for both the primal and the dual problem.

3. Nonsymmetric Lanczos process. The nonsymmetric Lanczos process constructs biorthogonal basis vectors of Krylov subspaces which are then used in methods such as QMR. We use the Lanczos process based on coupled two-term recurrences rather than that based on three-term recursions as used in [10] since, even though they are mathematically equivalent, the former is numerically more robust than the latter, as observed in [8]. Also, to simplify matters the look-ahead process [9] is not included.

- Initialization
  - Choose normalized initial vectors $\mathbf{v}_1$ and $\mathbf{w}_1$.
  - Set $\mathbf{p}_0 = \mathbf{q}_0 = \mathbf{0}$, $\epsilon_0 = \rho_1 = \xi_1 = 1, n = 1$.
- At iteration n:
  1. If $\epsilon_{n-1} = 0$, stop. Otherwise compute $\delta_n = \mathbf{w}_n^T \mathbf{v}_n$. If $\delta_n = 0$, then stop.
  2. Update

$$\mathbf{p}_n = \mathbf{v}_n - \mathbf{p}_{n-1}(\xi_n \delta_n / \epsilon_{n-1}),$$
$$\mathbf{q}_n = \mathbf{w}_n - \mathbf{q}_{n-1}(\rho_n \delta_n / \epsilon_{n-1}).$$

  3. Compute

$$\epsilon_n = \mathbf{q}_n^T \mathbf{A} \mathbf{p}_n,$$
$$\beta_n = \epsilon_n / \delta_n.$$

  Update

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{A}\mathbf{p}_n - \mathbf{v}_n \beta_n, \quad \rho_{n+1} = \|\tilde{\mathbf{v}}_{n+1}\|,$$
$$\tilde{\mathbf{w}}_{n+1} = \mathbf{A}^T \mathbf{q}_n - \mathbf{w}_n \beta_n, \quad \xi_{n+1} = \|\tilde{\mathbf{w}}_{n+1}\|.$$

  4. If $\rho_{n+1} = 0$ or $\xi_{n+1} = 0$, then stop. Else, update

$$\mathbf{v}_{n+1} = \tilde{\mathbf{v}}_{n+1} / \rho_{n+1},$$
$$\mathbf{w}_{n+1} = \tilde{\mathbf{w}}_{n+1} / \xi_{n+1}.$$

The result of the above iteration may be summarized compactly. First, we introduce the notation

$$\mathbf{V}_n \equiv \Big[\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \mathbf{v}_n\Big],$$
$$\mathbf{W}_n \equiv \Big[\mathbf{w}_1 \ \mathbf{w}_2 \ \cdots \mathbf{w}_n\Big],$$
$$\mathbf{P}_n \equiv \Big[\mathbf{p}_1 \ \mathbf{p}_2 \ \cdots \mathbf{p}_n\Big],$$
$$\mathbf{Q}_n \equiv \Big[\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \mathbf{q}_n\Big]. \tag{3.1}$$

Then, it may be seen from the Lanczos process that the above satisfies

$$\begin{aligned} \mathbf{V}_n &= \mathbf{P}_n \mathbf{U}_n, \\ \mathbf{W}_n &= \mathbf{Q}_n \boldsymbol{\Gamma}_n^{-1} \mathbf{U}_n \boldsymbol{\Gamma}_n, \\ \mathbf{A}\mathbf{P}_n &= \mathbf{V}_{n+1} \mathbf{L}_n, \\ \mathbf{A}^T \mathbf{Q}_n &= \mathbf{W}_{n+1} \boldsymbol{\Gamma}_{n+1}^{-1} \mathbf{L}_n \boldsymbol{\Gamma}_n, \end{aligned} \tag{3.2}$$

where the matrices $\boldsymbol{\Gamma}_n$, $\mathbf{U}_n$, and $\mathbf{L}_n$ are defined as

$$\boldsymbol{\Gamma}_n = \operatorname{diag}(\gamma_1, \gamma_2, \ldots, \gamma_n), \tag{3.3}$$

$$\mathbf{U}_n = \begin{bmatrix} 1 & \xi_2\delta_2/\epsilon_1 & 0 & \cdots & 0 \\ 0 & 1 & \xi_3\delta_3/\epsilon_2 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \xi_n\delta_n/\epsilon_{n-1} \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}, \tag{3.4}$$

$$\mathbf{L}_n = \begin{bmatrix} \beta_1 & 0 & 0 & \cdots & 0 \\ \rho_2 & \beta_2 & 0 & \cdots & 0 \\ 0 & \rho_3 & \beta_3 & \cdots & 0 \\ 0 & 0 & \rho_4 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \beta_n \\ 0 & \cdots & \cdots & 0 & \rho_{n+1} \end{bmatrix}, \tag{3.5}$$

the scalars $\xi_j, \delta_j, \epsilon_j, \rho_j$ are constants defined in the Lanczos process, and the constants γ_j satisfy the relation

$$\gamma_j = \begin{cases} 1, & j = 1, \\ \gamma_{j-1}\rho_j/\xi_j, & 1 < j \le n. \end{cases} \tag{3.6}$$

Furthermore, it may be verified that the vectors $\mathbf{v}_j$ and $\mathbf{w}_j$ satisfy the biorthogonality condition

$$\mathbf{W}_n^T \mathbf{V}_n = \operatorname{diag}(\delta_1, \delta_2, \ldots, \delta_n), \tag{3.7}$$

and the vectors $\mathbf{p}_j$ and $\mathbf{q}_j$ satisfy the $\mathbf{A}$-orthogonality condition

$$\mathbf{Q}_n^T \mathbf{A} \mathbf{P}_n = \operatorname{diag}(\epsilon_1, \epsilon_2, \ldots, \epsilon_n). \tag{3.8}$$

In the next section, we give a brief description of how QMR uses the Lanczos process to generate iterates that approximate the linear system (1.2).

4. Description of conventional QMR. Let the initial guess for the linear system (1.2) be $\mathbf{x}_0$. Then the initial residual is

$$\mathbf{r}_0 \equiv \mathbf{b} - \mathbf{A}\mathbf{x}_0.$$

At each iteration, QMR seeks an iterate $\mathbf{x}_n$ within the Krylov subspace

$$\begin{aligned} \mathbf{x}_n &\in \mathbf{x}_0 + \mathcal{K}_n(\mathbf{r}_0, \mathbf{A}) \\ &\in \mathbf{x}_0 + \operatorname{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0\}. \end{aligned} \tag{4.1}$$

With the initial vector taken to be the normalized initial residual, $\mathbf{v}_1 \equiv \mathbf{r}_0/\rho_1$, $\rho_1 \equiv \|\mathbf{r}_0\|$, it may be seen from the Lanczos iteration that

$$\text{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\} = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0\}.$$

Hence, from (4.1) $\mathbf{x}_n$ may be written as

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{V}_n \mathbf{U}_n^{-1} \mathbf{z}_n, \tag{4.2}$$

where the matrices $\mathbf{V}_n$, $\mathbf{U}_n$ are defined in (3.1) and (3.4), respectively, and the vector $\mathbf{z}_n$ is yet to be determined. Using the identities given in (3.2), it may be seen that

$$\begin{aligned} \mathbf{r}_n^{\text{pr}} &= \mathbf{b} - \mathbf{A}\mathbf{x}_n \\ &= \mathbf{r}_0 - \mathbf{V}_{n+1}\mathbf{L}_n\mathbf{z}_n. \end{aligned} \tag{4.3}$$

Using the fact that $\mathbf{v}_1 = \mathbf{r}_0/\rho_1$, and introducing an $(n+1) \times (n+1)$ diagonal weight matrix $\mathbf{\Omega}_n^{\text{pr}}$,

$$\mathbf{\Omega}_n^{\text{pr}} = \text{diag}(\omega_1^{\text{pr}}, \omega_2^{\text{pr}}, \ldots, \omega_{n+1}^{\text{pr}}),$$

(4.3) may be written as

$$\mathbf{r}_n^{\text{pr}} = \mathbf{V}_{n+1}\left(\mathbf{\Omega}_n^{\text{pr}}\right)^{-1}\left(\rho_1\omega_1^{\text{pr}}\mathbf{e}_1^{(n+1)} - \mathbf{\Omega}_n^{\text{pr}}\mathbf{L}_n\mathbf{z}_n\right). \tag{4.4}$$

Finally, $\mathbf{z}_n$ is chosen so that the 2-norm of the quasi residual is minimized:

$$\mathbf{z}_n = \arg\min_{\mathbf{z}} \left\| \rho_1\omega_1^{\text{pr}}\mathbf{e}_1^{(n+1)} - \mathbf{\Omega}_n^{\text{pr}}\mathbf{L}_n\mathbf{z} \right\|. \tag{4.5}$$

Since the matrix $\mathbf{L}_n$ has a bidiagonal structure, the above minimization may be done recursively, essentially performing QR decomposition of $\mathbf{L}_n$ using successive Givens rotations.

The original QMR algorithm is formulated with a diagonal weighting matrix $\mathbf{\Omega}_{\mathbf{n}}^{\text{pr}}$, and convergence has been shown for arbitrary weights $\omega_j^{\text{pr}} \neq 0$ [10]. Extension of the weight matrix to block diagonal form having upper triangular blocks has also been done [23]. However, in practice the weight matrix is usually set to unity owing to the lack of a better choice. Moreover, in practice the two starting vectors are usually taken to be the same and the fact that the same Lanczos iteration contains a dual problem is not utilized. The desire of not using $\mathbf{A}^T$ partly led to the development of transpose-free variants of QMR [7, 11].

5. Modifications for simultaneous dual solution. Here, we show the modifications to the conventional QMR which enable the solution of the dual problem (1.4) together with the primal problem (1.2) using the same Lanczos process. First, we observe that (3.2) implies not only

$$\mathbf{A}\mathbf{V}_n = \mathbf{V}_{n+1}\mathbf{L}_n\mathbf{U}_n, \tag{5.1}$$

but also

$$\mathbf{A}^T\mathbf{W}_n = \mathbf{W}_{n+1}\mathbf{\Gamma}_{n+1}^{-1}\mathbf{L}_n\mathbf{U}_n\mathbf{\Gamma}_n. \tag{5.2}$$

If we take $\mathbf{y}_0$ to be the zero vector, this suggests taking the starting vector $\mathbf{w}_1$ for the Lanczos process to be

$$\mathbf{w}_1 = \frac{\mathbf{g}}{\xi_1}, \quad \xi_1 = \|\mathbf{g}\|. \tag{5.3}$$

It may be observed that in the Lanczos process, given a certain $\mathbf{v}_1$, the choice of $\mathbf{w}_1$ is arbitrary so long as $\mathbf{w}_1^T\mathbf{v}_1 \neq 0$. Only in rare circumstances will the nonorthogonality condition not be satisfied. Note that this is a departure from the standard QMR, which chooses the starting vector for the Lanczos iteration to be $\mathbf{w}_1 = \mathbf{v}_1$.

Let us seek iterates for the dual problem of the form

$$\mathbf{y}_n = \mathbf{W}_n\boldsymbol{\Gamma}_n^{-1}\mathbf{U}_n^{-1}\boldsymbol{\Gamma}_n\mathbf{k}_n, \tag{5.4}$$

where the vector $\mathbf{k}_n$ is yet to be determined. Then, the dual residual $\mathbf{r}_n^{\mathrm{du}}$ is of the form

$$\begin{aligned}\mathbf{r}_n^{\mathrm{du}} &= \mathbf{g} - \mathbf{A}^T\mathbf{y}_n \\ &= \mathbf{W}_{n+1}\left(\boldsymbol{\Omega}_n^{\mathrm{du}}\right)^{-1}\left(\xi_1\omega_1^{\mathrm{du}}\mathbf{e}_1^{n+1} - \boldsymbol{\Omega}_n^{\mathrm{du}}\boldsymbol{\Gamma}_{n+1}^{-1}\mathbf{L}_n\boldsymbol{\Gamma}_n\mathbf{k}_n\right),\end{aligned} \tag{5.5}$$

where the dual weight parameter matrix $\boldsymbol{\Omega}_n^{\mathrm{du}}$ has been introduced. Analogous to the approach of QMR for the primal problem, we will seek $\mathbf{k}_n$ to minimize the 2-norm quasi residual

$$\mathbf{k}_n = \arg\min_{\mathbf{k}} \left\| \xi_1\omega_1^{\mathrm{du}}\mathbf{e}_1^{n+1} - \boldsymbol{\Omega}_n^{\mathrm{du}}\boldsymbol{\Gamma}_{n+1}^{-1}\mathbf{L}_n\boldsymbol{\Gamma}_n\mathbf{k} \right\|. \tag{5.6}$$

We note that the term $\boldsymbol{\Gamma}_{n+1}^{-1}\mathbf{L}_n\boldsymbol{\Gamma}_n$ in (5.5) is the analogue of $\mathbf{L}_n$ in (4.3). In fact, we see that

$$\boldsymbol{\Gamma}_{n+1}^{-1}\mathbf{L}_n\boldsymbol{\Gamma}_n = \begin{bmatrix} \beta_1 & 0 & \cdots & 0 \\ \xi_2 & \beta_2 & \cdots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & \ddots & \xi_n & \beta_n \\ 0 & \cdots & 0 & \xi_{n+1} \end{bmatrix}, \tag{5.7}$$

and comparison with $\mathbf{L}_n$ defined in (3.5) shows that $\boldsymbol{\Gamma}_{n+1}^{-1}\mathbf{L}_n\boldsymbol{\Gamma}_n$ is just $\mathbf{L}_n$ with the replacement $\rho_j \to \xi_j$. Thus, the adjoint iterates $\mathbf{y}_n$ may be obtained from the Lanczos vectors with the same procedure as the standard QMR method for the primal problem by replacing $\mathbf{v}_j \to \mathbf{w}_j$, $\mathbf{p}_j \to \mathbf{q}_j$, and $\rho_j \to \xi_j$.

6. Superconvergent QMR with preconditioning. With the use of preconditioners $\mathbf{M}_1$ and $\mathbf{M}_2$, the primal problem (1.2) effectively becomes

$$\begin{aligned} \mathbf{A}'\mathbf{x}' = \mathbf{b}', \ \text{ where } \mathbf{A}' &\equiv \mathbf{M}_1^{-1}\mathbf{A}\mathbf{M}_2^{-1}, \\ \mathbf{x}' &\equiv \mathbf{M}_2\mathbf{x}, \\ \mathbf{b}' &\equiv \mathbf{M}_1^{-1}\mathbf{b}. \end{aligned} \tag{6.1}$$

We require that the value of the primal linear functional be invariant under the preconditioning transformation; that is,

$$J^{\mathrm{pr}}(\mathbf{x}) = \mathbf{g}^T\mathbf{x} = \mathbf{g}'^T\mathbf{x}', \tag{6.2}$$

thereby obtaining the expression for $\mathbf{g}'$,

$$\mathbf{g}' = \mathbf{M}_2^{-T}\mathbf{g}. \tag{6.3}$$

Then, the adjoint for the preconditioned system, $\mathbf{y}'$, is just the solution to

$$\mathbf{A}'^{T}\mathbf{y}' = \mathbf{g}'. \tag{6.4}$$

From (6.4), it may be verified that

$$\mathbf{y}' = \mathbf{M}_1^{T}\mathbf{y}. \tag{6.5}$$

Furthermore, from (6.4) and (6.1),

$$\mathbf{b}^T\mathbf{y} = \mathbf{b}'^T\mathbf{y}'. \tag{6.6}$$

Equation (6.5) shows that the adjoint for the original system (1.4) may be recovered from that of the preconditioned system by simply multiplying with the matrix $\mathbf{M}_1^{-T}$ at the end of the iterations. In what follows, we will work exclusively with the preconditioned systems.

In achieving superconvergence, we seek to minimize

$$\check{\mathbf{y}}_{n+1}'^{T}\mathbf{r}_n^{\mathrm{pr}}, \tag{6.7}$$

where $\check{\mathbf{y}}'_{n+1}$ is an estimate of the adjoint available at the $(n+1)$th iteration and $\mathbf{r}_n^{\mathrm{pr}}$ is the primal residual at the nth iteration, both for the preconditioned system. Note that the check mark on $\check{\mathbf{y}}'_{n+1}$ is used to differentiate the adjoint estimates that we use to determine weight parameters and the adjoint estimates carried forward in our simultaneous primal-dual method.

The above provides a natural scaling for the QMR weight parameters. Specifically, by letting

$$\omega_j^{\mathrm{pr}} = \check{\mathbf{y}}_{n+1}'^{T}\mathbf{v}_j, \;\; 1 \le j \le n+1, \tag{6.8}$$

then

$$\begin{aligned}
\check{\mathbf{y}}_{n+1}'^{T}\mathbf{r}_n^{\mathrm{pr}} &= \check{\mathbf{y}}_{n+1}'^{T}\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_{n+1}\end{bmatrix}\left(\mathbf{\Omega}_n^{\mathrm{pr}}\right)^{-1}\left(\rho_1\omega_1^{\mathrm{pr}}\mathbf{e}_1^{(n+1)} - \mathbf{\Omega}_n^{\mathrm{pr}}\mathbf{L}_n\mathbf{z}_n\right)\\
&= \begin{bmatrix}\check{\mathbf{y}}_{n+1}'^{T}\mathbf{v}_1 & \mathbf{y}_{n+1}'^{T}\mathbf{v}_2 & \cdots & \check{\mathbf{y}}_{n+1}'^{T}\mathbf{v}_{n+1}\end{bmatrix}\left(\mathbf{\Omega}_n^{\mathrm{pr}}\right)^{-1}\left(\rho_1\omega_1^{\mathrm{pr}}\mathbf{e}_1^{(n+1)} - \mathbf{\Omega}_n^{\mathrm{pr}}\mathbf{L}_n\mathbf{z}_n\right)\\
&= \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \end{bmatrix}\left(\rho_1\omega_1^{\mathrm{pr}}\mathbf{e}_1^{(n+1)} - \mathbf{\Omega}_n^{\mathrm{pr}}\mathbf{L}_n\mathbf{z}_n\right).
\end{aligned}$$

Thus, with the ω_j^{pr} as chosen in (6.8), the linear error term (6.7) in the linear functional is approximately equal to the sum of the entries of the primal quasi residual. By selecting $\mathbf{z}_n$ to minimize this weighted primal quasi residual, the primal iterates obtained from our modified QMR approach should more accurately approximate the linear functional.

However, the scheme as described above could suffer from nonrecursivity because at each iteration n, all weights $\omega_j^{\mathrm{pr}}, 1 \le j \le n$ must be updated and all prior vectors

might need to be stored. To maintain recursivity, we introduce a truncated approximation to the adjoint, $\check{\mathbf{y}}'_{i,n+1}$, where the index i is an approximation parameter. We define $\check{\mathbf{y}}'_{i,n+1}$ as

$$\check{\mathbf{y}}'_{i,n+1} \equiv \left(\frac{\check{\mathbf{Y}}'^{T}_{i+1}\mathbf{v}_1}{\mathbf{w}_1^T\mathbf{v}_1}\right)\mathbf{w}_1 + \left(\frac{\check{\mathbf{Y}}'^{T}_{i+2}\mathbf{v}_2}{\mathbf{w}_2^T\mathbf{v}_2}\right)\mathbf{w}_2 + \cdots + \left(\frac{\check{\mathbf{Y}}'^{T}_{i+n+1}\mathbf{v}_{n+1}}{\mathbf{w}_{n+1}^T\mathbf{v}_{n+1}}\right)\mathbf{w}_{n+1},$$

where we take $\check{\mathbf{Y}}'_i$ to be the adjoint iterate from the standard QMR algorithm. Although other choices may be thought of, using this truncated approximation, we have

$$\check{\mathbf{y}}'^{T}_{i,n+1}\mathbf{v}_j = \check{\mathbf{Y}}'^{T}_{i+j}\mathbf{v}_j. \tag{6.9}$$

Associated with a certain choice of i is the storage needed for the i vectors $[\mathbf{p}_1 \cdots \mathbf{p}_i]$, each of the size of $\mathbf{x}'$, but no extra computational work is required. We will see that for our numerical experiments, a small i (≈ 3) works well enough.

With $\check{\mathbf{y}}'_{i,n+1}$ as defined in (6.9), we take the weight parameters to be

$$\begin{aligned}\omega_j^{\mathrm{pr}} &= \check{\mathbf{y}}'^{T}_{i,n+1}\mathbf{v}_j \\ &= \check{\mathbf{Y}}'^{T}_{i+j}\mathbf{v}_j.\end{aligned} \tag{6.10}$$

A similar strategy of weight parameter determination may be done for the dual problem. Analogous to (6.10), we take the dual weights to be

$$\omega_j^{\mathrm{du}} = \check{\mathbf{X}}'^{T}_{i+j}\mathbf{w}_j, \tag{6.11}$$

where similarly, $\check{\mathbf{X}}'_n$ is the primal solution estimate obtained at iteration n using the QMR method.

It is to be noted that although (6.10) and (6.11) imply forming the QMR iterates $\check{\mathbf{X}}'_n$ and $\check{\mathbf{Y}}'_n$ and performing inner products for the calculation of each weight, this is not necessary. Instead, they are equivalently but cheaply calculated from scalars obtained in the i Lanczos iterations ahead of the current primal and dual iterates.

7. Algorithm implementation. In this section, we describe an implementation of the superconvergent simultaneous primal-dual QMR method. The Lanczos iteration is carried i steps ahead of both the primal and the dual iterates. This method takes essentially the same computational work as the conventional QMR since the same number of Lanczos iterations is used. However, the former requires a constant extra storage of two sets of vectors $\mathbf{P} = [\mathbf{p}_1\, \mathbf{p}_2 \cdots \mathbf{p}_i]$ and $\mathbf{Q} = [\mathbf{q}_1\, \mathbf{q}_2 \cdots \mathbf{q}_i]$, where each vector is of the same size as $\mathbf{b}$. Additionally, three sets of i scalars need to be stored : $\beta = [\beta_1\, \beta_2 \cdots \beta_i]$, $\rho = [\rho_1\, \rho_2 \cdots \rho_i]$, and $\xi = [\xi_1\, \xi_2 \cdots \xi_i]$. After convergence by some criteria, the primal and dual solutions for the original systems are recovered by observing (6.1) and (6.5).

Algorithm with Lanczos Forward Index i.

- Initialization
  - Set initial guesses to be the zero vector : $\mathbf{x}_0^{\mathrm{pr}} = \mathbf{x}_0^{\mathrm{du}} = \mathbf{0}$.
  - Let $\rho_1 = \|\mathbf{M}_1^{-1}\mathbf{b}\|$, $\mathbf{v}_1 = \mathbf{M}_1^{-1}\mathbf{b}/\rho_1$. Let $\xi_1 = \|\mathbf{M}_2^{-T}\mathbf{g}\|$, $\mathbf{w}_1 = \mathbf{M}_2^{-T}\mathbf{g}/\xi_1$.
  - Check that $\mathbf{w}_1^T\mathbf{v}_1 \neq 0$, otherwise restart.
  - Set $\mathbf{p}_0 = \mathbf{q}_0 = \mathbf{d}_0^{\mathrm{pr}} = \mathbf{d}_0^{\mathrm{du}} = \mathbf{0}$.
  - Set $c_0^{\mathrm{pr}} = c_0^{\mathrm{du}} = \epsilon_0 = 1$, $\vartheta_0^{\mathrm{pr}} = \vartheta_0^{\mathrm{du}} = 0$, $\eta_0^{\mathrm{pr}} = \eta_0^{\mathrm{du}} = -1$.
  - Set $w\_c_0^{\mathrm{pr}} = w\_c_0^{\mathrm{du}} = 1$, $w\_\vartheta_0^{\mathrm{pr}} = w\_\vartheta_0^{\mathrm{du}} = 0$, $w\_\eta_0^{\mathrm{pr}} = w\_\eta_0^{\mathrm{du}} = -1$.

– Initialize vector storage $\mathbf{P} = [\mathbf{p}_1\, \mathbf{p}_2\, \cdots\, \mathbf{p}_i]$ and $\mathbf{Q} = [\mathbf{q}_1\, \mathbf{q}_2\, \cdots\, \mathbf{q}_i]$, scalar storage $\beta = [\beta_1\, \beta_2\, \cdots\, \beta_i]$, $\rho = [\rho_1\, \rho_2\, \cdots\, \rho_i]$, and $\xi = [\xi_1\, \xi_2\, \cdots\, \xi_i]$.
– Set counter values for weight parameter accumulation :
$k_0 = 1$
FOR $h = 1 : i$ DO
$m(h) = 2 - h$
END

- FOR $n = 1, 2, 3, \ldots,$ DO
1. If $\epsilon_n = 0$ or $\delta_n = 0$, stop. Otherwise, compute $\delta_n = \mathbf{w}_n^T \mathbf{v}_n$.
2. Update counter $k_n = (k_n \bmod i) + 1$. Update vectors

$$\begin{aligned} \mathbf{p}_n &= \mathbf{v}_n - \mathbf{p}_{n-1}(\xi_n \delta_n / \epsilon_{n-1}), \\ \mathbf{q}_n &= \mathbf{w}_n - \mathbf{q}_{n-1}(\rho_n \delta_n / \epsilon_{n-1}). \end{aligned}$$

Store $\mathbf{P}(:, k_n) = \mathbf{p}_n$, $\mathbf{Q}(:, k_n) = \mathbf{q}_n$.
3. Compute $\tilde{\mathbf{p}}_n = \mathbf{A}(\mathbf{M}_2^{-1}\mathbf{p}_n)$ and $\tilde{\mathbf{q}}_n = \mathbf{M}_1^{-T}\mathbf{q}_n$. Update $\epsilon_n = \tilde{\mathbf{q}}_n^T \tilde{\mathbf{p}}_n$, and set $\beta_n = \epsilon_n / \delta_n$.
Update

$$\begin{aligned} \tilde{\mathbf{v}}_{n+1} &= \mathbf{M}_1^{-1}\tilde{\mathbf{p}}_n - \beta_n \mathbf{v}_n, \\ \tilde{\mathbf{w}}_{n+1} &= \mathbf{M}_2^{-T}(\mathbf{A}^T \tilde{\mathbf{q}}_n) - \beta_n \mathbf{w}_n. \end{aligned}$$

Update $\rho_{n+1} = \|\tilde{\mathbf{v}}_{n+1}\|$, $\xi_{n+1} = \|\tilde{\mathbf{w}}_{n+1}\|$. Store $\beta(k_n) = \beta_n$, $\rho(k_n) = \rho_n$, $\xi(k_n) = \xi_n$.
If $\rho_{n+1} \neq 0$ and $\xi_{n+1} \neq 0$, update

$$\begin{aligned} \mathbf{v}_{n+1} &= \tilde{\mathbf{v}}_{n+1} / \rho_{n+1}, \\ \mathbf{w}_{n+1} &= \tilde{\mathbf{w}}_{n+1} / \xi_{n+1}. \end{aligned}$$

4. Update

$$\begin{aligned} w\_\vartheta_n^{\mathrm{pr}} &= \frac{\rho_{n+1}}{w\_c_{n-1}^{\mathrm{pr}} \beta_n},\ w\_c_n^{\mathrm{pr}} = \frac{1}{\sqrt{1 + (w\_\vartheta_n^{\mathrm{pr}})^2}} \\ w\_\eta_n^{\mathrm{pr}} &= -w\_\eta_{n-1}^{\mathrm{pr}} \frac{\rho_n (w\_c_n^{\mathrm{pr}})^2}{\beta_n (w\_c_{n-1}^{\mathrm{pr}})^2} \\ w\_\vartheta_n^{\mathrm{du}} &= \frac{\xi_{n+1}}{w\_c_{n-1}^{\mathrm{du}} \beta_n},\ w\_c_n^{\mathrm{du}} = \frac{1}{\sqrt{1 + (w\_\vartheta_n^{\mathrm{du}})^2}} \\ w\_\eta_n^{\mathrm{du}} &= -w\_\eta_{n-1}^{\mathrm{du}} \frac{\xi_n (w\_c_n^{\mathrm{du}})^2}{\beta_n (w\_c_{n-1}^{\mathrm{du}})^2}. \end{aligned}$$

5. Weight parameters $\omega_{n-i+1}^{\mathrm{pr}}$ and $\omega_{n-i+1}^{\mathrm{du}}$ calculation
FOR $h = 1 : i$ DO
IF $m(h) = 1$ THEN
$w\_q^{\mathrm{pr}}(h) = \delta_n, w\_d^{\mathrm{pr}}(h) = 0, w\_x^{\mathrm{pr}}(h) = 0$
$w\_q^{\mathrm{du}}(h) = \delta_n, w\_d^{\mathrm{du}}(h) = 0, w\_x^{\mathrm{du}}(h) = 0$
END
IF $1 < m(h) < i$ THEN
IF $n \neq 1$ THEN $w\_q^{\mathrm{pr}}(h) = -(\xi_n \delta_n / \epsilon_{n-1}) w\_q^{\mathrm{pr}}(h)$
$w\_d^{\mathrm{pr}}(h) = w\_\eta_n^{\mathrm{pr}} w\_q^{\mathrm{pr}}(h) + w\_d^{\mathrm{pr}}(h)(w\_\vartheta_{n-1}^{\mathrm{pr}} w\_c_n^{\mathrm{pr}})^2$
$w\_x^{\mathrm{pr}}(h) = w\_x^{\mathrm{pr}}(h) + w\_d^{\mathrm{pr}}(h)$

IF $n \neq 1$ THEN $w\_q^{\mathrm{du}}(h) = -(\rho_n \delta_n / \epsilon_{n-1}) w\_q^{\mathrm{du}}(h)$
$w\_d^{\mathrm{du}}(h) = w\_\eta_n^{\mathrm{du}} w\_q^{\mathrm{du}}(h) + w\_d^{\mathrm{du}}(h)(w\_\vartheta_{n-1}^{\mathrm{du}} w\_c_n^{\mathrm{du}})^2$
$w\_x^{\mathrm{du}}(h) = w\_x^{\mathrm{du}}(h) + w\_d^{\mathrm{du}}(h)$
END
IF $m(h) = i - 1$ THEN
$m(h) = 2 - i$
$\omega_{n-i+1}^{\mathrm{pr}} = w\_x^{\mathrm{du}}(h)$
$\omega_{n-i+1}^{\mathrm{du}} = w\_x^{\mathrm{pr}}(h)$
END
END

6. IF $n \geq i$ THEN

$$\vartheta_{n-i}^{\mathrm{pr}} = \frac{\omega_{n-i+1}^{\mathrm{pr}} \rho((k_n \bmod i) + 1)}{\omega_{n-i}^{\mathrm{pr}} c_{n-i-1}^{\mathrm{pr}} \beta((k_n \bmod i) + 1)}$$
$$c_{n-i}^{\mathrm{pr}} = \frac{1}{\sqrt{1 + (\vartheta_{n-i}^{\mathrm{pr}})^2}}$$
$$\eta_{n-i}^{\mathrm{pr}} = -\eta_{n-i-1}^{\mathrm{pr}} \frac{\rho((k_n \bmod i) + 1)(c_{n-i}^{\mathrm{pr}})^2}{\beta((k_n \bmod i) + 1)(c_{n-i-1}^{\mathrm{pr}})^2}$$
$$\mathbf{d}_{n-i}^{\mathrm{pr}} = \eta_{n-i}^{\mathrm{pr}} \mathbf{P}(:, (k_n \bmod i) + 1) + (\vartheta_{n-i-1}^{\mathrm{pr}} c_{n-i}^{\mathrm{pr}})^2 \mathbf{d}_{n-i-1}^{\mathrm{pr}}$$
$$\mathbf{x}_{n-i}^{\mathrm{pr}} = \mathbf{x}_{n-i-1}^{\mathrm{pr}} + \mathbf{d}_{n-i}^{\mathrm{pr}}$$
$$\omega_{n-i}^{\mathrm{pr}} = \omega_{n-i+1}^{\mathrm{pr}}$$

$$\vartheta_{n-i}^{\mathrm{du}} = \frac{\omega_{n-i+1}^{\mathrm{du}} \xi((k_n \bmod i) + 1)}{\omega_{n-i}^{\mathrm{du}} c_{n-i-1}^{\mathrm{du}} \beta((k_n \bmod i) + 1)}$$
$$c_{n-i}^{\mathrm{du}} = \frac{1}{\sqrt{1 + (\vartheta_{n-i}^{\mathrm{du}})^2}}$$
$$\eta_{n-i}^{\mathrm{du}} = -\eta_{n-i-1}^{\mathrm{du}} \frac{\xi((k_n \bmod i) + 1)(c_{n-i}^{\mathrm{du}})^2}{\beta((k_n \bmod i) + 1)(c_{n-i-1}^{\mathrm{du}})^2}$$
$$\mathbf{d}_{n-i}^{\mathrm{du}} = \eta_{n-i}^{\mathrm{du}} \mathbf{Q}(:, (k_n \bmod i) + 1) + (\vartheta_{n-i-1}^{\mathrm{du}} c_{n-i}^{\mathrm{du}})^2 \mathbf{d}_{n-i-1}^{\mathrm{du}}$$
$$\mathbf{x}_{n-i}^{\mathrm{du}} = \mathbf{x}_{n-i-1}^{\mathrm{du}} + \mathbf{d}_{n-i}^{\mathrm{du}}$$
$$\omega_{n-i}^{\mathrm{du}} = \omega_{n-i+1}^{\mathrm{du}}$$

END

7. FOR $h = 1 : i$ DO
$m(h) = m(h) + 1$
END

- Obtain primal and dual solutions for the original systems

$$\text{Primal solution} = (\mathbf{M}_2)^{-1} \mathbf{x}_n^{\mathrm{pr}},$$
$$\text{Dual solution} = (\mathbf{M}_1)^{-T} \mathbf{x}_n^{\mathrm{du}}.$$

8. Numerical experiments. Here, we show the results of numerical experiments conducted using MATLAB. The proposed method with weight parameters determined using (6.10) and (6.11) is denoted as superconvergent simultaneous QMR

(SSQMR). The method which solves both the primal and the dual problem but with all weight parameters set to unity is denoted as simultaneous QMR (SQMR). The conventional QMR method has unity weights and initial vectors are chosen to be the same: $\mathbf{w}_1 = \mathbf{v}_1$. QMR is separately applied to the primal and dual problems, and the results are compared to SSQMR and SQMR.

For all the examples shown here, the Lanczos forward index i for SSQMR is taken to be 3, since this has been shown to give good results. Also, the initial guesses $\mathbf{x}'_0$ and $\mathbf{y}'_0$ for all the methods are chosen to be the zero vector. No convergence criteria are used; rather, the number of iteration steps is specified in each case. The test problems are not excessively large so that the solutions for the linear systems are available up to the level of roundoff.

Example 1. In this example, we consider the second order finite difference discretization on a 51×51 grid of

$$\begin{aligned} \nabla^2 u &= \frac{1}{\pi} \exp^{-(x+2)^2-(y-1/2)^2}, \quad \Omega \in [0,1] \times [0,1], \\ u|_\Gamma &= 0, \end{aligned}$$

resulting in the matrix $\mathbf{A}$ with 12205 nonzero elements. The linear functional is the discretized approximation of

$$\int_0^1 \int_0^1 u(x,y) \sin(\pi x) \sin(\pi y) dx dy.$$

The left and right preconditioners were obtained from the incomplete LU factorization [21], of $\mathbf{A}$ using a drop tolerance of 2×10^{-2}, resulting in $\mathbf{L}$ and $\mathbf{U}$ having 12059 and 12057 nonzero elements, respectively.

From Figure 1 it may be seen that the primal and dual functional estimates obtained using standard QMR converge at roughly the same rate as the respective residual norms, showing that it is not superconvergent. SQMR obtains better functional estimates as a result of the better choice of starting vectors. In fact, the method appears to exhibit superconvergence. However, the behavior is not very consistent. On the other hand, SSQMR obtains functional estimates that are consistently superconverging at twice the order of residual convergence, and improvement in functional estimates over SQMR is clearly seen.

Example 2. In this example, we consider a linear system arising from a first order backward-Euler implicit scheme for solving a first order upwind discretization of the two-dimensional compressible flow Euler equations using the NASA Langley unstructured flow solver FUN2D [1]. The specific problem is the transonic flow around the NACA 0012 airfoil (freestream Mach number of 0.8 and 1.25 degree angle of attack). The mesh is composed of 1980 triangular elements with 990 nodes. The linear output is the airfoil drag functional linearized about the current iterate.

The matrix $\mathbf{A}$ has 108532 nonzero entries. The left and right preconditioners were obtained from the incomplete LU factorization of $\mathbf{A}$ using a drop tolerance of 10^{-2}, resulting in $\mathbf{L}$ and $\mathbf{U}$ having 88253 and 93588 nonzero elements, respectively.

Again, from Figure 2 we observe that the functional error convergence slopes of SSQMR are roughly twice that of conventional QMR, confirming the prediction made regarding superconvergence. In this example, the functional error of SQMR is close to that of SSQMR, showing that the unity parameter happens to be quite a good choice in this case. Still, SSQMR consistently gives better functional estimates than SQMR.

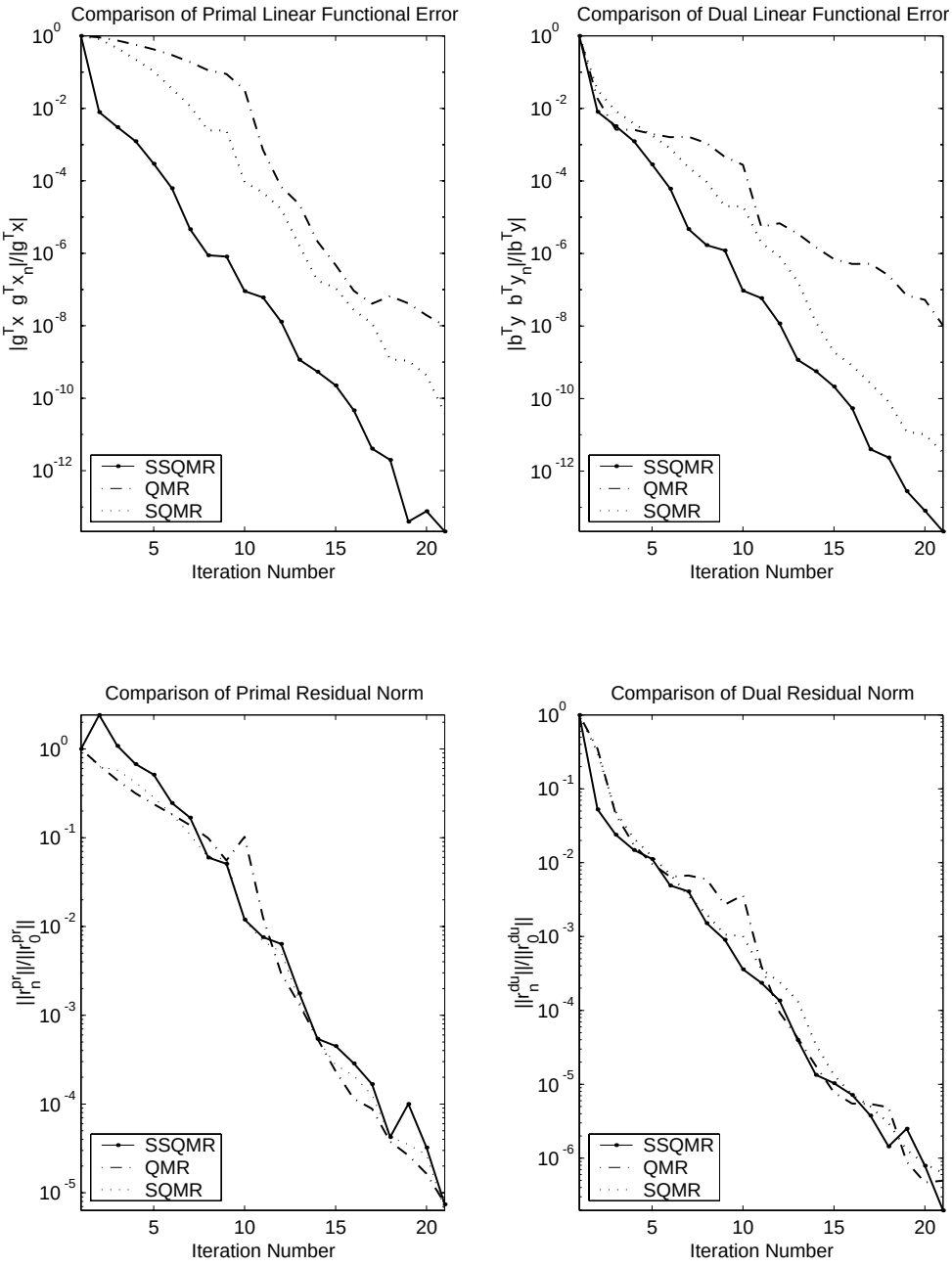

FIG. 1. *Plots from Example* 1: *Poisson problem.*

9. Conclusions. We show that the inherent structure of the nonsymmetric Lanczos process allows for the simultaneous solution of a dual problem associated with a desired functional output. Then, applying adjoint analysis to iterative methods, we show how iterates may be chosen so that they give superconvergent functional output estimates. This superconvergence property associated with the choice of starting vector of the Lanczos process appears not to have been studied in literature. We

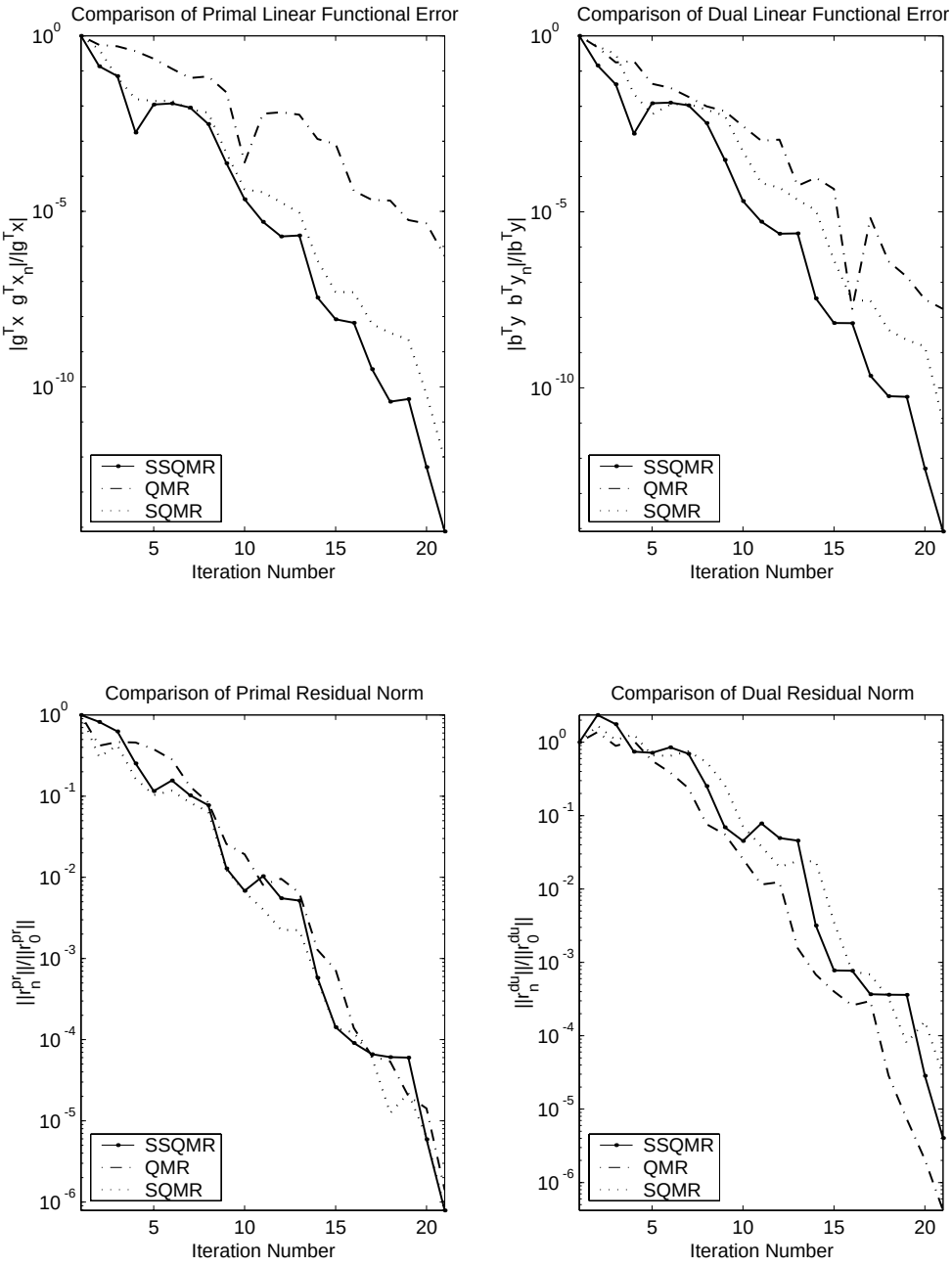

FIG. 2. *Plots from Example* 2: *Euler flow.*

modify the QMR method to construct a superconvergent variant (SSQMR), and numerical experiments demonstrate that the modified method gives the convergence of functional output to be twice the order of the residual norm, while giving residual behavior that is highly similar to QMR. This may lead to noticeable computational savings for applications where functional output is of particular interest. The inclu-

sion of the look-ahead procedure in the Lanczos process to increase the robustness of SSQMR and the extension of superconvergence to other Krylov methods remain subjects for further research.

REFERENCES

[1] W. K. Anderson and D. L. Bonhaus, *An implicit upwind algorithm for computing turbulent flows on unstructured grids*, Comput. & Fluids, 23 (1994), pp. 1–21.

[2] W. K. Anderson and V. Venkatakrishnan, *Aerodynamics Design Optimization on Unstructured Grids with a Continuous Adjoint Formulation*, AIAA paper 97-0643, AIAA, 1997.

[3] R. Becker and R. Rannacher, *Weighted a posteriori error control in finite element methods*, in Proceedings of the 2nd European Conference on Numerical Mathematics and Advanced Applications, Heidelberg, Germany, World Scientific, Singapore, 1998, pp. 621–637.

[4] B. Beckermann and A. B. J. Kuijlaars, *Superlinear convergence of conjugate gradients*, SIAM J. Numer. Anal., 39 (2001), pp. 300–329.

[5] T. F. Chan and W. L. Wan, *Analysis of projection methods for solving linear systems with multiple right-hand sides*, SIAM J. Sci. Comput., 18 (1997), pp. 1698–1721.

[6] J. Elliot and J. Peraire, *Aerodynamic Optimization on Unstructured Meshes with Viscous Effects*, AIAA paper 97-1849, AIAA, 1997.

[7] R. W. Freund, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM J. Sci. Comput., 14 (1993), pp. 470–482.

[8] R. W. Freund and N. M. Nachtigal, *An implementation of the QMR method based on coupled two-term recurrences*, SIAM J. Sci. Comput., 15 (1994), pp. 313–337.

[9] R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal, *An Implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices*, SIAM J. Sci. Comput., 14 (1993), pp. 137–158.

[10] R. W. Freund and N. M. Nachtigal, *QMR: A quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.

[11] R. W. Freund and T. Szeto, *A transpose-free quasi-minimal residual squared algorithm for non-Hermitian linear systems*, in Advances in Computer Methods for Partial Differential Equations - VII, R. Vichnevetsky, D. Knight, and G. Richter, eds., IMACS, New Brunswick, NJ, 1992, pp. 258–264.

[12] M. B. Giles and N. A. Pierce, *An introduction to the adjoint approach to design*, Flow Turbul. Combust., 65 (2000), pp. 393–415.

[13] A. Jameson, *Aerodynamic design via control theory*, J. Sci. Comput., 3 (1988), pp. 233–260.

[14] L. Machiels, *Output Bounds for Iterative Solutions of Linear Partial Differential Equations*, Technical Report UCRL-JC-137864, Lawrence Livermore National Laboratory, Livermore, CA, 2000.

[15] A. T. Patera and E. M. Rønquist, *A general output bound result: Application to discretization and iteration error estimation and control*, Math. Models Methods Appl. Sci., 11 (2001), pp. 685–712.

[16] J. Peraire and A. T. Patera, *Asymptotic a posteriori finite element bounds for the outputs of noncoercive problems: The Helmholtz and Burgers equations*, Comput. Methods Appl. Mech. Engrg., 171 (1999), pp. 77–86.

[17] N. A. Pierce and M. B. Giles, *Adjoint recovery of superconvergent functionals from PDE approximations*, SIAM Rev., 42 (2000), pp. 247–264.

[18] J. J. Reuther, A. Jameson, and J. J. Alonso, *Constrained multipoint aerodynamics shape optimization using an adjoint formulation and parallel computers*, I, J. Aircraft, 36 (1999), pp. 51–60.

[19] J. J. Reuther, A. Jameson, and J. J. Alonso, *Constrained multipoint aerodynamics shape optimization using an adjoint formulation and parallel computers*, II, J. Aircraft, 36 (1999), pp. 61–74.

[20] M. Rozložník and R. Weiss, *On the stable implementation of the generalized minimal error method*, J. Comput. Appl. Math., 98 (1998), pp. 49–62.

[21] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS, Boston, 1996.

[22] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.

[23] C. H. Tong, *A family of quasi-minimal residual methods for nonsymmetric linear systems*, SIAM J. Sci. Comput., 15 (1994), pp. 89–105.

[24] H. A. van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp.

631–644.
[25] H. A. Van der Vorst, *The superlinear convergence behaviour of GMRES*, J. Comput. Appl. Math., 48 (1993), pp. 327–341.
[26] D. A. Venditti and D. L. Darmofal, *Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow*, J. Comput. Phys., 164 (2000), pp. 204–227.
[27] L. B. Wahlbin, *Superconvergence in Galerkin Finite Element Methods*, Springer-Verlag, Berlin, Heidelberg, 1995.
[28] R. Weiss, *Error-minimizing Krylov subspace methods*, SIAM J. Sci. Comput., 15 (1994), pp. 511–527.