GRADIENT FLOW FOR GENERALIZED NORMAL EQUATION WITH APPLICATIONS TO LINEAR MATRIX EQUATIONS

DRAFT AS OF May 11, 2013

LIQI WANG*, BO YU†, AND MOODY T. CHU‡

Abstract. Solving a system of linear equations by its normal equation usually is highly unrecommended because this approach worsens the condition number and inflates the computational cost. For linear systems whose unknowns are matrices, such as the Sylvester equation, Lyapunov equation, Stein equation, and a variate of their generalizations, the formulation of the corresponding normal equation in the sense of tensor operators offers a common structure via gradient dynamics. This paper explains the setting of this framework and demonstrates its versatility by one simple ODE integrator that can handle almost all these types of problems.

Key words. linear matrix equation, tensor, adjoint, generalized normal equations, gradient flow

1. Introduction. The purpose of this paper is to bring forth the recognition that a linear matrix equations can be regarded as an operator equation, whence we can set up a general framework for numerical computation. So as to outline the structure more clearly, we begin with the very basic concept of matrix representation in linear algebra.

Given two finite dimensional vector spaces $\mathcal{U}$ and $\mathcal{V}$ over $\mathbb{R}$ with respective bases $\{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ and $\{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$, a prevailing fact in linear algebra is that any linear operator $\mathscr{L} : \mathcal{U} \longrightarrow \mathcal{V}$ can be represented by a matrix $L = [\ell_{ij}] \in \mathbb{R}^{m\times n}$ defined by the relationship

$$\mathscr{L}.\mathbf{a}_j = \sum_{i=1}^{m} \ell_{ij}\mathbf{b}_i, \quad j = 1, \ldots n \tag{1.1}$$

with respect to these bases, where the notation "." indicates the action of the operator . The representation refers to characterizing the action of $\mathscr{L}$ by describing its effect on the transformation of coordinates in the sense that

$$\mathscr{L}.\mathbf{x} = \mathscr{L}.\left(\sum_{j=1}^{n} x_j\mathbf{a}_j\right) = \sum_{j=1}^{n} x_j\left(\sum_{i=1}^{m}\ell_{ij}\mathbf{b}_i\right) = \sum_{i=1}^{m}\left(\sum_{j=1}^{n}\ell_{ij}x_j\right)\mathbf{b}_i.$$

It is thus conventional to denote this linear action $\mathscr{L}.\mathbf{x}$ by merely the matrix-to-vector multiplication $L\mathbf{x}$.

The very same notion is applicable to a linear operator $\mathscr{L} : \mathbb{R}^{p\times q} \longrightarrow \mathbb{R}^{m\times n}$. Let matrix entries be indicated by a double index $I = (i, j)$. Without causing ambiguity, let the same notation $E_I = [e^I_{st}]$, where

$$e^I_{st} = \begin{cases} 1, & \text{if } I = (s,t), \\ 0, & \text{otherwise,} \end{cases}$$

typify a standard basis element for both matrix spaces. Analogous to (1.1), assume the fundamental actions

$$\mathscr{L}.E_J = \sum_I \ell_{I,J}E_I, \quad J = (1,1), \ldots, (p,q)$$

on each of the pq basis elements, where the summation is over all double indices I relevant in $\mathbb{R}^{m\times n}$. Note that for each fixed J, $\ell_{:,J}$ is a matrix of size $m \times n$. From the relationship

$$\mathscr{L}.X = \mathscr{L}.\left(\sum_J x_J E_J\right) = \sum_J x_J\left(\sum_I \ell_{I,J}E_I\right) = \sum_I\left(\sum_J \ell_{I,J}x_J\right)E_I,$$

*School of Mathematical Sciences, Dalian University of Technology, Dalian, Liaoning, P. R. C., 116024. (lizzy@mail.dlut.edu.cn)

†School of Mathematical Sciences, Dalian University of Technology, Dalian, Liaoning, P. R. C., 116024. (yubo@dlut.edu.cn)

‡Department of Mathematics, North Carolina State University, Raleigh, NC 27695-8205. (chu@math.ncsu.edu) This research was supported in part by the National Science Foundation under grant DMS-1014666.

we conclude that the matrix representation of $\mathscr{L}$ is actual an order-4 tensor $[\ell_{I,J}]$ whose action on $X \in \mathbb{R}^{p\times q}$ should be interpreted as a tensor-to-matrix multiplication $\circledast$ defined by

$$\mathscr{L}.X \equiv \underbrace{\begin{bmatrix} \boxed{\ell_{11,:}} & \boxed{\ell_{12,:}} & \dots & \boxed{\ell_{1n,:}} \\ \boxed{\ell_{21,:}} & & & \\ \vdots & & & \\ \boxed{\ell_{m1,:}} & & \dots & \boxed{\ell_{mn,:}} \end{bmatrix}}_{L} \circledast X = [\langle \ell_{I,:}, X\rangle] \in \mathbb{R}^{m\times n}. \tag{1.2}$$

where $\ell_{I,:} \in \mathbb{R}^{p\times q}$ and

$$\langle \ell_{I,:}, X\rangle := \sum_J \ell_{I,J} x_J$$

is the Frobenius inner product (over $\mathbb{R}$) of matrices. There is nothing extraordinary about the operation $\circledast$ because, if so desired, we can get back to the usual matrix-to-vector multiplication by vectorizing these matrices.

Our emphasis here is not so much on the tensor representation of a linear map $\mathscr{L} : \mathbb{R}^{p\times q} \longrightarrow \mathbb{R}^{m\times n}$ and the associated tensor-to-matrix multiplication $\circledast$. For a linear matrix equation, such an approach would mean a mundane task of recasting the equation via the Kronecker product as a standard linear system $A\mathbf{x} = \mathbf{b}$ which is then solved by standard algorithms. Rather, we want to regard a linear matrix equations as it is without the vectorization. We want to bypass the usual requirement of either inversion or factorization when tackling the tensor equations. Our approach provides a general framework for almost all types of linear matrix equations.

A critical component in our discussion is the notion of the adjoint, denoted by $\mathscr{L}^\top$, with respect to the operator $\mathscr{L}$. It is known that the operator $\mathscr{L}^\top : \mathbb{R}^{m\times n} \to \mathbb{R}^{p\times q}$ must satisfy the adjoint condition

$$\langle \mathscr{L}.X, G\rangle = \langle X, \mathscr{L}^\top.G\rangle \tag{1.3}$$

for all $X \in \mathbb{R}^{p\times q}$ and $G \in \mathbb{R}^{m\times n}$. For real-valued matrix representations, the adjoint is simply the usual notion of matrix transpose. For tensor operators, we have a similar expression

$$\langle \mathscr{L}.X, G\rangle = \sum_I g_I \left(\sum_J \ell_{I,J} x_J\right) = \sum_J x_J \left(\sum_I g_I \ell_{I,J}\right) = \langle X, \mathscr{L}^\top.G\rangle. \tag{1.4}$$

Since each summation is double indexed, the notion of transpose for an order-4 tensor need be interpreted somewhat differently. In terms of (1.2), the representation of the adjoint operator $\mathscr{L}^\top$ is the transpose of the blocks which themselves are transposed. We shall demonstrate in subsequent discourse that the actual computation of $\mathscr{L}^\top$ need not be so involved.

2. Generalized normal equation.

Given a linear operator $\mathscr{L} : \mathbb{R}^{p\times q} \longrightarrow \mathbb{R}^{m\times n}$ and a matrix $Q \in \mathbb{R}^{m\times n}$, consider the equation

$$\mathscr{L}.X + Q = 0. \tag{2.1}$$

Assuming henceforth that Q is in the range space of $\mathscr{L}$, the goal is to find its preimage $X \in \mathbb{R}^{p\times q}$. Define the quadratic map $h : \mathbb{R}^{p\times q} \to \mathbb{R}$ via

$$h(X) := \frac{1}{2}\langle \mathscr{L}.X + Q, \mathscr{L}.X + Q\rangle. \tag{2.2}$$

Then the Fréchet derivative of h at $X \in \mathbb{R}^{p\times q}$ acting on $H \in \mathbb{R}^{p\times q}$ is given by

$$h'(X).H = \langle \mathscr{L}.H, \mathscr{L}.X + Q\rangle.$$

By the Riesz representation theorem, the gradient of h can be expressed as

$$\nabla h(X) = \mathscr{L}^\top. (\mathscr{L}.X + Q) \tag{2.3}$$

where $\mathscr{L}^\top : \mathbb{R}^{m\times n} \to \mathbb{R}^{p\times q}$ refers to the adjoint of the operator $\mathscr{L}$.

It is easy to argue that $h(X)$ is convex. Thus, the optimal objective value is zero and is unique. Any local minimizer is also a global minimizer. Indeed, the formulation (2.2) is precisely the typical linear least squares setting and the equation

$$\mathscr{L}^\top. (\mathscr{L}.X + Q) = 0 \tag{2.4}$$

is precisely the corresponding normal equation. The only difference is that the equation involves order-4 tensors and their actions on matrices. For this reason, we refer to (2.4) as the generalized normal equation.

Solving a least squares problem by means of a normal equation usually is regarded as a poor way for computation. When dealing with a general linear system whose unknown is a matrix, however, the lack of commutativity makes the conventional manipulations such as Gaussian elimination or other types of factorization even harder. For this reason, linear matrix equations are often handled individually by specifically designed algorithms. Existing results are widely scattered across different fields. A good discussion on general theory and some algorithms for matrix equations can be found in the books [16, 23]. In contrast, the generalized normal equation for a tensor equation is fairly easy to formulate. This is due to, in particular, the action of the adjoint operator is often a straightforward calculation. There is no need to concern about vectorization The gradient flow we propose offers a unified framework which does not involved inversion or factorization.

The gradient flow $X(t)$ is defined by the dynamical system

$$\frac{dX}{dt} = -\mathscr{L}^\top. (\mathscr{L}.X + Q). \tag{2.5}$$

Trivially, it can be seen that

$$\frac{dh(X(t))}{dt} = -\|\mathscr{L}^\top.(\mathscr{L}.X(t) + Q)\|_F^2,$$

implying that $X(t)$ converges to a global minimizer of $h(X)$. If the equation (2.1) does have a solution to begin with, as we have assumed, then the stationary point of (2.5) is a solution.

At first glance, the differential equation (2.5) is a linear system with constant coefficient. Keep in mind, however, that this constant coefficient in our context is an order-4 tensor whose action is more than just ordinary matrix multiplications. Though mathematically equivalent, we do not wish to invoke the vectorization. The convergence of the solution $X(t)$ should be easy to analyze via the variation of constants formula

$$X(t) = e^{-\mathscr{L}^\top.\mathscr{L}t}.X(0) - \left(\int_0^t e^{-\mathscr{L}^\top.\mathscr{L}(t-s)} ds\right).\mathscr{L}^\top Q. \tag{2.6}$$

where $e^{-\mathscr{L}^\top.\mathscr{L}t}$ should be interpreted as the semi-group generated by the operator $\mathscr{L}^\top.\mathscr{L}$ over $\mathbb{R}^{p\times q}$. For a stable linear system, the rate of convergence is typically measured by the notion of spectral gap

$$\mu := \min\left\{|\Re\lambda| \mid \lambda \text{ is an eigenvalue with nonzero real part}\right\}.$$

In our application the eigenvalue of the "positive semi-definite matrix" $\mathscr{L}^\top.\mathscr{L}$ in the sense of

$$\left(\mathscr{L}^\top.\mathscr{L}\right).Z = \lambda Z$$

for some nonzero $Z \in \mathbb{R}^{p\times q}$ are necessarily real and nonnegative. We shall work out some popular linear matrix equations to demonstrate the concepts mentioned above.

Example 1. Consider the generalized Sylvester equation

$$AXB + CXD + Q = 0 \tag{2.7}$$

where $A, C \in \mathbb{R}^{m\times p}$, $B, D \in \mathbb{R}^{q\times n}$, and $Q \in \mathbb{R}^{m\times n}$ are given. This problem is general enough to include the eight special cases listed under the subheading “generalized Sylvester" in Table 4.1 which arise in various important applications. For solvability, we assume a general minimum requirement that $mn \leq pq$, though in practice there are other restrictions, such as the sizes, symmetry, or positive definiteness, on the constant matrices. Some are listed in Table 4.1.

The linear operator $\mathscr{L} : \mathbb{R}^{p\times q} \longrightarrow \mathbb{R}^{m\times n}$ can be thought of as

$$\mathscr{L}.X = AXB + CXD.$$

Using the relationship (1.3), we can easily verify that the adjoint $\mathscr{L}^\top : \mathbb{R}^{m\times n} \to \mathbb{R}^{p\times q}$ acting on an arbitrary $G \in \mathbb{R}^{m\times n}$ is given by

$$\mathscr{L}^\top.G = A^\top GB^\top + C^\top GD^\top.$$

The gradient flow corresponding to (2.5) therefore is characterized by the differential system

$$\frac{dX}{dt} = -\left(A^\top(AXB + CXD + Q)B^\top + C^\top(AXB + CXD + Q)D^\top\right). \tag{2.8}$$

We propose to solve the generalized normal equation (2.4) associated with the linear system (2.7) by tracking the integral curve of the matrix differential equation (2.8) until reaching an equilibrium point which then is a solution to the generalized Sylvester equation (2.7). While the global convergence is guaranteed, note that no factorization or inversion is needed, except for eight matrix-to-matrix multiplications whenever X is updated. In this particular setting, the flow (2.7) serves as a unified scheme for handling all eight cases listed under generalized Sylvester in Table 4.1 each of which has received considerable interest in the literature and often algorithms are designed individually.

We are not interested in vectorizing (2.8) via the Kronecker product since, by doing that way, the constant coefficient “tensor" $\mathscr{L}^\top.\mathscr{L}$ becomes the $pq \times pq$ matrix $BB^\top \otimes A^\top A + BD^\top \otimes A^\top C + DB^\top \otimes C^\top A + DD^\top \otimes C^\top C$ which is considerably more involved. We intend to integrate (2.8) as it is. Indeed, our primary point is to solve the linear matrix equation (2.1) by integrating the generalized normal flow (2.5) in its natural form without vectorization.

3. Convergence rate. For a general discussion on spectral problems on matrix equations, we refer to the book [30]. But for the gradient dynamics (2.5), we can do a quick analysis as follows.

First, it is clear that the vector field defined in (2.5) is analytic in the variable X. Being an analytic gradient flow, the isolation of limit points is guaranteed by using the Łojasiewicz inequalities [8, 29].

To see how fast the flow converges to the limit point, note that the extreme values defined by

$$\sigma(\mathscr{L}^\top) := \max_{0\neq G\in\mathbb{R}^{m\times n}} \frac{\|\mathscr{L}^\top.G\|_F}{\|G\|_F}, \tag{3.1}$$

$$\theta(\mathscr{L}^\top) := \min_{0\neq G\in\mathbb{R}^{m\times n}} \frac{\|\mathscr{L}^\top.G\|_F}{\|G\|_F} \tag{3.2}$$

do exist over finite dimensional spaces, according to the Courant-Fischer theorem[1]. It follows that

$$-2\theta^2(\mathscr{L}^\top)h \geq \frac{dh}{dt} \geq -2\sigma^2(\mathscr{L}^\top)h. \tag{3.3}$$

[1] Upon vectorization, the Frobenius norm for matrices is precisely the Euclidean norm for vectors. So $\sigma(\mathscr{L}^\top)$ is precisely the operator norm $\|\mathscr{L}^\top\|$ induced by the 2-norm and is the largest “singular value" of the operator $\mathscr{L}^\top$. Likewise, $\theta(\mathscr{L}^\top)$ corresponds to the smallest singular value. Note also that it is possible that $\theta(\mathscr{L}^\top) = 0$.

The differential inequality implies that

$$e^{-2\theta^2(\mathscr{L}^\top)t}h(X(0)) \geq h(X(t)) \geq e^{-2\sigma^2(\mathscr{L}^\top)t}h(X(0)). \tag{3.4}$$

Thus, in order to achieve a relative improvement $h(X(t)(t) \leq \eta h(X(0))$ for a specified scalar η, we need a minimal length of integration

$$t \geq -\frac{\ln \eta}{2\sigma^2(\mathscr{L}^\top)}. \tag{3.5}$$

If $\theta(\mathscr{L}^\top) \neq 0$, then

$$t \geq -\frac{\ln \eta}{2\theta^2(\mathscr{L}^\top)}. \tag{3.6}$$

guarantees the relative improvement. This estimate suggests that if there is reasonable gap between 0 and $\theta(\mathscr{L}^\top)$, the interval of integration need not be long.

When $\theta(\mathscr{L}^\top) = 0$, the estimate (3.6) is useless. However, a more precise mathematical tool can be employed to analyze the flow (2.5) in greater details. Since some eigenvalues of $\mathscr{L}^\top.\mathscr{L}$ under such a circumstance must be zero, the space spanned by eigenvectors, say, $Z_1, \ldots, Z_k \in \mathbb{R}^{p\times q}$, corresponding to zero eigenvalues forms a center manifold which, in fact, is also the null space of $\mathscr{L}$. By the center manifold theorem [7], we know that the solution flow $X(t)$ can be expressed as

$$X(t) = \widehat{X} + \alpha_1 Z_1 + \ldots \alpha_k Z_k + O(e^{-\mu t})$$

for some scalars $\alpha_1, \ldots, \alpha_k$, where $\widehat{X}$ is a particular solution of (2.1) and μ is the smallest nonzero (positive) eigenvalue of $\mathscr{L}^\top\mathscr{L}$. In other words, $X(t)$ converges exponentially to an equilibrium point that solves (2.1). The rate of convergence is equal to the square of the smallest nonzero “singular value" of the operator $\mathscr{L}$. Note that this statement about rate of convergence remains true in the sense of (3.4) even when $\theta(\mathscr{L}^\top) \neq 0$.

Regardless of the rate of convergence, since the quantity $\mathscr{L}.X + Q \to 0$, the vector field in (2.5) is nearly zero when $t \to \infty$. From a numerical ODE prospective, the numerical integration can generally take significantly large step size to move toward the asymptotically stable limit point. The computational cost is not necessarily expensive.

Example 2. Consider the linear matrix equation

$$AXB + CYD + Q = 0 \tag{3.7}$$

for variables $(X, Y) \in \mathbb{R}^{p\times q} \times \mathbb{R}^{s\times t}$ with fixed $A \in \mathbb{R}^{m\times p}$, $B \in \mathbb{R}^{q\times n}$, $C \in \mathbb{R}^{m\times s}$, $D \in \mathbb{R}^{t\times n}$, and $Q \in \mathbb{R}^{m\times n}$. This problem can be cast as a specially structured Sylvester equation. Assume that the number mn of equations is no greater than the number $pq+\ell s$ of unknowns. Using the product topology and the induced Frobenius norm, we see that the adjoint corresponding to the linear operator $\mathscr{L} : \mathbb{R}^{p\times q} \times \mathbb{R}^{s\times t} \longrightarrow \mathbb{R}^{m\times n}$ defined by

$$\mathscr{L}.(X, Y) = AXB + CYD$$

is given by the map

$$\mathscr{L}^\top.G = \left(A^\top G B^\top, C^\top G D^\top\right).$$

According to the definition (3.1), we can calculate [17]

$$\sigma(\mathscr{L}^\top) = \|\mathscr{L}^\top\| = \sqrt{\|B\otimes A^\top\|_2^2 + \|D \otimes C^\top\|_2^2} = \sqrt{\|A\|_2^2\|B\|_2^2 + \|C\|_2^2\|D\|_2^2},$$

where $\|\cdot\|_2$ stands for the spectral norm and is the largest singular value of the corresponding matrix. Likewise, we have

$$\theta(\mathscr{L}^\top) = \sqrt{\theta^2(A^\top)\theta^2(B) + \theta^2(C^\top)\theta^2(D)}$$

where $\theta(\cdot)$ stands for the smallest singular value of the corresponding matrix. It is possible that $\theta(\mathscr{L}^\top) = 0$, due to the rectangular sizes of these matrices. For instance, if $B \in \mathbb{R}^{q\times n}$ and $D \in \mathbb{R}^{t\times n}$ are such that $q < n$ and $t < n$, then $\theta(B) = \theta(D) = 0$. The actual rate of convergence for $X(t)$, according to the center manifold theorem, is equal to

$$\mu = \min_{\sigma^2_{i_{A^\top}}\sigma^2_{i_B}+\sigma^2_{i_{C^\top}}\sigma^2_{i_D}\neq 0} \sigma^2_{i_{A^\top}}\sigma^2_{i_B} + \sigma^2_{i_{C^\top}}\sigma^2_{i_D},$$

where θ_{i_M} denotes an arbitrary singular values, including zeros, of the matrix M.

4. Linear Matrix Equations. Linear matrix equations arise in a variety of important applications, including control theory [5], completely integrable systems [6], Poisson equation solving, Lie algebra [14], invariant subspace computation, and so on [36]. Research efforts on this topic are extensive and discussions usually are focused on one special type of equation a time. This section contains a collection of problems and, far from being complete, a few major references. It would be of great interest, but mammoth, and is definitely beyond our capacity to review the development of widespread theories and algorithms on linear matrix equations, as the literature is so scattered and the techniques are so many. Instead, so that we can demonstrate the generality of our flow approach, we classify most problems found in the literature into four categories. Although there are a few stand-alone cases not belonging to any of these four categories and it is possible that there might be some others which we have missed, the extension to those cases should be similar.

Listed in Table 4.1 are Type I equations which can be expressed in the form

$$\underbrace{\sum_{i=1}^{k} A_i X B_i}_{\mathscr{L}.X} + Q = 0, \tag{4.1}$$

where the proper dimensions of all matrices are assumed. For the case $k = 2$, namely, the generalized Sylvester equation, the problem can be handled by either the Bartels-Stewart method or the Hessenberg-Schur method, both involving orthogonal similarity transformations, and a companion software package is available [15]. But for the general case, the factorization approach breaks down and little can be said about the general matrix G if (4.1) is reduced to the ordinary form $G\mathbf{x} = \mathbf{c}$ [24]. In contrast, the adjoint operator $\mathscr{L}^\top : \mathbb{R}^{m\times n} \longrightarrow \mathbb{R}^{p\times q}$ is given by

$$\mathscr{L}^\top.G = \sum_{j=1}^{k} A_j^\top G B_j^\top.$$

The gradient flow corresponding to (2.5) therefore is characterized by the differential system

$$\frac{dX}{dt} = -\sum_{j=1}^{k} A_j^\top \left(\sum_{i=1}^{k} A_i X B_i + Q \right) B_j^\top \tag{4.2}$$

whose flow $X(t)$ converges globally to either a solution of (4.1), if the equation is consistent, or the least squares solution. The flow (4.2) requires $4k$ matrix multiplications whenever X is updated and can handle all cases of k.

Sometimes the equation is further structured in practice. For instance, in the Lyapunov equation, both versions of continuous time and discrete time, often the coefficient matrix A is stable and the constant matrix

$$\sum_{i=1}^{k} A_i X B_i + Q = 0$$

$A_i \in \mathbb{R}^{m\times p}$, $B_i \in \mathbb{R}^{q\times n}$, $Q \in \mathbb{R}^{m\times n}$, $X \in \mathbb{R}^{p\times q}$

Equation		Remark	Ref.
Generalized Sylvester $AXB + CXD + Q = 0$		$mn \leq pq$	[15]
Lyapunov	$AX + XA^{\top} + Q = 0$	$m = p = q = n$, $Q = Q^{\top}$	
Generalized Lyapunov	$AXB + B^{\top}XA^{\top} + Q = 0$	$m = n$, $p = q$	
	$AXB + B^{\top}XA^{\top} + Q = 0$	$m = n = p = q$, X = symmetric	[35, 37]
Sylvester	$AX + XD + Q = 0$	$m = p$, $n = q$	[5]
Stein	$AXB - X + Q = 0$	$m = p$, $n = q$	[13, 22, 25, 34]
Discrete Lyapunov	$AXA^{\top} - X + Q = 0$	$m = p = q = n$, $Q = Q^{\top}$	
Generalize Discrete Lyapunov	$AXA^{\top} - CXC^{\top} + Q = 0$	$m = n$, $p = q$	
	$AXA^{\top} = B$	$B \succeq 0$, $X \succeq 0$	
$X - \sum_{i=1}^{\alpha} A_i^{\top} X A_i + \sum_{j=1}^{\beta} B_j^{\top} X B_j = I$			

TABLE 4.1
Linear matrix equations: Type I.

Q is symmetric and positive definite. Then the solution X is expected to be unique, symmetric and positive definite. Such a property usually can be realized inherently by the limit point of $X(t)$.

Listed in Table 4.2 are a few Type II equations which vary from those in (4.1) in that the transpose $X^{\top}$ of the unknown variable X also appears in the equations. The general form is expressed as

$$\underbrace{\sum_{i=1}^{k} A_i X B_i + \sum_{j=1}^{\ell} C_j X^{\top} D_j}_{\mathscr{L}.X} + Q = 0 \tag{4.3}$$

where ℓ need not be the same as k. Because of the similarity to the original Sylvester equation, for the ease of comparison we refer to each equation by the original basic name followed by a superscript $^{\top}$. Such a twist, though linear, changes both the computation and the theory significantly [6, 10]. In all, we still can easily formulate the adjoint operator associated with (4.3) as

$$\mathscr{L}^{\top}.G = \sum_{s=1}^{k} A_s^{\top} G B_s^{\top} + \sum_{t=1}^{\ell} D_t G^{\top} C_t$$

$$\sum_{i=1}^{k} A_i X B_i + \sum_{j=1}^{\ell} C_j X^\top D_j + Q = 0$$

$$A_i \in \mathbb{R}^{m\times p}, B_i \in \mathbb{R}^{q\times n}, Q \in \mathbb{R}^{m\times n}, C_j \in \mathbb{R}^{m\times q}, D_j \in \mathbb{R}^{p\times n}, X \in \mathbb{R}^{p\times q}$$

Equation		Remark	Ref.
Generalized Sylvester^T $AXB + CX^\top D + Q = 0$		$mn \leq pq$	
Lyapunov^T	$AX \pm X^\top A^\top + Q = 0$	$m = q = n$	[6, 14]
Generalized Lyapunov^T	$AXB \pm B^\top X^\top A^\top + Q = 0$		[28]
	$AXB \pm A^\top X^\top B^\top = C$		
Sylvester^T	$AX + X^\top D + Q = 0$	$m = q = n$	[10, 18]
Stein^T	$AXB - X^\top + Q = 0$		
Generalized Discrete Lyapunov^T	$AXA^\top \pm BX^\top B^\top + Q = 0$		

TABLE 4.2
Linear matrix equations: Type II.

from which the corresponding gradient flow (2.5) can be described straightforwardly.

Type III equations in the form

$$\underbrace{\sum_{i=1}^{k} A_i X B_i + \sum_{j=1}^{\ell} C_j Y D_j}_{\mathscr{L}.(X,Y)} + Q = 0 \tag{4.4}$$

involve two matrix variables X and Y of different sizes. Filling in zeros, if necessary, we may assume without loss of generality that $k = \ell$. A proper rearrangement

$$\sum_{i=1}^{k} [A_i, C_i] \begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} B_i \\ D_i \end{bmatrix} + Q = 0$$

shows that a Type III equation is a specially structured Type I equation. For this reason we identify each equation by the original basic name followed by a superscript ^{S}, referring to either the structure or the split of the variable $\left[\begin{smallmatrix} X & 0 \\ 0 & Y \end{smallmatrix}\right]$. As such, we mention that there are other types of structured linear matrix equations in the literature, including cases where the solution is limited to a certain subspace or manifold. For constrained linear matrix equations, all we need to do is to modify the gradient into the projected gradient and the idea discussed in this paper prevails. For problem (4.4), we shall regard $\mathscr{L} : \mathbb{R}^{p\times q} \times \mathbb{R}^{s\times t} \longrightarrow \mathbb{R}^{m\times n}$ as a linear map over the product topology equipped with the induced Frobenius inner product

$$\langle (X,Y), (Z,W) \rangle = \langle X, Z \rangle + \langle Y, W \rangle.$$

Then the adjoint $\mathscr{L}^\top : \mathbb{R}^{m\times n} \longrightarrow \mathbb{R}^{p\times q} \times \mathbb{R}^{s\times t}$ is given by

$$\mathscr{L}^\top.G = \left(\sum_{i=1}^{k} A_i^\top G B_i^\top, \sum_{j=1}^{\ell} C_j^\top G D_j^\top \right).$$

The gradient flow is governed by

$$\left\{\begin{array}{rcl} \frac{dX}{dt} & = & -\sum_{i=1}^{k} A_i^\top \left(\sum_{\tau=1}^{k} A_\tau X B_\tau + \sum_{\nu=1}^{\ell} C_\nu Y D_\nu + Q\right) B_i^\top, \\ \frac{dY}{dt} & = & -\sum_{j=1}^{\ell} C_j^\top \left(\sum_{\tau=1}^{k} A_\tau X B_\tau + \sum_{\nu=1}^{\ell} C_\nu Y D_\nu + Q\right) D_j^\top. \end{array}\right. \tag{4.5}$$

$$\sum_{i=1}^{k} A_i X B_i + \sum_{j=1}^{\ell} C_j Y D_j + Q = 0$$

$$A_i \in \mathbb{R}^{m\times p},\, B_i \in \mathbb{R}^{q\times n},\, Q \in \mathbb{R}^{m\times n},\, C_j \in \mathbb{R}^{m\times s},\, D_j \in \mathbb{R}^{t\times n},\, X \in \mathbb{R}^{p\times q},\, Y \in \mathbb{R}^{s\times t}$$

Equation		Remark	Ref.
Generalized Sylvester^S $AXB + CYD + Q = 0$		$mn \leq pq$	[27, 31, 32, 38]
Lyapunov^S	$AX + YA^\top + Q = 0$		
	$AX + YA + Q = 0$		
Sylvester^S	$AX + YD + Q = 0$		[3, 33, 42]
Generalized Discrete Lyapunov^S	$AXA^\top + BYB^\top + Q = 0$		[26]

TABLE 4.3
Linear matrix equations: Type III.

Finally, a generalization of Type III equation is the coupled matrix system of the form

$$\sum_{j=1}^{\alpha} A_{ij} X_j B_{ij} + Q_i = 0, \quad i = 1, \ldots, \beta, \tag{4.6}$$

which has been less studied, but begun to receive attention in recent years [11, 12, 39, 40]. Similar to the Type III equations, we shall regard $\mathscr{L} : \mathbb{R}^{p_1\times q_1} \times \ldots \times \mathbb{R}^{p_\alpha \times q_\alpha} \longrightarrow \mathbb{R}^{m_1\times n_1} \times \ldots \times \mathbb{R}^{m_\beta\times n_\beta}$ as one linear map via

$$\mathscr{L}(X_1, \ldots, X_\alpha) = \left(\sum_{j=1}^{\alpha} A_{1j} X_j B_{1j}, \ldots, \sum_{j=1}^{\alpha} A_{\beta j} X_j B_{\beta j}\right) \tag{4.7}$$

where the spaces are equipped with the induced Frobenius inner product. The adjoint equation should be interpreted as

$$\langle \mathscr{L}(X_1, \ldots, X_\alpha), (G_1, \ldots, G_\beta)\rangle = \langle (X_1, \ldots, X_\alpha), \mathscr{L}^\top (G_1, \ldots, G_\beta)\rangle, \tag{4.8}$$

whereas

$$\langle \mathscr{L}(X_1, \ldots, X_\alpha), (G_1, \ldots, G_\beta)\rangle = \sum_{i=1}^{\beta} \langle \sum_{j=1}^{\alpha} A_{ij} X_j B_{ij}, G_i\rangle = \sum_{j=1}^{\alpha} \langle X_j, \sum_{i=1}^{\beta} A_{ij}^\top G_i B_{ij}^\top \rangle.$$

Thus the action of $\mathscr{L}^\top : \mathbb{R}^{m_1\times n_1} \times \ldots \times \mathbb{R}^{m_\beta \times n_\beta} \longrightarrow \mathbb{R}^{p_1\times q_1} \times \ldots \times \mathbb{R}^{p_\alpha\times q_\alpha}$ is given by

$$\mathscr{L}^\top (G_1, \ldots G_\beta) = \left(\sum_{i=1}^{\beta} A_{i1}^\top G_i B_{i1}^\top, \ldots, \sum_{i=1}^{\beta} A_{i\alpha}^\top G_i B_{i\alpha}^\top\right).$$

The gradient flow is governed by the system

$$\frac{dX_j}{dt} = -\sum_{i=1}^{\beta} A_{ij}^{\top} \left(\sum_{\tau=1}^{\alpha} A_{i\tau} X_{\tau} B_{i\tau} + Q_i \right) B_{ij}^{\top}, \quad j = 1, \ldots \alpha. \tag{4.9}$$

$\sum_{j=1}^{\alpha} A_{ij} X_j B_{ij} + Q_i = 0, \quad i = 1, \ldots, \beta$ $A_{ij} \in \mathbb{R}^{m_i \times p_j}, B_i \in \mathbb{R}^{q_j \times n_i}, Q \in \mathbb{R}^{m_i \times n_i}, X_j \in \mathbb{R}^{p_j \times q_j}$

Equation		Remark	Ref.
Generalized Sylvester^C $\begin{cases} AXB + CYD + Q = 0 \\ EXF + RYS + T = 0 \end{cases}$		$mn \leq pq$	[9, 11]
Sylvester^C	$\begin{cases} AX + YD + Q = 0 \\ EX + YS + T = 0 \end{cases}$		[19]

TABLE 4.4
Linear matrix equations: Type IV.

5. Numerical techniques. Because of the important role they play in a wide range of applications, linear matrix equations have been extensively studied in the literature. Many numerical algorithms are available. Often these methods are designed for solving specific types of equations and are effective. At present, we certainly are not in a position to compare the efficiency of the gradient flow approach with these existent techniques. However, the framework we have proposed for gradient flows is still worth theoretical consideration due to its easy formulation and generalization to other applications [1, 2]. Additionally, we are indeed interested in following the flows for practical purpose. In this section, we outline a few possible numerical methods for gradient flows.

As a dynamical system, the gradient flow can be tracked by any available numerical ODE integrator. As reliable as this approach might be, one possible drawback is that these integrators usually work hard in finding uniform precision for all points along the solution curve, whereas what we are mainly interested in is the limit point. Taking advantage of the fact that the limit point is an *isolated* attractor, one possible strategy to reduce the overhead of computation is to set the local tolerance somewhat higher. The idea is to stay near the true trajectory, but not to strive for accuracy per se with the hope that the numerical solution will eventually be attracted to the equilibrium. Note that this strategy does not change the fact that we need theoretically an interval of length estimated by (3.6) for the flow to converge. The “faster" computation is due to the usage of high local error tolerance which might allow larger step sizes and, hence, less calculation efforts for the integration. We find that setting AbsTol $=$ RelTol $= 10^{-2}$ usually is sufficient for serving the purpose of tracking the solution curve. In the long run, the limit point compresses the flow, by its inherent power of attracting, to the equilibrium to within the machine precision.

Another straightforward approach is to apply the explicit Euler method

$$X_{k+1} = X_k - \epsilon_k \mathscr{L}^{\top} (\mathscr{L} X_k + Q) \tag{5.1}$$

with variable step size ϵ_k to be determined by a line search scheme. This amounts to the steepest descent method. It is trivial to see that if a constant step size ϵ is to be used, then ϵ is necessarily limited by

$$0 < \epsilon < \frac{2}{\sigma^2(\mathscr{L}^\top)},$$

implying slow convergence if the largest singular value of $\mathscr{L}^\top$ is large. Instead of trying to reinvent a good step size selection strategy such as that done in [39, 41], we would suggest taking advantage of the variable step size strategy already developed in the numerical ODE literature for the benefits that not only these methods are readily available, but also that these control strategies are proven to ensure precision and stability.

Along the same idea of staying near the trajectory, there are other specifically designed algorithms for following a gradient flow. The so called pseudo-transient continuation [21] applied to our gradient system works as follows. Assuming that an approximate solution X_k has already been computed, one implicit Euler step with step size ϵ_k to (2.5) yields an equation,

$$\left(I + \epsilon_k \mathscr{L}^\top.\mathscr{L}\right).X_{k+1} = X_k - \epsilon_k \mathscr{L}^\top.Q, \tag{5.2}$$

for the next step $\mathbf{X}_{k+1}$. Instead of solving (5.2) to high precision, which itself is another linear tensor equation, one single correction using any reason iteration scheme starting at $\mathbf{X}_k$ is accepted as X_{k+1}. The scheme may be written as

$$X_{k+1} = X_k - \left(\frac{1}{\epsilon_k} I_n + \nabla^2 h(X_k)\right)^{-1} \nabla h(X_k), \tag{5.3}$$

which is a special implicit upwind method popular for computing steady-state solutions in the PDE community. Note that for small values of ϵ_k the scheme (5.3) behaves like a steepest descent method, whereas for large values of ϵ_k it behaves like a Newton iteration. Taking into account the fact that $\nabla h(\mathbf{X})$ should have small norm near the optimal point $\mathbf{X}^*$, the so-called “switched evolution relaxation" strategy for selecting the step sizes, namely,

$$\epsilon_{k+1} = \epsilon_k \frac{\|\nabla h(X_k)\|}{\|\nabla h(X_{k+1})\|}, \tag{5.4}$$

seems to be able to capture the characteristics of being relatively large in the initial phase, and small in the terminal phase of the iteration. The convergence theory and implementation issues can be found in [20].

The treatment in the Ph.D. thesis [4] of the gradient flow is of a similar spirit. From a given point X_k, the algorithm calculates a curve $\boldsymbol{\gamma}_k(t)$, in closed form, that is an approximation to the gradient flow $X(t)$. It then does a search along this curve $\boldsymbol{\gamma}_k(t)$ for a point X_{k+1} that reduces the value of the objective function subject to certain predesignated criteria and then repeats the process until a limit point is found. Under mild assumptions the method is shown to converge to a critical point from any initial point and to converge quadratically in the neighborhood of a solution. After pasting all $\boldsymbol{\gamma}_k$'s together, this algorithm yields a piecewise smooth curve that approximates the gradient flow. It turns out that the approximate curve $\boldsymbol{\gamma}_k(t)$ is precisely the integral curve of the linearized gradient dynamics at X_k. Since our gradient dynamics (2.5) is already linear, such an algorithm applied to our problem leads to $\boldsymbol{\gamma}(t) = X(t)$, resulting in the breakdown of the curve search mechanism.

6. Numerical experiments. In this section, we report numerical results from some preliminary experiments to support our gradient flow approach.

Example 3. As a continuation of Example 1, we compute the expected length of integration needed for reducing the residual $\|\mathscr{L}.X + Q\|_F$ by a factor of 10^{-14} relative to the original error. To fix the idea, we demonstrate the case when $m = p = q = n = 10$ and use the same set of coefficients A, B, C, and D, whenever they appear, for all seven special cases under the genre of generalized Sylvester equation. We generate randomly 200 sets of coefficient matrices in $\mathbb{R}^{10\times 10}$ and compute the corresponding t based on (3.6). Depicted in Figure 6.1 are the boxplots of the logarithm of these lengths for the various equations. There are some outliers, but the inner-quartiles are approximately over the same range.

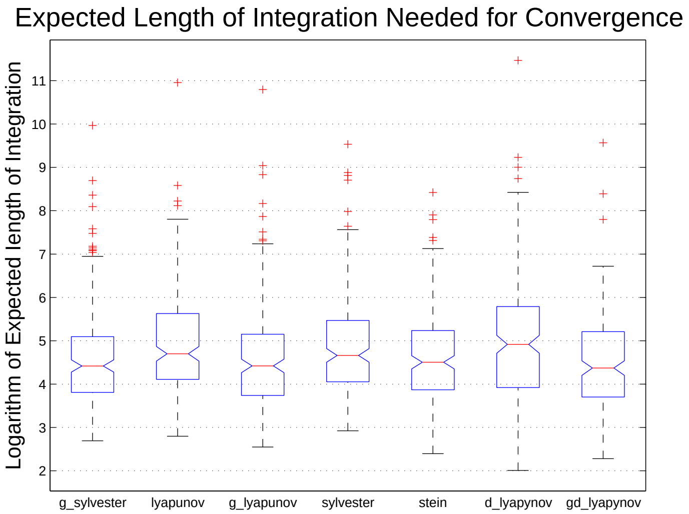

FIGURE 6.1. *Expected lengths of integration for generalized Sylvester equations to reduce residuals relatively by* 10^{-14}.

Example 4 The theoretical results from Example 3 indicate a long interval for integration and seem to suggest the need for lots of steps in order to reach to the equilibrium. In this example, we take a closer look at the application of a variable-step variable-order ODE method available from Matlab to the gradient dynamics (2.8). For the very same set of coefficient matrices, we run the ODE integrator twice with the local error tolerance is set at 10^{-2} and 10^{-10}, respectively. Based on these prescribed local error tolerance, step sizes are selected internally to meet both precision and stability. Since the high tolerance (10^{-2}) scheme computes an approximate solution curve which might drift longer, we allow one and half times of the theoretical estimate interval of integration for the low precision method. Plotted in Figure 6.2 are the number steps needed by the integrator to cover the intervals out of 200 randomly generated sets of coefficient matrices. We observe two important facts. First, even with the low tolerance (10^{-10}) scheme, it takes an order around 10^3 steps/iterations to complete the task of integration. The average step size is fairly large, confirming the statement we made earlier in the paragraph before Example 2. Second, even though the high tolerance scheme is forced to integrate over an interval that is one and half times longer than that for the low tolerance scheme, it still takes significantly fewer steps to complete the task. Indeed, the drawing in Figure 6.2 seems to suggest a linear correlation between steps take for convergence. The linear regression model is approximately $y = 12.4146x + 177.8944$, showing that the high tolerance scheme is using step sizes approximately twelve times larger than the low tolerance scheme.

Example 5. We have mentioned that using existent ODE integrators for the gradient flows has the disadvantage of working too hard in the transit state of acquiring precision for all points along the solution curve, whereas we are only interested in is the limit point. We suggest that using high tolerance scheme might alleviate the computation cost, as is demonstrated in Example 4. But then the question of precision becomes a concern. It is known that if a equilibrium point of a different system is asymptotically stable, then under mild conditions it is also an asymptotically stable equilibrium for perturbed systems nearby. With this in mind, there is a hope that the less precise approximate curve also converges to the same limit point of the original gradient flow. We carry out this experiment with the same 200 sets of random data used in Example 4 and compute the final residuals $\|\mathcal{L}.X + Q\|_F$ at the end of integration. Plotted in Figure 6.3 are the distributions of the absolute errors with the two prescribed local error tolerance 10^{-2} and 10^{-10}. It should be quite convincing that while the high tolerance scheme is considerably cheaper, with around 10^2 iterations, its final precision in finding the solution to the generalized Sylvester equation is surprisingly good. Even the outliers returns a precision at around 10^{-7}.

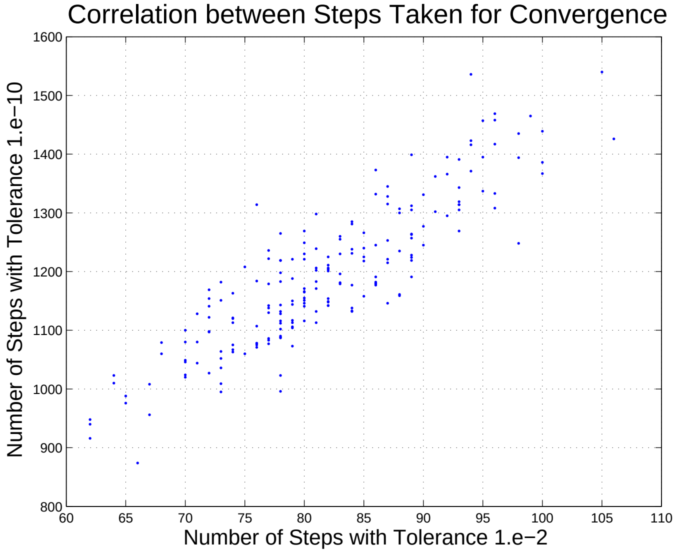

FIGURE 6.2. *Correlation between step taken between* 10^{-2} *and* 10^{-10}.

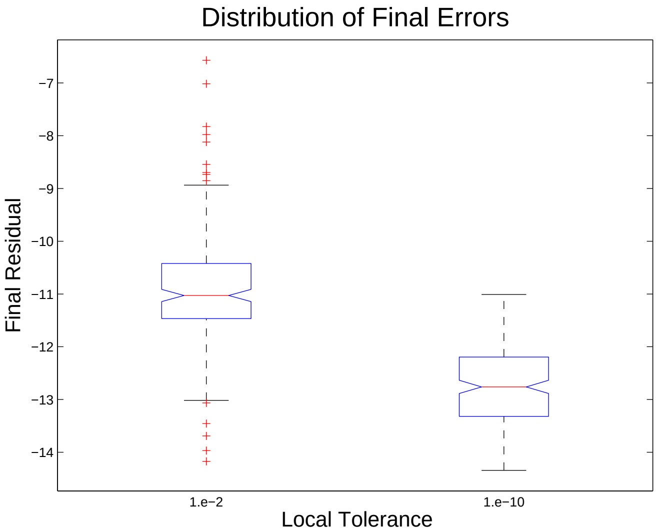

FIGURE 6.3. *Final residuals with step sizes* 10^{-2} *and* 10^{-10}.

7. Conclusion. Linear matrix equations can be regarded as order-4 linear tensor equations. The key ingredient in linear theory of computing the “adjoint" of a tensor operator is readily available. We thus propose a unifying framework that can handle almost all linear matrix equations via the notion of generalized normal equations. Though it is not advisable to solve a classical system of linear equations by its normal equation, as more direct methods such as matrix factorization are more efficient, forming the generalized normal equation for a matrix equation has the advantages of being straightforward, being uniform, working directly with the original sizes without Kronecker product or vectorization, avoiding inversion or factorization, and being easy to analyze the convergence behavior. This paper outlines the theory, exemplifies a collection of applications, suggests a few numerical procedures, and reports some numerical evidences.

REFERENCES

[1] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, Princeton, NJ, 2008.

[2] L. Ambrosio, N. Gigli, and G. Savaré, *Gradient flows in metric spaces and in the space of probability measures*, Lectures in Mathematics ETH Zürich, Birkhäuser Verlag, Basel, second ed., 2008.

[3] J. K. Baksalary and R. Kala, *The matrix equation $AX - YB = C$*, Linear Algebra Appl., 25 (1979), pp. 41–43.

[4] W. Behrman, *An efficient gradient flow method for unconstrained optimization*, PhD thesis, Stanford University, 1998.

[5] R. Bhatia and P. Rosenthal, *How and why to solve the operator equation $AX - XB = Y$*, Bull. London Math. Soc., 29 (1997), pp. 1–21.

[6] H. W. Braden, *The equations $A^TX \pm X^TA = B$*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 295–302 (electronic).

[7] J. Carr, *Applications of centre manifold theory*, vol. 35 of Applied Mathematical Sciences, Springer-Verlag, New York, 1981.

[8] R. Chill, *On the Łojasiewicz-Simon gradient inequality*, J. Funct. Anal., 201 (2003), pp. 572–601.

[9] H. Dai and P. Lancaster, *Linear matrix equations from an inverse problem of vibration theory*, Linear Algebra Appl., 246 (1996), pp. 31–47.

[10] F. De Terán, F. M. Dopico, N. Guillery, D. Montealegre, and N. Reyes, *The solution of the equation $AX+X^{\star}B = 0$*, Linear Algebra Appl., 438 (2013), pp. 2817–2860.

[11] M. Dehghan and M. Hajarian, *An efficient algorithm for solving general coupled matrix equations and its application*, Math. Comput. Modelling, 51 (2010), pp. 1118–1134.

[12] F. Ding and T. Chen, *On iterative solutions of general coupled matrix equations*, SIAM J. Control Optim., 44 (2006), pp. 2269–2284 (electronic).

[13] P. A. Fuhrmann, *A functional approach to the Stein equation*, Linear Algebra Appl., 432 (2010), pp. 3031–3071.

[14] S. R. Garcia and A. L. Shoemaker, *On the matrix equation $XA + AX^T = 0$*, Linear Algebra Appl., 438 (2013), pp. 2740–2746.

[15] J. D. Gardiner, A. J. Laub, J. J. Amato, and C. B. Moler, *Solution of the Sylvester matrix equation $AXB^\top + CXD^\top = E$*, ACM Trans. Math. Software, 18 (1992), pp. 223–231.

[16] N. J. Higham, *Functions of matrices*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. Theory and computation.

[17] R. A. Horn and C. R. Johnson, *Topics in matrix analysis*, Cambridge University Press, Cambridge, 1991.

[18] K. D. Ikramov, *On conditions for the unique solvability of the matrix equation $AX + X^TB = C$*, Dokl. Akad. Nauk, 430 (2010), pp. 444–447.

[19] B. Kågström, *A perturbation analysis of the generalized Sylvester equation $(AR-LB, DR-LE) = (C, F)$*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 1045–1060.

[20] C. T. Kelley and D. E. Keyes, *Convergence analysis of pseudo-transient continuation*, SIAM J. Numer. Anal., 35 (1998), pp. 508–523 (electronic).

[21] C. T. Kelley, L.-Z. Liao, L. Qi, M. T. Chu, J. P. Reese, and C. Winton, *Projected pseudotransient continuation*, SIAM J. Numer. Anal., 46 (2008), pp. 3071–3083.

[22] A. Klein and P. Spreij, *On Stein's equation, Vandermonde matrices and Fisher's information matrix of time series processes. I. The autoregressive moving average process*, Linear Algebra Appl., 329 (2001), pp. 9–47.

[23] M. Konstantinov, D.-W. Gu, V. Mehrmann, and P. Petkov, *Perturbation theory for matrix equations*, vol. 9 of Studies in Computational Mathematics, North-Holland Publishing Co., Amsterdam, 2003.

[24] P. Lancaster, *Explicit solutions of linear matrix equations*, SIAM Rev., 12 (1970), pp. 544–566.

[25] L. Lerer and A. C. M. Ran, *A new inertia theorem for Stein equations, inertia of invertible Hermitian block Toeplitz matrices and matrix orthogonal polynomials*, Integral Equations Operator Theory, 47 (2003), pp. 339–360.

[26] A.-P. Liao and Z.-Z. Bai, *Least squares symmetric and skew-symmetric solutions of the matrix equation $AXA^T + BYB^T = C$ with the least norm*, Math. Numer. Sin., 27 (2005), pp. 81–95.

[27] A.-P. Liao, Z.-Z. Bai, and Y. Lei, *Best approximate solution of matrix equation $AXB + CYD = E$*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 675–688.

[28] A.-P. Liao, Y. Lei, and X.-Y. Hu, *Least-squares solution with the minimum-norm for the matrix equation $A^TXB+B^TX^TA = D$ and its applications*, Acta Math. Appl. Sin. Engl. Ser., 23 (2007), pp. 269–280.

[29] S. Łojasiewicz and M.-A. Zurro, *On the gradient inequality*, Bull. Polish Acad. Sci. Math., 47 (1999), pp. 143–145.

[30] A. G. Mazko, *Matrix equations, spectral problems and stability of dynamic systems*, vol. 2 of Stability, Oscillations and Optimization of Systems, Cambridge Scientific Publishers, Cambridge, 2008.

[31] A. B. Özgüler, *The equation $AXB + CYD = E$ over a principal ideal domain*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 581–591.

[32] Z.-Y. Peng and Y.-X. Peng, *An efficient iterative method for solving the matrix equation $AXB + CYD = E$*, Numer. Linear Algebra Appl., 13 (2006), pp. 473–485.

[33] W. E. Roth, *The equations $AX - YB = C$ and $AX - XB = C$ in matrices*, Proc. Amer. Math. Soc., 3 (1952), pp. 392–396.

[34] F. C. Silva and R. Simões, *On the Lyapunov and Stein equations. II*, Linear Algebra Appl., 426 (2007), pp. 305–311.

[35] M. J. Todd, *Semidefinite optimization*, Acta Numer., 10 (2001), pp. 515–560.

[36] C. F. Van Loan, *The ubiquitous Kronecker product*, J. Comput. Appl. Math., 123 (2000), pp. 85–100. Numerical analysis 2000, Vol. III. Linear algebra.

[37] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[38] G. Xu, M. Wei, and D. Zheng, *On solutions of matrix equation $AXB + CYD = F$*, Linear Algebra Appl., 279 (1998), pp. 93–109.

[39] J.-J. Zhang, *A note on the iterative solutions of general coupled matrix equation*, Appl. Math. Comput., 217 (2011), pp. 9380–9386.

[40] B. Zhou, G.-R. Duan, and Z.-Y. Li, *Gradient based iterative algorithm for solving coupled matrix equations*, Systems Control Lett., 58 (2009), pp. 327–333.

[41] B. Zhou, J. Lam, and G.-R. Duan, *Gradient-based maximal convergence rate iterative method for solving linear matrix equations*, Int. J. Comput. Math., 87 (2010), pp. 515–527.

[42] K. Ziętak, *On a particular case of the inconsistent linear matrix equation $AX + YB = C$*, Linear Algebra Appl., 66 (1985), pp. 249–258.