An Implicit Riemannian Trust-Region Method for the Symmetric Generalized Eigenproblem ⋆

C. G. Baker[1,2], P.-A. Absil[3,4], and K. A. Gallivan[1]

[1] School of Computational Science, Florida State University, Tallahassee, FL, USA,
[2] Computational Mathematics & Algorithms, Sandia National Laboratories, Albuquerque, NM, USA,
[3] Département d'ingénierie mathématique, Université catholique de Louvain, Belgium,
[4] Peterhouse, University of Cambridge, UK

Abstract. The recently proposed Riemannian Trust-Region method can be applied to the problem of computing extreme eigenpairs of a matrix pencil, with strong global convergence and local convergence properties. This paper addresses inherent inefficiencies of an explicit trust-region mechanism. We propose a new algorithm, the Implicit Riemannian Trust-Region method for extreme eigenpair computation, which seeks to overcome these inefficiencies while still retaining the favorable convergence properties.

1 Introduction

Consider $n \times n$ symmetric matrices A and B, with B positive definite. The generalized eigenvalue problem

$$Ax = \lambda Bx$$

is known to admit n real eigenvalues $\lambda_1 \leq \ldots \leq \lambda_n$, along with associated B-orthonormal eigenvectors $v_1, \ldots, v_n$ (see [1]). We seek here to compute the p leftmost eigenvectors of the pencil (A, B). It is known that the leftmost eigenspace $\mathcal{U} = \mathrm{colsp}(v_1, \ldots, v_p)$ of (A, B) is the column space of any minimizer of the generalized Rayleigh quotient

$$f : \mathbb{R}_*^{n\times p} \to \mathbb{R} : Y \mapsto \mathrm{trace}\left((Y^T BY)^{-1}(Y^T AY)\right), \tag{1}$$

where $\mathbb{R}_*^{n\times p}$ denotes the set of full-rank $n \times p$ matrices.

⋆ This work was supported by NSF Grant ACI0324944. The first author was in part supported by the CSRI, Sandia National Laboratories. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy; contract/grant number: DE-AC04-94AL85000. The second author was partially supported by Microsoft Research through a Research Fellowship at Peterhouse, Cambridge.

This result underpins a number of methods based on finding the extreme points of the generalized Rayleigh quotient (see [2–7] and references therein). Here, we consider the recently proposed Riemannian Trust-Region (RTR) method [8, 9]. This method formulates the eigenvalue problem as an optimization problem on a Riemannian manifold, utilizing a trust-region mechanism to find a solution. Similar to Euclidean trust-region methods [10, 11], the RTR method ensures strong global convergence properties while allowing superlinear convergence near the solution. However, the classical trust-region mechanism has some inherent inefficiencies. When the trust-region radius is too large, valuable time may be spent computing an update that may be rejected. When the trust-region radius is too small, we may reject good updates lying outside the trust-region. A second problem with the RTR method is typical of methods where the outer stopping criterion is evaluated only after exiting the inner iteration: in almost all cases, the last call to the inner iteration will perform more work than necessary to satisfy the outer stopping criterion.

The inefficiencies resulting from the trust-region mechanism can be addressed by disabling the trust-region mechanism in such a way as to preserve the desired convergence properties. One recent approach [12] describes a filter-trust-region method, where a modified acceptance criterion seeks to encourage convergence to critical points. Another technique was investigated in [13], specifically aimed at the generalized eigenproblem. The authors propose a hybrid method, consisting of two phases. The first phase utilizes an inner iteration [6] allowing the trust-region mechanisms to be disabled while still guaranteeing global convergence. This phase, however, achieves only a linear rate of convergence. The second phase employs a more accurate model which enables a superlinear rate of convergence, but requires the trust-region mechanism to guarantee global convergence.

In the current paper, we explore solutions to both of the problems described above. We present an analysis providing us knowledge of the model fidelity at every step of the inner iteration, allowing our trust-region to be based directly on the trustworthiness of the model. We propose a new algorithm, the Implicit Riemannian Trust-Region (IRTR) method, exploiting this analysis. We present convergence analysis showing that the IRTR method preserves the global convergence properties of the RTR method. We also describe problems arising when the inner iteration is allowed to stop early as a result of satisfying the outer stopping criterion, and we propose techniques to handle these problems. Section 2 describes the RTR method and analyzes the trust-region mechanism. Section 3 describes the IRTR method. Section 4 discusses convergence of the RTR method in light of the new analysis, as well as proposing a convergence theory for the IRTR method. Section 5 presents numerical results comparing the IRTR method against the RTR method.

2 Riemannian Trust-Region Method with Newton Model

The RTR method can be used to minimize the generalized Rayleigh quotient (1). The right-hand side of this function depends only on $\text{colsp}(Y)$, so that f induces a real-valued function on the set of p-dimensional subspaces of $\mathbb{R}^n$. (This set is known as the Grassmann manifold, which can be endowed with a Riemannian structure [4, 14].) The RTR method iteratively computes the minimizer of f by (approximately) minimizing successive models of f. The minimization of the models is done via an iterative process, which is referred to as the *inner iteration*, to distinguish it with the principal *outer iteration*. We present here the process in a way that does not require a background in differential geometry; we refer to [15] for the mathematical foundations of the technique.

Let Y be a full-rank, $n \times p$ matrix. We desire a correction S of Y such that $f(Y+S) < f(Y)$. A difficulty is that corrections of Y that do not modify its column space do not affect the value of the cost function. This situation leads to unpleasant degeneracy if it is not addressed. Therefore, we require S to satisfy some complementarity condition with respect to the space $\mathcal{V}_Y := \{YM : M \in \mathbb{R}^{p\times p}\}$. Here, in order to simplify later developments, we impose complementarity via B-orthogonality, namely $S \in \mathcal{H}_Y$ where

$$\mathcal{H}_Y = \{Z \in \mathbb{R}^{n\times p} : Y^T B Z = 0\}.$$

Consequently, the task is to minimize the function

$$\hat{f}_Y(S) := \operatorname{trace}\left(((Y+S)^T B (Y+S))^{-1}((Y+S)^T A (Y+S))\right), \quad S \in \mathcal{H}_Y.$$

The RTR method constructs a model m_Y of $\hat{f}_Y$ and computes an update S which approximately minimizes m_Y, so that the inner iteration attempts to solve the following problem:

$$\min m_Y(S), \quad S \in \mathcal{H}_Y, \quad \|S\|_2 \le \Delta,$$

where Δ (the *trust-region radius*) denotes the region in which we trust m_Y to approximate $\hat{f}_Y$. The next iterate and trust-region radius are determined by the performance of m_Y with respect to $\hat{f}_Y$. This performance ratio is measured by the quotient:

$$\rho_Y(S) = \frac{\hat{f}_Y(0) - \hat{f}_Y(S)}{m_Y(0) - m_Y(S)}.$$

Low values of $\rho_Y(S)$ (close to zero) indicate that the model m_Y at S is not a good approximation to $\hat{f}_Y$. In this scenario, the trust-region radius is reduced and the update $Y+S$ is rejected. Higher values of $\rho_Y(S)$ allow the acceptance of $Y+S$ as the next iterate, and a value of $\rho_Y(S)$ close to one suggests good approximation of $\hat{f}_Y$ by m_Y, allowing the trust-region radius to be enlarged. The mechanism is described in more detail in Algorithm 1.

Usually, the model m_Y is chosen as a quadratic function approximating $\hat{f}_Y$. In the sequel, in contrast to [9] where the quadratic term of the model was unspecified, we assume that m_Y is the *Newton model*, i.e., the quadratic expansion of $\hat{f}_Y$ at $S=0$. Then, assuming from here on that $Y^T B Y = I_p$, we have

$$\begin{aligned} m_Y(S) &= \operatorname{trace}\left(Y^T A Y\right) + 2\operatorname{trace}\left(S^T A Y\right) + \operatorname{trace}\left(S^T\left(AS - BS(Y^T A Y)\right)\right) \\ &= \hat{f}_Y(0) + \operatorname{trace}\left(S^T \nabla \hat{f}_Y\right) + \frac{1}{2}\operatorname{trace}\left(S^T H_Y[S]\right), \end{aligned}$$

Algorithm 1 (Riemannian Trust-Region Algorithm [8, 9])
Data: A,B symmetric, B positive definite, $\rho' \in (0, \frac{1}{4})$
Input: Initial iterate $\mathcal{W}_0 = \mathrm{colsp}(Y_0)$
for $k = 0, 1, 2, \ldots$

    Model-based Minimization
    Generate Y_k *using a Rayleigh-Ritz procedure on* $\mathcal{W}_k$
    Compute $\nabla \hat{f}_{Y_k}$ *and check* $\|\nabla \hat{f}_{Y_k}\|_2$ *for convergence*
    Compute S_k *to approximately minimize* m_{Y_k} *such that* $\|S_k\|_2 \leq \Delta_k$
    Compute $\rho_k = \rho_{Y_k}(S_k)$

    Choose next trust-region radius
    if $\rho_k < \frac{1}{4}$
        $\Delta_{k+1} = \frac{1}{4}\Delta_k$
    elseif $\rho_k > \frac{3}{4}$ *and* $\|S_k\|_2 = \Delta_k$
        $\Delta_{k+1} = \min\{2\Delta_k, \Delta_{max}\}$
    else
        $\Delta_{k+1} = \Delta_k$
    end

    Generate next iterate
    if $\rho_k > \rho'$
        if *performing subspace acceleration*
            Compute new acceleration subspace $\mathcal{W}_{k+1}$ *from* $\mathcal{W}_k$ *and* S_k
        else
            Set $\mathcal{W}_{k+1} = \mathrm{colsp}(Y_k + S_k)$
        end
    else
        $\mathcal{W}_{k+1} = \mathcal{W}_k$
    end

end for.

---

where the gradient and the effect of the Hessian of $\hat{f}_Y$ are identified as

$$\nabla \hat{f}_Y = 2P_{BY}AY \qquad H_Y[S] = 2P_{BY}\left(AS - BS(Y^TAY)\right),$$

and where $P_{BY} = I - BY(Y^TBBY)^{-1}Y^TB$ is the orthogonal projector on the space perpendicular to the column space of BY. This model is minimized using a Steihaug-Toint truncated conjugate gradient method as described in [9] and discussed in Section 3.

Simple manipulation shows the following:

$$\begin{aligned}\hat{f}_Y(0) - \hat{f}_Y(S) &= \mathrm{trace}\left(Y^TAY - (I + S^TBS)^{-1}(Y+S)^TA(Y+S)\right) \\ &= \mathrm{trace}\left((I + S^TBS)^{-1}(S^TBS(Y^TAY) - 2S^TAY - S^TAS)\right).\end{aligned}$$

Consider the case where $p = 1$. The above equation simplifies to

$$\begin{aligned}\hat{f}_y(0) - \hat{f}_y(s) &= (1 + s^TBs)^{-1}\left(s^TBsy^TAy - 2s^TAy - s^TAs\right) \\ &= (1 + s^TBs)^{-1}\left(m_y(0) - m_y(s)\right),\end{aligned}$$

so that

$$\rho_y(s) = \frac{\hat{f}_y(0) - \hat{f}_y(s)}{m_y(0) - m_y(s)} = \frac{1}{1 + s^T B s}. \tag{2}$$

This allows the model performance ratio ρ_y to be constantly evaluated as the model minimization progresses, simply by tracking the B-norm of the current update vector. Furthermore, because the RTR method, when applied to computing extreme eigenspaces (as described in [9]), uses the 2-norm for measuring the trust-region radius, and because this is related to ρ by the above formula, we can make some statements about the behavior of the trust-region when $B = I$.

Assume that $p = 1$ and $B = I$. Given an iterate y_k, a trust-region radius Δ_k, and an update vector $s_k, \|s_k\|_2 \le \Delta_k$, the trust-region radius for the next iteration is given by the following:

$$\Delta_{k+1} = \begin{cases} \frac{1}{4}\Delta_k, & \text{if } \rho_{y_k}(s_k) < \frac{1}{4} \\ \min\{2\Delta_k, \Delta_{max}\}, & \text{if } \rho_{y_k}(s_k) > \frac{3}{4} \text{ and } \|s_k\|_2 = \Delta_k \\ \Delta_k, & \text{otherwise} \end{cases}.$$

Assume that the trust-region radius at iteration k satisfies the following:

$$\frac{1}{\sqrt{3}} < \Delta_k < \sqrt{3}.$$

It follows that

$$0 < \|s_k\|_2 < \sqrt{3}$$

which, with (2), yields

$$\frac{1}{4} < \rho_{y_k}(s_k) < 1.$$

Therefore, the trust-region radius will not be reduced for the next iteration and the next iterate will be accepted. Next, assume that $\rho_{y_k}(s_k) > \frac{3}{4}$. Then (2) requires

$$\|s_k\|_2 < \frac{1}{\sqrt{3}} < \Delta_k,$$

and the trust-region radius will not be increased for the next iteration. Therefore, for problems where $p = 1$ and $B = I$, initializing the trust-region radius in the range $\left(\frac{1}{\sqrt{3}}, \sqrt{3}\right)$ will guarantee acceptance of all computed iterates and ensure a static trust-region radius. This negates the need to compute ρ, while still guaranteeing the strong convergence properties of the RTR method.

3 Implicit Riemannian Trust-Region Method

In this section, we explore the possibility of selecting the trust-region as a sublevel set of the performance ratio ρ_Y. We dub this approach the Implicit Riemannian Trust-Region method.

3.1 Case $p = 1$

The analysis of ρ in the previous section shows that for the generalized Rayleigh quotient with $p = 1$, the performance of the model decreases as the iterate moves away from zero. Since the model is formed from the Taylor expansion of $\hat{f}$, this is to be expected, and it is the motivating intuition behind trust-region methods. However, in the case of the $p = 1$ generalized Rayleigh quotient, $\rho_y(s)$ has a simple relationship with $\|s\|_B$. Therefore, by monitoring the B-norm of the inner iterate, we can easily determine the value of ρ for a given inner iterate. Furthermore, the relationship between ρ and the B-norm of a vector, allows us to move along a search direction to a specific value of ρ. These two things, combined, enable us to redefine the trust-region based instead on the value of ρ.

The truncated conjugate gradient proposed in [9] for use in the simple RTR algorithm seeks to minimize the model m_Y within a trust-region defined explicitly as $\{s : \|s\|_2 \leq \Delta\}$. Here, we change the definition of the trust-region to $\{s : \rho_y(s) \geq \rho'\}$, for some $\rho' \in (0, 1)$. The necessary modifications to this algorithm are very simple. The definition of the trust-region occurs in three places: when detecting whether the trust-region has been breached; when constraining the update vector in the case that the trust-region was breached; and when constraining the update vector in the case that we have detected a direction of negative curvature. The first of these requires only that we monitor the B-norm of the iterate. The latter two cases require that the iterate s_j moves in the direction d_j, to the edge of the trust-region. The new inner iteration is listed in Algorithm 2, with the differences highlighted.

Having stated the definition of the implicit trust-region, based on ρ, we need a mechanism for following a search direction to the edge of the trust-region. That is, at some outer step k and given s_j and a search direction d_j, we wish to compute $s = s_j + \tau d_j$ such that $\rho_{y_k}(s) = \rho'$. We know that $\rho_{y_k}(s) = \left(1 + s^T B s\right)^{-1}$. Given ρ', we want

$$s^T B s = \frac{1}{\rho'} - 1 = s_j^T B s_j + \tau 2 s_j^T B d_j + \tau^2 d_j^T B d_j.$$

We wish to find a value of τ that satisfies this equation. Furthermore, because d_j is a direction of descent, we require that τ is positive. Denote our desired B-norm by

$$\Delta_{\rho'} = \sqrt{\frac{1}{\rho'} - 1}. \tag{3}$$

Solving the quadratic system yields

$$\tau_* = \frac{-d_j^T B s_j + \sqrt{(d_j^T B s_j)^2 + d_j^T B d_j (\Delta_{\rho'}^2 - s_j^T B s_j)}}{d_j^T B d_j}. \tag{4}$$

The previous iterate, s_j, was inside the trust-region, so that $s_j^T B s_j < \Delta_{\rho'}^2$. With this and the positive-definiteness of B, it is easily shown that τ_* is the unique positive solution to (4). A careful implementation precludes the need for

Algorithm 2 (Preconditioned Truncated CG (IRTR))
Data: A,B *symmetric,* B *positive definite,* $\rho' \in (0,1)$, *preconditioner* M
Input: Iterate y, $y^T B y = 1$
Set $s_0 = 0$, $r_0 = \nabla \hat{f}_y$, $z_0 = M^{-1} r_0$, $d_0 = -z_0$
for $j = 0, 1, 2, \ldots$

Check κ/θ ***stopping criterion***
if $\|r_j\|_2 \le \|r_0\|_2 \min\left\{\kappa, \|r_0\|_2^\theta\right\}$
return s_j

Check curvature of current search direction
if $d_j^T H_y[d_j] \le 0$
Compute τ *such that* $s = s_j + \tau d_j$ *satisfies* $\boxed{\rho_y(s) = \rho'}$
return s

Set $\alpha_j = (z_j^T r_j)/(d_j^T H_y[d_j])$

Generate next inner iterate
Set $s_{j+1} = s_j + \alpha_j d_j$

Check implicit trust-region
if $\boxed{\rho_y(s_{j+1}) < \rho'}$
Compute $\tau \ge 0$ *such that* $s = s_j + \tau d_j$ *satisfies* $\boxed{\rho_y(s) = \rho'}$
return s

Use CG recurrences to update residual and search direction
Set $r_{j+1} = r_j + \alpha_j H_y[d_j]$
Set $z_{j+1} = M^{-1} r_{j+1}$
Set $\beta_{j+1} = (z_{j+1}^T r_{j+1})/(z_j^T r_j)$
Set $d_{j+1} = -z_{j+1} + \beta_{j+1} d_j$

Check outer stopping criterion
$\boxed{\text{Compute } \|\nabla \hat{f}_{y+s_{j+1}}\|_2 \text{ and test}}$

end for.

any more matrix multiplications against B than are necessary to perform the iterations.

Another enhancement in Algorithm 2 is that the outer stopping criterion is tested during the inner iteration. This technique is not novel in the context of eigensolvers with inner iterations, having been proposed by Notay [16]. Our motivation for introducing this test is that, when it is absent, the final outer step may reach a much higher accuracy than specified by the outer stopping criterion, resulting in a waste of computational effort. Also, while Notay proposed a formula for the inexpensive evaluation of the outer norm based on the inner iteration, we must rely on a slightly more expensive, but less frequent, explicit evaluation of the outer stopping criterion.

The product of this iteration is an update vector s_j which is guaranteed to lie inside of the ρ-based trust-region. The result is that the ρ value of the new

Algorithm 3 (Implicit Riemannian Trust-Region Algorithm)
Data: A,B symmetric, B positive definite, $\rho' \in (0,1)$
Input: Initial subspace $\mathcal{W}_0$
for $k = 0, 1, 2, \ldots$

Model-based Minimization
Generate y_k *using a Rayleigh-Ritz procedure on* $\mathcal{W}_k$
Compute $\nabla \hat{f}_{y_k}$ *and check* $\|\nabla \hat{f}_{y_k}\|_2$
Compute s_k *to approximately minimize* m_{y_k} *such that* $\rho(s_k) \geq \rho'$ *(Algorithm 2)*

Generate next subspace
if *performing subspace acceleration*
Compute new acceleration subspace $\mathcal{W}_{k+1}$ *from* $\mathcal{W}_k$ *and* s_k
else
Set $\mathcal{W}_{k+1} = \mathrm{colsp}(y_k + s_k)$
end

end for.

iterate need not be explicitly computed, the new iterate can be automatically accepted, with an update vector constrained by model fidelity instead of a discretely chosen trust-region radius based on the performance of the last iterate. An updated outer iteration is presented in Algorithm 3, which also features an optional subspace acceleration enhancement à la Davidson [17].

This new algorithm modifies the trust-region framework to ensure that the model minimization occurs only in a range of user-specified fidelity. This is related to the stopping criterion proposed by Notay [16, pg. 34]. He derives a formula for evaluating the norm of the outer residual at each inner iteration, in the context of the related Jacobi-Davidson iterative eigensolver. He goes on to suggest a stopping criterion for the inner iteration, triggered when the ratio between successive inner residuals is less than the ratio between successive outer residuals produced using the update vectors. Similar to the IRTR and other trust-region mechanism, this suggestion attempts to halt the inner iteration when it becomes inefficient, as measured by the progress made in the global problem.

3.2 A Block Algorithm

The analysis of Section 2 seems to preclude a simple formula for ρ in the case that $p > 1$. We wish, however, to have a block algorithm. The solution is to decouple the block Rayleigh quotient into the sum of p separate rank-1 Rayleigh quotients, which can then be addressed individually using the IRTR strategy. This is done as follows.

Assume that our iterates satisfy $Y^T AY = \Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_p)$, in addition to $Y^T BY = I_p$. In fact, this is a natural consequence of the Rayleigh-Ritz process. Then given $Y = \left[y_1 \dots y_p\right]$, the model m_Y can be rewritten:

$$\begin{aligned} m_Y(S) &= \text{trace}\left(Y^T AY + 2S^T AY + S^T(AS - BS(Y^T AY))\right) \\ &= \text{trace}\left(\Sigma + 2S^T AY + S^T(AS - BS\Sigma)\right) \\ &= \sum_{i=1}^{p} \left(\sigma_i + 2s_i^T Ay_i + s_i^T(A - \sigma_i B)s_i\right) \\ &= \sum_{i=1}^{p} m_{y_i}(s_i). \end{aligned}$$

It should be noted that the update vectors for the decoupled minimizations must have the original orthogonality constraints in place. That is, instead of requiring only that $y_i^T Bs_i = 0$, we require that $Y^T Bs_i = 0$ for each s_i. This is necessary to guarantee that the next iterate, $Y + S$, has full rank, so that the Rayleigh quotient is defined.

As for the truncated conjugate gradient, the p individual IRTR subproblems should be solved simultaneously, with the inner iteration stopped as soon as any of the iterations satisfy one of the inner stopping criteria (exceeded trust-region or detected negative curvature). If only a subset of iterations are allowed to continue, then the κ/θ inner stopping criterion may not be feasible.

3.3 Difficulties with Early Stopping

As stated earlier, one of the problems with methods that utilize an inner iteration is that the last call to the inner iteration can proceed much longer than is necessary to satisfy the outer stopping criterion. A solution to this problem is to evaluate the outer stopping criterion from the inner iteration. As the purpose of this is simply to prevent the inner iteration from performing grossly excess work, this check need not be performed on every inner iteration.

Notay [16] devises a simple formula that relates the norm of the outer residual (a common measure of global convergence) to the norm of the inner residual and the coefficients of the conjugate gradient process. It is uncertain how numerically reliable this formula is in practice, as the conjugacy properties of CG are known to suffer as the number of iterations increases. This is especially likely to be true in the scenario in which we are interested: the last call to the model minimization CG routine often requires the largest number of iterations.

Furthermore, due to differences between the inner iteration in [16] and the current work, there is currently no similar formula relating the inner and outer residuals. Therefore, we propose explicitly forming the outer residual during the inner iteration. This requires more vector arithmetic than Notay's formula, although a clever implementation requires no additional matrix-vectors products against A or B.

Unfortunately, this benefit is not without its consequences. Early termination of the model minimization can result in update vectors S that are very small in norm, relative to those of the orthonormalized outer iterate Y. This causes problems when performing the Rayleigh-Ritz procedure, as $\hat{B} = \begin{bmatrix} Y & S \end{bmatrix}^T B \begin{bmatrix} Y & S \end{bmatrix}$ can become ill-conditioned, resulting in a loss of orthogonality of the computed Ritz vectors.

However, the Rayleigh-Ritz process is dependent only on the subspace, so that we can choose any basis to represent the subspace. In order to help alleviate the conditioning problems associated with the projected B matrix, we suggest two strategies. The first suggestion is to scale the vectors of S so that they each have unit B-norm. The second involves checking the orthogonality of the produced Ritz vectors, and choosing a smaller number of vectors if the loss in orthogonality is too high. The new basis is chosen using the singular value decomposition of $\hat{B}$. In numerical tests, the combination of these two strategies was sufficient to alleviate the problems described above.

4 Analysis of Convergence

The mechanisms of the IRTR method are sufficiently different from those of the RTR method that we must construct a separate convergence theory, albeit one that is readily adapted from the classical trust-region theory. This section presents a new definition for the Cauchy point and a proof of strong global convergence for the IRTR in the $p = 1$ case. The outline of a $p > 1$ convergence result is presented as well.

Given a model m_y, the Cauchy point is defined as the point which minimizes m_y along the direction of steepest descent, subject to the trust-region constraints. The IRTR redefines the concept of the trust-region; therefore, the Cauchy point is redefined as well. Theorem 2 gives a formula for the Cauchy point. Theorem 3 proves a bound on its decrease under the model. Both of these results are analogous to those from classical trust-region theory [18, 11].

Definition 1 *Cauchy Point*
Consider the Newton model m_y of $\hat{f}_y$ and $\rho' \in (0, 1)$, with

$$m_y(p) = \hat{f}_y(0) + \nabla \hat{f}_y^T p + \frac{1}{2} p^T H_y[p].$$

The Cauchy point p_y^C *is the point that minimizes m_y in the direction of steepest descent, subject to the trust-region constraint $\rho_y(p) \geq \rho'$.*

Theorem 2 *Cauchy Point Computation*
Given the Newton model m_y of $\hat{f}_y$ and $\rho' \in (0, 1)$, with

$$m_y(p) = \hat{f}_y(0) + \nabla \hat{f}_y^T p + \frac{1}{2} p^T H_y[p].$$

The Cauchy point (Definition 1) is given by

$$p_y^C = \tau_y p_y^S,$$

where $\Delta_{\rho'}$ *is defined as in* (3) *and*

$$p_y^S = -\Delta_{\rho'} \frac{\nabla \hat{f}_y}{\|\nabla \hat{f}_y\|_B}$$

$$\tau_y = \begin{cases} 1, & \text{if } \nabla \hat{f}_y^T H_y[\nabla \hat{f}_y] \leq 0 \\ \min\left\{1, \frac{1}{\Delta_{\rho'}} \frac{\|\nabla \hat{f}_y\|_2^2 \|\nabla \hat{f}_y\|_B^2}{\nabla \hat{f}_y^T H_y[\nabla \hat{f}_y]}\right\}, & \text{otherwise} \end{cases}$$

Proof. The direction of steepest descent of m_y is $-\nabla \hat{f}_y$. Following this direction to the edge of the trust-region gives us p_y^S, which is verified as follows:

$$\begin{aligned} \rho_y(p_y^S) &= \frac{1}{1 + (p_y^S)^T B (p_y^S)} \\ &= \left(1 + \Delta_{\rho'}^2 \frac{\nabla \hat{f}_y^T B \nabla \hat{f}_y}{\|\nabla \hat{f}_y\|_B^2}\right)^{-1} \\ &= \left(1 + \frac{1}{\rho'} - 1\right)^{-1} \\ &= \rho'. \end{aligned}$$

It remains to find the minimum of m_y along this direction. Consider the case where $\nabla \hat{f}_y^T H_y[\nabla \hat{f}_y] \leq 0$. Then $m_y(\tau p_y^S)$ decreases monotonically as τ increases, with a minimizer constrained by the trust-region at $\tau = 1$. Otherwise, when $\nabla \hat{f}_y^T H_y[\nabla \hat{f}_y] > 0$, we find by differentiating $m_y(\tau p_y^S)$ with respect to τ,

$$\frac{d}{d\tau} m_y(\tau p_y^S) = -\Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} + \tau \Delta_{\rho'}^2 \frac{\nabla \hat{f}_y^T H_y[\nabla \hat{f}_y]}{\|\nabla \hat{f}_y\|_B^2},$$

which has a unique root at

$$\tau_* = \frac{1}{\Delta_{\rho'}} \frac{\|\nabla \hat{f}_y\|_2^2 \|\nabla \hat{f}_y\|_B^2}{\nabla \hat{f}_y^T H_y[\nabla \hat{f}_y]}.$$

If $\tau_* > 1$, then the point $\tau_* p_y^S$ is outside of the trust-region. For the purpose of constructing the Cauchy point, τ_y must be constrained to 1. □

The next theorem concerns the decrease in m_y associated with its Cauchy point. This also is a standard result from classical trust-region theory, modified here only to reflect the new definition of the trust-region.

Theorem 3 *Cauchy Point Decrease*
Given the Newton model m_y *of* $\hat{f}_y$ *and* $\rho' \in (0, 1)$,

$$m_y(p) = \hat{f}_y(0) + \nabla \hat{f}_y^T p + \frac{1}{2} p^T H_y[p],$$

the decrease in m_y of the Cauchy point p_y^C satisfies

$$m_y(0) - m_y(p_y^C) \geq \frac{1}{2}\|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2}{\|H_y\|_2} \right\}.$$

Proof. Let p_y^C be the Cauchy point, with terms $\Delta_{\rho'}$ and τ_y defined as in (3) and Theorem 2. Then the decrease in the model under the Cauchy point is as follows:

$$m_y(0) - m_y(p_y^C) = \tau_y \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} - \frac{1}{2}\tau_y^2 \frac{\Delta_{\rho'}^2}{\|\nabla \hat{f}_y\|_B^2} \nabla \hat{f}_y^T H_y [\nabla \hat{f}_y].$$

Consider the case where $\nabla \hat{f}_y^T H_y [\nabla \hat{f}_y] \leq 0$. It follows that $\tau_y = 1$ and

$$\begin{aligned}
m_y(0) - m_y(p_y^C) &= \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} - \frac{1}{2}\Delta_{\rho'}^2 \frac{1}{\|\nabla \hat{f}_y\|_B^2\|} \nabla \hat{f}_y^T H_y [\nabla \hat{f}_y] \\
&\geq \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} + \frac{1}{2}\Delta_{\rho'}^2 \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B^2} \|H_y\|_2 \\
&\geq \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} \geq \|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2}{\|H_y\|_2} \right\} \\
&\geq \frac{1}{2}\|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2}{\|H_y\|_2} \right\},
\end{aligned}$$

and we have the desired result.

If instead $\nabla \hat{f}_y^T H_y [\nabla \hat{f}_y] > 0$, then

$$\tau_y = \min\left\{ 1, \frac{1}{\Delta_{\rho'}} \frac{\|\nabla \hat{f}_y\|_2^2 \|\nabla \hat{f}_y\|_B}{\nabla \hat{f}_y^T H_y [\nabla \hat{f}_y]} \right\}$$

and

$$\begin{aligned}
m_y(0) - m_y(p_y^C) &\geq \tau_y \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} - \frac{1}{2}\tau_y \frac{\Delta_{\rho'}^2}{\|\nabla \hat{f}_y\|_B^2} \nabla \hat{f}_y^T H_y [\nabla \hat{f}_y] \frac{1}{\Delta_{\rho'}} \frac{\|\nabla \hat{f}_y\|_2^2 \|\nabla \hat{f}_y\|_B}{\nabla \hat{f}_y^T H_y [\nabla \hat{f}_y]} \\
&\geq \tau_y \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} - \frac{1}{2}\tau_y \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} \\
&= \frac{1}{2}\tau_y \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2^2}{\|\nabla \hat{f}_y\|_B} \\
&= \frac{1}{2}\|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2^3}{\nabla \hat{f}_y^T H_y [\nabla \hat{f}_y]} \right\} \\
&\geq \frac{1}{2}\|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2}{\|H_y\|_2} \right\},
\end{aligned}$$

yielding again the desired result. □

The convergence theory of the RTR method [15] provides two results on global convergence. The stronger of these results states that the gradients of a sequence of iterates produced by the algorithm, converge to zero. Theorem 4 proves this for the IRTR method described in Algorithm 3.

Theorem 4 *Global Convergence*
Let $\{y_k\}$ be a sequence of iterates produced by Algorithm 3 with $\rho' \in (0,1)$ and without subspace acceleration. Suppose also that there exists $\beta > 0$ such that $\|H_{y_k}\|_2 \leq \beta$ and that f is bounded below on the level set

$$\{y : f(y) \leq f(y_0)\}.$$

Further suppose that each update s_k produces at least as much decrease in m_{y_k} as a fixed fraction of the Cauchy point. That is, for some constant $c_1 > 0$,

$$m_{y_k}(0) - m_{y_k}(s_k) \geq c_1 \|\nabla \hat{f}_y\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_y\|_2}{\|\nabla \hat{f}_y\|_B}, \frac{\|\nabla \hat{f}_y\|_2}{\|H_y\|_2} \right\}.$$

Then

$$\lim_{k\to\infty} \|\nabla f(y_k)\|_2 = 0.$$

Proof. Assume for the purpose of contradiction that the theorem does not hold. Then there exists $\epsilon > 0$ such that, for all $K > 0$, $\exists k \geq K$ such that

$$\|\nabla f(y_k)\|_2 > \epsilon.$$

From the workings of Algorithm 3,

$$\begin{aligned} f(y_k) - f(y_{k+1}) &= \hat{f}_{y_k}(0) - \hat{f}_{y_k}(s_k) = \rho_{y_k}(s_k)\left(m_{y_k}(0) - m_{y_k}(s_k)\right) \\ &\geq \rho'\left(m_{y_k}(0) - m_{y_k}(s_k)\right) \\ &\geq \rho' c_1 \|\nabla \hat{f}_{y_k}\|_2 \min\left\{ \Delta_{\rho'} \frac{\|\nabla \hat{f}_{y_k}\|_2}{\|\nabla \hat{f}_{y_k}\|_B}, \frac{\|\nabla \hat{f}_{y_k}\|_2}{\|H_{y_k}\|_2} \right\} \\ &\geq \rho' c_1 \|\nabla \hat{f}_{y_k}\|_2 \min\left\{ \Delta_{\rho'} \frac{1}{\|B\|_2}, \frac{\|\nabla \hat{f}_{y_k}\|_2}{\beta} \right\}, \end{aligned}$$

and for all $K > 0$, $\exists k \geq K$ such that

$$f(y_k) - f(y_{k+1}) \geq \rho' c_1 \epsilon \min\left\{ \Delta_{\rho'} \frac{1}{\|B\|_2}, \frac{\epsilon}{\beta} \right\} > 0.$$

But because f is bounded below and decreases monotonically with the iterates produced by the algorithm, we know that

$$\lim_{k\to\infty} f(y_k) - f(y_{k+1}) = 0,$$

and we have reached a contradiction. Hence, our original assumption must be false, and the desired result is achieved. □

It is easily shown that the when $p = 1$, the premises of this theorem hold true: the generalized Rayleigh quotient is bounded below by λ_1, the norm of the model Hessian is bounded above by $(\lambda_n - \lambda_1)$, and the truncated conjugate gradient method (Algorithm 2) first produces the Cauchy point and then improves upon it. One may note that this proof of convergence is much simpler than in the case of the classical trust region method and the RTR method. This results from basing the trust-region on the model performance ratio ρ. While demonstrated here on the generalized Rayleigh quotient, this is not expected to be possible for all objective functions.

Regarding the block version of the algorithm described in Section 3.2, Theorem 4 applies individually to each of the decoupled iterations; therefore, collectively, they approach zero. Furthermore, in the presence of a subspace acceleration scheme, the norm of the residual (gradient) is expected to be lower than without the subspace acceleration. A formal proof of this assertion should follow easily.

Therefore, we have a theory proving global convergence to critical points for the IRTR applied to the computation of extreme eigenspaces of a symmetric matrix pencil. As with classical trust-region methods, the method has stable convergence only to local minimizers. Therefore, in the case of the generalized Rayleigh quotient (which has a unique local minimizer when $\lambda_p < \lambda_{p+1}$), we expect global convergence to the global minimizer. This performance is observed in the numerical experiments in Section 5.

5 Numerical Results

The IRTR method seeks to overcome the inefficiencies of the RTR method, such as the rejection of computed updates and the limitations due to the discrete nature of the trust-region radius. We compare the performance of the IRTR with that of the classical RTR. The following experiments were performed in MATLAB (R14) under Mac OSX.

Figure 1 illustrates the benefit of checking the outer stopping criterion from the inner iteration. The outer iterations are denoted on each line by tick marks. In the experiments, the last model minimization, while solving the eigenvalue problem to high accuracy, requires a significant amount of work to do so. By monitoring the outer convergence, the method stops the last inner iteration when it has reached the target accuracy, avoiding a significant number of inner iterations. In the simple test, with no subspace acceleration and no preconditioner, the RTR method without outer criterion monitoring terminates after 7225 matrix multiplications, compared to 6525 and 4545 (respectively) multiplications for the RTR and IRTR with outer criterion monitoring. Similarly, when using a preconditioner and subspace acceleration, the unmonitored RTR requires 180 multiplications, compared to 135 and 120 (respectively) for the monitored RTR and IRTR.

The first problem is a standard eigenvalue problem, where the matrix A is a Laplacian of dimension $n = 10000$. Here, we are seeking the leftmost $p = 5$

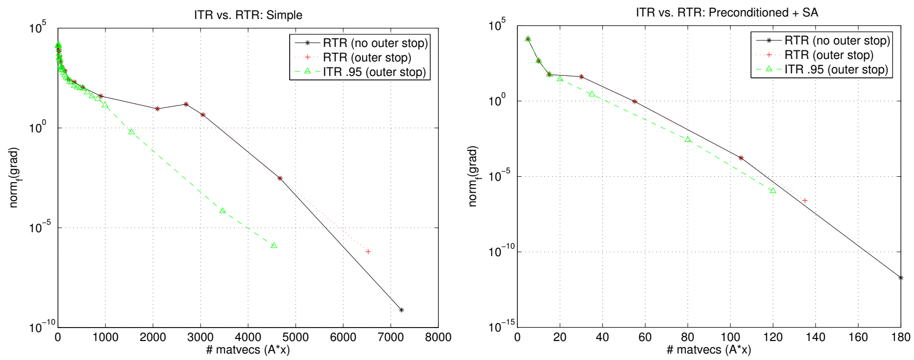

Fig. 1. Increased efficiency achievable by monitoring the outer stopping criterion.

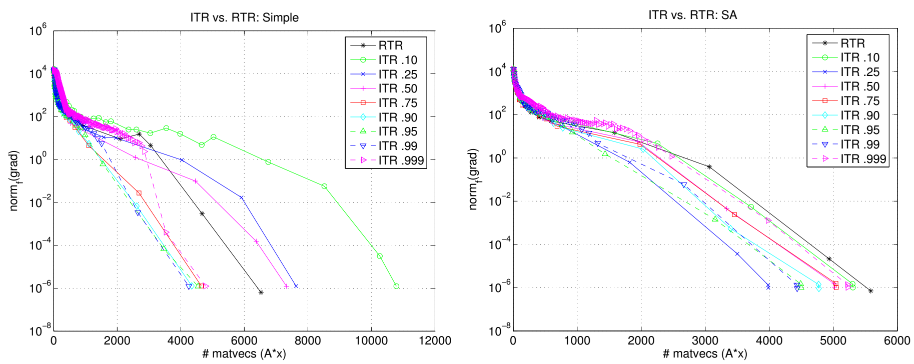

Fig. 2. Figures illustrating the efficiency of RTR vs. IRTR for different values of ρ', in the absence of a preconditioner.

eigenvalues of A. Four experiments are run: both with and without subspace acceleration, and with and without a preconditioned inner iteration. When in effect, the subspace acceleration strategy occurs over the 10-dimensional subspace colsp($[Y_k, S_k]$). The RTR is tested with a value of $\rho' = 0.1$, while the IRTR is run for multiple values of ρ' to illustrate the effect of this parameter on the efficiency of the method. The preconditioner is based on a complete factorization of A. The results are shown in Figures 2 and 3. These experiments demonstrate that in both of the above described scenarios, the IRTR method is able to achieve a greater efficiency than the RTR method.

Figure 4 considers a generalized eigenvalue problem with a preconditioned inner iteration. The matrices A and B are from the Harwell-Boeing collection BCSST24, where A is the stiffness matrix and B is the mass matrix from a structural engineering model of the Calgary Olympic Saddledome. The problem is of size $n = 3562$ and we are seeking the leftmost $p = 5$ eigenvalues of the matrix

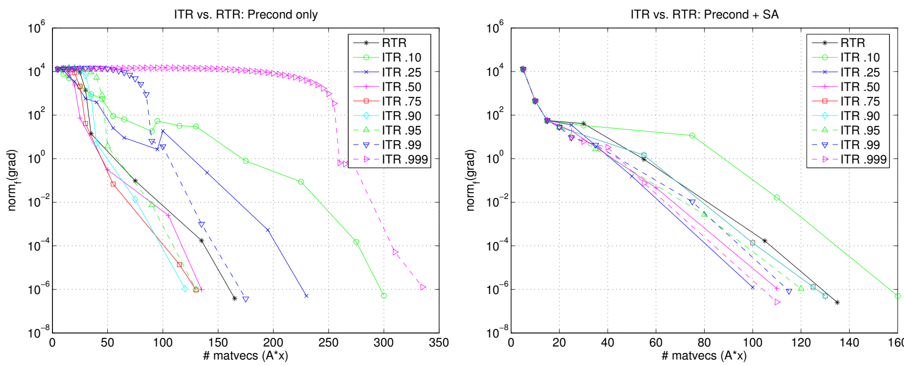

Fig. 3. Figures illustrating the efficiency of RTR vs. IRTR for different values of ρ', in the presence of a preconditioned inner iteration.

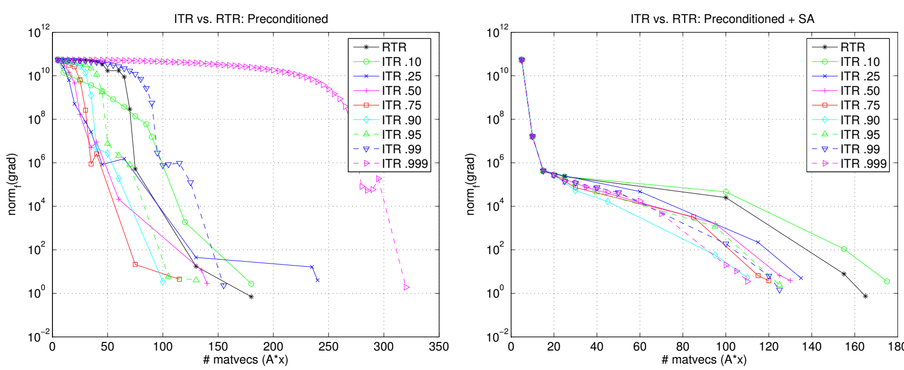

Fig. 4. Figures illustrating the efficiency of RTR vs. IRTR for different values of ρ', in the presence of a preconditioned inner iteration, for the BCSST24 data.

pencil (A, B). The inner iteration is preconditioned using an exact factorization of A. Two experiments are run: with and without subspace acceleration. The subspace acceleration mechanism is the same as in the previous set of experiments. The IRTR is demonstrated as before, with multiple values of ρ'. Again, it is shown that the IRTR method is able to outperform the RTR method.

6 Conclusion

This paper presents an optimization-based analysis of the symmetric, generalized eigenvalue problem which explores the relationship between the inner and outer iterations. The paper proposes the Implicit Riemannian Trust-Region method, which seeks to alleviate inefficiencies resulting from the inner/outer divide, while still preserving the strong convergence properties of the RTR method. This algorithm was shown in numerical experiments to be capable of greater efficiency than the RTR method.

Acknowledgments Useful discussions with Andreas Stathopoulos, Rich Lehoucq and Ulrich Hetmaniuk are gratefully acknowledged.

References

1. Stewart, G.W.: Matrix algorithms, Vol II: Eigensystems. Society for Industrial and Applied Mathematics, Philadelphia (2001)
2. Sameh, A.H., Wisniewski, J.A.: A trace minimization algorithm for the generalized eigenvalue problem. SIAM J. Numer. Anal. **19**(6) (1982) 1243–1259
3. Smith, S.T.: Optimization techniques on Riemannian manifolds. In: Hamiltonian and gradient flows, algorithms and control. Volume 3 of Fields Inst. Commun. Amer. Math. Soc., Providence, RI (1994) 113–136
4. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2) (1998) 303–353
5. Mongeau, M., Torki, M.: Computing eigenelements of real symmetric matrices via optimization. Comput. Optim. Appl. **29**(3) (2004) 263–287
6. Sameh, A., Tong, Z.: The trace minimization method for the symmetric generalized eigenvalue problem. J. Comput. Appl. Math. **123** (2000) 155–175
7. Knyazev, A.V.: Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method. SIAM J. Sci. Comput. **23**(2) (2001) 517–541
8. Absil, P.-A., Baker, C.G., Gallivan, K.A.: Trust-region methods on Riemannian manifolds with applications in numerical linear algebra. In: Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS2004), Leuven, Belgium, 5–9 July 2004. (2004)
9. Absil, P.-A., Baker, C.G., Gallivan, K.A.: A truncated-CG style method for symmetric generalized eigenvalue problems. to appear in Journal of Computational and Applied Mathematics (2006)
10. Moré, J.J., Sorensen, D.C.: Newton's method. In: Studies in numerical analysis. Volume 24 of MAA Stud. Math. Math. Assoc. America, Washington, DC (1984) 29–82
11. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer Series in Operations Research. Springer-Verlag, New York (1999)
12. Gould, N.I.M., Sainvitu, C., Toint, P.L.: A filter-trust-region method for unconstrained optimization. SIAM J. Optimization **16**(2) (2005) 314–357
13. Absil, P.-A., Baker, C.G., Gallivan, K.A., Sameh, A.: Adaptive model trust region methods for generalized eigenvalue problems. In Sunderam, V.S., van Albada, G.D., Peter M. A. Slot, e.a., eds.: International Conference on Computational Science. Volume 3514 of Lecture Notes in Compuer Science., Springer-Verlag (2005) 33–41
14. Absil, P.-A., Mahony, R., Sepulchre, R.: Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. Acta Appl. Math. **80**(2) (2004) 199–220
15. Absil, P.-A., Baker, C.G., Gallivan, K.A.: Trust-region methods on Riemannian manifolds. submitted (2005)

16. Notay, Y.: Combination of Jacobi-Davidson and conjugate gradients for the partial symmetric eigenproblem. Numer. Linear Algebra Appl. **9**(1) (2002) 21–44
17. Sleijpen, G.L.G., Van der Vorst, H.A.: A Jacobi-Davidson iteration method for linear eigenvalue problems. SIAM J. Matrix Anal. Appl. **17**(2) (1996) 401–425
18. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust-region methods. MPS/SIAM Series on Optimization. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, and Mathematical Programming Society (MPS), Philadelphia, PA (2000)