NORGES TEKNISK-NATURVITENSKAPELIGE
UNIVERSITET

A class of low complexity intrinsic schemes for orthogonal integration

by

Elena Celledoni and Brynjulf Owren

PREPRINT
NUMERICS NO. 1/2001

NORWEGIAN UNIVERSITY OF SCIENCE AND
TECHNOLOGY
TRONDHEIM, NORWAY

This report has URL http://www.math.ntnu.no/preprint/numerics/2001/N1-2001.ps
Address: Department of Mathematical Sciences, Norwegian University of Science and Technology,
N-7491 Trondheim, Norway.

A class of low complexity intrinsic schemes for orthogonal integration *

Elena Celledoni[†] Brynjulf Owren[‡]

February 14, 2001

Abstract

Numerical integration of ordinary differential equations on the orthogonal Stiefel manifold is considered. Points on this manifold are represented as $n \times k$ matrices with orthonormal columns, of particular interest is the case when $n >> k$. Mainly two requirements are imposed on the integration schemes. First, they should have arithmetic complexity of order nk^2. Second, they should be intrinsic in the sense that they only require the ODE vector field to be defined on the Stiefel manifold, as opposed to for instance projection methods. The design of the methods makes use of retractions maps. Two algorithms are proposed, one where the retraction map is based on the QR decomposition of a matrix, and one where it is based on the polar decomposition. Numerical experiments show that the new methods are superior to standard Lie group methods with respect to arithmetic complexity, and may be more reliable than projection methods, owing to their intrinsic nature.

AMS Subject Classification: 65L05

Key Words: time integration, geometric integration, numerical integration of ordinary differential equations on manifolds, numerical analysis, Stiefel manifold, homogeneous spaces.

1 Introduction

The elements of the orthogonal Stiefel manifold are often represented as $n \times k$ matrices with orthonormal columns, where $n \geq k$,

$$\mathcal{OS}(n,k) = \{Q \in \mathbb{R}^{n\times k} : Q^TQ = I_{k\times k}\}.$$

In particular, one has $\mathcal{OS}(n,n) = \mathcal{O}(n)$ the Lie group of $n \times n$ orthogonal matrices.

*This work was in part sponsored by The Norwegian Research Council under contract no. 111038/410, through the SYNODE project. WWW:A `http://www.math.ntnu.no/num/synode/`

[†]Email: `Elena.Celledoni@math.ntnu.no`, WWW: `http://www.math.ntnu.no/~elenac/`

[‡]Email: `Brynjulf.Owren@math.ntnu.no`, WWW: `http://www.math.ntnu.no/~bryn/`

Many applications involve computations with such matrices. One is the calculation of Lyapunov exponents, see [5, 11] for an overview. Another involves optimization problems in multivariate data analysis [18].

In this paper, we shall study the problem of approximating a solution of an ODE system on $\mathcal{OS}(n,k)$. We think of $\mathcal{OS}(n,k)$ as a special case of a manifold M , and we let TM denote the tangent bundle of M. A vector field on M is then a section $F : M \to TM$ which assigns to each $Q \in M$ a tangent vector $F(Q) \in T_Q M$. The ODE system is also allowed to be nonautonomous, thus the vector field may also depend on $t \in \mathbb{R}$,

$$\dot{Q} = F(t, Q), \quad Q(t_0) = Q_0 \in M.$$

As indicated above, the orthogonal Stiefel manifold is naturally embedded in the Euclidean space of all real $n \times k$ matrices, and it is also quite common to use this representation in computer programs. The situation is similar for other manifolds M embedded in a Euclidean space E. Nevertheless, one should distinguish numerical methods which are intrinsic and those which are extrinsic. The latter type of methods makes use of an extension of the vector field F to all of E, or at least to some neighborhood of M. Such methods include the projection methods, see for instance [13, 10]. More recently, there has been an increased interest in designing intrinsic integration methods for manifolds in general. Examples of such methods are the Crouch-Grossman methods [4] and the RKMK methods proposed by Munthe-Kaas, see e.g. [15]. At least when used in a naive manner, these intrinsic methods are not entirely satisfactory from the point of view of computational complexity when applied to problems on Stiefel manifolds with $n >> k$. The reason is that they usually demand operations to be performed on $n \times n$ matrices. In particular, one typically applies operations that cost $\mathcal{O}(n^3)$ flops. In comparison, the projection of a matrix in $\mathbb{R}^{n\times k}$ onto M by means of the QR factorization cost only $\mathcal{O}(nk^2)$ flops. In this paper, we will consider maps known as retractions to impose coordinates on the Stiefel manifold. The evaluation of this map and its (inverse) derivative map can be shown to have complexity $\mathcal{O}(nk^2)$. Thus, if the evaluation of the ODE vector field can be done with the same complexity, we obtain an integration method which has an overall complexity of $\mathcal{O}(nk^2)$.

We present numerical results which demonstrate the low cost compared to Lie group methods implemented in the standard way. We also give evidence to show that intrinsic methods in some cases are much preferable to extrinsic methods like projection methods.

2 Using retraction as a coordinate map

To begin with, let us consider in general a differentiable manifold M and a differential equation given by means of a vector field $F(t,y)$ such that for each t, $F(t,\cdot)$ is a vector field $M \to TM$, so

$$y'(t) = F(t, y(t)). \tag{1}$$

It has been proposed by Shub [17] to use a smooth mapping $\mathcal{R} : TM \to M$ in the design of Newton iterations on manifolds. For the restriction $\mathcal{R}_p$ of $\mathcal{R}$ to the tangent space T_pM of M at the point $p \in M$ we require that

1. $\mathcal{R}_p$ is defined in some open ball $B(0, r_p)$ of radius r_p about $0 \in T_pM$.

2. $\mathcal{R}_p(v) = p$ if and only if $v = 0 \in T_pM$.

3. $\mathcal{R}_p'\big|_0 = Id_{T_pM}$.

The essence of the idea we present here, lies in the fact that $\mathcal{R}_p$ serves to define local coordinates of the manifold M in a neighborhood of the point p. We can thus represent the solution of the differential equation in the form

$$y(t) = \mathcal{R}_p(\sigma(t)), \quad \sigma(t) \in T_pM. \tag{2}$$

By differentiating with respect to t and by using (1) we get

$$y'(t) = \mathcal{R}_p'\big|_{\sigma(t)} (\sigma'(t)) = F(t, \mathcal{R}_p(\sigma(t))).$$

For $\sigma(t)$ sufficiently close to $0 \in T_pM$ the map $\mathcal{R}_p'\big|_{\sigma(t)}$ is invertible so we obtain a differential equation for $\sigma(t)$ as follows

$$\sigma'(t) = \left(\mathcal{R}_p'\big|_{\sigma(t)}\right)^{-1} \Big(F(t, \mathcal{R}_p(\sigma(t)))\Big). \tag{3}$$

We may now approximate (3) by using a standard ODE solver, and we may subsequently transform the result back to M via (2).

This approach is intrinsic in the sense that it does not depend on whether M has been embedded in a bigger (say Euclidean) space with a corresponding extension of the vector field F.

Example 2.1 The procedure we have just described is very similar to the Lie group methods proposed by for instance Munthe-Kaas [15], Owren and Marthinsen [16], Diele et al. [7]. In the case where $M = G$ is a Lie matrix group, the Lie groups methods are equivalent to what we get in the approach above by setting

$$\mathcal{R}_p(v) = \Phi(v \cdot p^{-1}) \cdot p,$$

where Φ is some sufficiently smooth, mapping from $\mathfrak{g}$ to G and $\mathfrak{g}$ is the Lie algebra corresponding to the Lie group G. Typically one may use $\Phi = \exp$, the matrix exponential as Munthe-Kaas did in his first papers, but other choices are possible, as discussed in the other papers cited above.

For homogeneous spaces, of which the orthogonal Stiefel manifold is an example, the methods by Munthe-Kaas apply an action by a Lie group. $\mathcal{OS}(n,k)$ is acted upon by $\mathrm{SO}(n)$, the group of $n \times n$ matrices with unit determinant. As an example of a group action, one may use left multiplication

$$\Lambda(g, p) := \Lambda_p(g) = g \cdot p, \quad p \in \mathcal{OS}(n,k),\ g \in \mathrm{SO}(n).$$

The linear space of $n \times n$ skew-symmetric matrices, denoted $\mathfrak{so}(n)$, is mapped by the matrix exponential into $\mathrm{SO}(n)$, and by composing this with the action Λ one obtains a map $\lambda_p =$

$\Lambda_p \circ \exp : \mathfrak{so}(n) \to \mathcal{OS}(n,k)$. This map λ_p is a smooth map from some neighborhood of $0 \in \mathfrak{so}(n)$ onto some neighborhood of $p \in \mathcal{OS}(n,k)$. We thus get a representation of the solution $y(t)$ near the point $p \in \mathcal{OS}(n,k)$ quite similar to (2)

$$y(t) = \lambda_p(\sigma(t)).$$

By differentiation, this leads again to a differential equation for $\sigma(t)$, but now on the space $\mathfrak{so}(n)$ of skew-symmetric $n \times n$ matrices. The obvious drawback with this approach, is that we have replaced a differential system on the manifold $\mathcal{OS}(n,k)$ having dimension $nk - k(k-1)/2$, by an equation on a linear space which has dimension $n(n-1)/2$. Whenever $n >> k$ this may lead to a large increase in the number of degrees of freedom. The advantage in using the retraction approach is that we always obtain a coordinate mapping for the manifold with exactly the same number of degrees of freedom as the dimension of the manifold. $\diamond$

Example 2.2 In the context of a Riemannian manifold, one may use the exponential mapping as defined in terms of geodesics (geodesic flow) as a retraction map. Following for instance Chavel [3], we define

$$\mathcal{R}_p(v) = \exp_p(v) = \gamma_v(1),$$

where $\gamma_v(t)$ is the geodesic emanating from p with $\dot{\gamma}(0) = v$. It is known that $\exp_p$ is defined and of maximal rank in a neighborhood of $0 \in T_pM$. The derivative map of $\mathcal{R}_p$ is related to the Jacobi field Y satisfying the Jacobi equation, see [3, p. 70-82]. We let ∇ be the Levi-Civita connection with respect to the Riemannian metric on M, and $\mathbf{R}$ the corresponding curvature tensor. We consider the vector field defined along the geodesic γ, $\gamma(0) = p$, $\dot{\gamma}(0) = v$, satisfying the boundary value problem

$$\nabla_t^2 Y + \mathbf{R}(\dot{\gamma}, Y)\dot{\gamma} = 0, \quad Y(0) = 0, \quad Y(1) = w.$$

Then

$$\left(\mathcal{R}'_p\big|_v\right)^{-1}(w) = (\nabla_t Y)(0).$$

It is not at all clear to us how one should solve these equations in general, its apparent complexity may indicate that it is more useful to consider specialized cases rather than a general framework for Riemannian manifolds. On the other hand, one should keep in mind that for certain manifolds, at least the geodesics themselves can be computed cheaply, see for instance [9] for Stiefel and Grassman manifolds. $\diamond$

3 A retraction based on the reduced QR decomposition

The tangent spaces $T_P\mathcal{OS}(n,k)$ can be characterized in various different ways, we refer to [9] for an introduction. In the following we shall just need the following characterization

$$T_P\mathcal{OS}(n,k) = \{v \in \mathbb{R}^{n\times k} : P^T v \in \mathfrak{so}(k)\}.$$

This fact follows easily by letting $Q(t)$ be a smooth curve on $\mathcal{OS}(n,k)$ satisfying $Q(0)=P$ and $\dot{Q}(0)=v$ and thereafter differentiating the relation $Q^TQ=I$ at $t=0$.

Now let $\mathcal{S}(n,k)$ be the manifold of $n\times k$ matrices of rank k. Given any matrix $A\in\mathcal{S}(n,k)$ one can apply some kind of orthogonalization procedure to the columns of A and obtain a decomposition of the form $A=QR$ where $Q\in\mathcal{OS}(n,k)$ and $R\in\mathcal{T}_+(k)$ i.e. R being an upper triangular $k\times k$ matrix with positive diagonal elements. The complexity of this operation is $2nk^2$ flops [12, p 232]. The decomposition is unique as described above. We denote the QR decomposition map (coproduct) by $\mathrm{qr}:\mathcal{S}(n,k)\to\mathcal{OS}(n,k)\times\mathcal{T}_+(k)$, and let π_1 be the projection onto the first factor. For any vector $v\in T_P\mathcal{OS}(n,k)$ we define the retraction map $\mathcal{R}_P$ relative to $P\in\mathcal{OS}(n,k)$ as

$$\mathcal{R}_P(v)=(\pi_1\circ\mathrm{qr})(P+v).$$

In words, calculate the QR decomposition of the matrix $P+v$ and keep the matrix Q. In addition to being well defined, we can also show by construction that the inverse of $\mathcal{R}_P$ exists in some neighborhood of P. By writing

$$P+v=QR, \tag{4}$$

and show that for a given $Q\in\mathcal{OS}(n,k)$ sufficiently close to P in some sense to be made clear below, we can calculate v satisfying (4) by an explicit procedure. Looking at (4) columnwise, we have

$$v_j=\sum_{i=1}^{j}R_{ij}Q_i-P_j. \tag{5}$$

We take the inner product on each side by $P_m,\ m=1,\ldots j$ and exploit the skew-symmetry of the matrix P^Tv

$$\sum_{i=1}^{j}R_{ij}\langle P_m,Q_i\rangle=\delta_{mj}-\langle v_m,P_j\rangle,\quad m=1,\ldots,j, \tag{6}$$

where δ_{mj} is the Kronecker function. This is a linear system of j equations for $R_{1j},\ldots,R_{jj}$ and can be solved as long as the jth principal minor of P^TQ is nonsingular. One obtains successively v_j from (5). It is a crucial observation that $\langle v_j,P_j\rangle=0$ such that the right hand side of (6) depends only on $v_1,\ldots,v_{j-1}$. The arithmetic complexity of this algorithm is $\mathcal{O}(nk^2+k^3)$.

For the derivative mapping $\left(\mathcal{R}'_P|_v\right)^{-1}$, we first use the chain rule to infer that

$$\left(\mathcal{R}'_P\Big|_v\right)^{-1}=\left(\mathcal{R}_P^{-1}|_Q\right)'\quad\text{where}\quad Q=\mathcal{R}_P(v).$$

Thus, we let $Q(t)$ be a curve in $\mathcal{OS}(n,k)$ such that $Q(0)=Q\in\mathcal{OS}(n,k)$ and $\dot{Q}(0)=w\in T_Q\mathcal{OS}(n,k)$. By setting $v(t)=\mathcal{R}_P^{-1}(Q(t))$ we get the relation

$$P+v(t)=Q(t)\cdot R(t),$$

which we differentiate with respect to t, and set $t = 0$ to obtain $\dot{v} := \dot{v}(0) = \left(\mathcal{R}_P^{-1}|_Q\right)'(w)$. Due to the triangular structure of the matrix R, it makes sense to consider the differentiation columnwise, so we get

$$\dot{v}_j = \sum_{i=1}^{j} \left(w_i R_{ij} + Q_i \dot{R}_{ij}\right). \tag{7}$$

Here $R_{ij} = R_{ij}(0)$ and $\dot{R}_{ij} = \dot{R}_{ij}(0)$. We can solve for $R_{1j}, \ldots, R_{jj}$ by the procedure described above. However, in using retractions for solving ODEs on manifolds, we will see that the map $\left(\mathcal{R}_P^{-1}|_Q\right)'$ is always evaluated just after one has computed $Q = \mathcal{R}_P(v)$ and thus we would in practice store the R-matrix obtained as a by-product from this calculation.

Now we can use the fact that $\dot{v} \in T_P\mathcal{OS}(n,k)$ and take again the inner product with P_m

$$\sum_{i=1}^{j} \langle P_m, Q_i \rangle \dot{R}_{ij} = -\sum_{i=1}^{j} \langle P_m, w_i \rangle R_{ij} - \langle \dot{v}_m, P_j \rangle, \quad m = 1, \ldots, j.$$

Thus, we have a linear system of j equations for the unknowns $\dot{R}_{1j}, \ldots, \dot{R}_{jj}$ whose solution exists whenever the principial minors of P^TQ are nonsingular. Finally, we substitute the obtained values for $\dot{R}_{ij}$ into (7) to obtain the desired tangent matrix $\dot{v}$.

Note that the linear systems for R_{ij}, $i = 1, \ldots, j$ and $\dot{R}_{ij}$, $i = 1, \ldots, j$ are the same, thus one may use the same LU factorization of P^TQ. Note also that when the point $Q \in \mathcal{OS}(n,k)$ is "close" to the reference point P, we will have $P^TQ \approx I$ and the LU-factorization can be done without pivoting. All the k linear systems of equations can be solved by means of the same factorization.

See also Appendix A for the `Matlab` implementation of this algorithm.

Complexity. The evaluation of $\mathcal{R}_P$ involves one addition of two $n \times k$ matrices, and a reduced QR factorization. Using for instance the modified Gram-Schmidt algorithm [12], the cost of the QR decomposition is ca $2nk^2$ flops.

The computation of $\mathcal{R}_P^{-1}$ itself is not needed in our algorithm, but we count a complexity of

$$(4k^2 + k)n + \frac{4}{3}k^3 - \frac{1}{2}k^2 - \frac{5}{6}k$$

flops. Now, for the calculation of the derivative $\left(\mathcal{R}_P^{-1}|_Q\right)'$, we found that it requires

$$(7k^2 + k)n + 2k^3 + \frac{3}{2}k^2 + \frac{1}{2}k$$

flops in the case that the matrix $R \in \mathcal{T}_+(k)$ is already given where $\mathcal{R}_P^{-1}(Q) + P = QR$. This is a reasonable assumption when integrating ODEs on the Stiefel manifold, because in the use of integration methods, one first applies the retraction to a vector $v \in T_P\mathcal{OS}(n,k)$ to obtain $Q \in \mathcal{OS}(n,k)$ where the vector field F is to be evaluated, and the matrix R is obtained as a by-product.

4 A retraction based on the reduced polar decomposition

As an alternative to QR, one can use the reduced polar decomposition where a matrix $A \in \mathcal{S}(n,k)$ is factored as

$$A \to QH, \quad Q \in \mathcal{OS}(n,k),\ H \in \mathrm{Sym}^+(k),$$

where $\mathrm{Sym}^+(k)$ are the $k \times k$ symmetric positive definite matrices. Thus, in a very similar fashion to the retraction with QR decomposition above, we now define

$$\mathcal{R}_P(v) = (\pi_1 \circ \mathrm{pol})(P+v)$$

where $\mathrm{pol} : \mathcal{S}(n,k) \to \mathcal{OS}(n,k) \times \mathrm{Sym}^+(k)$ is the polar decomposition coproduct. It is well-known that for any matrix $\mathcal{S}(n,k)$ the factors Q and H above can be calculated via the reduced singular value decomposition, say $A = V\Sigma W^T$, $V \in \mathcal{OS}(n,k)$, Σ is $k \times k$ diagonal and non-singular, and $W \in \mathcal{O}(k)$. In this case, one obtains $Q = VW^T$ and $H = W\Sigma W^T$.

The derivative map $\dot{v} = \left(\mathcal{R}_P^{-1}|_Q\right)'(w)$ is obtained in a similar way as for the QR case, we consider the curve $P + v(t) = Q(t)H(t)$ of a continuous reduced polar decomposition. Differentiating, we get

$$\dot{v} := \dot{v}(0) = wH + Q\dot{H}, \tag{8}$$

where we have set $Q = Q(0) \in \mathcal{OS}(n,k)$, $w = \dot{Q}(0) \in T_Q\mathcal{OS}(n,k)$, $H = H(0) \in \mathrm{Sym}^+(k)$, $\dot{H} = \dot{H}(0) \in \mathrm{Sym}(k)$. Now, we multiply (8) by P^T from the left, and consider the symmetric part of the resulting equation. This leads to the following Lyapunov equation for $\dot{H}$

$$M\dot{H} + \dot{H}M^T + C = 0.$$

Here $M = P^TQ$ and $C = P^TwH + Hw^TP$. It is well known that this system has a unique solution $H \in \mathrm{Sym}(k)$ if and only if the eigenvalues of the matrix $M = P^TQ$ have nonzero real parts. So $\mathcal{R}_P'|_Q$ is invertible for every Q in some neighborhood of $P \in \mathcal{OS}(n,k)$. After solving for $\dot{H}$ we obtain $\dot{v}$ by subsituting it into (8). The matrix $H \in \mathrm{Sym}^+(k)$ can be obtained by solving another Lyapunov equation, but as in the QR case, it is for the applications we have in mind, feasible to assume that this matrix is already known whenever needed in the calcuation of $\left(\mathcal{R}_P^{-1}|_Q\right)'$.

See also Appendix A for `Matlab` implementations of these algorithms.

Complexity. Using the algorithm by Golub and Reinsch, it costs approximately $14nk^2 + \frac{22}{3}k^3$ flops to form the matrices V and Σ, and an additional $2nk^2$ flops to form U and then $k^3/2$ flops to form H. In conclusion, the dominating complexity terms for calculating $\mathcal{R}_P(v)$ is

$$16\,nk^2 + 7.83\,k^3$$

flops.

To obtain the corresponding map $\left(\mathcal{R}_P^{-1}|_Q\right)'$, the dominating cost consists in calculating 2 products of $n \times k$ with $n \times k$ matrices, then another 2 products of $n \times k$ with $k \times k$, each multiplication costing $2nk^2$ flops. The Lyapunov equation is only solved for a $k \times k$ matrix, and by using e.g. the Bartels-Stewart algorithm, the cost will be approximately $25k^3$ flops. For details about this and other algorithms for solving the Lyapunov equation, see [12, p. 367], [14] and the references therein. Summing up, we get approximately

$$8\,nk^2 + 25\,k^3$$

flops for calculating $\left(\mathcal{R}_P^{-1}|_Q\right)'(w)$.

5 Runge-Kutta methods based on retractions

We now consider in some more detail how to solve ODEs on manifolds by using a retraction map. We will now assume that the problem is given by the user in the form (1) meaning that there is a procedure, say, `F` available which for any $(t, y) \in \mathcal{D} \subset \mathbb{R} \times M$ returns the derivative $F(t, y) \in T_y M$. In addition, the user must provide an initial value $y_0 \in M$, and an initial stepsize h.

Suppose a Runge-Kutta method is given, with coefficients (a_{ij}) where $i, j = 1, \dots, s$ and (b_i), $i = 1, \dots, s$. In particular, the method is explicit if $a_{ij} = 0,\ j \geq i$. As usual we denote $c_i = \sum_j a_{ij}$.

The following algorithm integrates an ODE system on an arbitrary manifold in which a retraction $\mathcal{R}_p$ is defined.

Algorithm 5.1
Input y_0, stepsize h, and RK parameters a_{ij}, b_i.
for $n = 0, 1, \dots$
    for $i = 1, \dots, s$
        $u_i := h \sum_j a_{ij} k_j$
        $Y := \mathcal{R}_{y_n}(u_i)$
        $m_i := F(t_n + hc_i, Y)$
        $k_i := \left(\mathcal{R}_{y_n}^{-1}|_Y\right)'(m_i)$
    end for
    $v := h \sum_i b_i k_i$
    $y_{n+1} := \mathcal{R}_{y_n}(v)$
    [Update stepsize h if desired]
end for

$\diamond$

Note that if the Runge-Kutta coefficients are those of an explicit method, also Algorithm 5.1 will be explicit. In fact, in the approach presented here, explicit methods are recommended because implicitness may cause an even higher increase in computational complexity than what is the case for standard Runge-Kutta methods.

See also Appendix A for `Matlab` programs for the functions $\mathcal{R}_p$ and $\left(\mathcal{R}_p^{-1}|_Q\right)'$ when the retractions are based on the reduced QR and polar decompositions.

6 Numerical experiments

All the numerical experiments are performed in `Matlab` . We compare the methods presented in this paper, Runge-Kutta methods based on retractions, RKRqr (based on the qr decomposition) and RKRp (based on the polar decomposition), with the following other methods:

- Projection Runge-Kutta method (PRK): the method performs a projection based on the QR-factorization at each time step of an explicit Runge-Kutta method (extrinsic).
- The method recently proposed in [1] (SPRK): a splitting method applied to the perturbed problem obtained by adding the term $-\tau y(I_k - y^T y)$ to the original vector field. The parameter τ is suitably chosen in order to make $\mathcal{OS}(n,k)$ an attracting manifold. The resulting method is a projection method (extrinsic).
- Runge-Kutta Munthe-Kaas [15] methods (RKMK): generalization of the classical Runge-Kutta methods based on the use of the Lie group action action $\Lambda(g, Y_0) = g \cdot Y_0$ and $g = \exp(\sigma)$, for $\sigma \in \mathfrak{so}(n)$ and $Y_0 \in \mathcal{OS}(n,k)$ (intrinsic).

The results obtained with the RKMK methods are not reported in the plots, but we will comment on their performance in the experiments in various points of this section.

All the methods are based on the classical explicit Runge-Kutta method of order 4.

The numerical experiments are divided in two parts. In the first section we will compare the intrinsic methods with the extrinsic ones.

In the second section we will illustrate the performance of the methods based on retractions when applied to the computation of the Lyapunov exponents of a ring of oscillators.

6.1 Intrinsic versus extrinsic

As a first experiment we consider the following initial value problem:

$$y' = F(y) = A(y)y + \lambda y(I_k - y^T y), \tag{9}$$

for $y(0) \in \mathcal{OS}(n,k)$, $n = 1000$, $k = 4$. Here $A(y)$ is a banded skew-symmetric $n \times n$ matrix whose nonzero entries are

$$A_{i,i+m} = -A_{i+m,i} = y_{i,m}, \;\; 1 \le i \le n - m, \;\; 1 \le m \le k. \tag{10}$$

The initial value, y_0, is obtained as the first factor of the reduced QR-factorization of a random 1000×4 matrix, in `Matlab` we used the command `[y0,r]=qr(rand(n,k),0)`.

Note that the term $y(I_k - y^T y)$ in (9) is zero for any $y \in \mathcal{OS}(n,k)$, thus, the solution of (9) is independent of λ as long as $y_0 \in \mathcal{OS}(n,k)$. It might be desirable that the numerical approximation inherits this property.

In some cases it might be necessary or advantageous to rewrite (9) in a "strong skew-symmetric form", i.e. $F(t,y) = H(t,y)y$ with $H(t,y) \in \mathfrak{so}(n)$ (for the RKMK methods for example).

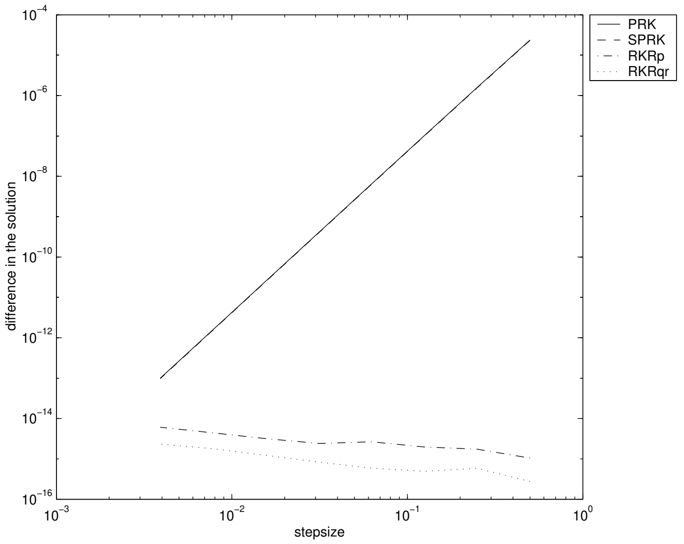

Figure 1: Difference in the solution for $\lambda = 0$ and $\lambda = 10$

Note that in the numerical experiments we always assume the value of the vector field $F(t, y)$ to be given by a black box program that we are not allowed to modify. Since $A(y)$ is a banded matrix the cost of computing F, as given in (9), is effectively $\mathcal{O}(nk^2)$ flops.

Given $F = F(t, y)$ in the tangent space at y to the Stiefel manifold we then consider the following strong skew-symmetric formulation

$$F = \left(\alpha(y)y^T - y\alpha^T(y)\right) y, \qquad \alpha(y) = y \;\mathrm{tril}(y^T F) + (F - yy^T F). \tag{11}$$

The formulation (11) is then computed just assuming the knowledge of F. Note that the formulation in the strong skew-symmetric form, as pointed out in [1] can be of crucial importance for extrinsic methods, as it makes $\mathcal{OS}(n, k)$ a strong invariant manifold. We apply the projection methods to this reformulation for the problem (9) in the case $\lambda \neq 0$.

In Figure (1), for different values of the step-size h, we plotted the norm of the difference of the numerical approximations produced by the methods after one time-step for the problem (9) with $\lambda = 0$ and $\lambda = 10$ respectively. The dependence on λ is quite evident for both the extrinsic methods, although the norm of the difference of the two numerical solutions decreases as the step-size goes to zero. Note that the lines of the PRK and the SPRK lie on top of each other in the plot.

For the intrinsic methods the global error remains more or less the same for the two different values of λ, and as we can see form the figure, the difference of the two approximate solutions is of the order of machine accuracy. Similar experiments gave analogous results for the RKMK methods.

Based on this experiment it seems that the use of intrinsic methods should be preferred to the use of extrinsic ones.

We note that the formulation (11) requires $\mathcal{O}(nk^2)$ flops.

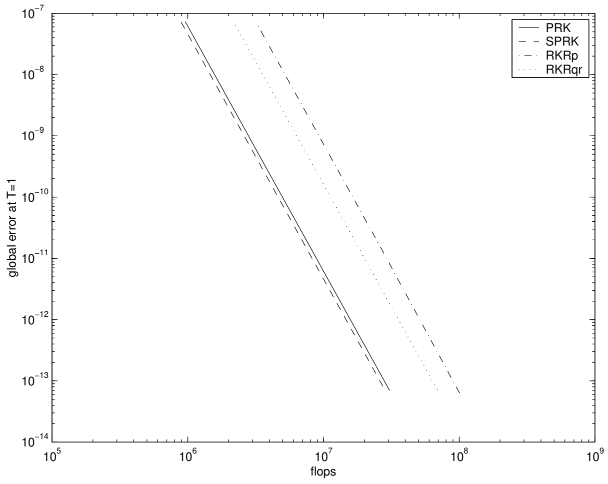

Figure 2: Global error versus the number of flops, $\lambda = 0$

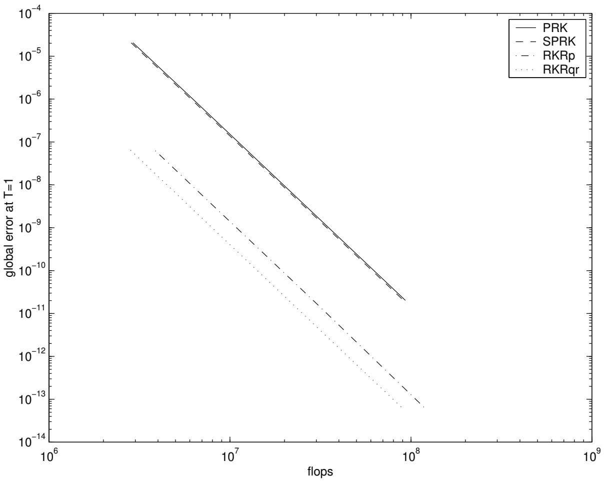

Figure 3: Global error versus the number of flops, $\lambda = 10$

In the next two figures we compare the cost of the four methods as applied to the problem (9). We plot the global error at $T = 1$ on the y-axis against the number of flops on the x-axis, in Figure (2) for $\lambda = 0$, in Figure (3) for $\lambda = 10$.

As we can notice in the case $\lambda = 0$, the achieved global error per amount of flops for the two extrinsic methods is lower in norm than for the intrinsic methods.

However for $\lambda = 10$ the RKRqr and the RKRp perform better then the two extrinsic methods.

The use of the RKMK methods in these experiments led to much more time consuming calculations. It seems that a naive implementation of these techniques causes unacceptable computational costs. At present new implementation techniques that can reduce the cost of the RKMK methods to a cost of $\mathcal{O}(nk^2)$ flops, are under investigation, [2].

6.2 Computing Lyapunov exponents

In the last numerical experiment we apply the RKRqr method to the computation of the Lyapunov exponents of the following ODEs system describing a ring of m Duffing oscillators:

$$\begin{array}{l} \ddot{y} + \alpha\left(y^2 - 1\right)\dot{y} + \omega^2 y = 0, \\ \ddot{x}_i + d\dot{x}_i + \beta\left[V'(x_i - x_{i-1}) - V'(x_{i+1} - x_i)\right] = \sigma y \delta_{i,1}, \quad i = 1, \ldots, m. \end{array} \tag{12}$$

The ring is forced externally by $y(t)$. Here $V(x) = (\frac{x^2}{2}) + (\frac{x^4}{4})$, $\delta_{i,j} = 0$ for $i \neq j$ and $\delta_{i,i} = 1$, and we impose periodic boundary conditions ($x_0 = x_m$ and $x_{m+1} = x_1$). In the experiments $m = 15$ and $\alpha = 1$, $\omega = 1.6$, $\beta = 1$, $\sigma = 2$, and $d = 0.4$. This test problem has been considered in [8], [6], [1].

The Lyapunov exponents of a continuous dynamical system $x' = F(x)$, ($x(t) \in \mathbb{R}^n$) provide a qualitative measure of its complexity, and can be defined as follows. Consider the linearization $A(t)$ of $x' = F(x)$ along a trajectory $x(t)$, and the solution U of the matrix problem

$$\dot{U} = A(t)U, \quad U(0) = U_0, \quad n \times n,$$

then the logarithms of the eigenvalues of the matrix

$$\Lambda = \lim_{t\to\infty} \left(U(t)^T U(t)\right)^{\frac{1}{2t}},$$

are the Lyapunov exponents for the given dynamical system. In [6] the authors describe a procedure for computing just k of the n Lyapunov exponents of a dynamical system. The strategy is based on solving a suitable initial value problem on $\mathcal{OS}(n,k)$ and computing a quadrature of the diagonal entries of a $k\times k$ matrix valued function. The initial value problem is defined as follows:

$$\dot{Q} = \left(A - QQ^T A + QSQ^T\right) Q,$$

with random initial value in $\mathcal{OS}(n,k)$ and

$$S_{k,j} = \begin{cases} (Q^T AQ)_{k,j}, & k > j, \\ 0, & k = j, \\ -(Q^T AQ)_{j,k}, & k < j, \end{cases} \qquad k, j = 1, \dots, p.$$

It can be shown that the i-th Lyapunov exponent λ_i can be obtained as

$$\lambda_i = \lim_{t\to\infty} \frac{1}{t} \int_0^t B_{i,i}(s)ds, \quad i = 1, \dots, k, \tag{13}$$

and

$$B = Q^T AQ - S.$$

In the numerical experiments we use the trapezoidal rule to approximate the integral (13) and compute λ_i ($i = 1, \dots, k$), we refer to the original paper [6] for further details on the method.

In our experiment we have considered a time interval $[0, 4000]$ and the step-size $h = 0.01$. The initial value for computing the trajectory $x(t)$ is an after transient numerical solution. The trajectory is computed numerically using a midpoint rule with stepsize $h/8$.

Considering the error introduced by substituting the integral (13) with the quadrature the results are in good agreement with those obtained in [8] and give the correct qualitative information about the considered dynamical system (all Lyapunov exponents are negative).

7 Conclusion

We have developed intrinsic integration methods of complexity $\mathcal{O}(n\,k^2)$ for solving ODEs on orthogonal Stiefel manifolds.

We would like to emphasize that in order to see this complexity in actual computations, it is important that the evaluation of the vector field F (as in $y' = F(t, y)$) does not involve cost of higher arithmetic complexity. It is for instance quite common to phrase problems on $\mathcal{OS}(n,k)$ in the form

$$\dot{y} = F(t, y) = A(t, y)y \tag{14}$$

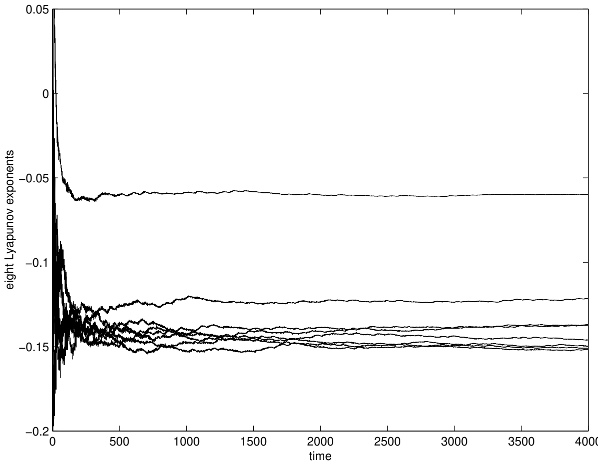

Figure 4: Lyapunov exponents for a ring of oscillators: RKRqr on the interval $[0, 4000]$

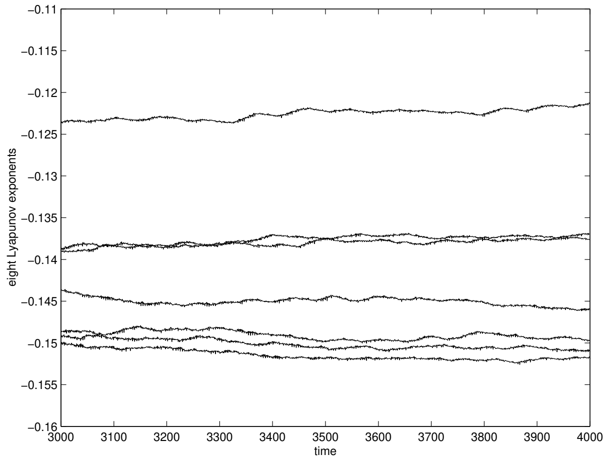

Figure 5: Lyapunov exponents for a ring of oscillators: RKRqr detail on the interval $[3000, 4000]$

where $A(t, y) : \mathbb{R} \times \mathcal{OS}(n, k) \to \mathfrak{so}(n)$. If there is no sparsity in $A(t, y)$, it seems impossible to calculate $F(t, y)$ in less than $\mathcal{O}(n^2\, k)$ flops, and the gain in using the presented type of integration methods may not be significant. However, since each tangent space $T_p\mathcal{OS}(n, k)$ only has dimension $d_{n,k} = nk - k(k-1)/2$ it is always possible to find local parametrizations of the vector field F using only $d_{n,k}$ degrees of freedom.

There is no doubt that the methods presented here seem to have something in common with projection methods. Although we have seen no precise definition of a projection method, we still claim that there are important differences between the methods presented here and those normally referred to as projection methods. We have therefore chosen to focus on intrinsicness as the main feature to distinguish the two types of methods. The term projection often indicates a map from higher to lower dimension, but in our case, the retraction map is seen as a map from each tangent space $T_p\mathcal{OS}(n, k)$ onto a neighborhood of $p \in \mathcal{OS}(n, k)$, and in this sense it is a local diffeomorphism. The main feature of the proposed methods is that they make no use of an extension of the vector field F to points not lying on the Stiefel manifold. It is certainly true that one can for all practical purposed find such an extension of the vector field F to all of $\mathbb{R}^{n\times k}$, for instance the imbedding theorem of Nash ensures this for Riemannian manifolds. However, the extension is not unique, and we find it unnatural that the numerical approximation produced should depend on the particular extension which is chosen. Our numerical experiments confirm that different extensions can lead to completely different numerical approximations for projection methods, whereas the ones presented here yield identical results modulo rounding errors.

Acknowledgments

The authors would like to thank Luciano Lopez and Nicoletta Del Buono for inspiring discussion on this topic, and Nick Higham for providing `Matlab` functions for the polar decom-

position.

A `Matlab` programs

A.1 The reduced QR decomposition

The following program calculates $Q = \mathcal{R}_P(V)$ satisfying $P + V = QR$. The matrix Q is the matrix with orthonormal columns in the reduced QR decomposition of $P + V$. The matrix R is upper triangular with positive diagonal elements and is also returned.

```
function [Q,R]=retr(X,V)
%
%
%
[Q,R]=qr(X+V,0);
[n,p]=size(X);

for k=1:p,
    if R(k,k)<0,
Q(:,k)=-Q(:,k);
R(k,k:p)=-R(k,k:p);
    end
end
```

The following program calculates $\left(\mathcal{R}_P^{-1}|_Q\right)'(W)$ given also the matrix R satisfying $QR = P + \mathcal{R}_P^{-1}(Q)$.

```
function [Vd]=dretinv(X,Q,W,R)
%
%  [Vd]=dretinv(X,Q,W,R)
%
%  X: Center point of coordinate system (on Stiefel manifold)
%  Q: Base point of derivative map
%  W: Argument to which derivative is applied
%  R: One has QR=X-V where V->Q is the retraction.
%
[n,p]=size(Q);
Vd=zeros(n,p);
M=X'*Q;
[L,U]=lu(M);
N=X'*W;

for j=1:p,
   r=R(1:j,j);
```

```
   bd=[-Vd(:,1:j-1)'*X(:,j);0];
   bd=bd-N(1:j,1:j)*r;
   c=L(1:j,1:j)\bd;
   rd=U(1:j,1:j)\c;
   Vd(:,j)=Q(:,1:j)*rd+W(:,1:j)*r;
end

return
```

A.2 The reduced polar decomposition

The following program calculates $Q = \mathcal{R}_P(V)$ satisfying $P + V = QH$. The matrix Q is the orthogonal factor in the reduced polar decomposition of $P + V$. The matrix H is symmetric, positive definite, and it is also returned.

```
function [Q,H]=pretr(P,V)
%PRETR Retraction map on Stiefel manifold based on reduced polar decomposition
%      [Q,H]=retr(P,V) computes matrices Q and H such that P+V=Q*H,
%      with H positive definite. The matrices
%      P,V,Q are all nxk and H is kxk.
%
[U,S,W]=svd(P+V,0);
Q=U*W';
H=W*S*W';

return
```

The following program calculates $\left(\mathcal{R}_P^{-1}|_Q\right)'(W)$ given also the matrix R satisfying $QR = P + \mathcal{R}_P^{-1}(Q)$.

```
function [Vd]=pdretinv(P,Q,W,H)
%DRETINV The derivative of the inverse of the retraction map on the
%        Stiefel manifold based on the reduced polar decomposition.
%        [Vd]=dretinv(P,Q,W,H) returns the derivative map applied
%        to the matrix W.
%  P: nxk matrix. Center point of coordinate system (on Stiefel manifold)
%  Q: nxp matrix. Base point of derivative map
%  H: One has QH=P+V where V->Q is the retraction. This matrix
%     is really redundant, but it saves computational cost and is
%     usually obtained as a byproduct from the retraction map.
%
M=P'*Q;
A=P'*W;
AH=A*H;
```

```
Hd=lyap(M,AH+AH');
Vd=W*H+Q*Hd;

return
```

References

[1] T. J. Bridges and S. Reich. Computing Lyapunov exponents on a Stiefel manifold. Technical report, University of Surrey, April 2000.

[2] E. Celledoni and B. Owren. Lie group methods for the integration of ODEs on the Stiefel manifold. In preparation.

[3] I. Chavel. *Riemannian geometry: a modern introduction.* Number 108 in Cambridge tracts in mathematics. Cambridge University Press, 1993.

[4] P. E. Crouch and R. Grossman. Numerical integration of ordinary differential equations on manifolds. *J. Nonlinear Sci.*, 3:1–33, 1993.

[5] L. Dieci, R. D. Russel, and S. Van Vleck. On the computation of Lyapunov exponents for continuous dynamical systems. *SIAM J. Numer. Anal.*, 34:402–423, 1997.

[6] L. Dieci and E. S. Van Vleck. Computation of a few Lyapunov exponents for continuous and discrete dynamical systems. *Applied Numerical Mathematics*, 17:275–291, 1995.

[7] F. Diele, L. Lopez, and R. Peluso. The Cayley transform in the numerical solution of unitary differential systems. *Adv. Comput. Math.*, 8(4):317–334, 1998.

[8] U. Dressler. Symmetry property of the Lyapunov spectra of a class of dissipative dynamical systems with viscous damping. *Phys. Rev. A*, 38(4):2103–2109, 1988.

[9] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Applic.*, 20(2):303–353, 1999.

[10] E. Eich-Soellner and C. Führer. *Numerical methods in multibody dynamics.* European Consortium for Mathematics in Industry. B. G. Teubner, Stuttgart, 1998.

[11] K. Geist, U. Parlitz, and W. Lauterborn. Comparison of different methods for computing Lyapunov exponents. *Prog. Theor. Phys.*, 83:875–893, 1990.

[12] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, 1996.

[13] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems.* Springer, 2nd edition, 1996.

[14] S. J. Hammarling. Numerical solution of the stable non-negative definite Lyapunov equation. *IMA J. Numer. Anal.*, 2:303–323, 1982.

[15] H. Munthe-Kaas. High order Runge–Kutta methods on manifolds. *Appl. Numer. Math.*, 29:115–127, 1999.

[16] B. Owren and A. Marthinsen. Integration methods based on canonical coordinates of the second kind. Technical Report Numerics No. 5/1999, The Norwegian University of Science and Technology, Trondheim, Norway, 1999.

[17] M. Shub. Some remarks on dynamical systems and numerical analysis. In L. Lara-Carrero and J. Lewowicz, editors, *Dynamical systems and partial differential equations*, VII ELAM, pages 69–92, 1986.

[18] N. Trendafilov. Projected gradient approach to multivariate data analysis, 2000. Submitted for publication.