Computing Optimal Model Perturbations: A Constraint Optimization Problem

T.L. van Noorden* J. Barkmeijer†

Abstract

We investigate the numerical computation of solutions to a special quadratic optimization problem with quadratic constraints. The optimization problem stems from the linear sensitivity analysis of systems of differential equations. This optimization problem has also been studied by Farrell et al. [4]. In their work the constraint on the optimal perturbation is formulated using a norm generated by an innerproduct. The optimization problem reduces in this case to the computation of the largest singular value of a linear operator. In the present paper, we consider a different norm in the constraints, which allows us to control the magnitude of the optimal perturbation at each instant in time. This norm does not have an associated innerproduct, and therefore the nature of the optimization problem changes. We investigate the properties of the posed constraint optimization problem, which turns out to have many similarities with the positive semi-definite standard eigenvalue problem. Based on these similarities we propose an effective algorithm for the computation of the optimal forcings for large scale systems.

1 Introduction

In this paper we discuss the efficient numerical solution of the following constraint optimization problem. Let A be a linear map from $X = \mathbb{R}^{np}$ into $Y = \mathbb{R}^n$. The linear map A can be written in block form

$$A = \begin{pmatrix} A_1 & A_2 & \dots & A_p \end{pmatrix}. \tag{1}$$

We think of the space X as p copies of $\mathbb{R}^n$, and write vectors in X as $x = (x_1^T, x_2^T, ..., x_p^T)^T$, where all x_i are in $\mathbb{R}^n$. Now we would like to find the maximum value of $x^T A^T A x$ over all vectors $x \in X$ that satisfy the constraint $||x_i|| \leq 1$ for $i = 1, ..., p$. We are particularly interested in cases where n is large. The constraint optimization problem can also be formulated in the following way: find

$$\max_{||x||_* \leq 1} ||Ax||_2^2, \tag{2}$$

*Universiteit Utrecht, Department of Mathematics, Budapestlaan 6, Utrecht, The Netherlands, email: noorden@math.uu.nl

†Royal Netherlands Meteorological Institute, PO Box 201, 3730 AE, De Bilt, The Netherlands

where $||\cdot||_2$ denotes the standard Euclidian norm in $\mathbb{R}^n$, and the norm $||\cdot||_*$ is given by

$$||x||_* = \max_{i=1,\ldots,p} ||x_i||_2 \tag{3}$$

The motivation for studying this optimization problem is the linear sensitivity analysis of large scale systems of differential equations with respect to time dependent perturbations. Suppose we study a nonlinear model, with model equations

$$\dot{x}(t) = F(x(t), t), \quad x(t) \in \mathbb{R}^n,$$

where the dimension n is large. Such large systems of equations are typically obtained after the spatial discretization of a system of partial differential equations. Suppose now $x^*(t)$ for $t \in [0, T]$ is a special solution of interest. In order to study the linear stability of this solution, we linearize the model equations around this special solution and obtain the so-called "tangent" model, which is given by

$$\dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t), t)y(t). \tag{4}$$

The linear behavior of perturbations of the initial condition of the solution $x^*(t)$ is determined by the eigenvalues of the Jacobian of the right hand side of the tangent model (4) (if the equations are autonomous) or of the monodromy matrix. When, however, the angles between eigenvectors become small, the eigenvalues alone do not tell everything about the behavior of solutions. In this case, the norm or the largest singular value of the appropriate linear map is the quantity that may be more important [15]. This norm is the solution of the constraint optimization problem

$$\max_{||y(0)||\leq 1} ||y(T)||, \text{ where } \dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t), t)y(t).$$

This is a well studied problem (see e.g. [15]). We have to study the map, say M, that maps a given initial conditions $y(0)$ to the value of the solution $y(T)$ at time T. To find the optimal disturbance, we need to find the maximal singular value of the map M. For symmetric problems, this reduces to finding the eigenvalues of M.

The problem of finding normalized perturbations of the initial conditions that maximize the norm of the solution at a given instant in time, is a well studied problem that has been treated for many different applications and in many different contexts [4, 1]. The computation of the norm requires the integration of the adjoint equations [1]. For large scale problems, the value of the norm can efficiently be computed with iterative eigenvalue methods.

We can also study the sensitivity of the solution of the tangent model (4) at time T with respect to perturbations in the right hand side of the differential equations:

$$\max_{||f||\leq 1} ||y(T)||, \text{ where } \dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t), t)y(t) + f(t), \quad y(0) = 0. \tag{5}$$

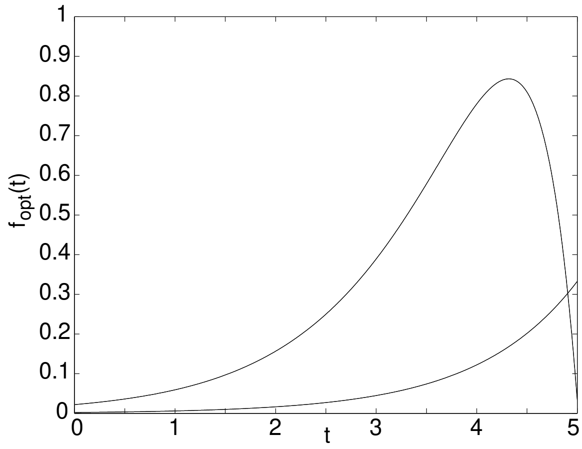

Figure 1: The two components of the optimal forcing f_{opt}.

In order to solve this optimization problem, we need to specify the norm used in the constraint put on the perturbations f. In [4], the authors study the constraint optimization problem (5), using the following norm on f

$$||f||^2 = \int_0^T f(t)^T f(t) dt. \tag{6}$$

This norm is obviously generated by an innerproduct. Therefore the solution of the constraint optimization problem (5) can be found by computing the largest singular value of the appropriate linear map.

It is instructive to look at the solution of the problem (5) using the norm given in (6) for the following simple example system

$$\dot{y}(t) = \begin{pmatrix} -1 & 10 \\ 0 & -2 \end{pmatrix} y(t) + f(t).$$

In the figure, the two components of the model perturbation $f_{opt}(t)$ that maximizes $y(T)$ is plotted. One can see that it has a distinctive peak near the end of the specified interval $[0, T]$. For some applications [1] this can be an undesirable feature. Sometimes also statistical data is available about the magnitude of the perturbations at different moments in time. (see also [1]). In this case it is important that the magnitude of the optimal perturbation is controlled over time.

Therefore we would like to be able to study the sensitivity of the solution at time T with respect to perturbations in the model equations, with the extra constraint that we can restrict the size of the forcing at each moment in time. In other words, we would like to solve the constraint optimization problem (5), but now using the norm on the perturbations f given by

$$||f|| = \sup_{t\in[0,T]} ||f(t)||_2, \tag{7}$$

where $||\cdot||_2$ is the standard Euclidian norm in $\mathbb{R}^n$. There does not exist an innerproduct associated with this norm. Therefore the solution of the optimization problem cannot be computed as the largest singular value of a linear map.

In this work, we will focus on the numerical solution of (5) using the norm given in (7). The natural first step is to discretize the forcing $f(t)$ in time: $0 < t_1 < ... < t_{p-1} < T$ and f_i, for $i = 1, ..., p$.

$$\max_{||f_i||\leq 1,\ i=1,...,p} ||y(T)||, \text{ where } \dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t),t)y(t) + k(t), \quad y(0) = 0,$$

and where

$$k(t) = f_i \text{ if } t_{i-1} \leq t < t_i.$$

The map that assigns to a collection f_i, the value $y(T)$, is a linear map. Denote this map by A. We can think of a collection of f_i's as a vector in $X = \mathbb{R}^{np}$, so that the map A is a linear map from $X = \mathbb{R}^{np}$ into $Y = \mathbb{R}^n$, which we can represent in the block form given in (1). Now we can state our problem as follows: compute

$$\max_{||f_i||\leq 1,\ i=1,...,p} \boldsymbol{f}^T A^T A \boldsymbol{f}, \tag{8}$$

where

$$\boldsymbol{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_p \end{pmatrix},$$

which is equivalent to the optimization problem stated in (2) and (3). Note that evaluating the action of A involves integrating the linearized model equations and evaluating the action of A^T involves integrating the adjoint linearized model equations [1]. It is clear that the operator $N := A^T A$ is positive semi-definite and has at most rank n.

In the remaining sections of this paper, we will study the optimization problem as stated in (2) and (3). In Section 2, we will study properties of solutions, and throughout the paper we will find many similarities with the studied optimization problem and the standard positive semi-definite eigenvalue problem. Based on the similarities with the symmetric eigenvalue problem, we study numerical methods that are variants of their eigenvalue counterparts. We show that there exists an analogue of the power method for our optimization problem. This methods usually converges slowly. We propose a numerical procedure to accelerate the convergence of the power method, which is in fact an adopted version of the Arnoldi or Lanczos method for eigenvalue problems [7]. The method we propose also falls in the class of accelerated inexact Newton schemes [5]. We focus on the case that the system of differential equations is large in dimension. This means that the integration of the equations and the integration of the adjoint equations is a computational expensive task. In fact it is the most demanding task. Therefore we design our solution algorithm such that we try to minimize the number of times we need to integrate the state and adjoint equations. The numerical procedure is discussed in Section 3, and is applied to some test problems in Section 3.4.

In this work we look at the optimization problem under consideration as a "modified" eigenvalue problem. Other ways to look at the problem would be to consider it as an optimal control problem [13] or as a convex optimization problem as discussed in [2]. There are also connections with the multiparameter eigenvalue problem [8].

Other numerical methods that could be used to numerically solve the constraint optimization problem, are the optimization methods developed and described by Smith et al [12]. These methods are optimization methods that work on Riemannian manifolds. The constraints in our optimization problem are such that we optimize over a direct sum of p n-dimensional spheres. This is a $p(n-1)$-dimensional hyper-surface in $\mathbb{R}^{np}$. These methods do not construct a search space. This is advantageous for the memory usage. However, we would like not only to find the global optimum, but also a subspace of perturbations, in which a few of the most amplified perturbation live (see Section 2.3). This is a feature of the Krylov subspace eigenvalue methods. Therefore we investigate the adaption of these kind of methods.

2 The Constraint Optimization Problem

2.1 Maximum is attained on the boundary

Let M be a linear operator from $\mathbb{R}^n$ to itself. We know that we can replace the smaller or equal sign in

$$\max_{||x||_2 \leq 1} ||Mx||_2$$

by an equality sign. We would like to be able to do the same in the constraint optimization problem given in (8). The following theorem tells us that we may.

Theorem 1. *Let the map A be a linear map from $\mathbb{R}^{np}$ into $\mathbb{R}^n$, given in the block form*

$$A = \begin{pmatrix} A_1 & A_2 & \dots & A_p \end{pmatrix}.$$

Suppose that A_i is invertible for $i = 1, ..., p$. Let N be the positive semi-definite operator given by $N := A^T A$, and suppose that $y = (y_1^T, y_2^T, ..., y_p^T)^T$, with $||y_i|| \leq 1$ for $i = 1, ..., p$, is such that

$$y^T N y = \max_{||x_i|| \leq 1,\ i=1,\dots,p} x^T N x,$$

then it holds that $||y_1|| = ||y_2|| = ... = ||y_p|| = 1$.

Proof: Denote $\lambda^* := y^T N y$, and think of the operator N as given in the block form

$$N = \begin{pmatrix} N_{11} & N_{12} & \dots & N_{1p} \\ N_{21} & N_{22} & \dots & N_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ N_{p1} & N_{p2} & \dots & N_{pp} \end{pmatrix},$$

where $N_{ij} \in \mathbb{R}^{n\times n}$. Suppose that not all of the sub-vectors of y have norm equal to one. We can permute, without loss of generality, the blocks of N in such a way that we have $||y_i|| = 1$ for $i = 1, ..., k$ and $||y_i|| < 1$ for $i = k+1, ..., p$. This means that the following constraint optimization problem attains a (local) maximum in y:

$$\max_{||x_i||\leq 1,\ i=1,\ldots,k} x^T N x.$$

Using the Euler-Lagrange approach, we know that $y = (\tilde{y}_1, \tilde{y}_2)$, with $\tilde{y}_1 = (y_1, ..., y_k)$ and $\tilde{y}_2 = (y_{k+1}, ..., y_p)$, should satisfy the equations

$$\begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{B}^T & \tilde{C} \end{pmatrix} \begin{pmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{pmatrix} = \begin{pmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{pmatrix}, \quad y_1^T y_1 = 1, \quad \ldots, \quad y_k^T y_k = 1,$$

where

$$\tilde{A} = \begin{pmatrix} N_{11} & \ldots & N_{1k} \\ \vdots & \ddots & \vdots \\ N_{k1} & \ldots & N_{kk} \end{pmatrix}, \quad \tilde{B} = \begin{pmatrix} N_{1k+1} & \ldots & N_{1p} \\ \vdots & \ddots & \vdots \\ N_{k(k+1)} & \ldots & N_{kp} \end{pmatrix},$$

and

$$\tilde{C} = \begin{pmatrix} N_{(k+1)(k+1)} & \ldots & N_{(k+1)p} \\ \vdots & \ddots & \vdots \\ N_{(k+1)p} & \ldots & N_{pp} \end{pmatrix}, \quad \Lambda_1 = \begin{pmatrix} \lambda_1 I & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \lambda_k I \end{pmatrix}.$$

Now define the number $\lambda_s := \max_{||x_i||=1,\ i=1,\ldots,k}(x^T \tilde{A} x)$ and let x, with $||x_i|| = 1$, $i = 1, ..., k$, be such that $x^T \tilde{A} x = \lambda_s$. Because we assumed that A_i is invertible for $i = 1, ..., p$, we can find a vector $\tilde{x}$ such that $\tilde{x}^T \tilde{C} \tilde{x} > 0$, $\tilde{x}^T B x \geq 0$ and $||\tilde{x}|| < 1$. For the vector $(x^T \ \ \tilde{x}^T)^T$ we have the following inequalities

$$\lambda^* \geq (x^T \ \ \tilde{x}^T) N \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} = x^T \tilde{A} x + 2\tilde{x}^T B x + \tilde{x}^T \tilde{C} \tilde{x} > \lambda_s.$$

Thus we find that $\lambda_s < \lambda^*$. On the other hand, using that y satisfies the Euler-Lagrange equations and the positive semi-definiteness of N, we obtain the inequality

$$\lambda^* = y^T N y = \tilde{y}_1^T \tilde{A} \tilde{y}_1 + \tilde{y}_1^T \tilde{B} \tilde{y}_2 = \tilde{y}_1^T \tilde{A} \tilde{y}_1 - \tilde{y}_2^T \tilde{C} \tilde{y}_2 \leq \lambda_s,$$

This is a contradiction, which proves the theorem. □

The above theorem tells us that we now may focus on the constraint optimization problem

$$\max_{||x_i||=1,\ i=1,\ldots,p} x^T N x.$$

The Euler-Lagrange approach tells us that if we would like to find a vector x, with $||x_i|| = 1$ for $i = 1, ..., p$, that maximizes $x^T N x$, we have to find the maximum of

$$\mathcal{F}(x, \lambda_1, \lambda_2, \ldots, \lambda_p) = x^T N x - \lambda_1(x_1^T x_1 - 1) - \ldots - \lambda_p(x_p^T x_p - 1).$$

After differentiating we obtain the system of equations

$$\begin{aligned}
\begin{pmatrix} N_{11} & N_{12} & \dots & N_{1p} \\ N_{21} & N_{22} & \dots & N_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ N_{p1} & N_{p2} & \dots & N_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} &= \begin{pmatrix} \lambda_1 x_1 \\ \lambda_2 x_2 \\ \vdots \\ \lambda_p x_p \end{pmatrix}, \\
x_1^T x_1 &= 1, \\
&\vdots \\
x_p^T x_p &= 1.
\end{aligned}$$

We will refer to this system of equations as the *Euler-Lagrange* equations, and in the following sections, we will concentrate on the properties of solutions to these equations and on methods to approximate them.

The analogue of Theorem 1 holds also in the continuous case, which means that for the optimization problem (5) using the norm given in (7) the the optimal forcing f satisfies $||f(t)|| = 1$ for all $t \in [0, T]$. See Appendix A for the statement of the theorem and a sketch of the proof.

2.2 Geometric Characterization of Solutions

For $p = 1$ the Euler-Lagrange equations reduce to the standard symmetric eigenvalue problem. In this case we know the number of solutions, we can define the spectrum of the problem, and we know that the different solutions vectors, or eigenvectors, are orthogonal to each other. In this section we would like to investigate whether we can generalize these concepts to the case $p > 1$.

For simplicity, we start by setting $p = 2$. Then we have to solve

$$\begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \lambda_1 I & 0 \\ 0 & \lambda_2 I \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{9}$$

$$x_1^T x_1 = 1 \tag{10}$$

$$x_2^T x_2 = 1 \tag{11}$$

First we focus at on equation (9). Define the family of matrices $K(\lambda_1, \lambda_2)$

$$K(\lambda_1, \lambda_2) := \begin{pmatrix} N_{11} - \lambda_1 I & N_{12} \\ N_{21} & N_{22} - \lambda_2 I \end{pmatrix}.$$

Consider the set U of pairs (λ_1, λ_2) where the matrix $K(\lambda_1, \lambda_2)$ is not invertible. The set U is the solution set of $\det(K(\lambda_1, \lambda_2)) = 0$. We can think of this set as the generalization of the spectrum of a matrix. Suppose now that the pair (μ_1, μ_2) is in the set U, and that μ_1 is not an eigenvalue of N_{11}. Then there is a vector (x_1, x_2) such that $(N_{11} - \mu_1 I)x_1 + N_{12}x_2 = 0$. Thus $x_1 = -(N_{11} - \mu_1 I)^{-1} N_{12} x_2$. If we substitute this into the equation $N_{21}x_1 + (N_{22} - \mu_2 I)x_2 = 0$, we obtain for x_2 the equation

$$(N_{22} - N_{21}(N_{11} - \mu_1 I)^{-1} N_{12}) x_2 = \mu_2 x_2.$$

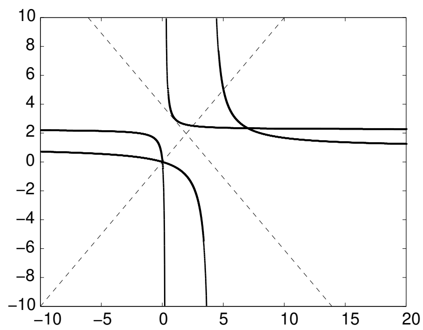

Figure 2: The solution curves in the (λ_1, λ_2)-plane of $\det(K(\lambda_1, \lambda_2)) = 0$.

From this equation we see that if the pair (μ_1, μ_2) belongs to U, then μ_2 is an eigenvalue of the matrix $(N_{22} - N_{21}(N_{11} - \mu_1 I)^{-1} N_{12})$. In general these eigenvalues change smoothly with μ_1, and therefore we see that the set U consists of curves. Note that not all of the points on these curves correspond to solutions of the Euler-Lagrange equations. In addition, we need that $||x_1|| = ||x_2|| = 1$.

Example 1. Let us investigate the set U for the following simple four by four example. Let A be given by

$$A = \begin{pmatrix} 2 & 0 & 1 & 0 \\ 0 & 0.5 & 0 & 1.5 \end{pmatrix}.$$

Then $N = A^T A$ is given by

$$N = \begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix} = \begin{pmatrix} 4 & 0 & 2 & 0 \\ 0 & 0.25 & 0 & 0.75 \\ 2 & 0 & 1 & 0 \\ 0 & 0.75 & 0 & 2.25 \end{pmatrix}.$$

In Figure 2, the solutions curves in the (λ_1, λ_2)-plane of $\det(K(\lambda_1, \lambda_2)) = 0$ are plotted. The set consists of four curves. The line $\lambda_1 = \lambda_2$ is also plotted in the figure. Intersections of the plotted curves with this line are the eigenvalues of N. Since N has only rank 2, the double eigenvalue in 0 is expected. $\square$

From Figure 2, we can see that for each curve we can think of λ_2 as a function of λ_1. Let us compute the derivative of λ_2 with respect to λ_1. If $x(t)$ is a parameterized set of eigenvalues of the set of symmetric matrices $B(t)$, then the derivative of $x(t)$ at $t = 0$ satisfies $\dot{x}(0) = x^T B'(0) x / x^T x$ (see for example [7]). If we apply this equations to the matrices $(N_{22} - N_{21}(N_{11} - \lambda_1 I)^{-1} N_{12})$, we obtain the following lemma.

Lemma 1.

$$\lambda_2'(\lambda_1) = \frac{-x_2^T N_{21}(N_{11} - \lambda_1 I)^{-2} N_{12} x_2}{x_2^T x_2} = -\frac{x_1^T x_1}{x_2^T x_2}.$$

This lemma tells us that solutions corresponding to points on the curves where the curve is tangent to the vector $(1, -1)$ satisfy $x_1^T x_1 = x_2^T x_2 = 1$. If we now again look at Figure 2, we see that there must be at least 4 solutions. For one of the solutions, the tangent line with slope -1 is plotted in the figure.

Now we would like generalize this result to the case $p > 2$. First define again the family of matrices $K(\lambda_1, \lambda_2, ..., \lambda_p)$

$$K(\lambda_1, \lambda_2, ..., \lambda_p) := \begin{pmatrix} N_{11} - \lambda_1 I & N_{12} & \dots & N_{1p} \\ N_{12}^T & N_{22} - \lambda_2 I & \dots & N_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ N_{1p}^T & N_{2p}^T & \dots & N_{pp} - \lambda_p I \end{pmatrix}.$$

The solutions to $h(\lambda_1, \lambda_2, ..., \lambda_p) := \det(K(\lambda_1, \lambda_2, ..., \lambda_p)) = 0$ form hyper-surfaces in $(\lambda_1, \lambda_2, ..., \lambda_p)$-space. They are the zero-level set of h.

Theorem 2. *The following two statements are equivalent:*

1) $(\lambda_1, \lambda_2, ..., \lambda_p)$ satisfies $h(\lambda_1, \lambda_2, ..., \lambda_p) = 0$ and $\nabla h(\lambda_1, \lambda_2, ..., \lambda_p) = \alpha(1, 1, ..., 1)$ for some real $\alpha \neq 0$.

2) there exists a vector $(x_1, x_2, ..., x_p)$ with $||x_i|| = 1$ for $i = 1, 2, ..., p$, that satisfies

$$\begin{pmatrix} N_{11} & N_{12} & \dots & N_{1p} \\ N_{21} & N_{22} & \dots & N_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ N_{p1} & N_{p2} & \dots & N_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \lambda_1 x_1 \\ \lambda_2 x_2 \\ \vdots \\ \lambda_p x_p \end{pmatrix}, \tag{12}$$

with dim Ker$(K(\lambda_1, \lambda_2, ..., \lambda_3)) = 1$.

Proof: First we will prove that statement 1 implies statement 2. Since $h(\lambda_1, \lambda_2, ..., \lambda_p) = 0$, we know that there exists a vector $(x_1, x_2, ..., x_p)$ such that eq. (12) holds. What we need to prove is that there is such a vector that satisfies $||x_i|| = 1$ for $i = 1, 2, ..., p$. First, we will show that $\mathrm{Ker}(K(\lambda_1, \lambda_2, ..., \lambda_p))$ is one dimensional. Fix $(\lambda_2, \lambda_3, ..., \lambda_p)$ for a moment and look at the generalized eigenvalue problem

$$\mathbf{N}x = \mu E x, \tag{13}$$

where

$$\mathbf{N} = \begin{pmatrix} N_{11} & N_{12} & \dots & N_{1p} \\ N_{21} & N_{22} - \lambda_2 I & \dots & N_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ N_{p1} & N_{p2} & \dots & N_{pp} - \lambda_p I \end{pmatrix}, \quad E = \begin{pmatrix} I & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}. \tag{14}$$

We have that $\det(\mathbf{N} - \mu E) = 0$ for $\mu = \lambda_1$ and also

$$\frac{d}{d\mu} \det(\mathbf{N} - \mu E)|_{\mu=\lambda_1} = \frac{\partial h}{\partial \lambda_1}(\lambda_1, \lambda_2, ..., \lambda_p) = \alpha \neq 0.$$

Therefore the vector $(x_1, x_2, ..., x_p)$ is a single eigenvector corresponding to the single eigenvalue λ_1 of (13). We may therefore conclude that $\mathrm{Ker}(K(\lambda_1, \lambda_2, ..., \lambda_p))$ is one dimensional. By the implicit function theorem, we know that we may write λ_1 as a function of $(\lambda_2, \lambda_3, ..., \lambda_p)$ such that

$$h(\lambda_1(\lambda_2, \lambda_3, ..., \lambda_p), \lambda_2, \lambda_3, ..., \lambda_p) = 0.$$

Now we would like to compute the partial derivative $\partial\lambda_1/\partial\lambda_p$. To this end it is instructive to consider the generalized eigenvalue problem

$$(\mathbf{N} + \lambda_p F)x = \lambda_1 Ex,$$

where $\mathbf{N}$ and E are as in (14) and

$$F = \begin{pmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \end{pmatrix}.$$

It is not hard to show that (see Lemma 1)

$$\frac{\partial\lambda_1}{\partial\lambda_p}(\lambda_2, \lambda_3, ..., \lambda_p) = -\frac{x^T F x}{x^T E x} = -\frac{x_p^T x_p}{x_1^T x_1}.$$

Likewise, we can show that

$$\frac{\partial\lambda_1}{\partial\lambda_i}(\lambda_2, \lambda_3, ..., \lambda_p) = -\frac{x_1^T x_1}{x_i^T x_i} \text{ for } i = 2, ..., p-1.$$

By implicit differentiation we know that

$$\nabla\lambda_1(\lambda_2, \lambda_3, ..., \lambda_p) = -\frac{1}{\partial h/\partial\lambda_1}(\partial h/\partial\lambda_2, \partial h/\partial\lambda_3, ..., \partial h/\partial\lambda_p).$$

Since we assumed that $\nabla h(\lambda_1, \lambda_2, ..., \lambda_p) = \alpha(1, 1, ..., 1)$ for some real $\alpha \neq 0$, we may now conclude that $x_1^T x_1 = x_2^T x_2 = ... = x_p^T x_p$. This proves that statement 1 implies statement 2.

Now assume that $(x_1, x_2, ..., x_p)$ satisfies (12). Fix for a moment $(\lambda_2, \lambda_3, ..., \lambda_p)$ and consider again the generalized eigenvalue problem (13). Since we assume that the kernel of $K(\lambda_1, \lambda_2, ..., \lambda_p)$ is one dimensional, the eigenvalue λ_1 of (13) is single and therefore

$$\frac{d}{d\mu}\det(\mathbf{N} - \mu E)|_{\mu=\lambda_1} = \frac{\partial h}{\partial\lambda_1}(\lambda_1, \lambda_2, ..., \lambda_p) \neq 0,$$

so that we can use the implicit function theorem to prove $\nabla h(\lambda_1, \lambda_2, ..., \lambda_p) = \alpha(1, 1, ..., 1)$ for some real $\alpha \neq 0$. This proves the converse. $\square$

In the theorem we proved or assumed that $\mathrm{Ker}(K(\lambda_1, \lambda_2, ..., \lambda_p))$ is one dimensional. It is also possible that a vector $(x_1, x_2, ..., x_p)$ satisfies (12), with the dimension of the kernel of $K(\lambda_1, \lambda_2, ..., \lambda_p)$ larger than one. In this case we can show that

$$\frac{\partial h}{\partial \lambda_i}(\lambda_1, \lambda_2, ..., \lambda_p) = 0, \tag{15}$$

for $i = 1, 2, ..., p$, so that $\nabla h(\lambda_1, \lambda_2, ..., \lambda_p) = (0, 0, ..., 0)$. This situation arises, for example, in Figure 2 on the intersection points of the curves.

Now we consider some examples and try to determine the number of solutions to the Euler-Lagrange equations.

Example 2. Consider the case $p = 1$. Then the Euler-Lagrange equations are reduced to a symmetric eigenvalue problem and we know that there exist n different solutions to the Euler-Lagrange equations. □

Example 3. If we consider the block diagonal case, where each diagonal block has n distinct eigenvalues, we see that there are $2^{p-1}n^p$ different solutions to the Euler-Lagrange equations. Note that in this case the matrix N has full rank. □

Example 4. We can also consider the case $n = 1$. In this case it is clear that all the vectors with coordinates equal to 1 or -1 satisfy the Euler-Lagrange equations. Up to scaling (with -1) there are 2^{p-1} such vectors. □

Example 5. Consider the case

$$N = \begin{pmatrix} A & aA \\ aA & a^2A \end{pmatrix},$$

where A is positive definite, $a \neq 1$ and $a \neq 0$. Assume that A has n distinct eigenvalues. The matrix N has rank n. Suppose that x_1, x_2 with λ_1 and λ_2 satisfy the Euler-Lagrange equations. Then it holds that $a\lambda_1 x_1 = \lambda_2 x_2$. Now there are two cases we need to consider separately: either $\lambda_1 = 0$ or $\lambda_1 \neq 0$. If $\lambda_1 \neq 0$, then either $x_1 = x_2$ or $x_1 = -x_2$ and it follows that x_1 should be an eigenvector of A. Thus there are $2n$ different solutions of the Euler-Lagrange equations with $\lambda_1 \neq 0$.

If $\lambda_1 = 0$, then $Ax_1 + aAx_2 = 0$ and since A is invertible, $x_1 = -ax_2$. This last equality is in contradiction with our assumption that x_1 and x_2 satisfy the Euler-Lagrange equations and thus $||x_1|| = ||x_2|| = 1$. Therefore the case $\lambda_1 = 0$ does not exist. The conclusion is that with N as given above, the Euler-Lagrange equations have $2n$ different solutions. □

The number of solutions to the Euler-Lagrange equations in the general case, is an open problem. Considering the examples presented above, and in previous sections, we also do not dare to conjecture.

We would also like to know whether the solution vectors, or "eigenvectors", of our constraint optimization problem are orthogonal to each other or not. Suppose that

$[(\lambda_1, ..., \lambda_p), (x_1, ..., x_p)]$ and $[(\mu_1, ..., \mu_p), (y_1, ..., y_p)]$ are solutions to the Euler-Lagrange equations. We know that holds

$$y^T N x = \sum_{i=1}^{p} \lambda_i y_i^T x_i.$$

We also have

$$x^T N y = \sum_{i=1}^{p} \mu_i y_i^T x_i.$$

Now we obtain by symmetry of N

$$\sum_{i=1}^{p} \lambda_i y_i^T x_i = \sum_{i=1}^{p} \mu_i y_i^T x_i.$$

For $p = 1$, thus for the standard symmetric eigenvalue problem, we obtain $\lambda y^T x = \mu y^T x$. From this equality follows the well known result that two eigenvectors corresponding to two different eigenvalues are orthogonal to each other. For $p > 1$, this result does not hold anymore. Therefore we propose in the next section a deflation strategy that computes a basis of a subspace of strongly amplified vectors and that guarantees the orthogonality of the computed basis vectors.

2.3 Deflation

In applications [1], it is often necessary not only to compute the vector that maximizes $x^T Ax$ under the given constraints, but to compute a subspace which contains vectors that are most strongly amplified by A. For the standard symmetric eigenvalue problem, it is clear how we, apart from the maximum eigenvalue, should define such a subspace that contains the most amplified perturbations. This subspace, say with dimension k, is spanned by the eigenvectors corresponding to the k largest eigenvalues. The way this subspace is computed in practice for large scale problems, is as follows. First compute the largest eigenvalue and corresponding eigenvector, say y_1, of A. Then compute the largest eigenvalue and corresponding eigenvector, say y_2 of the operator $(I - y_1 y_1^T)A(I - y_1 y_1^T)$, and next compute the largest eigenvalue of the operator $(I - y_2 y_2^T)(I - y_1 y_1^T)A(I - y_1 y_1^T)(I - y_2 y_2^T)$, etc.. This process of projecting the problem orthogonal to the already found vectors is called *deflation*. It is a well known procedure in eigenvalue computations, see for example [10].

In analogy with the deflation procedure for the standard symmetric eigenvalue problem, we propose the following deflation procedure for the constraint optimization problem under consideration. Once we have found the maximizing vector, say $x = (x_1^T, x_2^T, ..., x_p^T)^T$, we define the p-dimensional subspace spanned by the orthonormal basis

$$V_1 = \begin{pmatrix} x_1 & \bar{0} & \dots & \bar{0} \\ \bar{0} & x_2 & \dots & \bar{0} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{0} & \bar{0} & \dots & x_p \end{pmatrix},$$

where $\bar{0}$ denotes the n-dimensional vector with all components equal to 0. Now we project the whole problem onto the orthogonal complement of the space spanned by V_1

$$\max_{||x_i||\leq 1,\ i=1,\dots,k} x^T(I - V_1V_1^T)N(I - V_1V_1^T)x. \tag{16}$$

It is clear that the maximizing vector is in the orthogonal complement of the space spanned by V_1, by Theorem 1. If we now find the maximizing vector of the projected problem, we define the space V_2 analogously to V_1, and project on the the orthogonal complement of the space spanned by V_1 and V_2. We can carry on this process at most n times.

3 Numerical Solution

3.1 Power Method

In this section we discuss a generalization of the power method to compute solutions to the equations (9)-(11). Consider the map g defined by

$$g(x_1, x_2) = \begin{pmatrix} \frac{N_{11}x_1+N_{12}x_2}{||N_{11}x_1+N_{12}x_2||} \\ \frac{N_{21}x_1+N_{22}x_2}{||N_{21}x_1+N_{22}x_2||} \end{pmatrix}. \tag{17}$$

Fixed points of the map g satisfy

$$\begin{aligned} \begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &= \begin{pmatrix} \lambda_1 I & 0 \\ 0 & \lambda_2 I \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \\ x_1^T x_1 &= 1, \\ x_2^T x_2 &= 1, \end{aligned}$$

where $\lambda_1 = ||N_{11}x_1 + N_{12}x_2||$ and $\lambda_2 = ||N_{21}x_1 + N_{22}x_2||$. This means that if iterating the map g will converge to a fixed point, it will converge to a solution of the Euler-Lagrange equations (9)-(11).

It is now clear that it is important to investigate the stability of fixed points of the map g. Suppose (x_1^*, x_2^*) is a fixed point of g, and $\lambda_1^* = ||N_{11}x_1^* + N_{12}x_2^*||$ and $\lambda_2^* = ||N_{21}x_1^* + N_{22}x_2^*||$. Then the Jacobian of g in the fixed point (x_1^*, x_2^*) is given by

$$J_g(x_1^*, x_2^*) = \begin{pmatrix} \frac{1}{\lambda_1^*}(I - x_1^*x_1^{*T}) & 0 \\ 0 & \frac{1}{\lambda_2^*}(I - x_2^*x_2^{*T}) \end{pmatrix}\begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix}.$$

Suppose that $y = (y_1, y_2)$ is an eigenvector of $J_g(x_1^*, x_2^*)$ with eigenvalue $\mu \neq 0$. Then y satisfies

$$\begin{pmatrix} (I - x_1^*x_1^{*T}) & 0 \\ 0 & (I - x_2^*x_2^{*T}) \end{pmatrix}\begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \mu\begin{pmatrix} \lambda_1^* y_1 \\ \lambda_2^* y_2 \end{pmatrix}.$$

Thus we have $x_1^{*T}y_1 = x_2^{*T}y_2 = 0$. This means that $y = (y_1^T, y_2^T)^T$ is an eigenvector of the generalized eigenproblem

$$PNPy = \mu P\Lambda Py,$$

where P and Λ are given by

$$P = \begin{pmatrix} (I - x_1^* x_1^{*T}) & 0 \\ 0 & (I - x_2^* x_2^{*T}) \end{pmatrix}, \quad \Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

Since both PNP and $P\Lambda P$ are positive semi-definite, it holds that the eigenvalue μ is real and non-negative.

Now look at the function

$$G(a) = \left(\frac{x_1^{*T} + a y_1^T}{||x_1^* + a y_1||}, \frac{x_2^{*T} + a y_2^T}{||x_2^* + a y_2||} \right) \begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix} \begin{pmatrix} \frac{x_1^* + a y_1}{||x_1^* + a y_1||} \\ \frac{x_2^* + a y_2}{||x_2^* + a y_2||} \end{pmatrix},$$

Suppose that the in fixed point (x_1^*, x_2^*) a maximum of our constraint optimization problem is attained. Then the function G attains a maximum in $a = 0$. We call a fixed point (x_1^*, x_2^*) of G a *non-degenerate* maximum if for all $(y_1, y_2) \neq 0$ with $y_1^T x_1 = y_2^T x_2 = 0$ it holds that $G''(0) < 0$.

Theorem 3. *Non-degenerate maxima are stable fixed points of the map g.*

Proof: Suppose $y = (y_1, y_2)$ is an eigenvector of the Jacobian of g in the non-degenerate maximum $x^* = (x_1^*, x_2^*)$, with eigenvalue μ. We saw that y satisfies the generalized eigenvalue problem

$$PNPy = \mu P\Lambda P y.$$

Now consider the function $G(a)$ corresponding to y. Since x^* is non-degenerate, we know that $G''(0) < 0$. But we can also compute

$$G''(0) = 2y^T PNPy - 2y^T P\Lambda P y.$$

It follows that $0 \leq y^T PNPy < y^T P\Lambda P y$, and therefore $0 \leq \mu < 1$. Since we assumed that y was an arbitrary eigenvector of the the Jacobian of g, the vector x^* must be a stable fixed point. $\square$

The theorem is stated and proven for the case $p = 2$, but it is not hard to see that all the arguments are easily extended to the cases $p > 2$.

For the standard positive semi-definite eigenvalue problem, if the largest eigenvalue is simple, we know that there is only one maximum. It is to this global maximum that the power method always converges (with probability one). In other words, there are no local maxima other than the global maximum. Now we would like to know whether it is possible to have local maxima for our constraint optimization problem. The following example shows that local maxima may exist.

Example 6. Consider the matrix

$$B = \begin{pmatrix} 0.4 & -0.4 & 0.2 & -0.4 \\ -0.1 & 0.4 & -0.2 & -0.4 \end{pmatrix}.$$

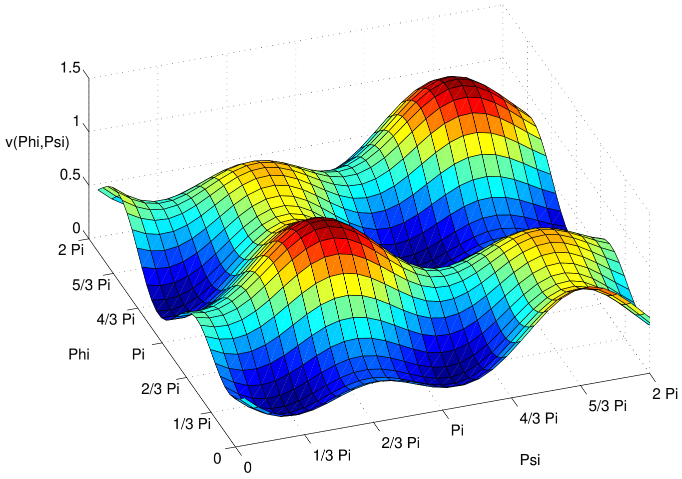

Figure 3: The value of $x^T N x$ as function of ϕ and ψ.

The matrix N given by $N = B^T B$ is a 4 by 4 matrix. Consider N in a 2 by 2 block form, where the blocks are 2 by 2 matrices. The sub-vectors of $x = (x_1, x_2)$ are 2 dimensional vectors. We would like to maximize over the set $||x_1|| = ||x_2|| = 1$. This set is the direct sum of two unit circles, which can be identified with a torus. We can parameterize these two unit circles with the angles ϕ and ψ. Now we will plot the value of

$$v(\phi, \psi) = (\cos\phi, \sin\phi, \cos\psi, \sin\psi) B^T B \begin{pmatrix} \cos\phi \\ \sin\phi \\ \cos\psi \\ \sin\psi \end{pmatrix},$$

for $0 \leq \phi < 2\pi$ and $0 \leq \psi < 2\pi$. The surface that is obtained in this way, is shown in Figure 3. Note that the surface is 2π-periodic. We clearly see that there is a global maximum and a second, local maximum. Both are stable fixed points of the power method. This means that we never know for certain when we have found a stable fixed point of the map g, whether this is the global maximum. This is, however, a common feature for the numerical solution of optimization problems.

3.2 Subspace Acceleration

For the standard eigenvalue problem $Nx = \lambda x$, the power iterations are known to converge to the largest eigenvalue of the operator N. This convergence can be very slow, and Krylov subspace methods such as the Lanczos method for positive definite operators and the Arnoldi method for general non-symmetric operators are successful attempts to accelerate the convergence of the power method.

The main idea behind these subspace acceleration methods is as follows. Let $U_m = \text{span}\{x, Nx, N^2x, N^{m-1}x\}$ be the Krylov subspace spanned by the first m iterates of the

(1) Supply $V = \boldsymbol{x}_0$, $W = N\boldsymbol{x}_0$, $y = 1$
(2) for $i = 0, 1, 2, 3, ...$

$$z = \Lambda(Wy)^{-1}Wy$$
$$v = (z - VV^T z)/||z - VV^T z||$$
$$V = [V, v]$$
$$W = [W, Nv]$$

Compute $\max_u(u^T V^T N V^T u)$
with $u^T V_1^T V_1 u = ... = u^T V_p^T V_p u = 1$

$$\boldsymbol{x}_{i+1} = Vu$$
$$r_{i+1} = (Wu - \Lambda(Wu)Vu)/||Wu||$$
$$y = u$$

Figure 4: The subspace accelerated power method.

power method, and let V_m be a orthonormal basis of U. The idea is to project the large, original, eigenvalue problem onto U_m:

$$V_m^T N V_m z = \lambda V_m^T V_m z = \lambda z, \tag{18}$$

and then to solve this m-dimensional eigenvalue problem: compute the largest eigenvalue of $V_m^T N V_m$.

We would like to apply the same idea to our constraint optimization problem. First we remind ourselves that we would like to solve

$$\begin{aligned} \begin{pmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &= \begin{pmatrix} \lambda_1 I & 0 \\ 0 & \lambda_2 I \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ x_1^T x_1 &= 1 \\ x_2^T x_2 &= 1 \end{aligned}$$

Suppose we have the first m iterates of the map g, defined as in (17), starting with initial vector $\boldsymbol{x}_0$: $\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(g(\boldsymbol{x}_0)..., g^{(m-1)}(\boldsymbol{x}_0)\}$.
Let $U_m = \text{span}\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(g(\boldsymbol{x}_0))..., g^{(m-1)}(\boldsymbol{x}_0)\}$ be the subspace spanned by these m iterates, and let

$$V = \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$$

be an orthonormal basis of U_m. The idea is again to project the original, high dimensional problem onto U_m: solve the m-dimensional optimization problem

$$\max_z (z^T V^T N V^T z) \text{ with } z^T V_1^T V_1 z = z^T V_2^T V_2 z = 1.$$

This can be solved using, e.g., again the Euler-Lagrange approach. The algorithm for the subspace accelerated power methods is given in Fig. 4 and the Newton algorithm for the

(1) Supply $B = V^T N V$, $W_j = V_j^T V_j$ for $j = 1, ..., p$, $i = 0$,
and $z_0 = (x_0, \lambda_0^{(1)}, ..., \lambda_0^{(p)})$
Compute $f = [2 \sum_j \lambda_0^{(j)} W_j x_0 - 2Bx_0; x_0^T W_1 x_0 - 1; ...; x_0^T W_p x_0 - 1]$
(2) while $||f||/||Bx|| > \epsilon$ and $i < m_{it}$ do
$f = [2 \sum_j \lambda_i^{(j)} W_j x_i - 2Bx_i; x_i^T W_1 x_i - 1; ...; x_i^T W_p x_i - 1]$
$J = 2[\sum_j \lambda_i^{(j)} W_j - B, W_1 x_i, ..., W_p x_i; [x_i^T W_1, \bar{0}]; ...; [x_i^T W_p, \bar{0}]]$
$z_{i+1} := (x_{i+1}, \lambda_{i+1}^{(1)}, ..., \lambda_{i+1}^{(p)}) = z_i - \delta J^{-1} f$
$i = i + 1$

Figure 5: Damped Newton algorithm for the projected problem, with damping δ (=0.5 in most experiments), convergence criterion ϵ and maximum number of Newton iteration m_{it}. (Note that the matrix J may not always be well conditioned. Using the pseudo-inverse of J instead of computing J^{-1} was in some cases beneficial. A thorough discussion of this topic is outside the scope of this paper.)

Euler-Lagrange equations of the projected problem is given in Fig. 5. In these algorithms, we use the following notation

$$\Lambda(x) = \begin{pmatrix} ||x_1||I & 0 & \dots & 0 \\ 0 & ||x_2||I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & ||x_p||I \end{pmatrix},$$

where $x = (x_1, x_2, ..., x_p)$.

In fact, the subspace U_m can be build up in two different ways:

1) $U_m = \text{span}\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(g(\boldsymbol{x}_0))..., g^{(m-1)}(\boldsymbol{x}_0)\}$. If we use these subspaces, then we know that the solution should converge, simply because the power iterations converge. When the power iterations converge slowly, then the differences between $g^{(m-1)}(\boldsymbol{x}_0)$ and $g^{(m)}(\boldsymbol{x}_0)$ may be very small, and there may be cancellation in the orthogonalization process for building an orthogonal basis of U_m. This may hamper the convergence.

2) $U_m = \text{span}\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(\boldsymbol{x}_1), ..., g(\boldsymbol{x}_{m-2})\}$, where $\boldsymbol{x}_i$ is the ith approximation produced in the subspace accelerated power method, with $\boldsymbol{x}_i \in U_{i+1}$. the subspace accelerated power method. The sequence $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, ...\}$, converges usually faster than the power method iterations, and therefore the differences between $\boldsymbol{x}_j$ and $\boldsymbol{x}_{j+1}$ may be larger than the differences in the power method sequence. This means that there is less chance for cancellation in the orthogonalization process. This may be the reason that this option is seen to perform better in the example problems. This option can also be implemented in a way that is slightly cheaper computationally than the first option.

Note that for the subspace accelerated method, we still need only one operator-vector multiplication with the operator N (which involves the integration of large linear and adjoint-linear systems) in each iteration of the method.

If we use the deflation technique discussed in Section 2.3, then we can use the subspace generated for the first "eigenvector" as a search space for the next "eigenvector". The algorithm for deflation in combination with the subspace accelerated power method is given in Fig. 6.

(1) Supply $V = \boldsymbol{x}_0$, $W = N\boldsymbol{x}_0$, $y = 1$, $U = [\,]$, $M = [\,]$, k_{min}. k_{max}
(2) for $j = 1, ..., k$
$\quad i = 0$
$\quad$ while no convergence
$\quad\quad z = \Lambda(Wy)^{-1}Wy$
$\quad\quad \bar{V} = [V, U]$
$\quad\quad v = (z - \bar{V}\bar{V}^T z)/||z - \bar{V}\bar{V}^T z||$
$\quad\quad V = [V, v]$
$\quad\quad W = [W, Nv]$
$\quad\quad M = [M, V^T z]$
$\quad\quad$ Compute $\max_u(u^T V^T N V^T u)$
$\quad\quad\quad$ with $u^T V_1^T V_1 u = ... = u^T V_p^T V_p u = 1$
$\quad\quad \boldsymbol{x}_{i+1} = Vu$
$\quad\quad r_{i+1} = (Wu - \Lambda(Wu)Vu)/||Wu||$
$\quad\quad y = u$
$\quad\quad i = i + 1$
$\quad\quad$ if $\dim(V) \geq k_{max}$
$\quad\quad\quad [Q, R] = qr(M(\text{end} - k_{min} : \text{end}), 0)$
$\quad\quad\quad V = VQ$
$\quad\quad\quad W = WQ$
$\quad\quad\quad M = [\,]$
$\quad\quad$ end if
$\quad$ end while

$\quad$ Construct $U = [U, Vy_1, ..., Vy_p]$
$\quad$ Compute orthonormal basis K for span$\{(I - UU^T)V\}$
$\quad$ $W = WV^T K$ and $V = K$ and $M = K^T VM$
$\quad$ Compute $\max_u(u^T V^T N V^T u)$
$\quad\quad$ with $u^T V_1^T V_1 u = ... = u^T V_p^T V_p u = 1$
$\quad$ $y = u$

Figure 6: The subspace accelerated power method with deflation.

3.3 Restarting the subspace accelerated power method

During the computing process the basis V of the subspace U_m keeps growing. If the number of iterations is large, then the memory requirements may become prohibitively large. In order to save on the memory costs, we can restart our method when the dimension of U_m exceeds a given number, say k_{max}. This means that we take the current best approximation as initial vector for a new run of the subspace accelerated power method.

A better way to restart the method may be not to throw away all of the subspace U_m, but to keep in storage a smaller subspace of U_m. Recall that the subspace U_m is given by

$$U_m = \text{span}\{u_1, u_2, ..., u_m\},$$

where $u_1 = \boldsymbol{x}_0$, $u_2 = g(\boldsymbol{x}_0)$,..., $u_m = g(\boldsymbol{x}_{m-2})$. In our numerical tests we experimented with keeping the k_{min}-dimensional subspace $\tilde{U}_m$ of U_m, given by

$$\tilde{U}_m = \text{span}\{u_{m-k_{min}+1}, ..., u_m\}.$$

Restarting techniques for eigenvalue computations can be found in, e.g., [14] and for linear systems in, e.g., [11].

3.4 Results

In this section, we discuss the results of the proposed numerical methods applied to test problems.

3.4.1 Test problem 1

First we consider the example where $N = A^T A$ is a 100 by 100 matrix with the matrix A a 50 by 100 matrix, given by $A = (B\ I)$, where B is the 50 by 50 tri-diagonal matrix with all ones on the diagonal an the sub- and super-diagonal, and I the 50 by 50 identity matrix. We fix $p = 2$, so that the dimension of the sub-vectors is 50. In Fig. 7, we see the convergence of the power method, when we start with the vector with all ones as components. On the y-axis the norm of the residual vector is given. The residual vector r_i is given by

$$r_i = (N\boldsymbol{x}_i - \Lambda(\boldsymbol{x}_i)\boldsymbol{x}_i)/||N\boldsymbol{x}_i||.$$

In Fig. 8, the convergence of two versions of the accelerated power method are given. For the gray line, the search subspace U_m is given by the iterates of the power method: $U_m = \text{span}\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(g(\boldsymbol{x}_0))..., g^{(m-1)}(\boldsymbol{x}_0)\}$. For the black line we build the search space as given by $U_m = \text{span}\{\boldsymbol{x}_0, g(\boldsymbol{x}_0), g(\boldsymbol{x}_1), ..., g(\boldsymbol{x}_{m-2})\}$, where $\boldsymbol{x}_i$ is the approximation of the "eigenvector" in the i-th iteration of the method. For the standard eigenvalue problem, these two approaches are mathematically equivalent (i.e., in exact arithmetic, but not computationally, due to rounding errors in the orthogonalization step). For our constraint optimization problem these two approaches are not equivalent in exact arithmetic. We see that the second option is performing better.

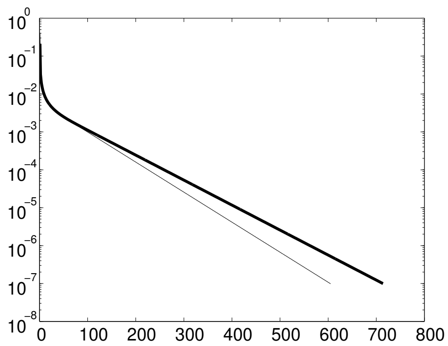

Figure 7: The norm of the residual versus the number of power method iterations for test problem 1, thick line: constraint optimization problem, thin line: largest eigenvalue.

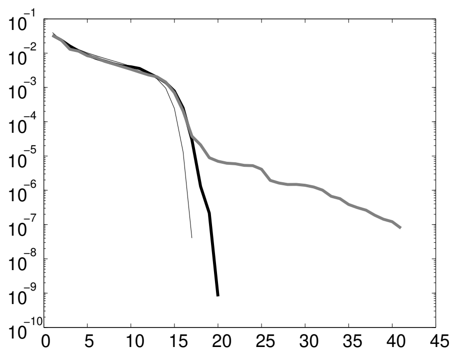

Figure 8: The norm of the residual versus the number of subspace accelerated power method iterations for the two different update strategies (thick lines) for test problem 1. The thin line is the convergence of the Lanczos method for the computation of the largest eigenvalue.

For comparison purposes, we also plotted in Figure 7 the convergence of the original power method for the computation of the largest eigenvalue of the matrix N. For the same reason we plotted the convergence of the Lanczos method in the Figure 8. In both cases the number of iterations for the eigenvalue computation and the constraint optimization problem are comparable.

As a comparison, we also solved the problem with the Matlab `fmincon` command, which is a SQP (sequential quadratic programming) method [3, 6]. This method needed 166 iterations and 517 matrix-vector products.

3.4.2 Test problem 2

During the iteration process the basis V of the subspace U_m keeps growing. In order to save on the memory costs, we can restart our subspace accelerated power method when the dimension of U_m exceeds k_{max}. For this test problem $N = M^T M$, with $M = (B\ E)$ where E is obtained by adding to the identity matrix 3's on the super-diagonal. For this

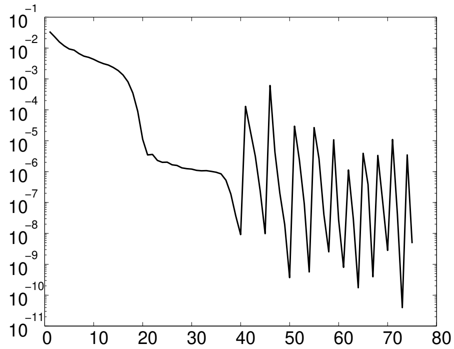

Figure 9: The norm of the residual versus the number of subspace accelerated power method iterations for test problem 2, with deflation used to compute 11 solutions.

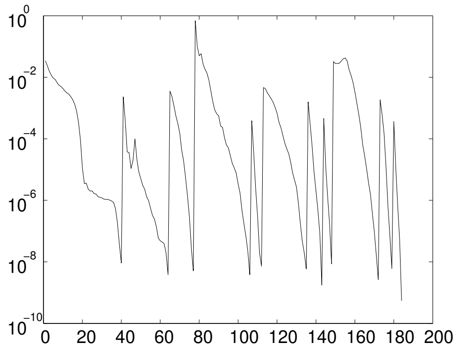

Figure 10: The norm of the residual versus the number of subspace accelerated power method iterations for test problem 2, with deflation used to compute 11 solutions and restarting with parameters $k_{max} = 40$ and $k_{min} = 10$.

test problem the power method needs 800 power iterations to obtain a single maximum. In Figure 9 the convergence of the subspace accelerated power method is depicted without restarting and in Figure 10 restarting is used with parameter values $k_{max} = 40$ and $k_{min} = 10$.

3.4.3 Test problem 3

This test problem is chosen to show that that the accelerated power method also works for a larger number of subvectors. Consider the 25 by 25 matrix A with -0.1 on the diagonal and -0.3 on the superdiagonal. We integrate the equation $\dot{y} = Ay + f(t)$, from $t = 0$ to $t = 10$, where we discretize $f(t)$ on 100 equidistant points. This means that we have 100 subvectors (so that $p = 100$), each of length 25. In Figure 11, the convergence behavior of the subspace accelerated power method is given, where we compute by deflation 6 solutions.

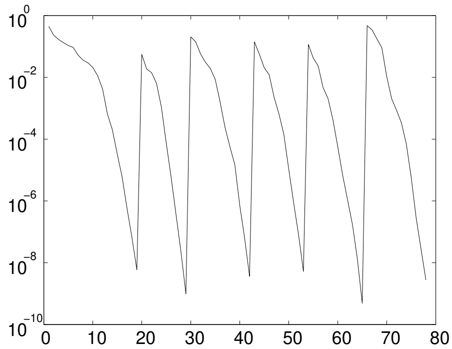

Figure 11: The norm of the residual versus the number of subspace accelerated power method iterations for test problem 3, with deflation used to compute 6 solutions.

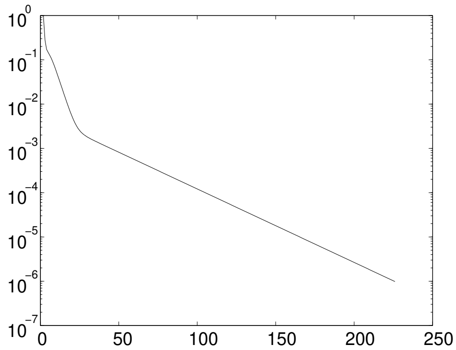

Figure 12: The norm of the residual versus the number of power method iterations for the Molteni-Marshall test problem with $p = 6$ and $T = 48$.

3.4.4 Test problem 4

The last test problem is a more realistic test problem. It is a nonlinear quasi-geostrophic model developed by Molteni and Marshall (see [9] for the model equations and details on the discretization) that describes the atmospheric dynamics. The model takes into account the spinning motion of the earth. The model is discretized using three vertical layers and horizontally the model is discretized using a spectral approximation. After the spatial discretization, a system of 1518 ordinary differential equations is obtained.

In the following numerical experiments, different values for the integration time, which is denoted by T (given in hours), are used, and also different values for the number of discretization intervals for the forcing $f(t)$, which is denoted by p.

In Fig. 12 the norm of the residual is plotted versus the number of iterations of the power method for $p = 6$ and $T = 48$. We see that the power method converges linearly, as we expected from the theory.

In Fig. 13, we see the norm of the residual plotted against the number of iterations of the accelerated power method ($p = 6$ and $T = 48$). Here we use the deflation technique as discussed in Section 2.3 to compute three "eigenvectors". Note that for the computa-

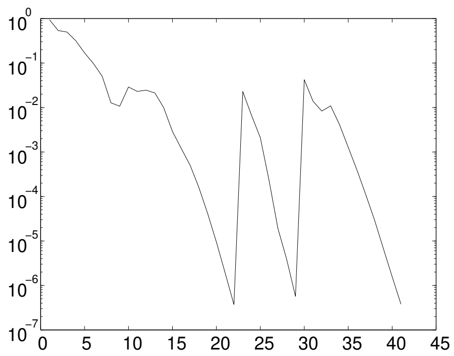

Figure 13: The norm of the residual versus the number of iterations of the accelerated power method for the Molteni-Marshall test problem with $p = 6$ and $T = 48$.

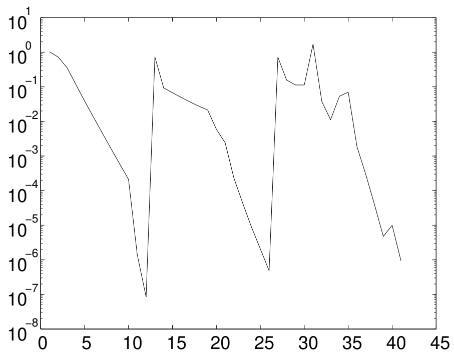

Figure 14: The norm of the residual versus the number of iterations of the accelerated power method for the Molteni-Marshall test problem with $p = 10$ and $T = 240$.

tion of the second and third "eigenvector" fewer iterations are needed than for the first "eigenvector".

In Fig. 14, we see the norm of the residual plotted against the number of iterations of the accelerated power method ($p = 10$ and $T = 240$), for two different convergence criteria. Here we also used deflation for the computation of three solutions. We see that for both the test problems, the subspace method accelerates the convergence of the power method considerably.

For the Molteni-Marshall test problem, we also experimented with computing more solutions. In Figure 15, 11 solutions are computed using deflation for $p = 6$ and $T = 24$. In Figure 16, a computation for the same values of T and p is shown, but now using a restarting strategy.

In the next experiments, we show the effect of the restarting strategy for the computation of a single solution. In Figure 17, four curves are shown. The fastest converging curve is the convergence of the subspace accelerated power method without restarting. The other three curves are obtained by restarting with $k_{max} = 20,\ 25,\ 30$. For all three computations $k_{min} = 10$. Figure 18 shows the convergence in case $k_{max} = 15$ and $k_{min} = 5$.

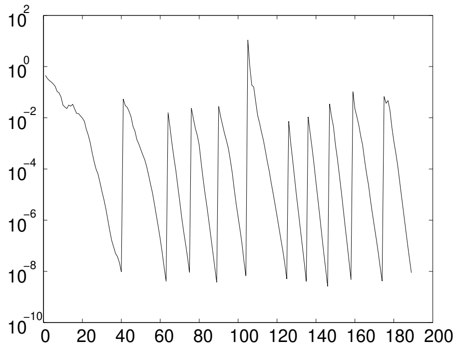

Figure 15: The norm of the residual versus the number of subspace accelerated power method iterations for the Molteni-Marshall test problem, with $p = 6$ and $T = 24$.

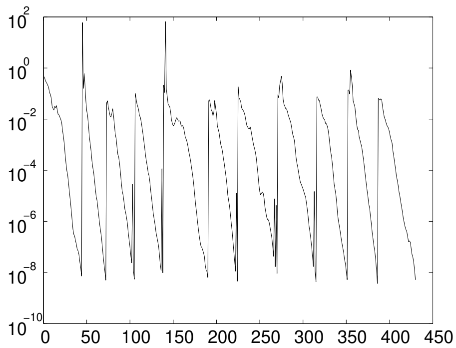

Figure 16: The norm of the residual versus the number of subspace accelerated power method iterations for the Molteni-Marshall test problem, with $p = 6$ and $T = 24$, using restarts with parameters $k_{min} = 10$ and $k_{max} = 30$.

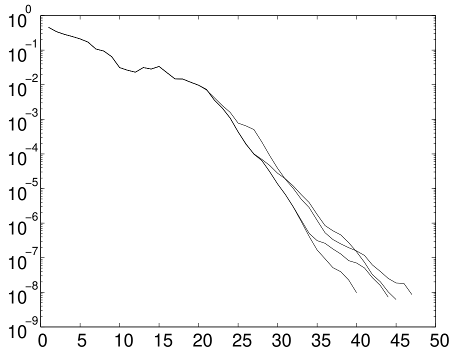

Figure 17: The norm of the residual versus the number of subspace accelerated power method iterations for the Molteni-Marshall test problem, with $p = 6$ and $T = 24$, using three different sets of restart parameters: $k_{max} = 20,\ 25,\ 30$. For all three computations $k_{min} = 10$.

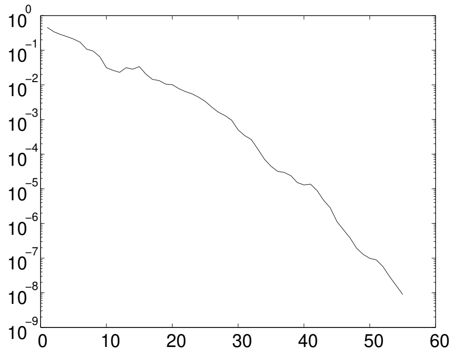

Figure 18: The norm of the residual versus the number of subspace accelerated power method iterations for the Molteni-Marshall test problem, with $p = 6$ and $T = 24$, using restart parameters $k_{min} = 5$ and $k_{max} = 15$.

(For all these computations we set $p = 6$ and $T = 24$.)

4 Conclusions

In this work, we have studied a constraint optimization problem that is motivated by the sensitivity analysis of continuous dynamical systems. After discretization, the optimal solution satisfies a set of equations that has many properties in common with the semi-definite eigenvalue problem. Therefore we presented adaptations of well known eigenvalue methods for the numerical solutions of the constraint optimization problem. The presented methods seem to be effective and are not much more expensive than solving the eigenvalue (or singular value) problem.

Ongoing research includes the careful consideration of the numerical solution of the projected optimization problem. The damped Newton method presented in this paper may not be the most efficient and reliable method. Another open problem is the deflation strategy. The deflation strategy presented in this paper is one of a number of possible choices. Which one is the most natural generalization of the eigenvalue case and which one is numerically most stable is subject to ongoing research.

APPENDIX

A Maximum is attained on the boundary (continuous case)

Consider the differential equation

$$\dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t), t)y(t) + f(t). \tag{19}$$

Using the variation of constants formula, solutions take the form

$$x(t) = M(0,t)x_0 + \int_0^t M(s,t)f(s)\,ds,$$

where $M(s,t)$ is the propagator from time s to time t. The operators $M(s,t)$ are invertible by uniqueness of solutions to the linear equations (19). We use the following notation

$$\boldsymbol{M}_{[a,b]}f = \int_a^b M(s,b)f(s)\,ds.$$

Definition 1. We call the equation (19) controllable if the map

$$f \to \boldsymbol{M}_{[a,b]}f$$

is onto for every a and b.

Theorem 4. *Suppose that (19) is controllable and that $f(t)$, with $||f(t)|| \leq 1$ for $t \in [0,T]$, is such that*

$$||\boldsymbol{M}_{[0,T]}f|| = \sup_{||g(t)||\leq 1,\ t\in[0,T]} ||\boldsymbol{M}_{[0,T]}g||,$$

then it holds that $||f(t)|| = 1$ almost everywhere in $[0,T]$.

Proof: Suppose that it does not hold that $||f(t)|| = 1$ almost everywhere in $[0,T]$. Then there exists an interval $[a,b] \subset [0,T]$ and $\delta > 0$ such that $||f(t)|| < 1 - \delta$ for $t \in [a,b]$. Because we assumed controllability, there exists a control $\tilde{f}$ such that

$$\boldsymbol{M}_{[a,b]}\tilde{f} = M(b,T)^{-1}\boldsymbol{M}_{[0,T]}f.$$

There exists an $\epsilon > 0$ such that for $g = f + \epsilon\tilde{f}$ holds that $||g(t)|| \le 1$ for $t \in [0, T]$. Now consider the solution of

$$\dot{y}(t) = \frac{\partial F}{\partial x}(x^*(t), t)y(t) + g(t), \;\; y(0) = 0.$$

This solution can be written as

$$\begin{aligned} y(T) &= M(0,T)y_0 + \int_0^T M(s,T)(f(s) + \epsilon\tilde{f}(s))\,ds = \int_0^T M(s,T)(f(s) + \epsilon\tilde{f}(s))\,ds \\ &= \boldsymbol{M}_{[0,T]}f + \epsilon(\int_0^a M(s,T)\tilde{f}(s)\,ds + \int_a^b M(s,T)\tilde{f}(s)\,ds + \int_b^T M(s,T)\tilde{f}(s)\,ds) \\ &= \boldsymbol{M}_{[0,T]}f + \epsilon M(b,T)\int_a^b M(s,b)\tilde{f}(s)\,ds = \boldsymbol{M}_{[0,T]}f + \epsilon M(b,T)\boldsymbol{M}_{[a,b]}\tilde{f} \\ &= (1+\epsilon)\boldsymbol{M}_{[0,T]}\tilde{f}. \end{aligned}$$

Now it follows that $||\boldsymbol{M}_{[0,T]}g|| > ||\boldsymbol{M}_{[0,T]}f||$. This is a contradiction, which proves the theorem. □

This theorem has connections with the bang-bang principle for minimum-time controls for linear systems, see [13]. This reference is also a good source for more details on controllability and optimal controls.

References

[1] J. Barkmeijer, T. Iversen, and T. N. Palmer, *Forcing singular vectors and other sensitive model perturbations*, Quart. J. Roy. Meteor. Soc., 129 (2003), pp. 2401–2424.

[2] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[3] T. Coleman, M. A. Branch, and A. Grace, *Optimization toolbox for use with Matlab, user's guide*, version 2 ed., 1999.

[4] P. J. Farrell, B. F. Ioannou, *Optimal excitation of linear dynamical systems by distributed forcing*, J. Atmos, Sci., (2003 (submitted)).

[5] D. R. Fokkema, G. L. G. Sleijpen, and H. A. Van der Vorst, *Accelerated inexact Newton schemes for large systems of nonlinear equations*, SIAM J. Sci. Comput., 19 (1998), pp. 657–674 (electronic).

[6] P. E. Gill, W. Murray, and W. M. H., *Practical Optimization*, Academic Press, London, 1981.

[7] G. H. Golub and C. F. Van Loan, *Matrix computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, third ed., 1996.

[8] M. E. Hochstenbach and B. Plestenjak, *A Jacobi-Davidson type method for a right definite two-parameter eigenvalue problem*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 392–410 (electronic).

[9] J. Marshall and F. Molteni, *Towards a dynamical understanding of planetary-scale flow regimes*, J. Atmos. Sci., 50 (1993), pp. 1792–1818.

[10] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Algorithms and Architectures for Advanced Scientific Computing, Manchester University Press, 1992.

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

[12] S. T. Smith, *Optimization techniques on Riemannian manifolds*, in Hamiltonian and gradient flows, algorithms and control, vol. 3 of Fields Inst. Commun., Amer. Math. Soc., 1994, pp. 113–136.

[13] E. D. Sontag, *Mathematical Control Theory*, Springer-Verlag New York, Inc., 1998.

[14] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Analysis and Applications, 13 (1992), pp. 357–385.

[15] L. N. Trefethen, A. E. Trefethen, S. C. Reddy, and T. A. Driscoll, *Hydrodynamic stability without eigenvalues*, Science, 261 (1993), pp. 578–584.