# A Natural Approach to the Numerical Integration of Riccati Differential Equations

Jeremy Schiff[1] and S. Shnider[2]
Department of Mathematics and Computer Science
Bar Ilan University
Ramat Gan 52900, Israel
[1]e-mail: schiff@math.biu.ac.il
[2]e-mail: shnider@math.biu.ac.il

Revised version, August 1998

**Abstract**

This paper introduces a new class of methods, which we call Möbius schemes, for the numerical solution of matrix Riccati differential equations. The approach is based on viewing the Riccati equation in its natural geometric setting, as a flow on the Grassmanian of $m$-dimensional subspaces of an $(n+m)$-dimensional vector space. Since the Grassmanians are compact differentiable manifolds, and the coefficients of the equation are assumed continuous, there are no singularities or intrinsic instabilities in the associated flow. The presence of singularities and numerical instabilitites is an artefact of the coordinate system, but since Möbius schemes are based on the natural geometry, they are able to deal with numerical instability and pass accurately through the singularities. A number of examples are given to demonstrate these properties.

## 1 Introduction

The matrix Riccati differential equation [1] is the equation

$$\dot{y} = a(t)y + b(t) - yc(t)y - yd(t), \tag{1}$$

where the unknown $y(t)$ is an $n \times m$ matrix function, and the known coefficients $a(t), b(t), c(t)$, $d(t)$ are $n \times n$, $n \times m$, $m \times n$, and $m \times m$ matrix functions respectively. The coefficient functions are all assumed continuous in the interval of interest, and, where required, differentiable to the appropriate order. One of the properties of Riccati equations is the existence of movable singularities, i.e. singularities whose position depends on the initial conditions. In applications to boundary value problems [2] it may be necessary to integrate through singularities.

Our aim in this paper is to show that the initial value problem for (1) can be effectively integrated *even through singularities* via explicit, one-step numerical schemes of the form

$$y_{i+1} = \left(\tilde{\alpha}(t_i,h)y_i + \tilde{\beta}(t_i,h)\right)\left(\tilde{\gamma}(t_i,h)y_i + \tilde{\delta}(t_i,h)\right)^{-1} . \tag{2}$$

Here $y_i$ is an approximation to $y(t_i)$ and (2) specifies how to construct an approximation $y_{i+1}$ to $y(t_{i+1})$, where $t_{i+1} = t_i + h$. The functions $\tilde{\alpha}(t_i,h)$,$\tilde{\beta}(t_i,h)$, $\tilde{\gamma}(t_i,h)$,$\tilde{\delta}(t_i,h)$ are constructed

from the coefficient functions $a(t), b(t), c(t), d(t)$ by formulae of the form:

$$\begin{aligned} \tilde{\alpha}(t_i, h) &= \mathrm{I}_n + ha(t_i) + o(h) \\ \tilde{\beta}(t_i, h) &= hb(t_i) + o(h) \\ \tilde{\gamma}(t_i, h) &= hc(t_i) + o(h) \\ \tilde{\delta}(t_i, h) &= \mathrm{I}_m + hd(t_i) + o(h) \end{aligned} \tag{3}$$

($\mathrm{I}_n$ denotes the $n \times n$ identity matrix). Schemes of this type can be constructed with arbitrary order. Moreover, by correct construction of $\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\delta}$ from $a, b, c, d$, problems of stiffness can be avoided.

It is known that one can integrate through singularities either by changing coordinates [3], or by integrating a larger linear system associated with the Riccati equation. Recursion formulae of type (2), which we call Möbius schemes (since they use generalized Möbius transformations), accomplish this directly in the original variables. Substantial attention has been paid in the literature to the numerical integration of Riccati equations: see [4], [5], [6], [7], [8], [9] and references therein. Despite this, it seems the only scheme of the form (2) that has previously been used is the modified Davison-Maki method of Kenney and Leipnik [5]. Had they used their method to integrate through a singularity they would have succeeded, and may well have developed the ideas we describe here. In [3], Keller and Lentini also noticed a recursion of the form (2) arising naturally as an iterative scheme for solving Riccati equations. We emphasize that standard approaches (e.g. Runge-Kutta methods) applied to Riccati equations are not of the form (2), and this is why they fail to integrate through singularities.

A full explanation of the rationale for Möbius schemes and why they can pass singularities, requires a geometric viewpoint, which we present in section 2. (This geometric viewpoint is in fact necessary to understand in what sense the solution of a Riccati equation can be extended through a singularity.) However from a conventional viewpoint, we can understand Möbius schemes as an efficient method of solving via linearization (cf. [3]). Recall that if the $n \times m$ matrix $u(t)$ and the $m \times m$ matrix $v(t)$ solve the linear system

$$\begin{pmatrix} \dot{u(t)} \\ v(t) \end{pmatrix} = \begin{pmatrix} a(t) & b(t) \\ c(t) & d(t) \end{pmatrix} \begin{pmatrix} u(t) \\ v(t) \end{pmatrix}, \tag{4}$$

then $y(t) = u(t)v(t)^{-1}$ solves the Riccati equation (1). Now, since (4) is a linear system, any of the standard numerical methods (explicit or implicit) for integration also has linear form

$$\begin{pmatrix} u_{i+1} \\ v_{i+1} \end{pmatrix} = \begin{pmatrix} \tilde{\alpha}(t_i, h) & \tilde{\beta}(t_i, h) \\ \tilde{\gamma}(t_i, h) & \tilde{\delta}(t_i, h) \end{pmatrix} \begin{pmatrix} u_i \\ v_i \end{pmatrix}, \tag{5}$$

where $\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\delta}$ obey the conditions (3). Defining $y_i = u_i v_i^{-1}$, an elementary manipulation gives the formula (2) for $y_i$. Thus we evidently can work with (2), at least away from singularities. The main point of this paper, as will be presented in section 2, is that there is a much deeper geometric reason for using Möbius schemes, which gives them much wider applicability.

The contents of the rest of this paper are as follows: we conclude the introduction by giving a simple example of the use of a Möbius scheme to integrate a Riccati equation through a singularity, and the application of this to a boundary value problem. (Apart from here, in the rest of the paper we focus on the initial value problem for the Riccati equation, and

do not discuss at the moment applications to linear boundary value problems from which Riccati problems arise.) Section 2 describes the geometry behind Möbius schemes. In section 3 we discuss the construction of methods of the form (2), i.e. how to chose $\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\delta}$ from given $a, b, c, d$, and in particular how to do this to avoid stiffness. Section 4 explores the issue of stiffness in greater detail. Sections 5 and 6 report the results of application of Möbius schemes: In section 5 we look at four problems, taken from [8], two of which are autonomous and two of which are time dependent, and for which, in all cases, we are seeking nonsingular solutions (though as we shall see, singularities are often lurking nearby). In section 6 we reconsider the integration of two of the problems from section 5 in the case where we are seeking singular solutions. Section 7 contains some concluding comments.

**A Simple Example** Consider the boundary value problem

$$\ddot{x} + x = 0, \qquad t \in [0, L], \quad x(0) = 0, \quad \dot{x}(L) = 1 \ . \tag{6}$$

Provided $L \neq (n + \frac{1}{2})\pi$ for some integer $n$, there is a unique solution $x(t) = \sin t / \cos L$. In the invariant imbedding approach to this problem, we first reformulate the equation as a first order system; writing, say $u(t) = x(t)$, $v(t) = \dot{x}(t)$, we have

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \ , \qquad u(0) = 0, \quad v(L) = 1 \ . \tag{7}$$

Introducing now the function $y(t)$ defined, where $v(t) \neq 0$, by $u(t) = y(t)v(t)$, elementary manipulations show we can solve the problem by the following steps:

1. Integrate the Riccati problem

$$\dot{y} = y^2 + 1, \qquad\qquad y(0) = 0, \tag{8}$$

forwards from $t = 0$ to $t = L$, to find $y(t)$.

2. Integrate the linear problem

$$\dot{v} = -yv, \qquad\qquad v(L) = 1, \tag{9}$$

backwards from $t = L$ to $t = 0$ to find $v(t)$.

3. Reconstruct $x(t)$ from $x(t) = u(t) = y(t)v(t)$.

The exact solution of the Riccati problem is $y(t) = \tan t$, so this approach is usually only considered valid if $L < \pi/2$; otherwise a singularity must be passed (the sense in which $y(t) = \tan t$ is the unique solution of the initial value problem for all $t$ will become clear in section 2). Standard numerical methods cannot pass the singularity; for example the Euler method and the backward Euler method with stepsize $h$ give, respectively, the recursions

$$y_{i+1} = y_i + h(y_i^2 + 1), \qquad y_0 = 0 \tag{10}$$

$$y_{i+1} = \frac{1}{2h}\left(1 - \sqrt{1 - 4hy_i - 4h^2}\right), \qquad y_0 = 0 \tag{11}$$

which evidently fail (the first because it defines a monotonically increasing sequence $\{y_i\}$, and the second because the recursion is not defined for $y_i > 1/4h$). The first order Möbius scheme for Riccati equations, that we will define later on, leads, however, to the recursion

$$y_{i+1} = \frac{y_i + h}{1 - hy_i}, \qquad\qquad y_0 = 0. \tag{12}$$

Since this expresses the fact that $y_{i+1}$ is obtained from $y_i$ by an $i$-independent Möbius transformation, it is straightforward to solve this recursion explicitly, obtaining

$$y_i = \frac{1}{\sqrt{-1}} \frac{(1+\sqrt{-1}h)^i - (1-\sqrt{-1}h)^i}{(1+\sqrt{-1}h)^i + (1-\sqrt{-1}h)^i} = \tan(i \tan^{-1} h). \tag{13}$$

Setting $i = t/h$, we have the numerical solution using stepsize $h$:

$$y_h(t) = \tan r_h t, \qquad r_h = (\tan^{-1} h)/h \approx 1 - h^2/3. \tag{14}$$

*For small $h$ this solution is evidently both qualitatively and quantitatively accurate through many singularities of the solution.* Furthermore, we can use it to solve the original boundary value problem; using the Euler method for the linear problem (9) gives the recursion

$$v_{i+1} = (1 - h y_i) v_i, \qquad v_N = 1 \ , \tag{15}$$

where we have assumed $L = Nh$ for some integer $N$. This can be explicitly solve to give

$$v_i = \frac{\cos(i \tan^{-1} h)}{(1+h^2)^{(N-i)/2} \cos(N \tan^{-1} h)} \ , \tag{16}$$

giving a numerical solution of the original boundary value problem

$$x_h(t) = \frac{\sin r_h t}{(1+h^2)^{(L-t)/2h} \cos r_h L} \ . \tag{17}$$

For small $h$ the validity of this result is not restricted to $L < \pi/2$.

The reader will have no problem programming the recursions (12) and (15) and checking there are no numerical stability problems for generic $L$ and $h$. It is possible that a problem could arise that, for some $i$, $y_i = \tan(i \tan^{-1} h)$ may be too large for the computer to handle properly, giving an overflow error. In practice we have not encountered problems of this nature. Note that close to a singularity there are almost always large absolute errors in the computed $y_i$, compared to the corresponding exact values $y(ih)$. These are to be expected, but do not impair the accuracy once the singularity has been passed, as will be explained in the next section.

## 2 Riccati Equations as Flows on A Grassmanian

An $n \times m$ matrix $y$ defines an $m$-dimensional subspace of $\mathbf{R}^{n+m}$; denoting the coordinates of $\mathbf{R}^{n+m}$ by $z_1, \ldots, z_{n+m}$, the subspace associated with $y$ is the space of solutions of the $n$ equations

$$\begin{pmatrix} z_1 \\ \vdots \\ z_n \end{pmatrix} = y \cdot \begin{pmatrix} z_{n+1} \\ \vdots \\ z_{n+m} \end{pmatrix} \ . \tag{18}$$

Not all $m$-dimensional subspaces of $\mathbf{R}^{n+m}$ arise this way, but a dense open subset of the collection of all such subspaces does. This collection forms a topological space, in fact a differentiable manifold, known as the Grassmanian $Gr(m, m+n)$. The topology of the Grassmanian can be obtained by making it into a metric space using the distance function

$$d(p_1, p_2) = \sup_{u \in p_1, \ ||u||=1} \ \sup_{v \in p_2, \ ||v||=1} ||u - v|| \ , \tag{19}$$

where $||\cdot||$ denotes the standard Euclidean norm on $\mathbf{R}^{n+m}$. The manifold structure of $Gr(m, m+n)$ comes from its representation as $O(n+m)/O(n) \times O(m)$, which also demonstrates that it is compact. [10]

There is a natural action of $GL(m+n)$ on $\mathbf{R}^{m+n}$, and this induces a $GL(m+n)$ action on $Gr(m, m+n)$. Let us first consider the effect of an $GL(m+n)$ transformations infinitesimally close to the identity:

$$\begin{pmatrix} z_1 \\ \vdots \\ z_{n+m} \end{pmatrix} \to \begin{pmatrix} z'_1 \\ \vdots \\ z'_{n+m} \end{pmatrix} = \begin{pmatrix} z_1 \\ \vdots \\ z_{n+m} \end{pmatrix} + \epsilon \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} z_1 \\ \vdots \\ z_{n+m} \end{pmatrix} , \tag{20}$$

where $a, b, c, d$ are $n \times n$, $n \times m$, $m \times n$, and $m \times m$ matrices respectively. Assuming that the point $(z_1, \ldots, z_{n+m})$ lies in the subspace defined by $y$ (i.e. the coordinates satisfy (18)), a brief calculation (ignoring $O(\epsilon^2)$) shows that the point $(z'_1, \ldots, z'_{n+m})$ lies in the subspace defined by $y'$, where

$$y' = y + \epsilon(ay + b - ycy - yd) . \tag{21}$$

We thus have the classic result (see, for example, [12]) that *the Riccati equation corresponds to a flow by infinitesimal $GL(m+n)$ transformations on the Grassmanian $Gr(m, m+n)$.*

A similar calculation gives the full $GL(m+n)$ action on the Grassmanian. A general $GL(m+n)$ transformation on $\mathbf{R}^{m+n}$ takes the form

$$\begin{pmatrix} z_1 \\ \vdots \\ z_{n+m} \end{pmatrix} \to \begin{pmatrix} z'_1 \\ \vdots \\ z'_{n+m} \end{pmatrix} = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} z_1 \\ \vdots \\ z_{n+m} \end{pmatrix} , \tag{22}$$

where $\alpha, \beta, \gamma, \delta$ are $n \times n$, $n \times m$, $m \times n$, and $m \times m$ matrices respectively, with $\det \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \neq 0$. The relation (18) implies

$$\begin{pmatrix} z'_1 \\ \vdots \\ z'_n \end{pmatrix} = y' \cdot \begin{pmatrix} z'_{n+1} \\ \vdots \\ z'_{n+m} \end{pmatrix} . \tag{23}$$

where

$$y' = (\alpha y + \beta)(\gamma y + \delta)^{-1} , \tag{24}$$

provided $\det(\gamma y + \delta) \neq 0$. If $\det(\gamma y + \delta) = 0$, this does not mean the action of the $GL(m+n)$ transformation $\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$ is not defined on the plane $y$; it is defined, but the image point does not lie in the dense open subset of $Gr(m, m+n)$ which can be written in the form (18).

We now combine the results of the last two paragraphs. Since the Riccati equation corresponds to a flow by infinitesimal $GL(m+n)$ transformations, the solution at time $t_0 + h$ must be given by a transformation of the form (24) on the solution at time $t_0$, In other words, there must exist matrices $\alpha(t_0, h), \beta(t_0, h), \gamma(t_0, h), \delta(t_0, h)$ such that

$$y(t_0 + h) = (\alpha(t_0, h)y(t_0) + \beta(t_0, h))(\gamma(t_0, h)y(t_0) + \delta(t_0, h))^{-1} , \tag{25}$$

The matrices $\alpha(t_0, h), \beta(t_0, h), \gamma(t_0, h), \delta(t_0, h)$ define a generalized Möbius transformation that generates the solution at $t_0 + h$ from the solution at $t_0$, *independent of the value of $y(t_0)$*. This provides the justification for (2).

The flow on $Gr(m, m+n)$ underlying the Riccati equation should be considered as the fundamental object. Since it is the flow associated with a continuous vector field on a compact manifold ($a(t), b(t), c(t), d(t)$ are assumed to be continuous), it can be extended for all time [11], and there are no singularities. The singularities in solutions of the Riccati equation for $y$ arise simply as result of the flow leaving the dense open subset where the "local coordinate" $y$ can be used, corresponding to values of $h$ in (25) for which $\det(\gamma(t_0, h)y(t_0) + \delta(t_0, h)) = 0$. Generically, the vanishing of the determinant occurs only for isolated values of $h$, and consequently *generic solutions of Riccati equations have at most pointlike singularities, and can be extended past these.* Formula (25) specifies how to do this: it is valid even when there is a singularity in the interval $(t, t+h)$.

This explains why, in principle, Möbius schemes of the form (2) can jump singlarities, provided we can adequately approximate $\alpha(t_0, h), \beta(t_0, h), \gamma(t_0, h), \delta(t_0, h)$ (this is the topic of the next section). But there remain several concerns that need to be addressed. First, near a singularity, close $y$ orbits must diverge, i.e. there is an intrinsic instability. Like the singularities themselves, *this is a coordinate artefact; the underlying flow on $Gr(m, m+n)$ has no intrinsic instability.* Orbits that are close on $Gr(m, m+n)$ (in a natural metric on $Gr(m, m+n)$, such as (19)) can appear to be diverging when expressed in terms of $y$. This phenomenon was evident in the simple example in the introduction, where large absolute errors in $y$ were present close to the singularity, but disappeared soon thereafter. Thus the intrinsic instability need not concern us; but (2) also involves inversion of a matrix which near a singularity will certainly be ill-conditioned, and we have to worry about numerical instability. (Of course, the chance that the matrix becomes genuinely singular — corresponding to landing exactly on a singularity — is negligible.) In fact, we claim that since Möbius schemes can tolerate large "apparent" errors (deviations in $y$) close to singularities, numerical instability is also of limited concern. Nevertheless, we chose in our numerical work, only to implement Möbius schemes with some form of error control; the effect of this was that when very high accuracy was demanded, our algorithms failed to pass singularities, but for moderate accuracy, excellent results could still be obtained.

We conclude this section with a note on the title of this paper. Because the map from $y_i$ to $y_{i+1}$ described by (2) takes the form of a $y_i$-independent Möbius transformation, which is the known form of the transformation from $y(t_i)$ to $y(t_{i+1})$, Möbius schemes are *natural* schemes for the integration of Riccati equations (in the same way as it is natural to integrate the linear system (4) via linear schemes of the form (5)).

## 3 Construction of Algorithms and Eliminating Stiffness

As explained in section 2, to construct Möbius schemes requires approximation of the matrices $\alpha(t_0, h), \beta(t_0, h), \gamma(t_0, h), \delta(t_0, h)$ appearing in (25). We do this using a linearization result for the Riccati equation that is slightly less well-known than the one presented in Section 1. Assemble the coefficients of the Riccati equation (1) into a single $(n+m) \times (n+m)$ matrix:

$$A(t) := \begin{pmatrix} a(t) & b(t) \\ c(t) & d(t) \end{pmatrix}. \tag{26}$$

If the $(n+m) \times (n+m)$ matrix $\Gamma(t)$ satisfies

$$\dot{\Gamma}(t) = A(t)\Gamma(t), \qquad \Gamma(t_0) = \mathrm{I}_{n+m}, \tag{27}$$

and we decompose $\Gamma(t)$ into sub-blocks of dimensions $n \times n$, $n \times m$, $m \times n$, and $m \times m$ via

$$\Gamma(t) = \begin{pmatrix} \alpha(t) & \beta(t) \\ \gamma(t) & \delta(t) \end{pmatrix}, \tag{28}$$

then it is straightforward to check that

$$y(t) = (\alpha(t)Y + \beta(t))(\gamma(t)Y + \delta(t))^{-1} \tag{29}$$

satisfies the Riccati equation (1), with initial condition $y(t_0) = Y$. The solution to (27) can be written down using the the time ordered exponential [13]:

$$\Gamma(t) = \mathrm{I}_{n+m} + \sum_{r=1}^{\infty} \int \dots \int_{t_0 \le s_1 \le \dots \le s_r \le t} A(s_r) \dots A(s_1) ds_1 \dots ds_r. \tag{30}$$

So, in particular, we have (with a slight modification of notation)

$$y(t_0 + h) = (\alpha(t_0, h)y(t_0) + \beta(t_0, h))(\gamma(t_0, h)y(t_0) + \delta(t_0, h))^{-1}, \tag{31}$$

where

$$\begin{aligned} \Gamma(t_0, h) &:= \begin{pmatrix} \alpha(t_0, h) & \beta(t_0, h) \\ \gamma(t_0, h) & \delta(t_0, h) \end{pmatrix} \\ &= \mathrm{I}_{n+m} + \sum_{r=1}^{\infty} \int \dots \int_{0 \le s_1 \le \dots \le s_r \le h} A(t_0 + s_r) \dots A(t_0 + s_1) ds_1 \dots ds_r. \end{aligned} \tag{32}$$

For the purposes of numerical analysis, we want formulae for approximating $y(t_0 + h)$ given $y(t_0)$, for small $h$. We see this reduces to finding approximations $\tilde{\Gamma}(t_0, h) = \begin{pmatrix} \tilde{\alpha}(t_0, h) & \beta(t_0, h) \\ \tilde{\gamma}(t_0, h) & \tilde{\delta}(t_0, h) \end{pmatrix}$ to the time ordered exponential $\Gamma(t_0, h)$ for small $h$. Any such approximation gives us a Möbius scheme (2). This observation, we note, was already made by Kenney and Leipnik in their construction of the modified Davison-Maki method [5].

Before we construct approximations to $\Gamma(t_0, h)$, we emphasize again that equation (31) is an exact formula, that holds for all $h$ except those for which $\det(\gamma(t_0, h)y(t_0) + \delta(t_0, h)) = 0$ (this relation defines the singular points of the solution). So if we have an approximation $\tilde{\Gamma}(t_0, h)$ to $\Gamma(t_0, h)$, then this can be used to construct an approximation to $y(t_0 + h)$ from an approximation to $y(t_0)$, regardless of the presence of a singularity in the interval $(t_0, t_0 + h)$. *This is the key observation that allows us to integrate through singularities.* There remains the possibility that we might select $h$ such that $\tilde{\gamma}(t_0, h)y(t_0) + \tilde{\delta}(t_0, h)$ is ill-conditioned for inversion, so we must carefully impose an error control procedure.

In (32), the $r$-th term in the sum is of order $h^r$, so to estimate the time ordered exponential to order $h^p$, we need only consider the first $p$ terms of the sum. To obtain a fourth order formula we retain the first four terms in the sum, and estimate each of these to the necessary order using the Taylor series expansion of $A(t + s) - A(t)$. This gives the approximation

$$\begin{aligned} \tilde{\Gamma}_4(t_0, h) &= \mathrm{I}_{n+m} + hA + \tfrac{h^2}{2}(A' + A^2) + \tfrac{h^3}{6}(A'' + 2A'A + AA' + A^3) \\ &+ \tfrac{h^4}{24}\left(A''' + AA'' + 3(A')^2 + 3A''A + 3A'A^2 + 2AA'A + A^2A' + A^4)\right), \end{aligned} \tag{33}$$

where all occurrences of $A$ and its derivatives are evaluated at $t_0$. Truncating at order $h^1$ gives the first order approximation

$$\tilde{\Gamma}_1(t_0, h) = \mathrm{I}_{n+m} + hA(t_0), \tag{34}$$

and truncating at order $h^2$ gives us a second order approximation, which differs only by $O(h^3)$ terms from the simple formula

$$\tilde{\Gamma}_2(t_0, h) = \mathrm{I}_{n+m} + hA(t_0 + \tfrac{h}{2}) + \tfrac{h^2}{2}A(t_0 + \tfrac{h}{2})^2. \tag{35}$$

This latter approximation, which is convenient because no derivatives of $A$ appear, was the main one we used in calculations. A fourth order approximation containing no derivatives, but involving evaluations of $A$ at three different points, can be obtained by polynomial approximating of the integrands in the first four terms of the sum in (32). The first order approximation (34) was used in the "simple example" in the introduction.

At this stage a subtlety arises. Specifying $A(t)$ uniquely determines the related Riccati equation, but the converse is *not* true. Equation (1) is evidently unaffected by the substitutions

$$\begin{aligned} a(t) &\rightarrow a(t) + p(t)\ \mathrm{I}_n \\ d(t) &\rightarrow d(t) + p(t)\ \mathrm{I}_m, \end{aligned} \tag{36}$$

where $p(t)$ is a scalar function. This can be written in the equivalent form

$$A(t) \rightarrow A(t) + p(t)\ \mathrm{I}_{n+m}. \tag{37}$$

We usually specify a particular Riccati equation by giving one possible choice of the matrix $A(t)$; but we should always recall it can be changed in the manner just described. Under the replacement (37),

$$\Gamma(t_0, h) \rightarrow \exp\left(\int_0^h p(t_0 + s)ds\right)\Gamma(t_0, h), \tag{38}$$

i.e. $\Gamma(t_0, h)$ undergoes an overall rescaling; this leaves the $y(t_0+h)$ in equation (31) invariant, as expected. Looking at the approximations $\tilde{\Gamma}_4(t_0, h)$, $\tilde{\Gamma}_2(t_0, h)$, $\tilde{\Gamma}_1(t_0, h)$ introduced above, we see that since we have truncated the infinite sum, changing $A(t)$ by (37) does not just have the effect of an overall rescaling. For example, we find

$$\tilde{\Gamma}_2(t_0, h) \rightarrow (1 + hp + \tfrac{1}{2}h^2p^2)\tilde{\Gamma}_2(t_0, h) - \tfrac{1}{2}h^3pA(p\mathrm{I}_{n+m} + A + \tfrac{1}{2}hpA), \tag{39}$$

where here it is understood that $p$ and $A$ are to be evaluated at $t_0 + \frac{1}{2}h$. It follows that replacement of $A$ via (37) does not leave a given approximation scheme invariant; therefore to fully specify an integration method for a Riccati equation, we have to discuss the choice of $A$.

This freedom turns out to be an enormous advantage. (37) describes a time dependent shift of the spectrum of $A(t)$. It is well-known that for the linear system Eq. (27) the occurrence of stiffness is associated with eigenvalues of $A(t)$ with negative real part and large absolute value, while accuracy is determined, in general, by the largest eigenvalue in absolute value. This suggests that for low accuracy integrations, where stiffness may be an issue, we should translate the spectrum of $A(t)$ so that there are no eigenvalues with negative real part, but for high accuracy, we should shift the spectrum to minimize the largest absolute value of the eigenvalues. Although it requires some modification (as we shall see in the next section), numerical examples support this basic philosophy well. In particular we found our methods, after suitable shifts, gave excellent low cost/low accuracy results for stiff systems. There is much room for further research here, and ultimately we would like to have an adaptive

algorithm, that adjusts the freedom in $A(t)$ in response to required accuracy, much like existing algorithms for stepsize control.

The observation that "approximations to $\Gamma(t_0, h)$ need not have the same properties as $\Gamma(t_0, h)$" can also be exploited to find many approximations other than the ones $\tilde{\Gamma}_2$ and $\tilde{\Gamma}_4$ we have written above. In particular, let us consider the identities

$$\Gamma(t_0, h) \;=\; \Gamma(t_0 + h, -h)^{-1} \;=\; \Gamma(t_0 + h, -\tfrac{h}{2})^{-1}\Gamma(t_0, \tfrac{h}{2}). \tag{40}$$

When we approximate the time ordered exponentials appearing in these expressions by keeping only a certain number of terms in the relevant sums, the above identities are no longer true to all orders, and thus each of these three expressions gives different approximations to $\Gamma(t_0, h)$. The third expression gives particularly attractive Padé approximant formulae: for example we have the second and fourth order approximations

$$\begin{aligned}
\tilde{\Gamma}_{2\ \text{Padé}} &= \left(\mathrm{I}_{n+m} - \tfrac{h}{2}A\right)^{-1}\left(\mathrm{I}_{n+m} + \tfrac{h}{2}A\right) \\
\tilde{\Gamma}_{4\ \text{Padé}} &= \left(\mathrm{I}_{n+m} - \tfrac{h}{2}A - \tfrac{h^3}{24}\left(\tfrac{1}{2}A'' + A'A - AA' - A^3\right)\right)^{-1} \times \\
&\quad \left(\mathrm{I}_{n+m} + \tfrac{h}{2}A + \tfrac{h^3}{24}\left(\tfrac{1}{2}A'' + A'A - AA' - A^3\right)\right),
\end{aligned} \tag{41}$$

where here $A$ and its derivatives are all evaluated at $t_0 + \frac{h}{2}$. We have not used this formula, because it requires substantial symbolic computing to implement, but the efficiency of using such approximations certainly merits investigation. In greater generality, many other possible schemes for constructing approximations $\tilde{\Gamma}$ exist, and we have only selected the simplest for the purpose of demonstrating the efficacy of Möbius schemes.

## 4 Eliminating Stiffness: An Example

To investigate more concretely the spectrum-translating strategy suggested in the last section for handling stiffness, we consider application of the simplest first order Möbius scheme to the constant coefficient Riccati differential equation. Thus we are approximating solutions of

$$\dot{y} = ay + b - ycy - yd \tag{42}$$

via the recursion

$$y_{i+1} = \left[(I_n + ha)y_i + hb\right]\left[hcy_i + (I_m + hd)\right]^{-1} \tag{43}$$

*Claim.* The fixed points of (43) coincide (precisely, for all $h$) with the critical points of (42). In the generic case, the asymptotic behavior of (43) near its fixed points reproduces qualitatively the asymptotic behavior of (42) near its critical points, but only provided $h$ is sufficiently small.

*Proof.* The critical points of (42) are points $y = Y$, where $Y$ is any solution to the algebraic Riccati equation $aY + b - YcY - Yd = 0$. A simple manipulation shows that $y_n = Y$ is a fixed point of the recursion (43) also precisely when $Y$ obeys this algebraic Riccati equation.

We now look at the differential equation and the recursion in the neighborhood of a critical/fixed point $Y$. The "genericity" assumption that we make is that the $n \times n$ matrix $a - Yc$ and the $m \times m$ matrix $(d + cY)^T$ each have a complete set of eigenvectors, i.e. that we can find $n$ linearly independent $n-$dimensional vectors $v_r$, $r = 1, \ldots, n$, such that

$$(a - Yc)v_r = \lambda_r v_r \qquad \text{for some scalar } \lambda_r\ , \tag{44}$$

and $m$ linearly independent $m-$dimensional vectors $w_s$, $s = 1, \ldots, m$, such that

$$(d + cY)^T w_s = \mu_s w_s \qquad \text{for some scalar } \mu_s \ . \tag{45}$$

The $mn$ rank one matrices of the form $v_r w_s^T$ span the space of $n \times m$ matrices.

Consider first the differential equation in the neighborhood of $Y$. Writing $y = Y + \epsilon$, we have

$$\dot{\epsilon} = (a - Yc)\epsilon - \epsilon(d + cY) + O(\epsilon^2) \ . \tag{46}$$

Expanding $\epsilon$ via

$$\epsilon = \sum_{r=1}^{n} \sum_{s=1}^{m} \epsilon_{rs} v_r w_s^T \ , \tag{47}$$

we find

$$\dot{\epsilon}_{rs} = (\lambda_r - \mu_s)\epsilon_{rs} + O(\epsilon^2) \qquad \begin{matrix} r = 1, \ldots, n \\ s = 1, \ldots, m \end{matrix} \ . \tag{48}$$

The $mn$ numbers $\lambda_r - \mu_s$ determine the behavior of the orbits of the equation (42) near the critical point $Y$. We note that under a spectrum translation in the equation, viz. $a \to a + pI_n$, $d \to d + pI_m$ for some constant $p$, we have $\lambda_r \to \lambda_r + p$ and $\mu_s \to \mu_s + p$ and the differences $\lambda_r - \mu_s$ are left unchanged, as expected.

Turning now to the recursion, we write $y_i = Y + \epsilon_i$, and a simple calculation gives

$$\epsilon_{i+1} = [I_n + h(a - Yc)]\, \epsilon_i\, [I_m + h(cY + d)]^{-1} + O(\epsilon_i^2) \ . \tag{49}$$

Expanding

$$\epsilon_i = \sum_{r=1}^{n} \sum_{s=1}^{m} \epsilon_{i,rs} v_r w_s^T \ , \tag{50}$$

we have

$$\epsilon_{i+1,rs} = \left( \frac{1 + h\lambda_r}{1 + h\mu_s} \right) \epsilon_{i,rs} \qquad \begin{matrix} r = 1, \ldots, n \\ s = 1, \ldots, m \end{matrix} \ . \tag{51}$$

The $mn$ numbers $(1 + h\lambda_r)/(1 + h\mu_s)$ determine the behavior of the recursion near the fixed point. Since

$$\frac{1 + h\lambda_r}{1 + h\mu_s} = 1 + h(\lambda_r - \mu_s) + o(h) \ , \tag{52}$$

for sufficiently small $h$ we are guaranteed to qualitatively reproduce the asymptotic behavior of the differential equation near $Y$. But evidently for larger $h$, this need not be the case. •

Stiffness is the phenomenon that to obtain qualitatively correct behavior from a numerical scheme near a stable critical point of an ODE we are forced to use very small $h$. In our case, from (48), the critical point $y = Y$ is stable if all the differences $\lambda_r - \mu_s$ have negative real part. To obtain qualitatively correct behavior from the recursion we need

$$\left| \frac{1 + h\lambda_r}{1 + h\mu_s} \right| < 1 \tag{53}$$

for all $r, s$. Assuming $\mathrm{Re}(\lambda_r - \mu_s) < 0$, this condition is automatic if $|\mu_s| \geq |\lambda_r|$, and otherwise implies

$$h < \frac{2\mathrm{Re}(\mu_s - \lambda_r)}{|\lambda_r|^2 - |\mu_s|^2} \ . \tag{54}$$

Stiffness can happen, but the criterion for stiffness is not the eigenvalues of $A$ having negative real part and large absolute value, as was tentatively suggested in the previous section (we

will see confirmation of this in the numerical example in section 5.5). Nevertheless, *spectrum translation can resolve stiffness*. Under spectrum translation we have

$$\frac{1+h\lambda_r}{1+h\mu_s} \to \frac{1+h(\lambda_r+p)}{1+h(\mu_s+p)} = \frac{1+\frac{h}{1+hp}\lambda_r}{1+\frac{h}{1+hp}\mu_s} , \tag{55}$$

showing spectrum translation relaxes the condition on the stepsize $h$ (instead of $h$ being small, $h/(1+hp)$ must be small). Unfortunately it is not easy to determine from this analysis (particuarly in the time dependent context) what $p$ should be chosen for a given equation. In our numerical studies we took $p$ to be the largest eigenvalue of $A$ in absolute value, which seemed to work well.

We conclude this section with several notes:

1. In the introduction we presented Möbius schemes as an implementation of linearization techniques for Riccati equations in the original variables. We have now shown that stiffness can occur for Möbius schemes, but its occurence does not coincide with the occurence of stiffness for the associated linear system. This shows that there can be substantial differences in the effectiveness of the two types of method. See also section 5.5.

2. From the above analysis we see that *if the recursion (43) converges, it converges to a solution of the algebraic Riccati equation* $aY + b - YcY - Yd = 0$. Under the circumstances that this algebraic Riccati equation has a unique stable solution (a solution for which all the differences $\lambda_r - \mu_s$ have negative real part), as happens in a number of cases of interest, implementing the recursion (43), using *arbitrary* starting value $y_0$ looks like a potentially useful method for finding this stable solution (one could of course exploit the higher order methods of section 2 as well). This method for solving the algebraic Riccati equation has many similarities to the recursive method for computing the sign function of the matrix $A$, and using this to solve the associated algebraic Riccati equation [14]. We thank one of the referees for pointing this out.

3. It is interesting to generalize the above analysis for higher order Möbius schemes. Unfortunately, for a general Möbius scheme applied to the general constant coefficient Riccati equation, the fixed points of the recursion need not coincide exactly with the critical points of the ODE. If we restrict to the case $b = 0$, and look only at the critical point at $y = 0$, which will be a fixed point for all the Möbius schemes we have considered, then a general theory can be developed, analogous to the standard stability theory for linear ODEs. Stability of a scheme, for a given $h$, is determined by the eigenvalues of $a$ and $d$, which now are exactly the eigenvalues of $A$; the region of stability is changed by spectrum translation, and it is possible to estimate the spectrum translation needed to produce stability for a given $h$. For the general case ($b \neq 0$), none of this is possible without a priori knowledge of the critical point $Y$. (If $Y$ is known, then substituting $z = y - Y$ the equation is brought to $b = 0$ form $\dot{z} = (a - Yc)z - zcz - z(d + cY)$.)

# 5 Examples 1: Nonsingular Solutions

## 5.1 Numerical Procedure

In the following examples, we performed numerical integrations of Riccati equations using Möbius schemes (2). The approximation $\tilde{\Gamma}(t_i, h)$ used was $\tilde{\Gamma}_2$ from (35) in all examples. In

addition $\tilde{\Gamma}_4$ from (33) was used in the third example below (section 5.4). Each Riccati equation was specified by an initial choice of $A(t)$, and the effects of various spectrum translations of the form (37) were considered.

A standard stepsize control procedure was used throughout. At each step, the program receives the current value $t$ of the independent variable, the computed value of $y(t)$, and $h$, the last stepsize used. Using this data the program computes two approximations to $y(t+h)$, the first, $y_1$, by application of the given numerical method once with stepsize $h$, and the second, $y_2$, by application of the given numerical method twice, with stepsize $h/2$. A local error estimate is then obtained using the simple absolute error

$$||y_1 - y_2|| = \sum_{a=1}^{n} \sum_{b=1}^{m} |y_{1ab} - y_{2ab}|. \tag{56}$$

The use of a pure absolute error formula was intended to be extremely constraining when integrating through singularities. Once $||y_1 - y_2||$ is computed, it is compared with a preset tolerance $\Delta$. If the error exceeds $2\Delta$, the step is rejected, and all calculations are repeated using a smaller stepsize $h_{\text{new}}$, obtained from $(h_{\text{new}}/h)^{p+1} = \Delta/||y_1 - y_2||$, where $p$ is the order of the method. Otherwise, the step is accepted, with the approximation to $y(t+h)$ taken to be the extrapolated value $(2^p y_2 - y_1)/(2^p - 1)$. The stepsize $h$ is passed on to the next step, unless $||y_1 - y_2|| < \Delta/2$, in which case it is updated to $h_{\text{new}}$ via the formula given above.

This stepsize control procedure should be sensitive to the problem of ill-conditioning in the matrices that need to be inverted to construct $y_{i+1}$ from $y_i$.

### 5.2 Problem 1: A constant coefficient example

As a first example we look at example 1 from [8], in which

$$A(t) = \begin{pmatrix} 0 & 0 & 0 & 1 \\ -10 & -1 & 10 & 0 \\ 0 & 1 & 0 & 0 \\ 100 & 0 & -100 & -1 \end{pmatrix}, \tag{57}$$

and we consider the problem on $t > 0$ with initial condition $y(0) = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}$. The exact solution is

$$y(t) = \frac{1}{1 + 2.331e^{-22t} + .42e^{-11t} - 1.21e^{-21t}} \begin{pmatrix} 1 + 2.1e^{-22t} - 1.89e^{-11t} - 1.21e^{-21t} & .11 - .231e^{-22t} + .121e^{-21t} \\ -2.541e^{-22t} & -.1 - .21e^{-22t} + .189e^{-11t} + .121e^{-21t} \end{pmatrix}.$$

In practice we consider integration over the interval $0 < t < 5$; by $t = 5$ the solution is extremely close to its asymptotic value $y(\infty) = \begin{pmatrix} 1 & .110 \\ 0 & -.1 \end{pmatrix}$. The eigenvalues of the matrix $A(t)$ given above are $10, 0, -1, -11$. We considered using this $A(t)$, and three other choices of $A(t)$, translated by $+10$ $\text{I}_4$, $+20$ $\text{I}_4$, and $-10$ $\text{I}_4$. Results, given as a log-log plot of number of steps used against inverse tolerance, are displayed in Figure 1. In all cases the integration was found to be reliable over the large range of tolerances displayed, with the tolerance giving a good measure of the magnitude of global error. For tight tolerances all 4 methods showed the expected linear increase of the number of steps required with $1/(\text{tolerance})^{1/3}$, with best

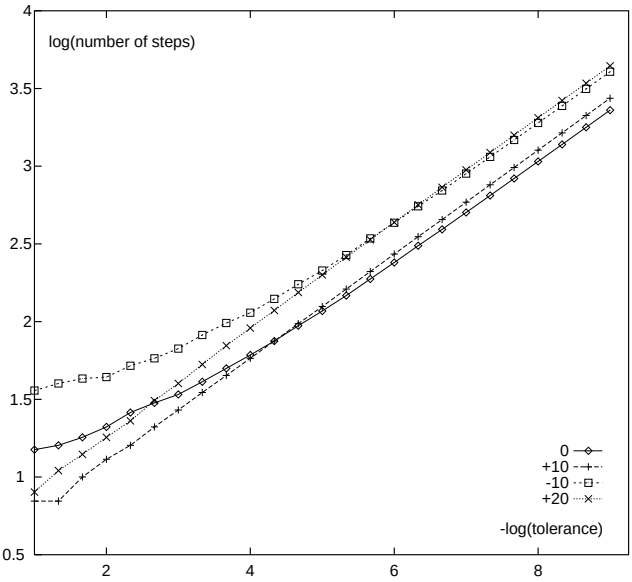


Figure 1: Results for Problem 1. The legend indicates how much the matrix $A(t)$ has been shifted. Base 10 logarithms are used.

performance by the original choice of $A(t)$, for which the eigenvalues are most symmetrically placed about the origin. For broader tolerances there is a marked difference in behavior between the methods for which there are large negative eigenvalues (i.e. the initial choice of $A(t)$, and the choice with $A(t)$ shifted by $-10$ $\mathrm{I}_4$) and the methods for which there are no large negative eigenvalues. For the former, there are signs of stiffness — stepsize choice becomes erratic, and the decrease of the number of steps as tolerance is broadened is not as rapid as expected, because stability issues, and not just accuracy issues, are relevant. So, as we have predicted, the question of which method is optimal depends on the desired tolerance; for low accuracy, stiff methods are best, for higher accuracy the choice of $A(t)$ with eigenvalues symmetric about the origin is preferable. Of course, in the current example, where the negative eigenvalues are not particularly large, these effects are small, but they are most definitely present.

### 5.3 Problem 2: Dieci's knee problem

Here we consider example 3 from [8], i.e. the scalar equation

$$\dot{y} = 1 + \frac{y(y-t)}{\epsilon}, \tag{58}$$

for which

$$A(t) = \begin{pmatrix} -\frac{t}{2\epsilon} & 1 \\ -\frac{1}{\epsilon} & \frac{t}{2\epsilon} \end{pmatrix}. \tag{59}$$

Here $\epsilon$ is a small positive number. We are given an initial condition at $t = -1$, and it is desired to integrate over the interval $-1 < t < 1$.

This is a difficult problem for numerical analysis. The $\epsilon$ dependence can be scaled out, and the solutions take the form $y(t) = \sqrt{\epsilon}Y(t/\sqrt{\epsilon})$ where $Y(T)$ is plotted in Figure 2. From this figure the general behavior can be seen: for $t < 0$ the solutions are attracted (possibly

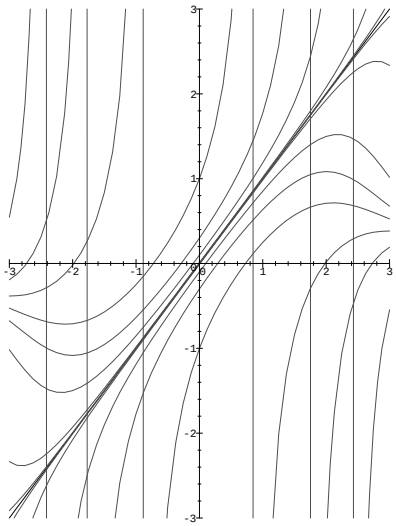

Figure 2: $Y(T)$ as a function of $T = t/\sqrt{\epsilon}$ for Problem 2. The solutions of Problem 2 are given by $y(t) = \sqrt{\epsilon}Y(T)$. The vertical lines indicate asymptotes of solutions.

through a singularity) towards $y(t) = t$, which is an exact solution. For $t > 0$ the solution $y(t) = t$ becomes unstable, and solutions are attracted towards an asymptotic solution $y(t) = \epsilon/t + O(\epsilon^2)$ ($Y(T) = 1/T + \ldots$). What is less evident from the figure, because of the rescaling, is just how rapidly solutions are pulled towards the stable asymptotic solutions in $t < 0$ and $t > 0$. If we take initial condition $y(-1) = -1 - m$, the exact solution takes the form

$$y(t) = t - \frac{me^{(t^2-1)/2\epsilon}}{1 + \frac{m}{\epsilon}\int_{-1}^{t} e^{(s^2-1)/2\epsilon} ds}, \tag{60}$$

and from this

$$|y(0)| \leq |m|\ e^{-1/2\epsilon}. \tag{61}$$

For small $\epsilon$ and moderate $m$, this is very, very small, and, bearing in mind that whatever integration scheme we use has some error, $y(0)$ will be essentially indistinguishable from 0. What this means is that once we reach $t = 0$, all information about the initial condition will have been lost.

If all information about the initial condition has been lost, all that we can ask from a numerical method is qualitatively correct behavior, i.e. that for $t > 0$ it should reach the appropriate asymptotic solution $y(t) = \epsilon/t + O(\epsilon^2)$. Now, from Figure 2, we see that solutions with $y(0) > 0$ reach the asymptotic solution via a singularity, while solutions with $y(0) < 0$ do not. Since we have no control over whether $y(0) > 0$ or $y(0) < 0$, an apparent *sine qua non* for dealing with this problem is the ability to pass through singularities. In fact, the situation is a little better than this: any method, while integrating up the stable solution $y(t) = t$ in $t < 0$, will have a systematic error, whose sign will be typically purely method dependent, i.e. not dependent on choice of stepsize/error tolerances. Should the systematic error be negative, the method will report $y(0) < 0$ and there will be no need to go through a singularity. Of course, it is doubtful that there exists a method not suited to dealing with singularities, which happens to correctly avoid the need to do so in *all* problems. The argument given here is an explanation of how methods that cannot handle singularities might

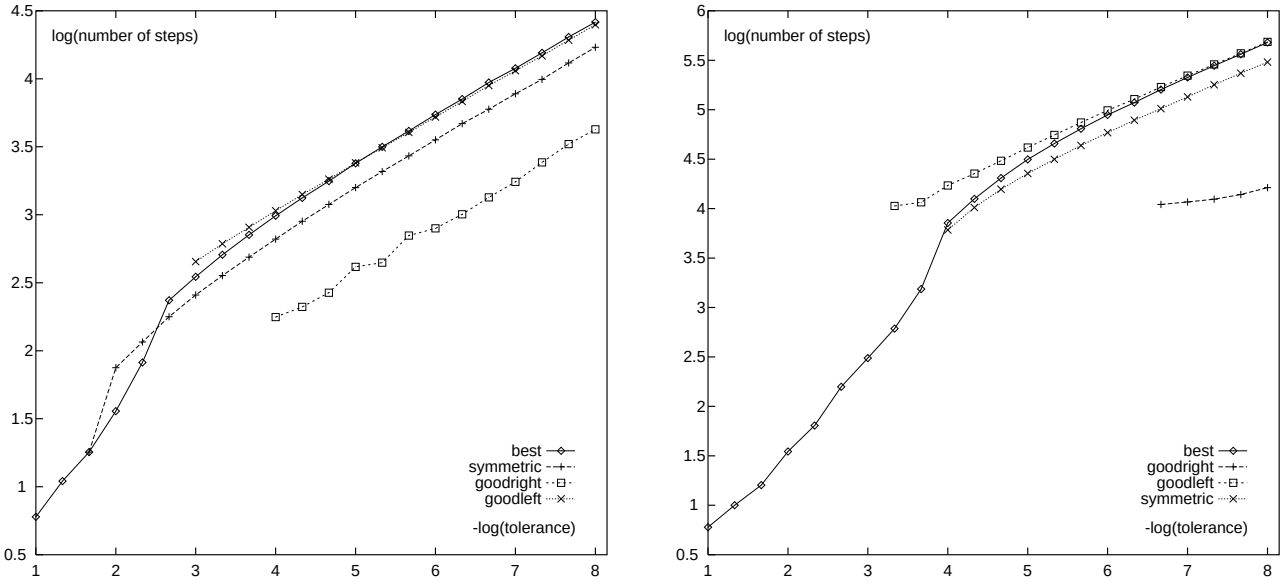


Figure 3: Results for Problem 2, $\epsilon = 10^{-3}$ (left) and $\epsilon = 10^{-5}$ (right). For methods other than "best", reliable integrations could not be obtained at low accuracy.

still produce correct results in the current problem. It is certainly not a justification for their generally applicability.

Having established that we are only looking for qualitatively correct behavior in this problem, particularly in the presence of singularities, let us present results. We took initial condition $y(-1) = -1.1$, and considered 4 choices of $A(t)$,

$$\begin{aligned}
\text{"symmetric"} \quad &: \quad A(t) = \begin{pmatrix} -\frac{t}{2\epsilon} & 1 \\ -\frac{1}{\epsilon} & \frac{t}{2\epsilon} \end{pmatrix} \\
\text{"goodright"} \quad &: \quad A(t) = \begin{pmatrix} 0 & 1 \\ -\frac{1}{\epsilon} & \frac{t}{\epsilon} \end{pmatrix} \\
\text{"goodleft"} \quad &: \quad A(t) = \begin{pmatrix} -\frac{t}{\epsilon} & 1 \\ -\frac{1}{\epsilon} & 0 \end{pmatrix} \\
\text{"best"} \quad &: \quad A(t) = \begin{pmatrix} -\frac{t}{2\epsilon} & 1 \\ -\frac{1}{\epsilon} & \frac{t}{2\epsilon} \end{pmatrix} + \left| \frac{t}{2\epsilon} \right| \, \mathrm{I}_2.
\end{aligned}$$

The "best" method has no negative eigenvalues, and agrees with the "goodleft" method for $t < 0$ and the "goodright"method for $t > 0$. The "symmetric" method has the two eigenvalues of $A(t)$ symmetrically placed around the origin. Results for $\epsilon = 10^{-3}$ and $\epsilon = 10^{-5}$ are presented in Figure 3.

All methods except "best" cannot be reliably run for broad error tolerances. On the other hand, "best" gives excellent, reliable results in this region. To illustrate this, we present the runs for this method at tolerances $0.08, 0.09, 0.10, 0.11$ for $\epsilon = 10^{-5}$ in Table 1. The method produces — to within the tolerance level — the correct behavior for both $t < 0$ and $t > 0$, with, in fact, surprising accuracy in the latter region. Bearing in mind the complexity of this problem, these are remarkable results. To demonstrate further the reliability of our method, in Table 2 we present a similar set of runs obtained for a modified Dieci knee problem [15], the equation

$$\dot{y} = \epsilon + \frac{y(y-t)}{\epsilon} \tag{62}$$

| tolerance 0.11 | | tolerance 0.10 | | tolerance 0.09 | | tolerance 0.08 | |
|---|---|---|---|---|---|---|---|
| t | y | t | y | t | y | t | y |
| -0.5688 | -0.7125 | -0.6778 | -0.7852 | -0.7008 | -0.8005 | -0.7246 | -0.8164 |
| -0.1376 | -0.2812 | -0.3556 | -0.4629 | -0.4016 | -0.5013 | -0.4492 | -0.5410 |
| 0.2937 | 0.04597 | -0.0333 | -0.1406 | -0.1024 | -0.2021 | -0.1738 | -0.2655 |
| 0.7249 | 1.483e-5 | 0.2889 | 0.01102 | 0.1968 | 0.03435 | 0.1016 | 0.01050 |
| 2.0186 | 0.512e-5 | 0.7560 | 1.398e-5 | 0.4960 | 2.155e-5 | 0.3770 | 2.291e-5 |
| | | 2.1575 | 0.474e-5 | 1.3936 | 0.739e-5 | 1.2032 | 0.803e-5 |

Table 1: Samples runs of method "best" for the Dieci knee problem

| tolerance 0.11 | | tolerance 0.10 | | tolerance 0.09 | | tolerance 0.08 | |
|---|---|---|---|---|---|---|---|
| t | y | t | y | t | y | t | y |
| -0.5688 | -0.7124 | -0.6778 | -0.7851 | -0.7008 | -0.8004 | -0.7246 | -0.8163 |
| -0.1376 | -0.2810 | -0.3556 | -0.4628 | -0.4017 | -0.5012 | -0.4492 | -0.5409 |
| 0.2936 | 0.04563 | -0.0334 | -0.1402 | -0.1025 | -0.2018 | -0.1739 | -0.2653 |
| 0.7248 | -0.76e-8 | 0.2888 | 0.01016 | 0.1967 | 0.03382 | 0.1015 | 0.01303 |
| 2.0183 | 0.51e-8 | 0.7675 | 1.00e-8 | 0.4959 | -5.66e-8 | 0.3769 | -10.22e-8 |
| | | 2.2037 | 0.46e-8 | 1.3934 | 0.74e-8 | 1.2030 | 0.80e-8 |

Table 2: Similar results for the modified Dieci knee problem

(run with identical initial condition and $\epsilon = 10^{-4}$), for which the stable solution for $t < 0$ takes the form $y(t) = t + \epsilon/t + \ldots$ and the stable solution for $t > 0$ takes the form $y(t) = \epsilon^2/t + \ldots$.

Several other features of Figure 3 deserve mention. First, at high accuracy, the "goodright" method outperforms the others substantially, in needing far fewer steps. This is a coincidence. A detailed analysis shows that for this method in the $t < 0$ region, the dominant systematic error term present for the other methods is absent. Consequently it integrates far better in this region if there are no stability problems. However there are stability problems for broad tolerances, for which a reliable integration could not be achieved. Of the other 3 methods, "symmetric" works best, as expected. Second, we note the strange behavior of the plots for the method "best" for broad tolerances. Despite the absence of stiffness, there is no reason to expect anything approaching linear behavior of the plot except in the region where the typical stepsize is substantially less than the natural scale of the problem, $\sqrt{\epsilon}$. In fact the onset of the linear regime in the plots we have obtained is perfectly within expectations. One contributory factor to the rapid increase in the number of steps as the tolerance is tightened is the fact that the solutions obtained with the method "best" go through a singularity to latch on to the correct asymptotic solution for $t > 0$. For the reasons we have explained, there is no reason why they should not do this.

### 5.4 Problem 3: A matrix time dependent example

Here we consider example 4 from [8], for which

$$A(t) = \begin{pmatrix} 0 & \frac{t}{2\epsilon} & \frac{1}{2} & 1 \\ 0 & 0 & 0 & 1 \\ \frac{1}{\epsilon} & 0 & -\frac{t}{2\epsilon} & 0 \\ 0 & \frac{1}{\epsilon} & 0 & 0 \end{pmatrix}, \tag{63}$$

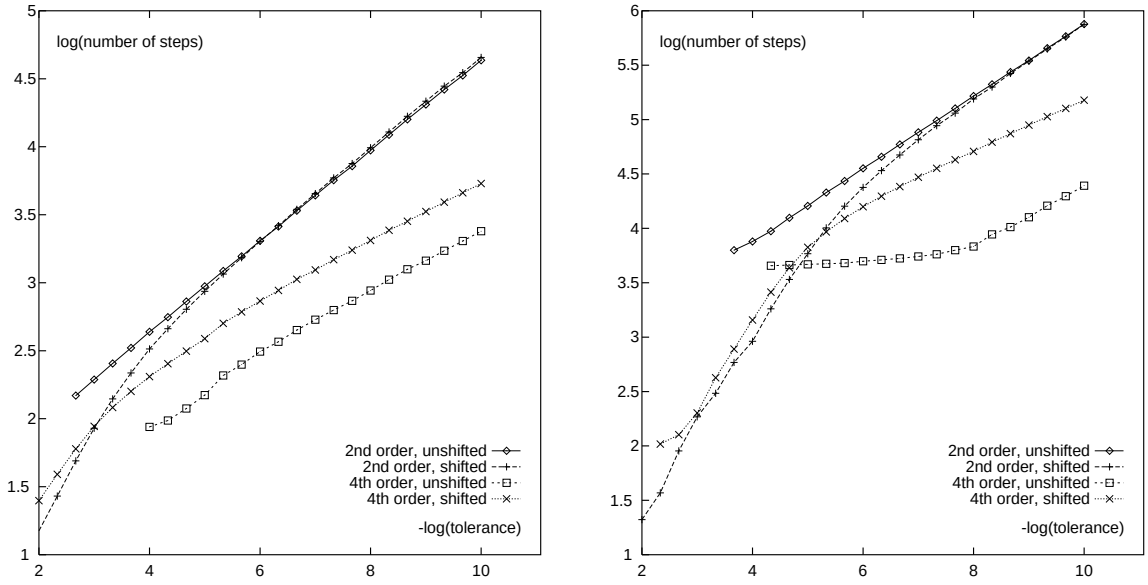


Figure 4: Results for Problem 3, $\epsilon = 10^{-3}$ (left) and $\epsilon = 10^{-5}$ (right)

and we integrate over the interval $-1 < t < 1$ with initial condition

$$y(-1) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{64}$$

Straightforward calculations show this equation has the following stable solutions:

$$y(t) = \begin{pmatrix} -\epsilon/t + O(\epsilon^2) & (t/2) + \sqrt{\epsilon} + O(\epsilon) \\ 0 & \sqrt{\epsilon} \end{pmatrix} \qquad t < 0 \tag{65}$$

$$y(t) = \begin{pmatrix} t/2 & \sqrt{\epsilon} \\ 0 & \sqrt{\epsilon} \end{pmatrix} \qquad t > 0. \tag{66}$$

The eigenvalues of $A(t)$ as given above are

$$\pm\frac{1}{\sqrt{\epsilon}}, \ \frac{-t \pm \sqrt{t^2 + 8\epsilon}}{4\epsilon}. \tag{67}$$

We therefore implemented our methods with two choices of $A(t)$, the one given above, and the above shifted by $(t/2\epsilon)\mathrm{I}_4$ in the region $t > 0$. We note the latter choice leaves a negative eigenvalue $-1/\sqrt{\epsilon}$ for $t < 0$, but removes the much larger negative eigenvalue that appears for $t > 0$. We used $\epsilon = 10^{-3}$ and $\epsilon = 10^{-5}$, and considered both the second order and fourth order methods given in Section 3. Results are displayed in Figure 4, and are completely in line with expectations. Having left a moderate negative eigenvalue in even the shifted method, we found the broadest tolerance usable for the shifted method was not as low as in the example of the Dieci knee problem, but still we could get very fast, reliable low-accuracy integration for tolerance $10^{-2}$.

### 5.5 Problem 4: A large matrix example and (the absence of) stiffness

Finally, we consider example 1 from [7] (which is equivalent to example 6 from [8]). Here $y(t)$ is an $N \times N$ matrix, $k$ is a positive real, and we consider the equation

$$\dot{y} = k^2 \, \mathrm{I}_N - y^2 \qquad t > 0. \tag{68}$$

The general solution is

$$y(t) = k\left(\tanh kt\ \mathrm{I}_N + \frac{y(0)}{k}\right)\left(\mathrm{I}_N + \frac{\tanh kt}{k}\ y(0)\right)^{-1}. \tag{69}$$

We consider two choices of the matrix $A(t)$ for this equation, both constant:

$$A_1 = \begin{pmatrix} 0 & k^2\ \mathrm{I}_N \\ \mathrm{I}_N & 0 \end{pmatrix} \tag{70}$$

$$A_2 = \begin{pmatrix} k\mathrm{I}_N & k^2\ \mathrm{I}_N \\ \mathrm{I}_N & k\mathrm{I}_N \end{pmatrix}. \tag{71}$$

$A_1$ has eigenvalues $\pm k$, each repeated $N$ times, and $A_2$ has eigenvalues $0, 2k$, each repeated $N$ times. Applying our method with constant stepsize $h$ gives the recursions

$$y_{i+1} = (\alpha_a(h)y_i + \beta_a(h))\,(\gamma_a(h)y_i + \delta_a(h))^{-1} \tag{72}$$

where

$$\begin{aligned} \tilde{\Gamma}_{2,a}(h) &= \begin{pmatrix} \alpha_a(h) & \beta_a(h) \\ \gamma_a(h) & \delta_a(h) \end{pmatrix} \\ &= \mathrm{I}_{2N} + hA_a + \frac{h^2}{2}A_a^2, \qquad a = 1, 2 \qquad (73) \\ &= \begin{cases} \left(1 + \frac{h^2k^2}{2}\right)\begin{pmatrix} \mathrm{I}_N & \frac{hk^2}{1+h^2k^2/2}\mathrm{I}_N \\ \frac{h}{1+h^2k^2/2}\mathrm{I}_N & \mathrm{I}_N \end{pmatrix} & a = 1 \\ (1 + hk + h^2k^2)\begin{pmatrix} \mathrm{I}_N & \frac{hk^2(1+hk)}{1+hk+h^2k^2}\mathrm{I}_N \\ \frac{h(1+hk)}{1+hk+h^2k^2}\mathrm{I}_N & \mathrm{I}_N \end{pmatrix} & a = 2 \end{cases}. \qquad (74) \end{aligned}$$

It is straightforward to diagonalize and compute powers of the matrices $\tilde{\Gamma}_{2,a}(h)$, and thereby solve the recursions. For both $a = 1, 2$ the solution takes the form

$$y_i = k\left(\tanh i\overline{kh}\ \mathrm{I}_N + \frac{y(0)}{k}\right)\left(\mathrm{I}_N + \frac{\tanh i\overline{kh}}{k}\ y(0)\right)^{-1}, \tag{75}$$

where

$$\overline{kh} = \begin{cases} \frac{1}{2}\ln\left(\frac{1+hk+h^2k^2/2}{1-hk+h^2k^2/2}\right) & a = 1 \\ \frac{1}{2}\ln(1 + 2hk + 2h^2k^2) & a = 2 \end{cases}. \tag{76}$$

The meaning of these results is that if the recursions were implemented exactly, their solutions follow the solution curves of the differential equation, but the propagation rate down the curve is not necessarily correct; the ratio of the observed rate to the correct rate is $\overline{kh}/kh$. For small enough $h$ ($hk << 1$), this is approximately 1, but for larger $h$ it can be substantially less than 1. (Other Möbuis schemes allow the asymptotic solution to be approached faster than the real rate, for suitable values of $h$. For the second order Padé approximant method mentioned at the end of Section 3, we find $\overline{kh} = \ln|(1 + hk/2)/(1 - hk/2)|$, which can exceed $kh$.)

The above results do not automatically imply that Möbius schemes will be successful for equation (68). Depending on the value of $y(0)$, the exact solution curve can tend to a variety of limit points as $t \to \infty$, but only one of these is stable (in the sense of section 3), the limit $y \to k\mathrm{I}_N$, and a good integration scheme should always reach this limit. We numerically integrated (68) for the cases $N = 10$ and 50, $k = 10$ and 1000, over the interval $0 < t < 5$,

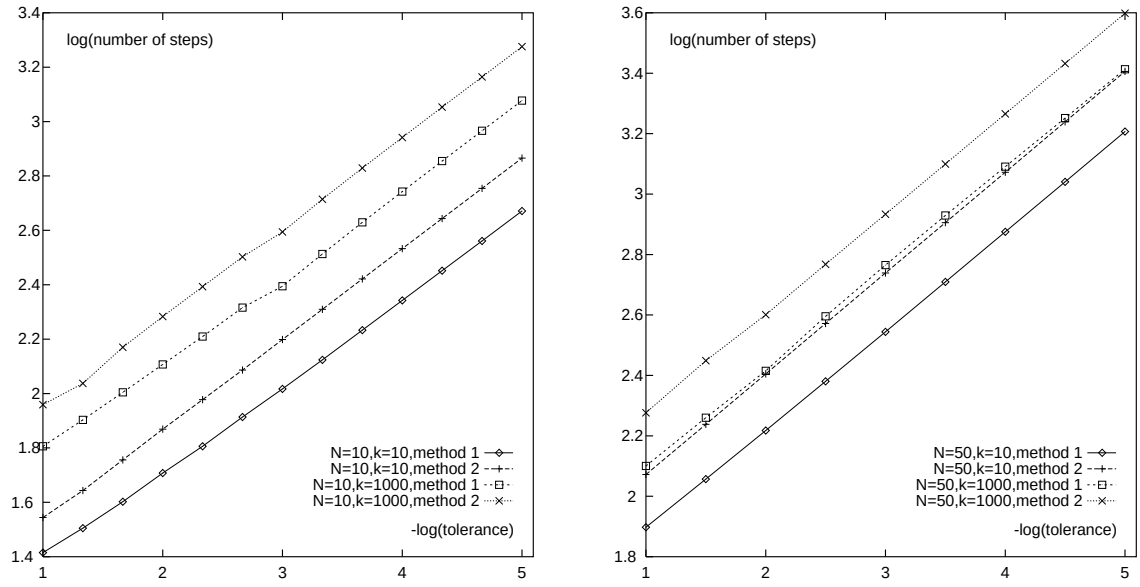


Figure 5: Results for Problem 4, $N = 10$ (left) and $N = 50$ (right)

via method 1 (using $A_1$) and method 2 (using $A_2$), with the usual stepsize control technique. The initial condition chosen was $y(0) = U\ \mathrm{diag}(1, 2, \ldots, N)\ U^{-1}$, where $U$ was an $N \times N$ matrix of random numbers between 0 and 1. Correct limit behavior was always achieved, and results on the number of steps required are presented in Figure 5.

In figure 5, no signs of stiffness are evident, even for $k = 1000$ with method 1, for which the matrix $A = A_1$ has large negative eigenvalues. As was discussed in section 3, the true stiffness criterion for Möbius schemes is somewhat different, and in fact for the simple first order Möbius scheme the criterion of qualitatively correctness with the choice $A = A_1$ can easily be verified to be $|(1 - hk)/(1 + hk)| < 1$, which holds for all $hk > 0$.

Figure 5 conceals one minor subtlety. For method 1, we see from Eq. (76) that $\overline{hk}$ is bounded as a function of $hk$; this means that irrespective of the stepsize chosen, we go no more than a certain distance down the solution curve. For method 2, this is not true. In particular, if given the option of taking a "giant" step, it can move directly, in one step, to the asymptotic solution, and stepsize control will not prevent this, as it is actually doing the correct thing. Thus the results for method 2 turned out to be dependent on the initial $h$ fed to the stepsize routine. The results presented used a small initial $h$ of 0.1; increasing the initial $h$ over a certain threshold would make the integration finish in (essentially) one step.

# 6 Examples 2: Singular Solutions

In this section we illustrate integration through singularities, using two of the equations of the previous section.

First, we consider the Dieci knee problem, equation (58), with $\epsilon = 1$, and initial condition $y = 0$ at $t = -1$. From Figure 2, we see that the corresponding solution passes through a singularity at some positive $t$. Integrating numerically, using method "best" from Section 5.3, we found, for a wide range of tolerances, that the numerical solution reproduced the correct behavior; relevant data is reproduced in Table 3. For tolerance $10^{-10}$ and smaller, we found

| tolerance | y changes from positive to negative | |
|---|---|---|
| | between | and |
| $10^{-1}$ | 0.3168888327 | 0.5167669143 |
| $10^{-2}$ | 0.4309325461 | 0.4457121281 |
| $10^{-3}$ | 0.4383197809 | 0.4401761636 |
| $10^{-4}$ | 0.4391577864 | 0.4392703998 |
| $10^{-5}$ | 0.4392164717 | 0.4392278245 |
| $10^{-6}$ | 0.4392223764 | 0.4392238105 |
| $10^{-7}$ | 0.4392228711 | 0.4392232076 |
| $10^{-8}$ | 0.4392231157 | 0.4392231199 |

Table 3: Numerical integration through a singularity in the Dieci knee problem

that the numerical method got "stuck" in the sense that it took smaller and smaller step sizes as it approached the singularity, and failed to pass it. This can be attributed directly to the manner in which we we computed error estimates, *viz.* equation (56). An absolute error formula is too constraining when dealing with quantities of large magnitude. We made the decision to use absolute error estimates precisely to demonstrate that despite this we could integrate through singularities. Using relative error estimates (for large values of $y$) allows us to integrate even at tighter tolerance; we have tested tolerances down to $10^{-13}$. For smaller $\epsilon$ in the Dieci knee problem, the minimum tolerance permitted when using only absolute error estimates decreases (for $\epsilon = 10^{-4}$ we could use tolerance $10^{-11}$ without problem); this is because as $\epsilon$ is decreased the approach to singularity is more rapid.

At low tolerances we expect to get so close to the singularity that the issue of machine number handling may become relevant. This could cause failure of the method, or at the least a loss of accuracy. In practice we have yet to see this.

Moving to the problem of Section 5.5, equation (68), the exact solution (69) shows there are singularities whenever $\tanh kt = -k/\lambda$, where $\lambda$ is an eigenvalue of $y(0)$. The singularity is a pole of order equal to the multiplicity of the corresponding eigenvalue $\lambda$. We ran method 1 from section 5.5 on this problem, taking $N = 3$, $k = 10$, and $y(0) = U\ D\ U^{-1}$, where $U$ is a matrix of random numbers between 0 and 1, and $D = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$. We took $(\lambda_1, \lambda_2, \lambda_3)$ to be $(-20, -30, -40)$, $(-20, -20, -30)$ and $(-20, -20, -20)$. (In the last case, since $D$ was a multiple of the identity, $U$ played no role). The singularity associated with eigenvalue $-20$ occurs at $t \approx 0.0549306144$, that associated with eigenvalue $-30$ occurs at $t \approx 0.0346573590$, and that associated with eigenvalue $-40$ at $t \approx 0.0255412812$. Results are shown in Table 4. The tolerances used give an idea of when the method works despite the use of absolute error estimates; using relative errors it can be substantially extended. The effect of the order of the pole on the ease with which it can be passed is not currently clear to us.

# 7 Concluding Comments

In conclusion, there are some open issues we would like to mention. The underlying idea of this paper comes from group theory — that if a differential equation tells us that a function is evolving via infinitesimal group transformations of a certain type, the recursions used in numerical simulations should be group transformations of the same type. However, certain aspects of our implementation are not natural, in the group theoretic sense, in particular the error estimate and extrapolation procedure described in Section 5.1. In the notation of that

| eigenvalues | tolerance | y changes from positive to negative between | and |
|---|---|---|---|
| $(-20,-30,-40)$ | $10^{-1}$ | 0.025304245 | 0.025724295 |
| | | 0.034544911 | 0.034767249 |
| | | 0.054868592 | 0.054977271 |
| | $10^{-3}$ | 0.025539348 | 0.025543016 |
| | | 0.034656892 | 0.034657922 |
| | | 0.054930411 | 0.054930951 |
| | $2 \times 10^{-5}$ | 0.025541235 | 0.025541319 |
| | | 0.034657355 | 0.034657369 |
| | | 0.054930607 | 0.054930617 |
| | $10^{-5}$ | 0.025541260 | 0.025541300 |
| | | 0.034657358 | 0.034657361 |
| | | fails to pass next singularity | |
| $(-20,-20,-30)$ | $10^{-1}$ | 0.034525551 | 0.034809718 |
| | | 0.054826677 | 0.055057440 |
| | $10^{-3}$ | 0.034655129 | 0.034659164 |
| | | 0.054929646 | 0.054931847 |
| | $10^{-5}$ | 0.034657350 | 0.034657370 |
| | | 0.054930595 | 0.054930630 |
| $(-20,-20,-20)$ | $10^{-1}$ | 0.054678140 | 0.055058566 |
| | $10^{-3}$ | 0.054926294 | 0.054933468 |
| | $10^{-5}$ | 0.054930602 | 0.054930637 |
| | $10^{-6}$ | 0.054930609 | 0.054930617 |

Table 4: Numerical integration through singularities in Problem 4

section, the natural way to estimate the error in $y_1, y_2$ is to estimate the minimal deviation from the identity element of those group elements transforming $y_1$ to $y_2$. We intend to return to this issue. Similarly, extrapolation should be done in a group theoretic — not necessarily linear — way. In practice, we do not expect these issues to have very serious effect on the results of implementations, but they are of theoretical interest, not just for the numerical integration of Riccati equations, but also for the host of other equations for which symmetric methods are relevant.

It is also of interest to construct multistep or multivalue methods for Riccati equations, with the expectation that these may be computationally cheaper for higher order calculations. Presumably this involves exploitation of higher dimensional representations of Möbius transformations. Also, it is of interest to apply our methods to so-called symmetric Riccati equations, for which various different transformation groups are relevant.

## Acknowledgments

J.S. gratefully acknowledges a number of very useful conversations with David Kessler, and support from the Rashi Foundation as a Guastella Fellow. It is the authors' wish that no one derive, directly or indirectly, military benefit from this work. Please copy this wish if you cite this paper. Some further information on the numerical procedures used in this paper can be found at the URL **http://www.cs.biu.ac.il:8080/~schiff/riccati.html**

## References


[1] See, for example, W.T.Reid, *Riccati Differential Equations*, Academic Press (1972).

[2] See, for example, U.M.Ascher, R.M.Mattheij and R.D.Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, Prentice-Hall (1988).

[3] See, for example, H.B.Keller and M.Lentini, *Invariant Imbedding, the box scheme, and an equivalence between them*, SIAM J.Numer.Anal. **19** 942-962 (1982).

[4] E.J.Davison and M.C.Maki, *The Numerical Solution of the Matrix Riccati Differential Equation*, IEEE Trans.Automat.Contr., **AC-18** 71-73 (1973).

[5] C.S.Kenney and R.B.Leipnik, *Numerical Integration of the Differential Matrix Riccati Equation*, IEEE Trans.Automat.Contr. **AC-30** 962-970 (1985).

[6] D.W.Rand and P.Winternitz, *Nonlinear superposition principles: a new numerical method for solving matrix Riccati equations*, Comp.Phys.Commun. **33** 305-328 (1984); M.Sorine and P.Winternitz, *Superposition laws for solutions of differential matrix Riccati equations arising in control theory*, IEEE Trans.Automat.Contr. **AC-30** 266-272 (1985).

[7] C.H.Choi and A.J.Laub, *Efficient Matrix-valued Algorithms for Solving Stiff DREs*, IEEE Trans.Automat.Contr. **AC-35** 770-776 (1990).

[8] L.Dieci, *Numerical Integration of the Differential Riccati Equation and some Related Issues*, SIAM J.Numer.Anal. **29** 781-815 (1992).

[9] L.Jodar and E.Ponsoda, *Non-autonomous Riccati-type matrix differential equations: existence interval, construction of continuous numerical solutions and error bounds*, IMA J.Numer.Anal. **15** 61-74 (1995); J.L.Morera, G.Rubio and L.Jodar, *Accurate Numerical Integration of Stiff Differential Riccati Equations*, Appl.Math.and Comp. **72** 183-203 (1995).

[10] L.Auslander and R.E.MacKenzie, *Introduction to Differentiable Manifolds*, McGraw-Hill (1963).

[11] Y.Choquet-Bruhat, C.DeWitt-Morette and M.Dillard-Bleick, *Analysis, Manifolds and Phyics*, North-Holland (revised edition, 1982).

[12] M.A.Shayman, *Phase Portrait of the Matrix Riccati Equation*, SIAM J.Control.Optim. **24** 1-65 (1986).

[13] R.P.Feynman, *An Operator Calculus Having Applications in Quantum Electrodynamics*, Phys.Rev. **84** 108-128 (1951).

[14] See, for example, C.S.Kenney and A.J.Laub, *The Matrix Sign Function*, IEEE Trans.Automat.Contr. **40** 1330-1348 (1995).

[15] L.Dieci and D.Estep, *Some Stability Aspects of Schemes for the Adaptive Integration of Stiff Initial Value Problems*, SIAM J.Sci.Statist.Comput. **12** 1284-1303 (1991).