Applications of Mathematics

Vladimír Janovský; O. Liberda

Continuation of invariant subspaces via the Recursive Projection Method

Applications of Mathematics, Vol. 48 (2003), No. 4, 241–255

Persistent URL: http://dml.cz/dmlcz/134531

48 (2003) APPLICATIONS OF MATHEMATICS No. 4, 241–255

CONTINUATION OF INVARIANT SUBSPACES VIA THE RECURSIVE PROJECTION METHOD

V. Janovský, O. Liberda, Praha

(Received May 31, 2001)

Abstract. The Recursive Projection Method is a technique for continuation of both the steady states and the dominant invariant subspaces. In this paper a modified version of the RPM called projected RPM is proposed. The modification underlines the stabilization effect. In order to improve the poor update of the unstable invariant subspace we have applied subspace iterations preconditioned by Cayley transform. A statement concerning the local convergence of the resulting method is proved. Results of numerical tests are presented.

Keywords: steady states, pathfollowing, stability exchange, unstable invariant subspace

MSC 2000: 65H17, 47H17

1. Introduction

We consider a parameter dependent dynamical system

$$\dot{u} = G(u, \lambda), \tag{1}$$

where $G\colon\ \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is a sufficiently smooth vector field. In order to find the steady states of (1), i.e. to solve the system of nonlinear equations

$$G(u, \lambda) = 0, \tag{2}$$

we can use a standard predictor-corrector continuation, see e.g. [8], [1]. To indicate the stability exchange in the course of pathfollowing is much harder. The Recursive Projection Method (RPM), see [11], could serve this purpose.

The classical RPM computes the steady states of (1) as parameter dependent fixed points of a mapping F, namely

$$F(u, \lambda) = u. \tag{3}$$

At each continuation step, the state space $\mathbb{R}^n$ is split as

$$\mathbb{R}^n = \mathbb{P} \oplus \mathbb{Q}, \quad \mathbb{Q} = \mathbb{P}^{\perp}$$

where $\mathbb{P}$ is the invariant subspace of $F_u(u, \lambda)$ containing all unstable modes. Denoting by P and Q the orthogonal projectors on $\mathbb{P}$ and $\mathbb{Q}$ respectively, the problem (3) can be reformulated as follows: Find $p \in \mathbb{P}$ and $q \in \mathbb{Q}$ such that

$$p = PF(p + q, \lambda), \tag{4}$$

$$q = QF(p + q, \lambda). \tag{5}$$

Under certain assumptions, see [11], QF is contractive. Therefore, fixed points of (5) can be computed via Picard iterations. In order to find p, Newton-like methods are suggested. The choice of F also includes the possibility to use any black-box ODE solver in the Picard iteration step, see [11]. This step can be interpreted as a dynamical simulation. As far as the stability exchange is concerned, the path of $\mathbb{P}$'s is supposed to be one of the outputs of RPM. Therefore, the unstable modes can be extracted from a small dimensional $\mathbb{P}$ at a reasonable cost.

The presence of modes with large negative real parts (which are irrelevant for the bifurcation analysis) can slow down the RPM algorithms rapidly. An attempt to eliminate the influence of these modes by a preconditioning was made in [3].

Note that the technique for a continuation of limit cycles, [10], originated from [11].

According to our own experience with RPM, [6], the detection of Hopf bifurcation points is not reliable. From the theoretical point of view, the relationship between the spectra $\sigma(G_u)$ and $\sigma(F_u)$ is not clear (except for a very simple black-box solver, certainly without an adaptive time-stepping); this is crucial for the detection of Hopf bifurcation points. In general, the weak point of RPM is a poor approximation of $\mathbb{P}$.

In [7] we introduced a modification of RPM which we refer to as *Projected* RPM; note that we were inspired by [2]. Projected RPM attempts to solve (2) directly without a fixed-point reformulation (3). Let us point out the advantages:

- In the dynamical simulation step, a suitably projected vector field is integrated. It underlines the stabilisation effects.
- The differential G_u is often explicitly available. It is not true for F_u due to the machinery behind the fixed-point formulation.

Concerning an update of $\mathbb{P}$: In the original RPM, subspace iterations were suggested. However, subspace iterations tend to span an invariant subspace corresponding to the eigenvalues with large moduli. Unfortunately, these need not be the rightmost eigenvalues.

Therefore we propose to use subspace iterations preconditioned by Cayley transform, see [4]. The effect is that the rightmost modes are mapped outside the unit circle.

The paper is organized as follows: In Section 2 the Projected RPM is formulated; a pseudocode is given. Important parts of the algorithm such as a correction of the basis of $\mathbb{P}$ and changing the dimension of $\mathbb{P}$ are discussed in detail. In Section 3 we state and prove a theorem about the local convergence of the Projected RPM. The last section contains results of a numerical experiment.

2. Projected RPM

First we introduce some notation. Let $\Gamma \subset \mathbb{R}^n \times \mathbb{R}$ be the solution set to (2) and let $(u^*, \lambda^*) \in \Gamma$ be an arbitrary point. By A we denote the Jacobian matrix

$$A = D_u G(u^*, \lambda^*).$$

Stability properties of the steady state u^* are determined by the eigenvalues $\mu_1, \ldots, \mu_n$ of A. Let us arrange them decreasingly with respect to their real parts. Consider $\gamma < 0$ such that

$$\Re\mu_1 \geqslant \ldots \geqslant \Re\mu_m \geqslant \gamma > \Re\mu_{m-1} \geqslant \ldots \geqslant \Re\mu_n. \tag{6}$$

Let us call this particular γ the *stability boundary*. We say that a particular eigenvector v of A exceeds the stability boundary γ if the relevant eigenvalue μ satisfies $\Re\mu \geqslant \gamma$.

Let $\mathbb{P}$ be the invariant subspace corresponding to the first m eigenvalues and let Z_1 be an orthonormal basis in $\mathbb{P}$. Thus $Z_1 \in \mathbb{R}^{n\times m}$ satisfies the equations

$$AZ_1 = Z_1\Lambda_1, \tag{7}$$

$$Z_1^T Z_1 = I_m, \tag{8}$$

where $I_m \in \mathbb{R}^{m\times m}$ is the identity matrix and $\Lambda_1 \in \mathbb{R}^{m\times m}$ is a square matrix with eigenvalues $\mu_1, \ldots, \mu_m$. Further let us define the orthogonal complement

$$\mathbb{Q} = \mathbb{P}^\perp,$$

and let Z_2 be an orthonormal basis in $\mathbb{Q}$, i.e.

$$Z_2^T Z_2 = I_{n-m}.$$

From the definition of $\mathbb{Q}$ it follows that

$$Z_1^T Z_2 = 0 \in \mathbb{R}^{m\times(n-m)}, \quad Z_2^T Z_1 = 0 \in \mathbb{R}^{(n-m)\times m}. \tag{9}$$

Note that $Z_1 Z_1^T$ and $Z_2 Z_2^T$ are orthogonal projectors.

For each vector $u \in \mathbb{R}^n$ there exist unique vectors $p \in \mathbb{R}^m$ and $q \in \mathbb{R}^{n-m}$ such that $u = Z_1 p + Z_2 q$. The assertion is proved by setting $p = Z_1^T u$ and $q = Z_2^T u$. In particular, $u^* = Z_1 p^* + Z_2 q^*$.

2.1. Coupled iterations.

System (2) can be split into two systems

$$G^P(p, q, \lambda) \equiv Z_1^T G(Z_1 p + Z_2 q, \lambda) = 0, \tag{10}$$

$$G^Q(p, q, \lambda) \equiv Z_2^T G(Z_1 p + Z_2 q, \lambda) = 0, \tag{11}$$

where

$$G^P\colon\ \mathbb{R}^m \times \mathbb{R}^{n-m} \times \mathbb{R} \longrightarrow \mathbb{R}^m \quad \text{and} \quad G^Q : \mathbb{R}^m \times \mathbb{R}^{n-m} \times \mathbb{R} \longrightarrow \mathbb{R}^{n-m}$$

are projections of $G(u, \lambda)$ onto $\mathbb{R}^m$ and $\mathbb{R}^{n-m}$, respectively.

Let $u^{(k)}$ be the k-th approximation to u^*, $u^{(k)} = Z_1 p^{(k)} + Z_2 q^{(k)}$. Let $\lambda = \lambda^*$. To define $u^{(k+1)}$ we perform one step of coupled iteration: Equation (10) is solved with respect to the unknown p by Newton's method. Namely, we set

$$p^{(k+1)} = p^{(k)} - (D_p G^P(p^{(k)}, q^{(k)}, \lambda))^{-1} G^P(p^{(k)}, q^{(k)}, \lambda). \tag{12}$$

Equation (11) is solved by means of dynamical simulation: Consider the parameter dependent dynamical system

$$\dot{q}(t) = G^Q(p, q(t), \lambda), \tag{13}$$

$$q(0) = q. \tag{14}$$

Let $\psi(t, q; p, \lambda)$ be the relevant flow. For a fixed time increment Δt we set

$$q^{(k+1)} = \psi(\Delta t, q^{(k)}; p^{(k)}, \lambda). \tag{15}$$

In practice we approximate the flow by means of a blackbox solver for ODEs. For the numerical stability of this step it is important that the projected vector field G^Q does not contain unstable modes.

Let us resume that one step of coupled iteration can be written in the form

$$p^{(k+1)} = F^P(p^{(k)}, q^{(k)}, \lambda), \tag{16}$$

$$q^{(k+1)} = F^Q(p^{(k)}, q^{(k)}, \lambda), \tag{17}$$

$$u^{(k+1)} = Z_1 p^{(k+1)} + Z_2 q^{(k+1)}, \tag{18,}$$

where

$$F^P(p, q, \lambda) = p - (D_p G^P(p, q, \lambda))^{-1} G^P(p, q, \lambda) \tag{19}$$

and

$$F^Q(p, q, \lambda) = \psi(\Delta t, q; p, \lambda). \tag{20}$$

In fact, the basis Z_2 need not be computed at all. Indeed, according to the equation (18) we only need to evaluate the quantity $Z_2 q$. To this end we set $v(t) = Z_2 q(t)$. After differentiating v with respect to t and inserting into (13), (14) we arrive at the dynamical system

$$\dot{v}(t) = Z_2 Z_2^T G(Z_1 p + v(t), \lambda) = (I - Z_1 Z_1^T) G(Z_1 p + v(t), \lambda), \tag{21}$$

$$v(0) = Z_2 q = v. \tag{22}$$

Let $\varphi(t, v; p, \lambda)$ be the relevant flow of the above dynamical system. Then $Z_2 q^{(k+1)}$ in (18) reads

$$Z_2 q^{(k+1)} = \varphi(\Delta t, v; p^{(k)}, \lambda) \tag{23}$$

with $v = u^{(k)} - Z_1 p^{(k)}$.

2.2. Pathfollowing.

We intend to use the coupled iterations (16)–(18) in the context of a standard pathfollowing $(u(s), \lambda(s))$ of (2). The idea is to project $(u(s), \lambda(s)) \in \mathbb{R}^{(n+1)}$ to $(p(s), \lambda(s)) \in \mathbb{R}^{(m+1)}$.

We consider a predictor-corrector continuation. Let (u_0, λ_0) be a point on Γ. Let $(\dot{u}, \dot{\lambda})$ be the unit tangent vector (or its approximation) at (u_0, λ_0). Let $(u^{(0)}, \lambda^{(0)})$ be the relevant *predictor*, i.e.

$$u^{(0)} = u_0 + \dot{u}\,\delta s, \quad \lambda^{(0)} = \lambda_0 + \dot{\lambda}\,\delta s \tag{24}$$

where $\delta s > 0$ is the arclength increment. Note that in the actual computation we have used the secant method.

Let $A = D_u G(u_0, \lambda_0)$. Let m modes of A exceed the stability boundary. These modes span an invariant subspace. Let Z_1 be an orthonormal basis of this subspace. Finally, let $\dot{p} = Z_1^T \dot{u}$.

We generate the sequence $(u^{(k)}, \lambda^{(k)})$ of *correctors* via the following recurrence: Instead of (12), we require

$$D_p G^P(p^{(k)}, q^{(k)}, \lambda^{(k)})(p^{(k+1)} - p^{(k)}) + D_\lambda G^P(p^{(k)}, q^{(k)}, \lambda^{(k)})(\lambda^{(k+1)} - \lambda^{(k)}) = -G^P(p^{(k)}, q^{(k)}, \lambda^{(k)}) \tag{25}$$

with the constraint

$$\dot{p}^T (p^{(k+1)} - p^{(k)}) + \dot{\lambda}(\lambda^{(k+1)} - \lambda^{(k)}) = 0, \tag{26}$$

see [11]. The linear system (25), (26) is to be solved for $p^{(k+1)}$ and $\lambda^{(k+1)}$. Motivated by (18) and (23), we set

$$u^{(k+1)} = Z_1 p^{(k+1)} + \varphi(\Delta t, u^{(k)} - Z_1 p^{(k)}; p^{(k)}, \lambda^{(k)}). \tag{27}$$

The corrector step is resumed in Section 2.4.

2.3. Approximation of the basis Z_1.

After the predictor-corrector step we need to update Z_1. Note that also the dimension m, see (6), may change from time to time.

In [11] it was suggested to use *subspace iterations* (in principle) to modify Z_1. These iterations converge to *dominant* modes of A which are not necessarily the rightmost ones.

The remedy is to use the so called *Cayley transform* of A. The resulting algorithm is referred to as subspace iterations preconditioned by the Cayley transform in literature. In the following we will outline the basic idea behind it, for details see [4].

The Cayley transform of a matrix A is defined by

$$\mathcal{C}(A) = (A - \alpha_1 I)^{-1}(A - \alpha_2 I)$$

for $\alpha_2 < \alpha_1$, $\alpha_1 \notin \sigma(A)$. The most important property is that it maps the eigenvalues with $\Re\mu < \frac{1}{2}(\alpha_1 + \alpha_2)$ inside the unit circle and the eigenvalues with $\Re\mu > \frac{1}{2}(\alpha_1 + \alpha_2)$ outside the unit circle.

Given the stability boundary $\gamma < 0$, we fix some $\omega > 0$ and

$$\alpha_1 = \gamma + \omega, \quad \alpha_2 = \gamma - \omega. \tag{28}$$

Let m eigenvectors of A exceed the stability boundary γ; let Z_1 be an orthonormal basis of the relevant invariant subspace.

Initialize $Q^{(0)} \in \mathbb{R}^{n\times m}$ randomly (it should not be deficient in Z_1). We consider the following iterative process:

do
$\mathtt{V}^{(\mathtt{k})} = \mathcal{C}(\mathtt{A})\mathtt{Q}^{(\mathtt{k}-1)}$;
$\mathtt{Q}^{(\mathtt{k})} = \mathrm{orth}(\mathtt{V}^{(\mathtt{k})})$;
until convergence

Then $Q^{(k)} \longrightarrow Z_1$.

In the above loop and in the sequel, $\mathtt{orth(V)} \in \mathbb{R}^{n\times m}$ is the orthonormalisation of a given $V \in \mathbb{R}^{n\times m}$ via the modified Gramm-Schmidt process, see e.g. [5].

So far the revision. In our context, $A = D_u(u_0, \lambda_0)$ where (u_0, λ_0) was just updated by the predictor-corrector, see Section 2.2. Obviously, we do not know the relevant Z_1 and m. On the other hand, we are able to recall Z_1 and m at the previous continuation step. They may play the role of initial guesses.

We propose the following

procedure $\mathtt{CorrectBasisViaCayley}(\mathtt{u}_0, \lambda_0, \mathtt{Z}_1, \mathtt{m})$
$\mathtt{A} = \mathtt{G}_\mathtt{u}(\mathtt{u}_0, \lambda_0)$;
Augment $\mathtt{Z}_1$ by $\mathtt{r}$ randomly initialized columns;
$\mathtt{Q}^{(0)} = \mathrm{orth}(\mathtt{Z}_1)$;
$\mathcal{C}(\mathtt{A}) = (\mathtt{A} - \alpha_1\mathtt{I})^{-1}(\mathtt{A} - \alpha_2\mathtt{I})$
for $\mathtt{k} = 1, \ldots, \mathtt{k}_{\max}$ **do**
$\mathtt{V}^{(\mathtt{k})} = \mathcal{C}(\mathtt{A})\mathtt{Q}^{(\mathtt{k}-1)}$;
$\mathtt{Q}^{(\mathtt{k})} = \mathtt{orth}(\mathtt{V}^{(\mathtt{k})})$;
end for
$\mathtt{Q} = \mathtt{Q}^{(\mathtt{k}_{\max})}$; $\mathtt{H} = \mathtt{Q}^{\mathrm{T}}\mathtt{A}\mathtt{Q}$;
Compute $\sigma(\mathtt{H}) = \{\mu_1, \ldots, \mu_{\mathtt{m}+\mathtt{r}}\}$;
Find $\mathtt{z}_\mathtt{k}$: $\mathtt{H}\mathtt{z}_\mathtt{k} = \mu_\mathtt{k}\mathtt{z}_\mathtt{k}$, $\|\mathtt{z}_\mathtt{k}\| = 1$ as $\mathtt{k} = 1, \ldots, \mathtt{m}+\mathtt{r}$;
$\mathtt{m}_{\mathtt{new}} = 0$, empty $\mathtt{Z}_1$;
for $\mathtt{k} = 1, \ldots, \mathtt{m}+\mathtt{r}$ **do**
if $\Re\mu_\mathtt{k} > \gamma$ **then**
Augment $\mathtt{Z}_1$ by the column $\mathtt{Q}\,\mathtt{z}_\mathtt{k}$;
$\mathtt{m}_{\mathtt{new}} = \mathtt{m}_{\mathtt{new}} + 1$;
end if;
end for;
$\mathtt{m} = \mathtt{m}_{\mathtt{new}}$;

The procedure returns Z_1 and m: m equals the number of modes that exceed the stability boundary. The particular eigenvectors of A are stored as columns of $Z_1 \in$

$\mathbb{R}^{n\times m}$. Therefore, $Z_1 = \texttt{orth}(\texttt{Z}_1)$ would be the actual update of Z_1. Parameters to be fixed are $\texttt{r}$, $\texttt{k}_{\texttt{max}}$ and (28).

In the heart of the algorithm is the loop of subspace iterations preconditioned by the Cayley transform. The matrix $Q \in \mathbb{R}^{n\times(m+r)}$ contains an approximation of an invariant subspace of A; $\texttt{k}_{\texttt{max}}$ controls the quality of the approximation. The action of A on Q is represented by the matrix $H \in \mathbb{R}^{(m+r)\times(m+r)}$. Since the size of H is comparatively small, we use the QR Algorithm, see e.g. [5], to find the spectrum $\sigma(H)$ and the relevant eigenvectors z_k. We accept only those modes that do not exceed the stability boundary γ.

Recall that one of the aims is to indicate the *stability exchange*: We just count the modes of H with $\Re\mu_k \geqslant 0$.

Let us discuss the performance of the algorithm. Assume that there was no need to update m. We could set $\texttt{r} = \texttt{0}$ and the algorithm would converge in few steps. Troubles start when a) some of the modes in the "old" Z_1 converge to modes that *do not* exceed the stability boundary, b) a *new* mode exceeds the stability boundary. Let us call the trouble making eigenvectors the *transition modes*. The natural idea is to augment the initial Z_1 by r randomly initialized columns. One expects that transition modes would contribute to the invariant subspace $Q \in \mathbb{R}^{n\times(m+r)}$. Therefore, these eigenvectors could be extracted from $\sigma(H)$ and be "recognized". This strategy requires a larger $k_{\max}$.

Let us fix a finite $k_{\max}$. One should be aware of the fact that not all eigenvectors of $\sigma(H)$ are approximated with the same precision. The *domain of confidence* of the Cayley transform, see [4], would indicate which modes are approximated best/worst. Parameters should be set in such a way that the eigenvectors z of H with $\Re\mu \geqslant 0$ should be particularly well-approximated. Obviously, parameter tuning is problem-dependent.

2.4. Code of the Projected RPM.

Combining all the things together, i.e. coupled iterations, the pseudo-arclength condition and the `CorrectBasisViaCayley` procedure, we arrive at the following corrector step:

procedure $\texttt{ProjectedRPM}\ (\texttt{u}^{(0)}, \lambda^{(0)}, \texttt{Z}_1, \texttt{m})$
$\texttt{G} = \texttt{G}(\texttt{u}^{(0)}, \lambda^{(0)})$;
$\texttt{p}^{(0)} = \texttt{Z}_1^{\texttt{T}}\texttt{u}^{(0)}; \quad \texttt{v}^{(0)} = \texttt{u}^{(0)} - \texttt{Z}_1\texttt{p}^{(0)}$;
$\texttt{H} = \texttt{Z}_1^{\texttt{T}}\texttt{G}_{\texttt{u}}(\texttt{u}^{(0)}, \lambda^{(0)})\texttt{Z}_1$;
$\texttt{k} := \texttt{0}$;
while $(\|\texttt{G}\| > \texttt{tolRes})$ **do**
$\quad \texttt{v}^{(\texttt{k}+\texttt{1})} = \texttt{RK}(\Delta\texttt{t}, \texttt{p}^{(\texttt{k})}, \texttt{v}^{(\texttt{k})}, \lambda^{(\texttt{k})}, \texttt{tol})$;

$$\begin{pmatrix} p^{(k+1)} \\ \lambda^{(k+1)} \end{pmatrix} = \begin{pmatrix} p^{(k)} \\ \lambda^{(k)} \end{pmatrix} - \begin{pmatrix} H & Z_1^T G_\lambda \\ \dot{p} & \dot{\lambda} \end{pmatrix}^{-1} \begin{pmatrix} Z_1^T G \\ 0 \end{pmatrix};$$

$\mathtt{u}^{(\mathtt{k}+1)} = \mathtt{Z}_1\mathtt{p}^{(\mathtt{k}+1)} + \mathtt{v}^{(\mathtt{k}+1)}$;

$\mathtt{G} = \mathtt{G}(\mathtt{u}^{(\mathtt{k}+1)}, \lambda^{(\mathtt{k}+1)})$;

$\mathtt{H} = \mathtt{Z}_1^{\mathtt{T}}\mathtt{G}_{\mathtt{u}}(\mathtt{u}^{(\mathtt{k}+1)}, \lambda^{(\mathtt{k}+1)})\mathtt{Z}_1$;

$\mathtt{k} = \mathtt{k} + 1$;

endwhile;

Update $\mathtt{u}_0 = \mathtt{u}^{(\mathtt{k}+1)}$, $\lambda_0 = \lambda^{(\mathtt{k}+1)}$;

$\mathtt{CorrectBasisViaCayley}(\mathtt{u}_0, \lambda_0, \mathtt{Z}_1, \mathtt{m})$;

$\mathtt{Z}_1 = \mathtt{orth}(\mathtt{Z}_1)$;

end

In order to approximate the flow φ, see (21), (22), we have employed a standard Runge-Kutta integrator with automatic step control:

$$\varphi(\Delta t, v; p, \lambda) \approx \mathrm{RK}(\Delta t, p, v, \lambda, tol).$$

The last parameter sent to the Runge-Kutta procedure is the local discretization error tolerance *tol*. The following table lists the parameters used in the above algorithm together with the particular values we have used in the computations (see Section 4):

$\delta\mathtt{s} = 0.15$	—arclength increment
$\Delta\mathtt{t} = 0.1$	—time increment in the dynamical simulation
$\mathtt{tolRes} = 10^{-4}$	—stopping criterion in corrector loop
$\mathtt{tol} = 10^{-3}$	—see Runge-Kutta integrator
$\gamma = -5.0$	—stability boundary
$\alpha_1 = \gamma + 1.4$	—see (28)
$\alpha_2 = \gamma - 1.4$	—see (28)
$\mathtt{r} = 6$	—augmenting $\mathtt{Z}_1$
$\mathtt{k}_{\max} = 11$	—number of loops in preconditioned subspace iterations

3. Convergence analysis

We will provide the proof of a local convergence of the iterations defined in Section 2.1. We are aware that this is just the first step towards a justification of our algorithm. Consider u^*, p^*, q^*, λ^*, $A = D_u G(u^*, \lambda^*)$, $\mathbb{P}$, Z_1, Z_2, m and γ being defined as in Section 2.1.

Lemma 1. *The matrix $Z_2^T A Z_1$ satisfies $Z_2^T A Z_1 = 0 \in \mathbb{R}^{(n-m)\times m}$.*

P r o o f. The stamement follows from (7) and (9). □

As a preliminary step, let us investigate properties of the matrix

$$A^Q \in \mathbb{R}^{(n-m)\times(n-m)}$$

being defined as the projection of A on $\mathbb{R}^{n-m}$,

$$A^Q = Z_2^T A Z_2.$$

Lemma 2. *The matrix A^Q has $n-m$ eigenvalues $\mu_{m+1}, \ldots, \mu_n$.*

P r o o f. Let the columns of the matrix $\tilde{Z}_2 \in \mathbb{R}^{n\times(n-m)}$ form a basis in the invariant subspace of A corresponding to the last $n-m$ eigenvalues, i.e. $A\tilde{Z}_2 = \tilde{Z}_2\Lambda_2$ with $\sigma(\Lambda_2) = \{\mu_{m+1}, \ldots, \mu_n\}$. Note that the columns of both Z_1 and $\tilde{Z}_2$ together form a basis of $\mathbb{R}^n$ and hence

$$\mathbb{R}^n = \mathbb{P} \oplus \tilde{Z}_2(\mathbb{R}^{n-m}). \tag{29}$$

As a next step we will show that the columns of the matrix

$$V_2 = Z_2^T \tilde{Z}_2 \in \mathbb{R}^{(n-m)\times(n-m)}$$

span an invariant subspace of A^Q. Indeed,

$$A^Q V_2 = Z_2^T A Z_2 Z_2^T \tilde{Z}_2 = Z_2^T A(Z_1 Z_1^T + Z_2 Z_2^T)\tilde{Z}_2 = Z_2^T A\tilde{Z}_2 = V_2\Lambda_2.$$

Here we have used Lemma 1. Moreover, $Z_1 Z_1^T$ and $Z_2 Z_2^T$ are projectors, therefore $Z_1 Z_1^T + Z_2 Z_2^T = I$. The latter identity in the chain above is a simple consequence of the decomposition $\mathbb{R}^n = Z_1(\mathbb{R}^m) \oplus Z_2(\mathbb{R}^{n-m})$. To complete the proof it suffices to show that the columns of V_2 are linearly independent.

Suppose that there exists a vector $0 \neq q \in \mathbb{R}^{n-m}$ such that $V_2 q = Z_2^T \tilde{Z}_2 q = 0$. Therefore, $\tilde{Z}_2 q \in \mathbb{P}$. Due to the splitting (29), $\tilde{Z}_2 q = 0$. Since $\tilde{Z}_2$ is a full rank matrix, $q = 0$, which is a contradiction. □

Before we prove the main result of this section we need to evaluate three differentials. This is done in the following three lemmas.

Lemma 3. *Let G be from the class $\mathcal{C}^2$. Assume that* $\det(D_p G^P(p^*, q^*, \lambda^*)) \neq 0$. *Then*

$$D_p F^P(p^*, q^*, \lambda^*) = 0 \in \mathbb{R}^{m\times m}.$$

P r o o f. Differentiating F^Q with respect to p we obtain

$$D_p F^Q = I - D_p[(D_p G^P)^{-1}]G^P - (D_p G^P)^{-1} D_p G^P = -D_p[(D_p G^P)^{-1}]G^P.$$

For the particular (p^*, q^*, λ^*) we have $G^P = 0$. □

Lemma 4. *Let G be from the class $\mathcal{C}^1$. Then*

$$D_q F^Q(p^*, q^*, \lambda^*) = \mathrm{e}^{Z_2^T A Z_2 \Delta t}.$$

P r o o f. By definition (20), $D_q F^Q(p, q, \lambda^*) = D_q \psi(\Delta t, q; p, \lambda)$. Consider the directional derivative

$$z(t) = D_q \psi(t, q; p, \lambda^*) \cdot \delta q$$

as a function of time t for an arbitrary fixed $\delta q \in \mathbb{R}^{n-m}$. For the existence, see e.g. [9], Theorem 13.1.4. The particular $z(\Delta)t$ can be computed by integrating the *equation in variations*, see e.g. [9], eq. (1.11) on page 232:

$$\begin{aligned} \dot{z}(t) &= D_q F^Q(p, q(t), \lambda^*) z(t), \\ z(0) &= \delta q \end{aligned}$$

where $q(t) = \psi(t, q; p, \lambda^*)$. At the steady state (p^*, q^*, λ^*), we have $q(t) \equiv q^*$ and

$$\dot{z}(t) = Z_2^T A Z_2 z(t), \tag{30}$$

$$z(0) = \delta q. \tag{31}$$

Hence we can deduce that $z(\Delta t) = \mathrm{e}^{Z_2^T A Z_2 \Delta t} \delta q$. □

Lemma 5. *Let G be from the class $\mathcal{C}^1$. Then*

$$D_p F^Q(p^*, q^*, \lambda^*) = 0 \in \mathbb{R}^{(n-m) \times m}.$$

P r o o f. In accordance with definition (20), $D_p F^Q(p, q, \lambda^*) = D_p \psi(\Delta t, q; p, \lambda^*)$. We consider the variation

$$v(t) = D_p \psi(t, q; p, \lambda^*) \cdot \delta p$$

as a function of time t for an arbitrary fixed $\delta p \in \mathbb{R}^m$. Under the conditions on the smoothness of the mapping G the existence of the differential with respect to p is guaranteed, see [9], Theorem 14.2.1. Then according to [9], eq. (2.8), (2.9) on page 245, $v(t)$ satisfies the differential equation

$$\begin{aligned} \dot{v}(t) &= D_q G^Q(p, q(t), \lambda^*) v(t) + D_p G^Q(p, q(t), \lambda^*) \cdot \delta p, \\ v(0) &= 0. \end{aligned}$$

At the steady state (p^*, q^*, λ^*) we have $q(t) \equiv q^*$. Therefore, $v(t)$ satisfies

$$\dot{v}(t) = Z_2^T A Z_2 v(t) + Z_2^T A Z_1 \delta p = Z_2^T A Z_2 v(t), \tag{32}$$

$$v(0) = 0. \tag{33}$$

Clearly the only solution of the above system is $v(t) = 0$. □

Theorem. *Let G be from the class $\mathcal{C}^2$; let $\det(D_p G^P(p^*, q^*, \lambda^*)) \neq 0$. Then the iterations* (16), (17) *are locally convergent with*

$$\lim_{k\to\infty} p^{(k)} = p^*, \quad \lim_{k\to\infty} q^{(k)} = q^*.$$

P r o o f. Let us consider a composite mapping $H\colon\ \mathbb{R}^m \times \mathbb{R}^{n-m} \to \mathbb{R}^m \times \mathbb{R}^{n-m}$ defined as

$$H(p, q, \lambda^*) = \begin{pmatrix} F^P(p, q, \lambda^*) \\ F^Q(p, q, \lambda^*) \end{pmatrix}.$$

To prove local convergence it is sufficient to show that the spectral radius satisfies the condition

$$\varrho(D_{p,q} H(p^*, q^*, \lambda^*)) < 1. \tag{34}$$

For the differential $D_{p,q} H(p^*, q^*, \lambda^*)$ we obtain

$$D_{p,q} H(p^*, q^*, \lambda^*) = \begin{pmatrix} 0 & D_q F^P(p^*, q^*, \lambda^*) \\ 0 & \mathrm{e}^{Z_2^T A Z_2 \Delta t} \end{pmatrix}$$

due to Lemmas 3, 4 and 5. Lemma 1 yields that the eigenvalues of the Jacobian matrix of H are $0, \ldots, 0, \mathrm{e}^{\mu_{m+1}\Delta t}, \ldots, \mathrm{e}^{\mu_n \Delta t}$; there are m zeroes at the beginning of the list. Since

$$\Re\mu_i < 0, \quad i = m+1, \ldots, n$$

we get $\mathrm{e}^{\mu_i \Delta t} < 1$ and thus (34) holds. □

4. Numerical experiment

In order to compare the performance of both the original RPM and the Projected RPM we have chosen the same example as in [11], i.e. the nonsymmetric system of PDEs

$$\frac{\partial v}{\partial t} = \frac{1}{5}\frac{\partial^2 v}{\partial x^2} + \lambda(v - w) + \frac{1}{5}\lambda^2 \mathrm{e}^v,$$
$$\frac{\partial w}{\partial t} = \frac{1}{5}\frac{\partial^2 w}{\partial x^2} + \lambda(v + w) + \frac{1}{5}\lambda^2 \mathrm{e}^w$$

with

$$v(0) = v(1) = 0, \quad w(0) = w(1) = 0.$$

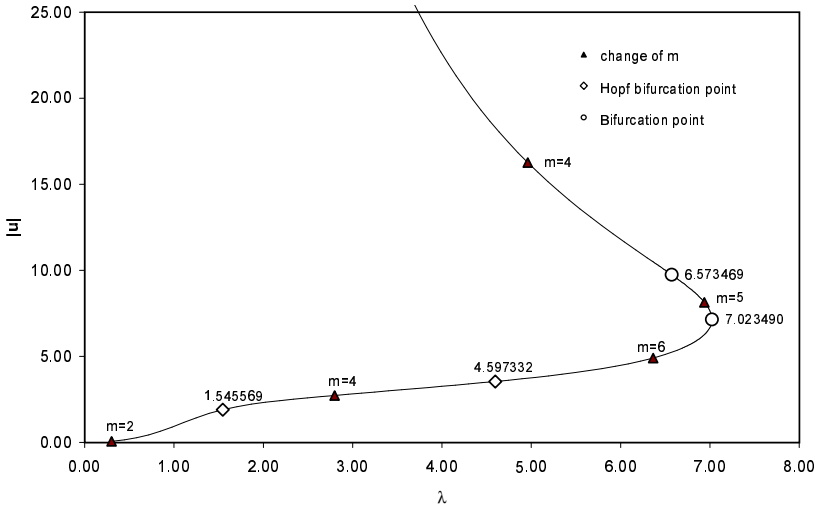

Figure 1. Solution path of nonsymmetric system in for $nx = 40$ computed by the Projected RPM.

The above system is discretized by finite differences. The number of mesh points is denoted by nx. Notice that the dimension of the resulting problem is $n = 2 * nx$.

Fig. 1 depicts the solution curve computed by Projected RPM for $nx = 40$. Discontinuities of m are marked. All four bifurcation points were safely detected: two Hopf bifurcation points, one turning point and one symmetry breaking bifurcation point. Note that the original RPM failed to detect the Hopf bifurcation point for $\lambda \approx 4.5$.

λ	$\|u\|$	μ	$\lvert\frac{\hat{\mu}-\mu}{\hat{\mu}}\rvert$
1.3918	1.6824	$-0.1566 + 1.3861\mathrm{i}$	$1.5{\cdot}10^{-9}$
1.5456	1.903	$0.0767 + 1.5363\mathrm{i}$	$2.5{\cdot}10^{-9}$
4.4477	3.4694	$-0.2733 + 4.3172\mathrm{i}$	$1.4{\cdot}10^{-7}$
4.5973	3.5427	$0.0821 + 4.4481\mathrm{i}$	$4.8{\cdot}10^{-7}$
7.0235	7.1372	-0.2391	$1.1{\cdot}10^{-4}$
7.0230	7.2211	0.8011	$1.5{\cdot}10^{-4}$
6.6935	9.2843	0.6832	$1.2{\cdot}10^{-4}$
6.5597	9.7910	-0.0401	$4.0{\cdot}10^{-5}$

Table 1. Accuracy of the computed eigenvalues near bifurcation points.

Tab. 1 refers to the stability exchange: While following the solution path $(u(s), \lambda(s))$ numerically, we produce and process the path of the relevant $\mu \in$

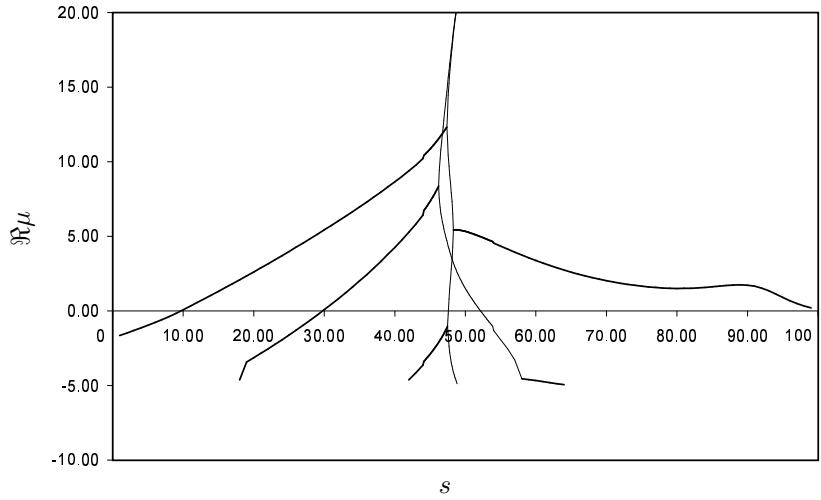

Figure 2. Eigenvalue movie.

$\sigma(H(u(s), \lambda(s)))$, see `CorrectBasisViaCayley` in Section 2.3. Two consecutive points on the numerical branch with μ just crossing the imaginary axis indicate stability exchange. For comparison, we computed $\sigma(A)$ via QR and marked by $\hat{\mu}$ the eigenvalue closest to the imaginary axis. From the table we can conclude that all the eigenvalues needed for the detection of the bifurcation points were computed within a satisfactory accuracy.

Finally, Fig. 2 shows the eigenvalue movie, i.e. the dependence of the real parts of the rightmost eigenvalues on the pseudo-arclength parameter. Bold lines correspond to the complex conjugate pairs of eigenvalues.

References

[1] *E. L. Allgower. K. Georg*: Numerical Continuation Methods. Springer Verlag, New York, 1990.

[2] *J. Bošek, V. Janovský*: A note on the recursive projection method. Proceedings of GAMM96. Z. Angew. Math. Mech. (1977), 437-440.

[3] *B. D. Davidson*: Large-scale continuation and numerical bifurcation for partial differential equations. SIAM J. Numer. Anal. *34* (1997), 2001–2027.

[4] *T. J. Garratt, G. Moore and A. Spence*: A generalised Cayley transform for the numerical detection of Hopf bifurcation points in large systems. Contributions in numerical mathematics, World Sci. Ser. Appl. Anal. (1993), 177–195.

[5] *G. H. Golub, Ch. F. van Loan*: Matrix Computations. The Johns Hopkins University Press, Baltimore, 1996.

[6] *V. Janovský, O. Liberda*: Recursive Projection Method for detecting bifurcation points. Proceedings SANM'99. Union of Czech Mathematicians and Physicists, 1999, pp. 121–124.

[7] *V. Janovský, O. Liberda*: Projected version of the Recursive Projection Method algorithm. Proceedings of 3rd Scientific Colloquium. Institute of Chemical Technology, Prague, 2001, pp. 89–100.
[8] *M. Kubíček, M. Marek*: Computational Methods in Bifurcation Theory and Dissipative Structures. Springer, 1983.
[9] *J. Kurzweil*: Ordinary Differential Equations. Elsevier, 1986.
[10] *K. Lust, D. Roose*: Computation and bifurcation analysis of periodic solutions of large–scale systems. IMA Preprint Series #1536, Feb. 1998, IMA, University of Minnesota.
[11] *G. M. Shroff, H. B. Keller*: Stabilization of unstable procedures: the Recursive Projection Method. SIAM J. Numer. Anal. *30* (1993), 1099–1120.

Authors' address: *V. Janovský, O. Liberda*, Faculty of Mathematics and Physics, Charles University, Sokolovská 83, 186 00 Prague 8, Czech Republic, e-mail: `{janovsky, liberda}@karlin.mff.cuni.cz`.