Model-based methods for computing extreme eigenpairs of definite pencils

Pierre-Antoine Absil

Christopher G. Baker

Kyle A. Gallivan

School of Computational Science,

Florida State University

FoCM conference, Santander, 7 July 2005

The Problem : Leftmost Eigenpair of Matrix Pencil

Given $n \times n$ matrix pencil (A, B), $A = A^T$, $B = B^T \succ 0$ with (unknown) eigen-decomposition

$$A\,[v_1|\ldots|v_n] = B\,[v_1|\ldots|v_n]\,\mathrm{diag}(\lambda_1, \ldots, \lambda_n)$$

$$[v_1|\ldots|v_n]^T\,B\,[v_1|\ldots|v_n] = I, \quad \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_n.$$

The problem is to compute the "leftmost" eigenvector $\pm v_1$.

Strategy: Optimization on Manifold

1. Rewrite the leftmost eigenpair problem as minimizing a cost function (the Rayleigh quotient) on a manifold.
2. Select a suitable method of optimization on manifolds.
3. Apply the method to the leftmost eigenpair problem.

Previous work

- Bradbury and Fletcher [BF66]: nonlinear CG on the projective space using particular coordinate systems.
- Helmke, Moore, Mahony [HM94, MHM96]: steepest descent on the sphere and on the projective space.
- Smith [Smi94], Edelman, Arias and Smith [EAS98]: Newton method and nonlinear CG on the sphere (yields single vector iteration) and on the Grassmann manifold (yields block iteration).

Strategy: Optimization on Manifold

1. Rewrite the leftmost eigenpair problem as minimizing a cost function (the Rayleigh quotient) on a manifold.
2. Select a suitable method of optimization on manifolds.
3. Apply the method to the leftmost eigenpair problem.

The ideal leftmost component algorithm

Given (A, B), $A = A^T$, $B = B^T \succ 0$ with (unknown) eigenvalues $0 < \lambda_1 \leq \ldots \lambda_n$ and associated eigenvectors $v_1, \ldots, v_n$.

1. Global convergence:
   - Convergence to some eigenvector for all initial conditions.
   - Stable convergence to the "leftmost" eigenvector $\pm v_1$ only.
2. Superlinear (cubic) local convergence to $\pm v_1$.
3. "Matrix-free" (no factorization of A, B) but possible use of preconditioner.
4. Minimal storage space required.

Strategy: Optimization on Manifold

1. Rewrite the leftmost eigenpair problem as minimizing a cost function (the Rayleigh quotient) on a manifold.
2. Select a suitable method of optimization on manifolds.
3. Apply the method to the leftmost eigenpair problem.

Strategy: Optimization on Manifold

1. Rewrite the leftmost eigenpair problem as minimizing a cost function (the Rayleigh quotient) on a manifold.
2. Define and analyze *trust-region methods* on manifolds.
3. Apply the method to the leftmost eigenpair problem.

Outline

- Rewrite computation of leftmost eigenpair as an optimization problem (on a manifold).
- Use a model-trust-region scheme to solve the problem.
  $\rightsquigarrow$ Global convergence.
- Take the exact quadratic model (at least, close to the solution).
  $\rightsquigarrow$ Superlinear convergence.
- Solve the trust-region subproblems using the (Steihaug-Toint) truncated CG (tCG) algorithm.
  $\rightsquigarrow$ "Matrix-free", preconditioned iteration.
  $\rightsquigarrow$ Minimal storage of iteration vectors.

The Problem : Leftmost Eigenpair of Matrix Pencil

Given $n \times n$ matrix pencil (A, B), $A = A^T$, $B = B^T \succ 0$ with (unknown) eigen-decomposition

$$A\,[v_1|\ldots|v_n] = B\,[v_1|\ldots|v_n]\,\mathrm{diag}(\lambda_1, \ldots, \lambda_n)$$

$$[v_1|\ldots|v_n]^T\,B\,[v_1|\ldots|v_n] = I, \quad \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_n.$$

The problem is to compute the “leftmost” eigenvector $\pm v_1$.

Leftmost eigendirection computation $\equiv$ optimization on $\mathbb{R}^n$

Cost function: Rayleigh quotient

$$f : \mathbb{R}^n_0 \to \mathbb{R} : y \mapsto \frac{y^T A y}{y^T B y}$$

Minimizers: $\{\alpha v_1 : \alpha \neq 0\}$.

Difficulty: the minimizers are not isolated.

(For example, Newton on f yields the iteration $x \mapsto 2x$!)

Leftmost eigenvector computation $\equiv$ optimization on manifold

Manifold: ellipsoid

$$\mathcal{M} = \{y \in \mathbb{R}^n : y^T B y = 1\}$$

Cost function:

$$f : \mathcal{M} \to \mathbb{R} : y \mapsto y^T A y$$

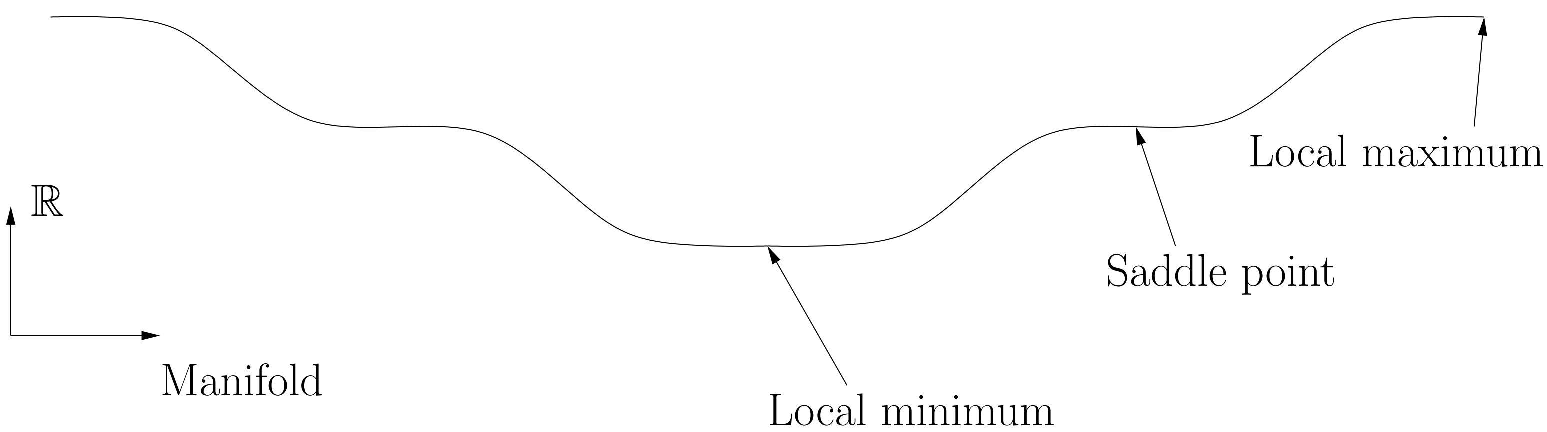
Properties:

- $\pm v_1, \ldots, \pm v_n$ are the stationary points of f.
- $\pm v_1$ are the local and global minimizers of f.

The shape of the cost function

Cost function:

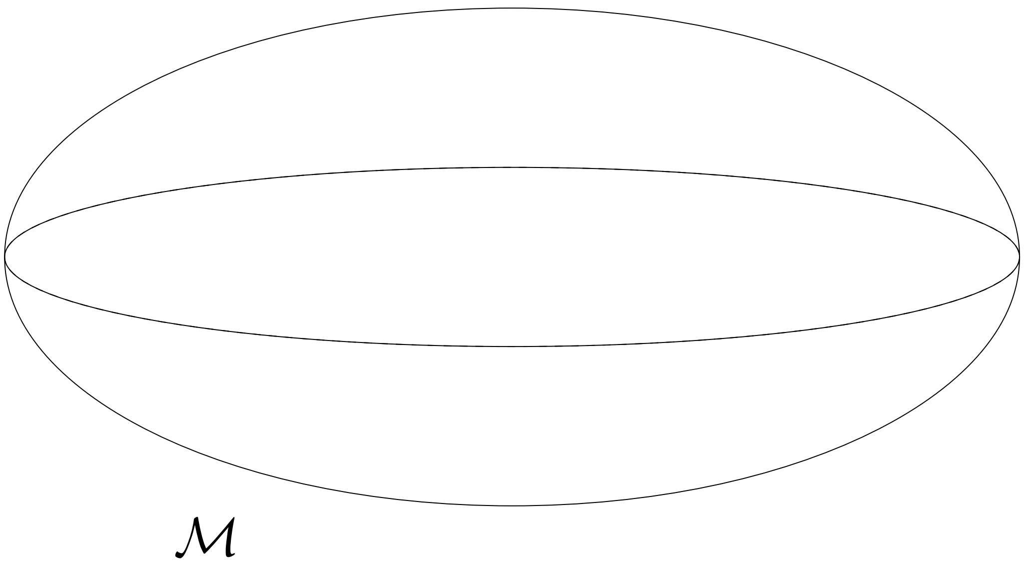
$$f : \mathcal{M} \to \mathbb{R} : y \mapsto y^T A y$$



The points $\pm v_1$, corresponding to the leftmost eigenvector, are the only minima. The other stationary points are the other eigenvectors.

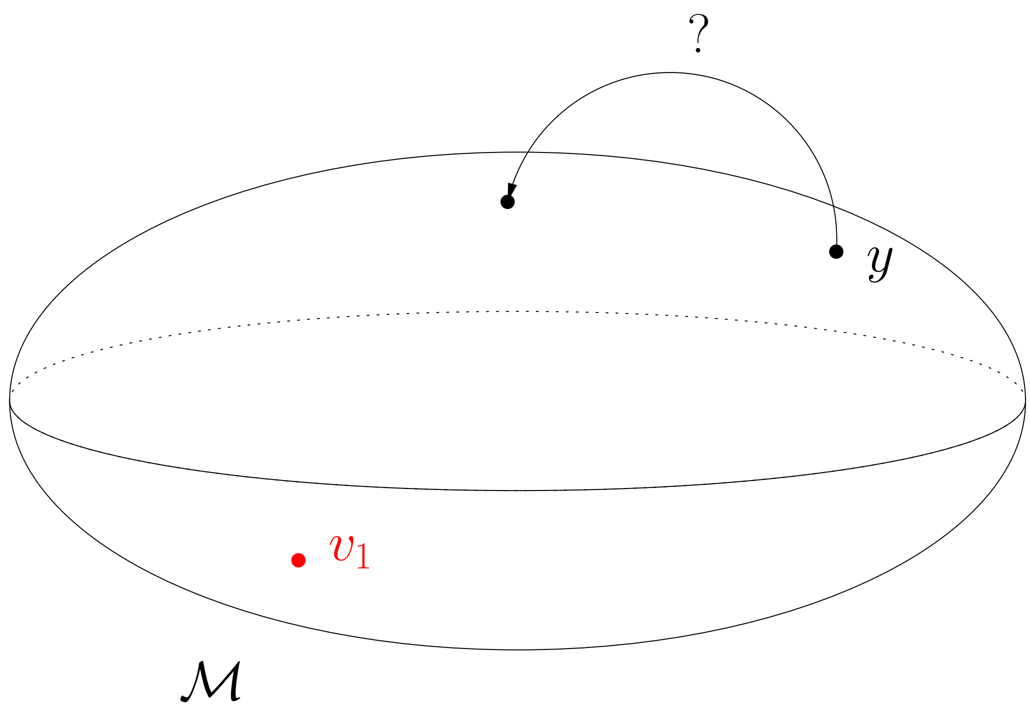
The Manifold: ellipsoid

Manifold: ellipsoid $\mathcal{M} = \{y \in \mathbb{R}^n : y^T B y = 1\}$



Cost function: $f : \mathcal{M} \to \mathbb{R} : y \mapsto y^T A y$

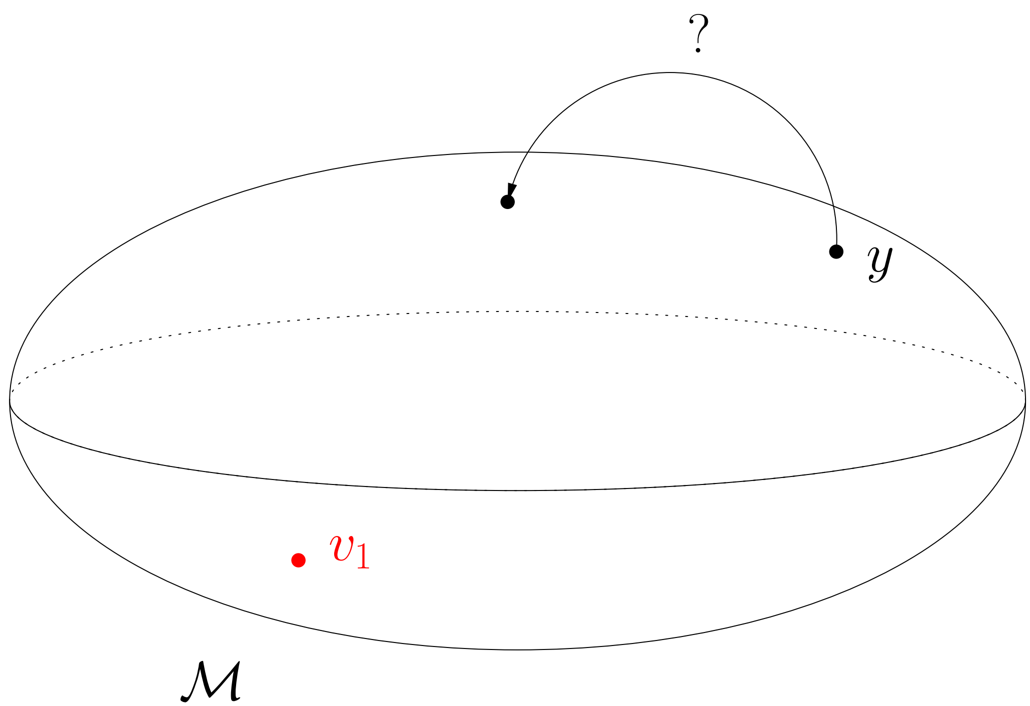
Iteration on the manifold



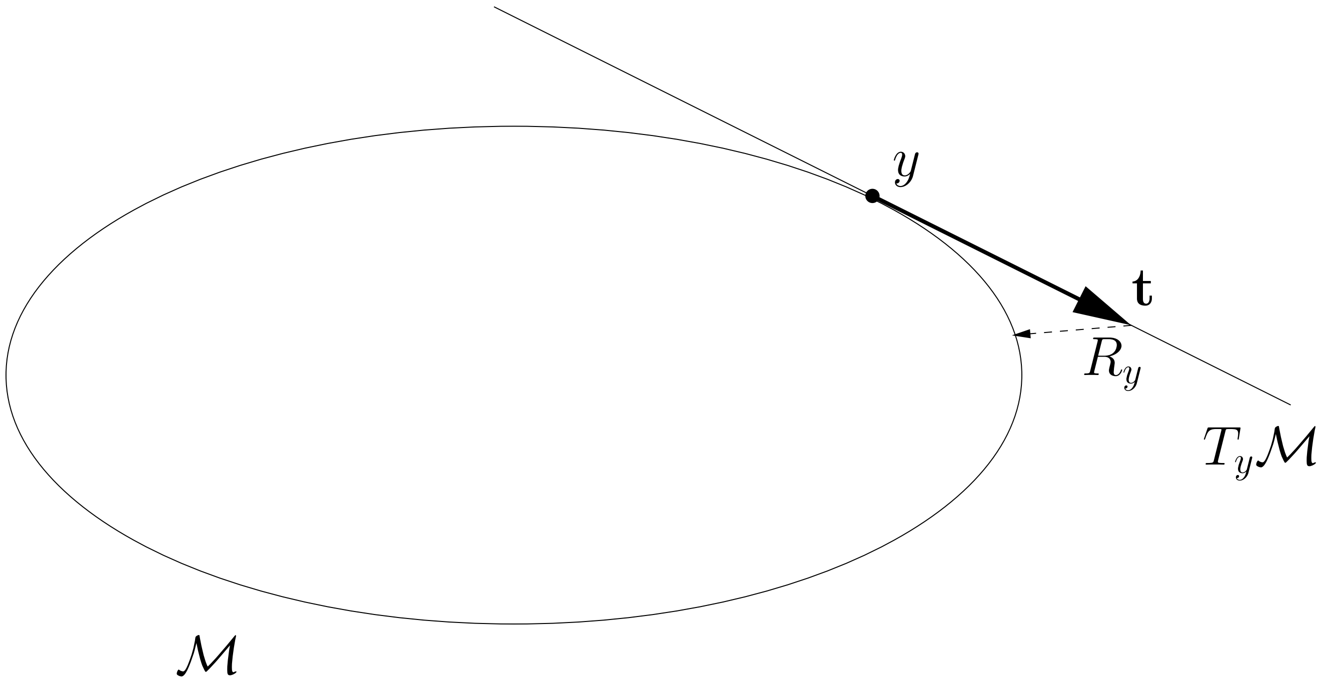
Outline

- Rewrite computation of leftmost eigenpair as an optimization problem (on a manifold).
- Use a model-trust-region scheme to solve the problem.
  $\rightsquigarrow$ Global convergence.
- Take the exact quadratic model (at least, close to the solution).
  $\rightsquigarrow$ Superlinear convergence.
- Solve the trust-region subproblems using the (Steihaug-Toint) truncated CG (tCG) algorithm.
  $\rightsquigarrow$ "Matrix-free", preconditioned iteration.
  $\rightsquigarrow$ Minimal storage of iteration vectors.

Iteration on the manifold



Tangent space and retraction (2D picture)

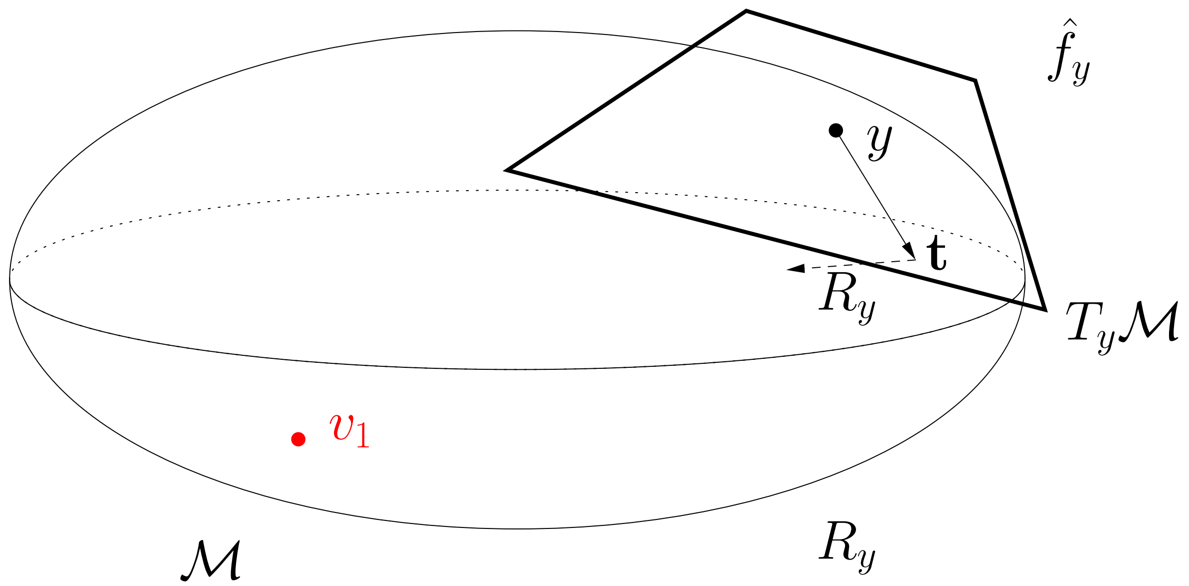


Tangent space: $T_y\mathcal{M} := \{\mathbf{t} \in \mathbb{R}^n : y^T B\mathbf{t} = 0\}$.

Retraction: $R_y\mathbf{t} := (y + \mathbf{t})/\|y + \mathbf{t}\|_B$.

Lifted cost function: $\hat{f}_y(\mathbf{t}) := f(R_y\mathbf{t}) = \frac{(y+\mathbf{t})^T A(y+\mathbf{t})}{(y+\mathbf{t})^T B(y+\mathbf{t})}$.

Tangent space and retraction



Tangent space: $T_y\mathcal{M} := \{\mathbf{t} \in \mathbb{R}^n : y^T B\mathbf{t} = 0\}$.

Retraction: $R_y\mathbf{t} := (y + \mathbf{t})/\|y + \mathbf{t}\|_B$.

Lifted cost function: $\hat{f}_y(\mathbf{t}) := f(R_y\mathbf{t}) = \frac{(y+\mathbf{t})^T A(y+\mathbf{t})}{(y+\mathbf{t})^T B(y+\mathbf{t})}$.

Outline

- Rewrite computation of leftmost eigenpair as an optimization problem (on a manifold).
- Use a model-trust-region scheme to solve the problem.
  $\rightsquigarrow$ Global convergence.
- Take the exact quadratic model (at least, close to the solution).
  $\rightsquigarrow$ Superlinear convergence.
- Solve the trust-region subproblems using the (Steihaug-Toint) truncated CG (tCG) algorithm.
  $\rightsquigarrow$ "Matrix-free", preconditioned iteration.
  $\rightsquigarrow$ Minimal storage of iteration vectors.

Quadratic model

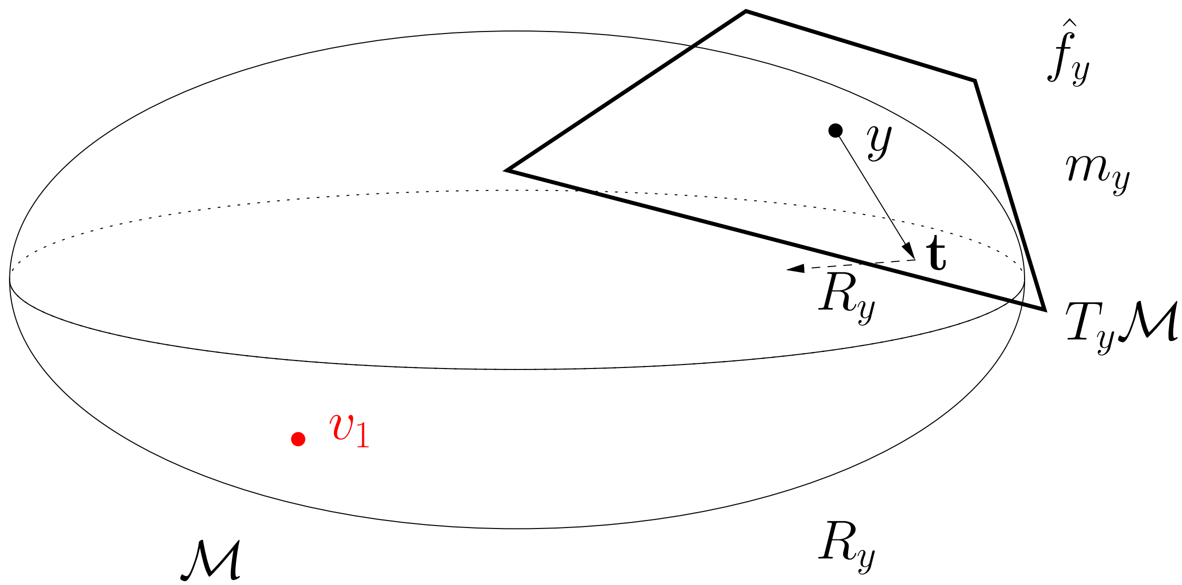
$$\hat{f}_y(\mathbf{t}) = \frac{y^TAy}{y^TBy} + 2\frac{y^TA\mathbf{t}}{y^TBy} + \frac{1}{y^TBy}\left(\mathbf{t}^TA\mathbf{t} - \frac{y^TAy}{y^TBy}\mathbf{t}^TB\mathbf{t}\right) + \dots$$
$$= f(y) + 2\langle PAy, \mathbf{t}\rangle + \frac{1}{2}\langle 2P(A - f(y)B)P\mathbf{t}, \mathbf{t}\rangle + \dots$$

where $\langle u, v\rangle = u^Tv$ and $P = I - By(y^TB^2y)^{-1}y^TB$.

Model:

$$m_y(\mathbf{t}) = f(y) + 2\langle PAy, \mathbf{t}\rangle + \frac{1}{2}\langle P(A - f(y)B)P\mathbf{t}, \mathbf{t}\rangle, \quad y^TB\mathbf{t} = 0.$$

Quadratic model



$$m_y(\mathbf{t}) = f(y) + 2\langle PAy, \mathbf{t}\rangle + \frac{1}{2}\langle P(A - f(y)B)P\mathbf{t}, \mathbf{t}\rangle, \quad y^T B\mathbf{t} = 0.$$

Model minimization

Model:

$$m_y(\mathbf{t}) = f(y) + 2\langle PAy, \mathbf{t}\rangle + \frac{1}{2}\langle P(A - f(y)B)P\mathbf{t}, \mathbf{t}\rangle, \quad y^T B\mathbf{t} = 0. \tag{1}$$

Newton method: Compute the stationary point of the model, i.e., solve

$$P(A - f(y)B)P\,\mathbf{t} = -PAy.$$

Instead, compute (approximately) the minimizer of m_y within a trust-region

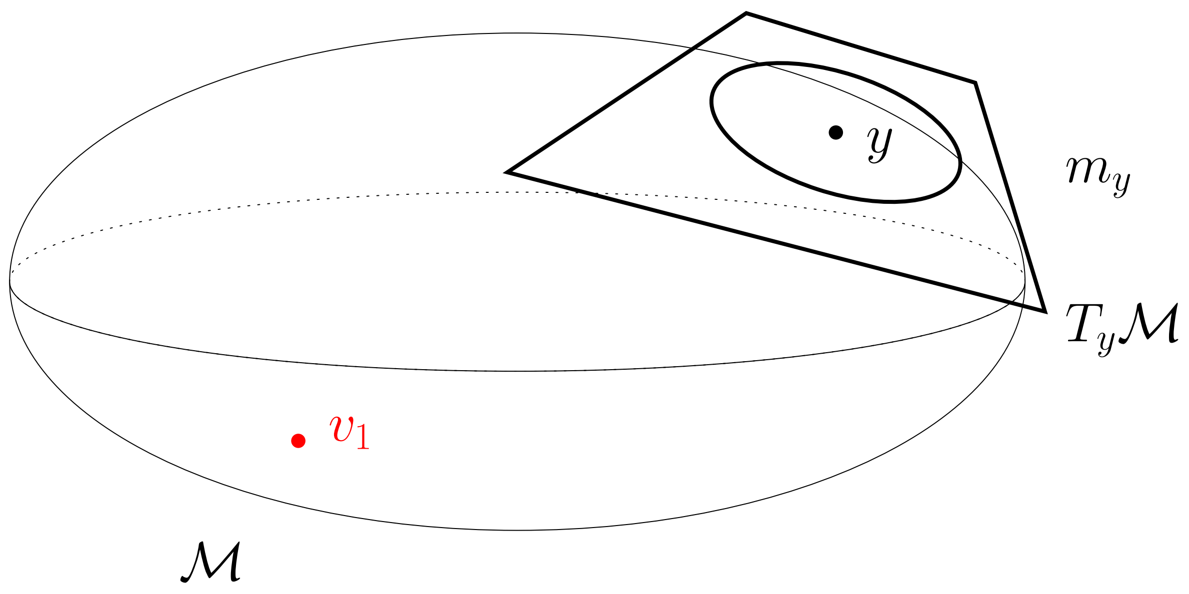
$$\{\mathbf{t} \in T_x\mathcal{M} : \mathbf{t}^T\mathbf{t} \le \Delta^2\}.$$

Trust-region subproblem

Minimize

$$m_y(\mathbf{t}) = f(y) + 2\langle PAy, \mathbf{t}\rangle + \frac{1}{2}\langle P(A - f(y)B)P\mathbf{t}, \mathbf{t}\rangle, \quad y^T B\mathbf{t} = 0.$$

subject to $\mathbf{t}^T\mathbf{t} \leq \Delta^2$.



Outline

- Rewrite computation of leftmost eigenpair as an optimization problem (on a manifold).
- Use a model-trust-region scheme to solve the problem.
  $\rightsquigarrow$ Global convergence.
- Take the exact quadratic model (at least, close to the solution).
  $\rightsquigarrow$ Superlinear convergence.
- Solve the trust-region subproblems using the (Steihaug-Toint) truncated CG (tCG) algorithm.
  $\rightsquigarrow$ "Matrix-free", preconditioned iteration.
  $\rightsquigarrow$ Minimal storage of iteration vectors.

Truncated CG method for the TR subproblem (1)

Let $\langle \cdot, \cdot \rangle$ denote the standard inner product and let $\mathcal{H}_{x_k} := P(A - f(x_k)B)P$ denote the Hessian operator.

Initializations:

Set $\mathbf{t}_0 = 0$, $r_0 = P_{x_k} A x_k = A x_k - B x_k (x_k^T B^2 x_k)^{-1} x_k^T B A x_k$, $\delta_0 = -r_0$;

Then repeat the following loop on j:

Check for negative curvature

if $\langle \delta_j, \mathcal{H}_{x_k} \delta_j \rangle \leq 0$

Compute τ such that $\mathbf{t} = \mathbf{t}_j + \tau \delta_j$ minimizes $m(\mathbf{t})$ in (1) and satisfies $\|\mathbf{t}\| = \Delta$;

return $\mathbf{t}$;

Truncated CG method for the TR subproblem (2)

Generate next inner iterate

Set $\alpha_j = \langle r_j, r_j \rangle / \langle \delta_j, \mathcal{H}_{x_k} \delta_j \rangle$;

Set $\mathbf{t}_{j+1} = \mathbf{t}_j + \alpha_j \delta_j$;

Check trust-region

if $\|\mathbf{t}_{j+1}\| \geq \Delta$

Compute $\tau \geq 0$ such that $\mathbf{t} = \mathbf{t}_j + \tau \delta_j$ satisfies $\|\mathbf{t}\| = \Delta$;

return $\mathbf{t}$;

Truncated CG method for the TR subproblem (3)

Update residual and search direction

Set $r_{j+1} = r_j + \alpha_j \mathcal{H}_{x_k} \delta_j$;

Set $\beta_{j+1} = \langle r_{j+1}, r_{j+1} \rangle / \langle r_j, r_j \rangle$;

Set $\delta_{j+1} = -r_{j+1} + \beta_{j+1} \delta_j$;

$j \leftarrow j + 1$;

Check residual

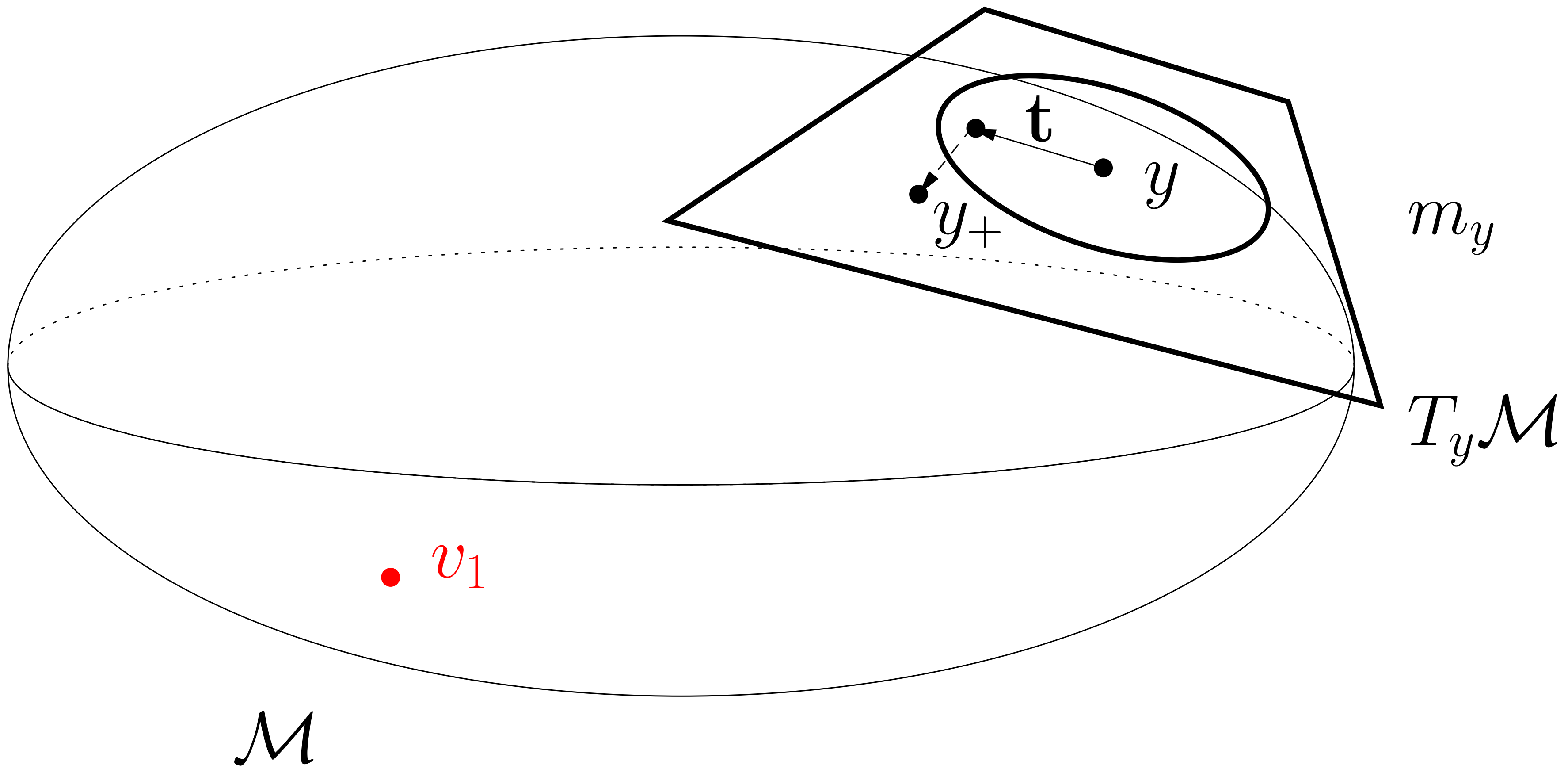
If

$$\|r_j\| \leq \|r_0\| \min \left(\|r_0\|^{\theta}, \kappa \right) \tag{2}$$

for some prescribed θ and κ

return $\mathbf{t}_j$;

Overall iteration



The outer iteration – manifold trust-region (1)

Data: symmetric $n \times n$ matrices A and B, with B positive definite.

Parameters: $\bar{\Delta} > 0$, $\Delta_0 \in (0, \bar{\Delta})$, and $\rho' \in (0, \frac{1}{4})$.

Input: initial iterate $x_0 \in \{y : y^T B y = 1\}$.

Output: sequence of iterates $\{x_k\}$ in $\{y : y^T B y = 1\}$.

Initialization: $k = 0$

Repeat the following:

The outer iteration – manifold trust-region (2)

- Obtain $\mathbf{t}_k$ using the Steihaug-Toint truncated conjugate-gradient method to approximately solve the trust-region subproblem

$$\min_{x_k^T B \mathbf{t}=0} m_{x_k}(\mathbf{t}) \quad \text{s.t. } \|\mathbf{t}\| \leq \Delta_k, \tag{3}$$

where m is defined in (1).

The outer iteration – manifold trust-region (3)

- Evaluate

$$\rho_k = \frac{\hat{f}_{x_k}(0) - \hat{f}_{x_k}(\mathbf{t}_k)}{m_{x_k}(0) - m_{x_k}(\mathbf{t}_k)} \tag{4}$$

where $\hat{f}_{x_k}(\mathbf{t}) = \frac{(x_k+\mathbf{t})^T A(x_k+\mathbf{t})}{(x_k+\mathbf{t})^T B(x_k+\mathbf{t})}$.

- Update the trust-region radius:

if $\rho_k < \frac{1}{4}$

$\Delta_{k+1} = \frac{1}{4}\Delta_k$

else if $\rho_k > \frac{3}{4}$ **and** $\|\mathbf{t}_k\| = \Delta_k$

$\Delta_{k+1} = \min(2\Delta_k, \bar{\Delta})$

else

$\Delta_{k+1} = \Delta_k$;

The outer iteration – manifold trust-region (4)

- Update the iterate:

  if $\rho_k > \rho'$

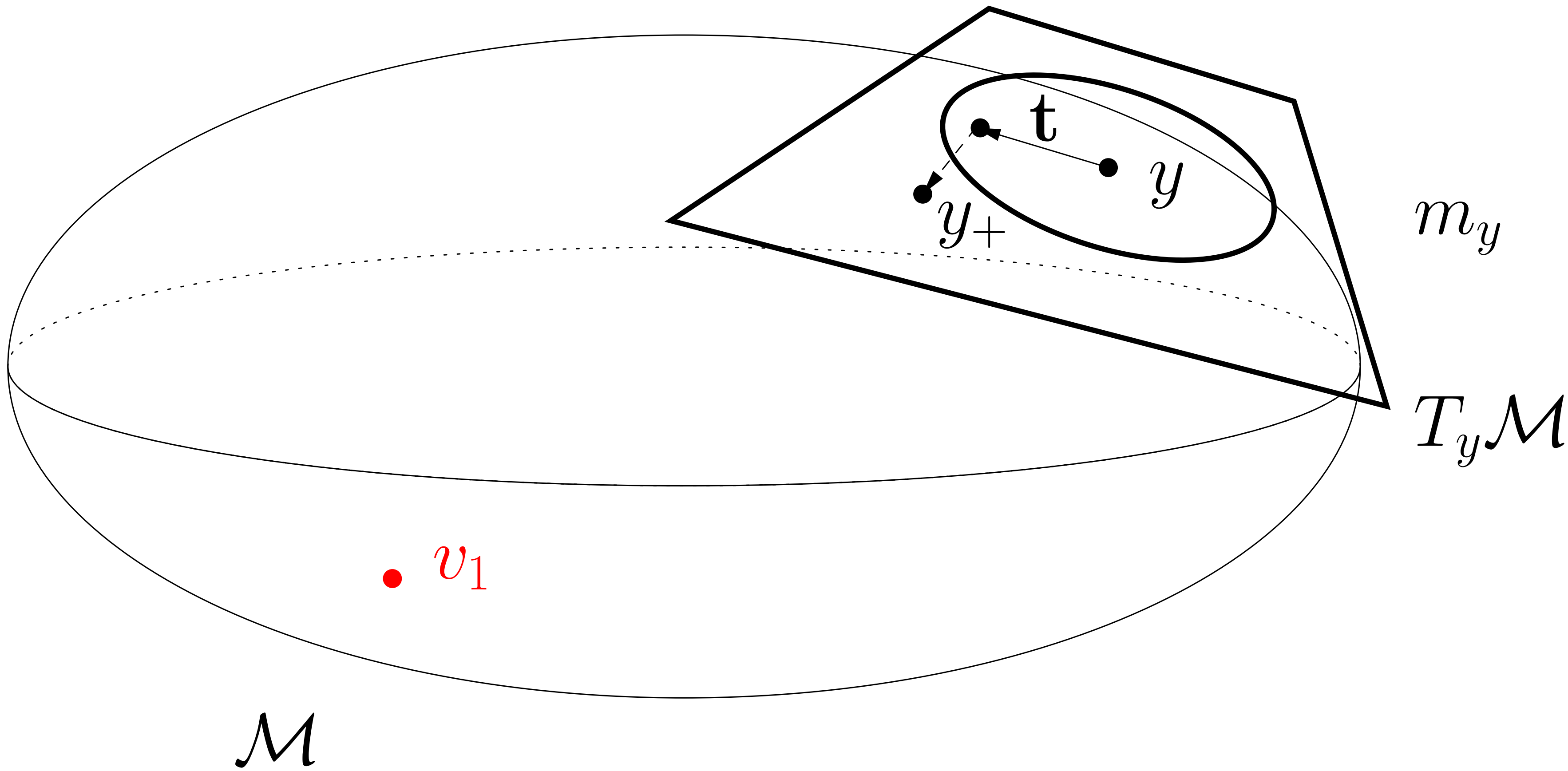
$$x_{k+1} = (x_k + \mathbf{t}_k)/\|x_k + \mathbf{t}_k\|_B; \tag{5}$$

  else

  $x_{k+1} = x_k;$

  $k \leftarrow k + 1$

Convergence



Convergence

Theorem:

Let (A, B) be an $n \times n$ symmetric/positive-definite matrix pencil with eigenvalues $\lambda_1 < \lambda_2 \leq \ldots \leq \lambda_{n-1} \leq \lambda_n$ and an associated B-orthonormal basis of eigenvectors $(v_1, \ldots, v_n)$.

Let $\mathcal{S}_i = \{y : Ay = \lambda_i By,\ y^T By = 1\}$ denote the intersection of the eigenspace of (A, B) associated to λ_i with the set $\{y : y^T By = 1\}$.

...

Convergence (global)

(i) Let $\{x_k\}$ be a sequence of iterates generated by the Algorithm. Then $\{x_k\}$ converges to the eigenspace of (A, B) associated to one of its eigenvalues. That is, there exists i such that $\lim_{k\to\infty} \mathrm{dist}(x_k, \mathcal{S}_i) = 0$.

(ii) Only the set $\mathcal{S}_1 = \{\pm v_1\}$ is stable.

Convergence (local)

(iii) There exists $c > 0$ such that, for all sequences $\{x_k\}$ generated by the Algorithm converging to $\mathcal{S}_1$, there exists $K > 0$ such that for all $k > K$,

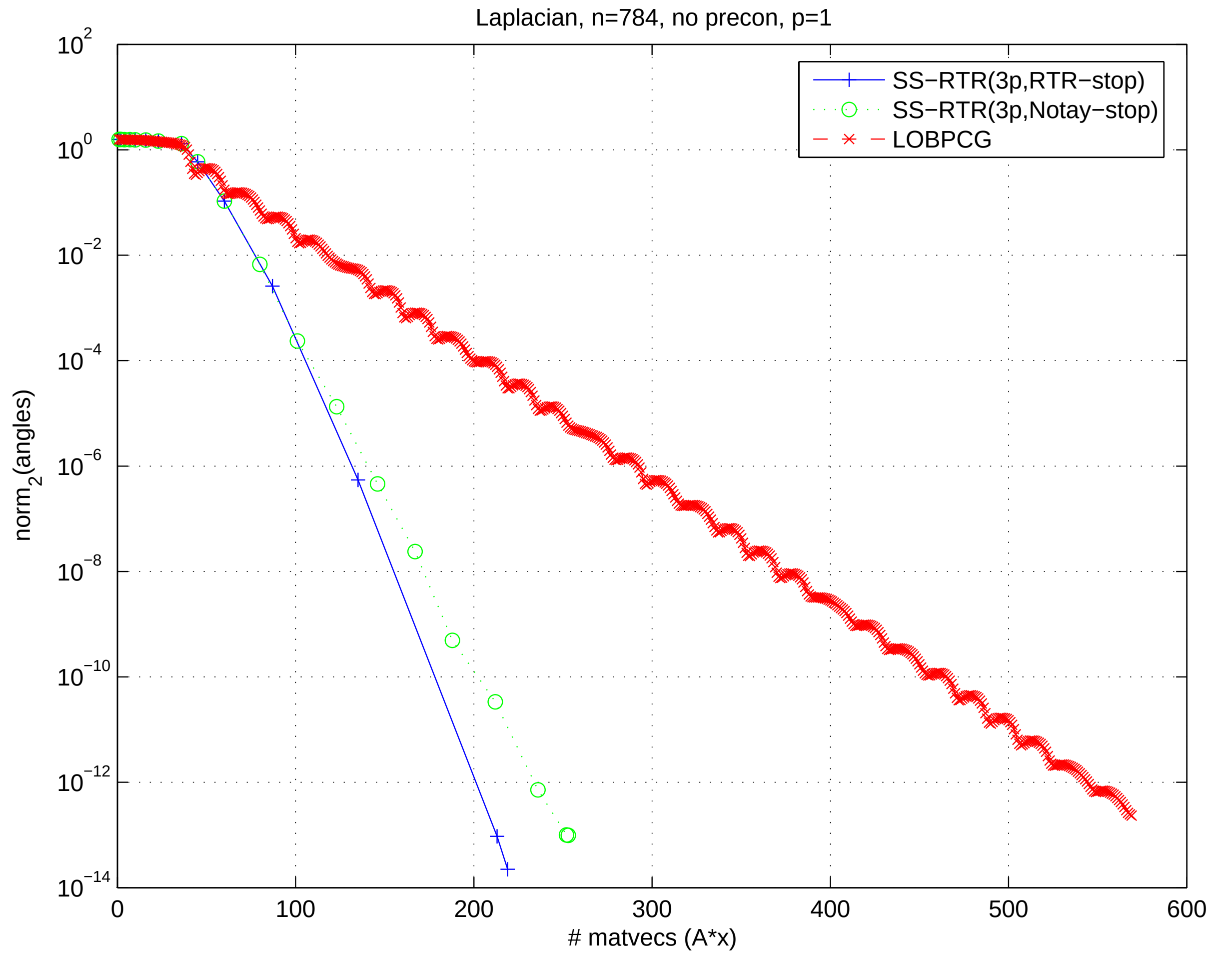
$$\mathrm{dist}(x_{k+1}, \mathcal{S}_1) \leq c\,(\mathrm{dist}(x_k, \mathcal{S}_1))^{\min\{\theta+1,2\}} \tag{6}$$

with $\theta > 0$.

Outline

- Rewrite computation of leftmost eigenpair as an optimization problem (on a manifold).
- Use a model-trust-region scheme to solve the problem.
  $\rightsquigarrow$ Global convergence.
- Take the exact quadratic model (at least, close to the solution).
  $\rightsquigarrow$ Superlinear convergence.
- Solve the trust-region subproblems using the (Steihaug-Toint) truncated CG (tCG) algorithm.
  $\rightsquigarrow$ "Matrix-free", preconditioned iteration.
  $\rightsquigarrow$ Minimal storage of iteration vectors.

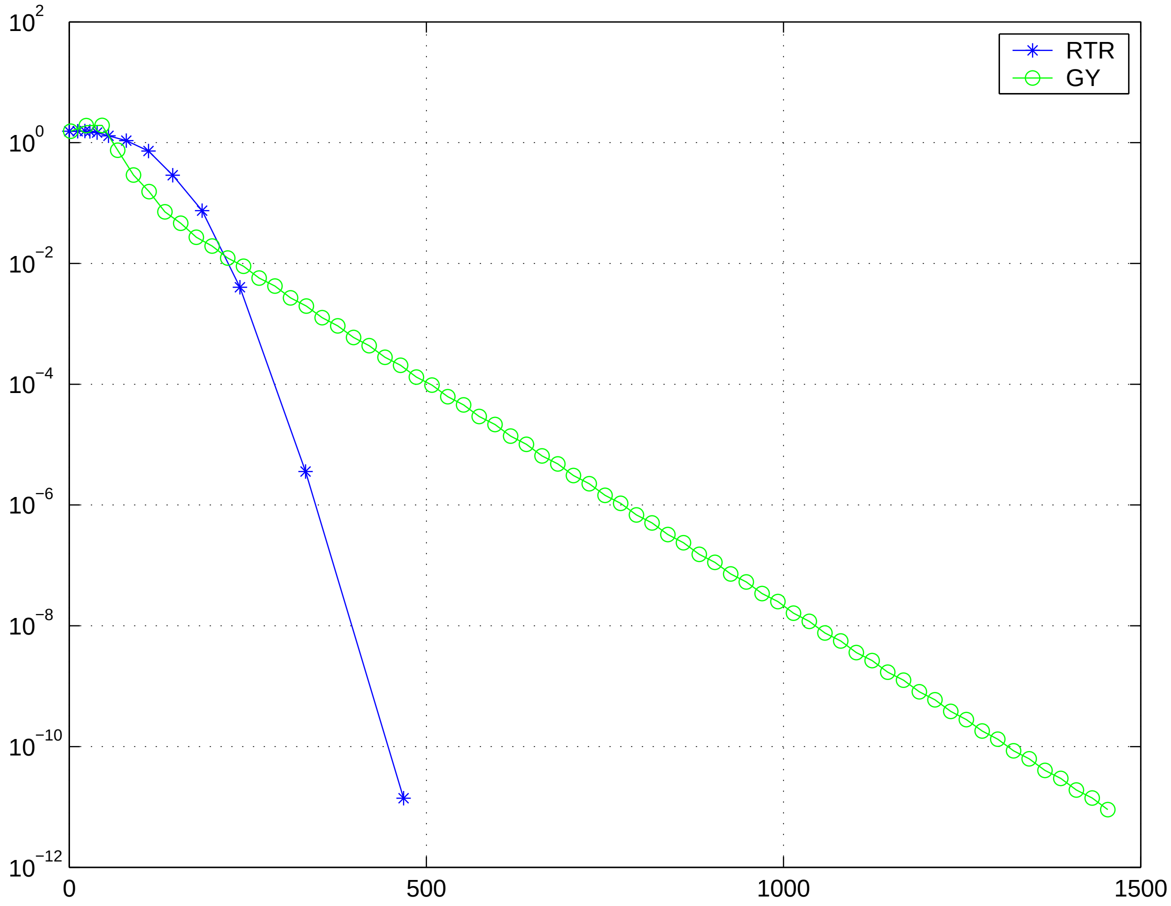
Numerical experiments: RTR vs LOBPCG



Distance to target versus matrix-vector multiplications.

Symmetric/positive-definite generalized eigenvalue problem.

Numerical experiments: RTR vs Krylov [GY02]



Distance to target versus matrix-vector multiplications.

Symmetric/positive-definite generalized eigenvalue problem.

Key points

1. Minimize the Rayleigh quotient on a manifold in order to have isolated minima.

2. Do not try to solve Jacobi correction equations. Instead, (approximately) minimize a model within a trust-region.

3. To obtain superlinear convergence, use an appropriate stopping criterion (2) in the inner iteration.

Current and future work

- When approximately minimizing the model, monitor the eigenproblem residual and the cost function.
- Use subspace acceleration to improve efficiency.

Conclusions

- "Brute-force" application of the Riemannian Trust-Region scheme [ABG04b, ABG05] to the extreme symmetric generalized eigenproblem.
- Local and global convergence results inherited from the Riemannian scheme.
- Promising numerical results.
- Several enhancements are possible...

For more details on the topic of this talk, see [ABG04a].

References

[ABG04a] P.-A. Absil, C. G. Baker, and K. A. Gallivan, *A truncated-CG style method for symmetric generalized eigenvalue problems*, submitted, 2004.

[ABG04b] ———, *Trust-region methods on Riemannian manifolds with applications in numerical linear algebra*, Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS2004), Leuven, Belgium, 5–9 July 2004, 2004.

[ABG05] ———, *Trust-region methods on Riemannian manifolds*, submitted, March 2005.

[BF66] W. W. Bradbury and R. Fletcher, *New iterative methods for solution of the eigenproblem*, Numer. Math. **9** (1966), 259–267.

[EAS98] A. Edelman, T. A. Arias, and S. T. Smith, *The geometry of* *algorithms with orthogonality constraints*, SIAM J. Matrix Anal. Appl. **20** (1998), no. 2, 303–353.

[GY02] G. H. Golub and Q. Ye, *An inverse free preconditioned Krylov subspace method for symmetric generalized eigenvalue problems*, SIAM J. Sci. Comput. **24** (2002), no. 1, 312–334 (electronic).

[HM94] U. Helmke and J. B. Moore, *Optimization and dynamical systems*, Springer, 1994.

[MHM96] R. E. Mahony, U. Helmke, and J. B. Moore, *Gradient algorithms for principal component analysis*, J. Austral. Math. Soc. Ser. B **37** (1996), 430–450.

[Smi94] Steven T. Smith, *Optimization techniques on Riemannian manifolds*, Hamiltonian and gradient flows, algorithms and control, Fields Inst. Commun., vol. 3, Amer. Math. Soc., Providence, RI, 1994, pp. 113–136. MR MR1297990 (95g:58062)