Variationnal Data assimilation, Earth, Atmosphere, Ocean

Serge Gratton
University of Toulouse
CERFACS-IRIT, Toulouse, France

Joint work with S. Gurol, P. Jiranek, D. Titley-Peloquin, Ph.L. Toint, J. Tshimanga (CERFACS-IRIT)
O. Titaud, I. Mirouze, A. Weaver (CERFACS)
L. Berre, G. Desroziers, H. Varella (Météo-France)
R. Biancale, L. Seoane, W. Zerhouni (CNES)

ADTAO seminar, Toulouse 2013

The ADTAO project

- Is funded by RTRA-STAE
- Is a 4 year project
- Involves international experts : I.S. Duff, J.Mandel, A. Moore, Ph.L. Toint, L.N. Vicente,
- Recruited 6 postdocs : J. Tshimanga, L. Seoane, W. Zerhouni, H. Varella, P. Jiranek, D. Titley-Peloquin
- Is a collaboration between 5 entities : CERFACS, CNES, IRIT, Mteo-France, Obervatoire Midi-Pyrnes
- Produced 25 journal papers, 2 international conferences, operational softwares for NEMOVAR, GINS, ROMS, Arpège-IFS

- Introduction to data assimilation
- Dual iterative solvers
- Summary and ongoing related work

- Introduction to data assimilation
- Dual iterative solvers
- Summary and ongoing related work

A dynamical integration model predicts the state of the system given the state at an earlier time.

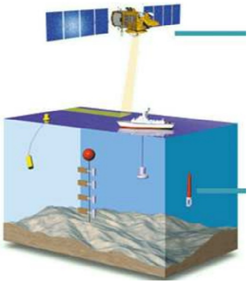
$\longrightarrow$ integrating may lead to very large prediction errors
(inexact physics, discretization errors, approximated parameters)

A dynamical integration model predicts the state of the system given the state at an earlier time.

⟶ integrating may lead to very large prediction errors
(inexact physics, discretization errors, approximated parameters)

Observational data are used to improve accuracy of the forecasts.

⟶ but the data are inaccurate (measurement noise, under-sampling)



Solve a large-scale non-linear weighted least-squares problem:

$$\min_{x\in\mathbb{R}^n} \ \frac{1}{2}\|x - x_b\|^2_{B^{-1}} + \frac{1}{2}\sum_{j=0}^{N} \left\|\mathcal{H}_j(\mathcal{M}_j(x)) - y_j\right\|^2_{R_j^{-1}}$$

where

- $x \equiv x(t_0)$ is the control variable
- $\mathcal{M}_j$ are model operators: $x(t_j) = \mathcal{M}_j(x(t_0))$
- $\mathcal{H}_j$ are observation operators: $y_j \approx \mathcal{H}_j(x(t_j))$
- the obervations y_j and the background x_b are noisy
- B and R_j are covariance matrices

Project desciption:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|x - x_b\|^2_{B^{-1}} + \frac{1}{2}\sum_{j=0}^{N}\left\|\mathcal{H}_j(\mathcal{M}_j(x)) - y_j\right\|^2_{R_j^{-1}}$$

where

- Improving B : CERFACS and CNES, using diffusion operator and ensemble techniques (Talks of L. Berre, A. Weaver)
- Optimization algorithms : CERFACS, IRIT, CNES dual algorithms and Ensemble Kalman filters (Talks of Ph. Toint, L. Vicente)
- Modelling : CNES

Solve a large-scale non-linear weighted least-squares problem:

$$\min_{x \in \mathbb{R}^n} \; \frac{1}{2}\|x - x_b\|^2_{B^{-1}} + \frac{1}{2}\sum_{j=0}^{N} \left\|\mathcal{H}_j(\mathcal{M}_j(x)) - y_j\right\|^2_{R_j^{-1}}$$

Typically solved using a truncated Gauss-Newton algorithm (known as 4D-Var in the DA community).

Solve a large-scale non-linear weighted least-squares problem:

$$\min_{x\in\mathbb{R}^n} \ \frac{1}{2}\|x - x_b\|^2_{B^{-1}} + \frac{1}{2}\sum_{j=0}^{N} \left\|\mathcal{H}_j(\mathcal{M}_j(x)) - y_j\right\|^2_{R_j^{-1}}$$

Typically solved using a truncated Gauss-Newton algorithm (known as 4D-Var in the DA community).

$\longrightarrow$ linearize $\mathcal{H}_j(\mathcal{M}_j(x^{(k)} + \delta x^{(k)})) \approx \mathcal{H}_j(\mathcal{M}_j(x^{(k)})) + H_j^{(k)}\delta x^{(k)}$

$\longrightarrow$ solve the linearized subproblem

$$\min_{\delta x^{(k)}\in\mathbb{R}^n} \ \frac{1}{2}\|\delta x^{(k)} - (x_b - x^{(k)})\|^2_{B^{-1}} + \frac{1}{2}\left\|H^{(k)}\delta x^{(k)} - d^{(k)}\right\|^2_{R^{-1}}$$

$\longrightarrow$ update $x^{(k+1)} = x^{(k)} + \delta x^{(k)}$

- Introduction to data assimilation
- Dual iterative solvers
- Summary and ongoing related work

Exploiting the structure: Dual Approach

- The **exact solution** can be rewritten from duality theory or using the Sherman-Morrison-Woodbury formula

$$x_b - x_k + BH_k^T \underbrace{(H_k B H_k^T + R)^{-1}(d_k - H_k(x_b - x_k))}_{\text{Lagrange mult. : requires solving a linear system } \textbf{iteratively} \text{ in } \mathbb{R}^m}$$

Exploiting the structure: Dual Approach

- The **exact solution** can be rewritten from duality theory or using the Sherman-Morrison-Woodbury formula

$$x_b - x_k + BH_k^T \qquad \underbrace{(H_k B H_k^T + R)^{-1}(d_k - H_k(x_b - x_k))}_{\text{Lagrange mult. : requires solving a linear system \textbf{iteratively} in } \mathbb{R}^m}$$

- If $m << n$, then performing the minimization in $\mathbb{R}^m$ can **reduce memory and computational cost**.

Exploiting the structure: Dual Approach

- The **exact solution** can be rewritten from duality theory or using the Sherman-Morrison-Woodbury formula

$$x_b - x_k + BH_k^T \qquad \underbrace{(H_k B H_k^T + R)^{-1}(d_k - H_k(x_b - x_k))}_{\text{Lagrange mult. : requires solving a linear system } \textbf{iteratively} \text{ in } \mathbb{R}^m}$$

- If $m << n$, then performing the minimization in $\mathbb{R}^m$ can **reduce memory and computational cost**.

Preconditioned CG algorithm

Initialization

- $r_0 = A\delta x_0 - b,\ z_0 = Fr_0,\ p_0 = z_0$

For $i = 0, 1, \ldots$

1. $q_i = (B^{-1} + H^T R^{-1} H) p_i$
2. $\alpha_i = < r_i, z_i > / < q_i, p_i >$ Compute the step-length
3. $\delta x_{i+1} = \delta x_i + \alpha_i p_i$ Update the iterate
4. $r_{i+1} = r_i - \alpha_i q_i$; Update the residual
5. $r_{i+1} = r_{i+1} - RZ^T r_{i+1}$ Re-orthogonalization
6. $z_{i+1} = F\ r_{i+1}$ Update the preconditioned residual
7. $\beta_i = < r_{i+1}, z_{i+1} > / < r_i, z_i >$ Ensure A-conjugate directions
8. $R = [R, r/\beta_i]$ Re-orthogonalization
9. $Z = [Z, z/\beta_i]$ Re-orthogonalization
10. $p_{i+1} = z_{i+1} + \beta_i p_i$ Update the descent direction

Theorem

Suppose that

1. $\mathbf{BH}^{\mathrm{T}}\mathbf{G} = \mathbf{FH}^{\mathrm{T}}$.
2. $\mathbf{v}_0 = \mathbf{x}^b - \mathbf{x}_0$.

$\rightarrow$ *vectors* $\widehat{\mathbf{r}}_i$, $\widehat{\mathbf{p}}_i$, $\widehat{\mathbf{v}}_i$, $\widehat{\mathbf{z}}_i$ *and* $\widehat{\mathbf{q}}_i$ *such that*

$$\begin{aligned} \mathbf{r}_i &= \mathbf{H}^{\mathrm{T}}\widehat{\mathbf{r}}_i, \\ \mathbf{p}_i &= \mathbf{BH}^{\mathrm{T}}\widehat{\mathbf{p}}_i, \\ \mathbf{v}_i &= \mathbf{v}_0 + \mathbf{BH}^{\mathrm{T}}\widehat{\mathbf{v}}_i, \\ \mathbf{z}_i &= \mathbf{BH}^{\mathrm{T}}\widehat{\mathbf{z}}_i, \\ \mathbf{q}_i &= \mathbf{H}^{\mathrm{T}}\widehat{\mathbf{q}}_i \end{aligned}$$

Initialization steps

given $\mathbf{v}_0$; $\mathbf{r}_0 = (\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H} + \mathbf{B}^{-1})\mathbf{v}_0 - \mathbf{b}$, ...

Loop: WHILE

1. $\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{q}}_{i-1} = \mathbf{H}^{\mathrm{T}}(\mathbf{R}^{-1}\mathbf{H}\mathbf{B}^{-1}\mathbf{H}^{\mathrm{T}} + \mathbf{I}_m)\widehat{\mathbf{p}}_{i-1}$
2. $\alpha_{i-1} = \mathbf{r}_{i-1}^{\mathrm{T}}\mathbf{z}_{i-1} \,/\, \widehat{\mathbf{q}}_{i-1}^{\mathrm{T}}\widehat{\mathbf{p}}_{i-1}$
3. $\mathbf{B}\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{v}}_i = \mathbf{B}\mathbf{H}^{\mathrm{T}}(\mathbf{v}_{i-1} + \alpha_{i-1}\widehat{\mathbf{p}}_{i-1})$
4. $\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{r}}_i = \mathbf{H}^{\mathrm{T}}(\mathbf{r}_{i-1} + \alpha_{i-1}\widehat{\mathbf{q}}_{i-1})$
5. $\mathbf{B}\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{z}}_i = \mathbf{F}\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{r}}_i = \mathbf{B}\mathbf{H}^{\mathrm{T}}\mathbf{G}\widehat{\mathbf{r}}_i$ $\quad FH^T = BH^T G$
6. $\beta_i = (\mathbf{r}_i^{\mathrm{T}}\mathbf{z}_i \,/\, \mathbf{r}_{i-1}^{\mathrm{T}}\mathbf{z}_{i-1})$
7. $\mathbf{B}\mathbf{H}^{\mathrm{T}}\widehat{\mathbf{p}}_i = \mathbf{B}\mathbf{H}^{\mathrm{T}}(-\widehat{\mathbf{z}}_i + \beta_i\widehat{\mathbf{p}}_{i-1})$

Initialization

$\lambda_0 = 0$, $\widehat{r}_0 = R^{-1}(d - H(x_b - x))$,
$\widehat{z}_0 = G\widehat{r}_0$, $\widehat{p}_1 = \widehat{z}_0$, $k = 1$

Loop on k

1. $\widehat{q}_i = \widehat{A}\widehat{p}_i$
2. $\alpha_i = < \widehat{r}_{i-1}, \widehat{z}_{i-1} >_C / < \widehat{q}_i, \widehat{p}_i >_C$
3. $\lambda_i = \lambda_{i-1} + \alpha_i \widehat{p}_i$
4. $\widehat{r}_i = \widehat{r}_{i-1} - \alpha_i \widehat{q}_i$
5. $\beta_i = < \widehat{r}_{i-1}, \widehat{z}_{i-1} >_C / < \widehat{r}_{i-2}, \widehat{z}_{i-2} >_C$
6. $\widehat{z}_i = G\widehat{r}_i$
7. $\widehat{p}_i = \widehat{z}_{i-1} + \beta_i \widehat{p}_{i-1}$

- $\widehat{A} = R^{-1}HBH^T + I_m$
- G is the preconditioner.
- C is the inner-product.
- RPCG Algorithm: $C = HBH^T$ identical CG on original system : preserves monotonic decrease of quadratic cost and exploit geometry
- G should be symmetric w.r.t. to C $(FH^T = BH^T G)$

Initialization

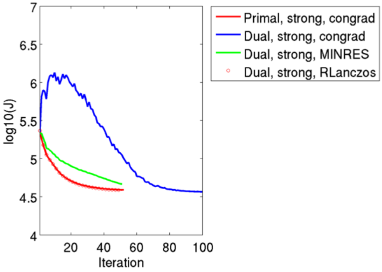
$\lambda_0 = 0$, $\widehat{r}_0 = R^{-1}(d - H(x_b - x))$,
$\widehat{z}_0 = G\widehat{r}_0$, $\widehat{p}_1 = \widehat{z}_0$, $k = 1$

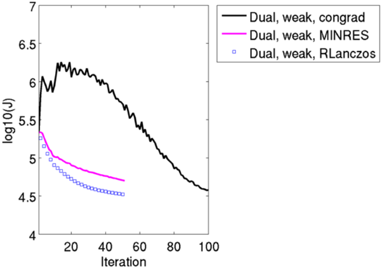
Loop on k

1. $\widehat{q}_i = \widehat{A}\widehat{p}_i$
2. $\alpha_i = < \widehat{r}_{i-1}, \widehat{z}_{i-1} >_C \; / < \widehat{q}_i, \widehat{p}_i >_C$
3. $\lambda_i = \lambda_{i-1} + \alpha_i \widehat{p}_i$
4. $\widehat{r}_i = \widehat{r}_{i-1} - \alpha_i \widehat{q}_i$
5. $\beta_i = < \widehat{r}_{i-1}, \widehat{z}_{i-1} >_C \; / < \widehat{r}_{i-2}, \widehat{z}_{i-2} >_C$
6. $\widehat{z}_i = G\widehat{r}_i$
7. $\widehat{p}_i = \widehat{z}_{i-1} + \beta_i \widehat{p}_{i-1}$

- $\widehat{A} = R^{-1}HBH^T + I_m$
- G is the preconditioner.
- C is the inner-product.
- RPCG Algorithm: $C = HBH^T$ identical CG on original system : preserves monotonic decrease of quadratic cost and exploit geometry
- G should be symmetric w.r.t. to C $(FH^T = BH^T G)$
- Best known reference : PSAS Algorithm for $C = R$

- Observations: SST (Sea Surface Temperature) and SSH(Sea Surface Height) observations from satellites. Sub-surface hydrographic observations from floats.
- Number of observations (m): 10^5
- Number of state variables (n): 10^6 for strong constraint and 10^7 for weak constraint.
- Computation: 64 CPUs





- It is **possible to maintain** the one-to-one correspondance between primal and dual iterates, under the assumption that

$$F_{k-1} H_k^T = B H_k^T G_{k-1}$$

where F_{k-1} is a preconditioner for a primal solver and G_{k-1} is a preconditioner for a dual solver (Gratton and Tshimanga 2009).

- The preconditioner G_{k-1} needs to be **symmetric** in $H_k B H_k^T$ inner product.

- Use **Preconditioned Conjugate Gradient method (PCG)**

  $\rightarrow$ Preconditioning with the quasi-Newton Limited Memory Preconditioner (Morales and Nocedal 2000) (Gratton, Sartenaer and Tshimanga 2011)

- For **linear case**, Gratton, Gurol and Toint (2012) derive the quasi-Newton LMP in dual space which generates **mathematically equivalent iterates** to those of primal approach.

The quasi-Newton LMP in dual space (Linear case)

- The **quasi-Newton LMP**: The descent directions p_i, $i = 1, ..., l$ generated by a CG method are used.

The quasi-Newton LMP in dual space (Linear case)

- The **quasi-Newton LMP**: The descent directions p_i, $i = 1, ..., l$ generated by a CG method are used.

$$F_i = \left(I_n - \frac{p_i p_i^T A}{p_i^T A p_i} \right) F_{i-1} \left(I_n - \frac{A p_i p_i^T}{p_i^T A p_i} \right) + \frac{p_i p_i^T}{p_i^T A p_i},$$

$\rightarrow F = F_l.$

The quasi-Newton LMP in dual space (Linear case)

- The **quasi-Newton LMP**: The descent directions p_i, $i = 1, ..., l$ generated by a CG method are used.

$$F_i = \left(I_n - \frac{p_i p_i^T A}{p_i^T A p_i}\right) F_{i-1} \left(I_n - \frac{A p_i p_i^T}{p_i^T A p_i}\right) + \frac{p_i p_i^T}{p_i^T A p_i},$$

$\rightarrow F = F_l$.

- The corresponding quasi-Newton preconditioner in dual space is given as

$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}\right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}\right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1} HBH^T$ and $\widehat{p}_i$ is the search direction.
$\rightarrow G = G_l$.

The quasi-Newton LMP in dual space (Linear case)

- The **quasi-Newton LMP**: The descent directions p_i, $i = 1, ..., l$ generated by a CG method are used.

$$F_i = \left(I_n - \frac{p_i p_i^T A}{p_i^T A p_i}\right) F_{i-1} \left(I_n - \frac{A p_i p_i^T}{p_i^T A p_i}\right) + \frac{p_i p_i^T}{p_i^T A p_i},$$

$\rightarrow F = F_l$.

- The corresponding quasi-Newton preconditioner in dual space is given as

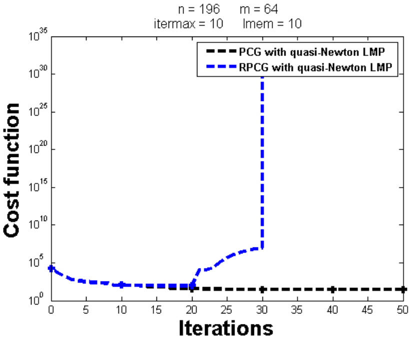
$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}\right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}\right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1}HBH^T$ and $\widehat{p}_i$ is the search direction.
$\rightarrow G = G_l$.

$\rightarrow$ This preconditioner satisfies the relation: $FH^T = BH^T G$ and it is symmetric in the C inner product (Gratton, Gurol and Toint 2012).

The quasi-Newton LMP in dual space (Nonlinear case)

- For **nonlinear case**, inheriting the previous preconditioner may not be possible!



The quasi-Newton LMP in dual space (Nonlinear case)

Solution:

- **Re-generate** the pairs and the preconditioner using the current inner product.

The quasi-Newton LMP in dual space (Nonlinear case)

Solution:

- **Re-generate** the pairs and the preconditioner using the current inner product.

$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1}HBH^T$ and $\widehat{p}_i$ is the search direction.

The quasi-Newton LMP in dual space (Nonlinear case)

Solution:

- **Re-generate** the pairs and the preconditioner using the current inner product.

$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1} HBH^T$ and $\widehat{p}_i$ is the search direction.

$\rightarrow$ It is **costly** for large-scale problems.

The quasi-Newton LMP in dual space (Nonlinear case)

Solution:

- **Re-generate** the pairs and the preconditioner using the current inner product.

$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

  where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1}HBH^T$ and $\widehat{p}_i$ is the search direction.

  $\rightarrow$ It is **costly** for large-scale problems.

- **Define a criterion** on whether we precondition the system or not by using a measure on symmetry.

  $\rightarrow$ Sensitive to the threshold value.

The quasi-Newton LMP in dual space (Nonlinear case)

Solution:

- **Re-generate** the pairs and the preconditioner using the current inner product.

$$G_i = \left(I_m - \frac{\widehat{p}_i \widehat{p}_i^T \widehat{A} C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) G_{i-1} \left(I_m - \frac{\widehat{A} \widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i} \right) + \frac{\widehat{p}_i \widehat{p}_i^T C}{\widehat{p}_i^T \widehat{A} C \widehat{p}_i}$$

  where $i = 1, ..., l$, $C = HBH^T$, $\widehat{A} = I_m + R^{-1}HBH^T$ and $\widehat{p}_i$ is the search direction.

  $\rightarrow$ It is **costly** for large-scale problems.

- **Define a criterion** on whether we precondition the system or not by using a measure on symmetry.

  $\rightarrow$ Sensitive to the threshold value.

Objective:

- A robust algorithm that handles this sensitivity
- A globally convergent algorithm
- Most cases are not highly nonlinear : $\rightarrow$ Perturb as little as possible the preconditioner of the linear case, and check a posteriori

- Global convergence can be ensured by inserting the Gauss-Newton strategy in a trust region framework.

- Trust-region method simply solves the following problem at iteration k:

$$\min_{\delta x_k \in \mathbb{R}^n} J(\delta x_k) = \frac{1}{2}\|\delta x_k - x_b + x_k\|^2_{B^{-1}} + \frac{1}{2}\|H_k \delta x_k - d_k\|^2_{R^{-1}}$$

$$\text{subject to } \|\delta x_k\|_{F_k^{-1}} \leq \Delta_k \text{ (primal approach)}$$

where Δ_k is the trust region radius.

Trust-region in dual space

- The preconditioner G_{k-1} that is inherited from previous iteration may **not be symmetric in the current inner product** and may **not be positive-definite in the full dual space** (merely in one direction).

Trust-region in dual space

- The preconditioner G_{k-1} that is inherited from previous iteration may **not be symmetric in the current inner product** and may **not be positive-definite in the full dual space** (merely in one direction).
- We need to adapt the strategy in the trust region algorithm.

Trust-region in dual space

- The preconditioner G_{k-1} that is inherited from previous iteration may **not be symmetric in the current inner product** and may **not be positive-definite in the full dual space** (merely in one direction).
- We need to adapt the strategy in the trust region algorithm.

TR Step calculation with a **flexible (Steihaug-Toint) RPCG algorithm**

1. Check the positive-definiteness along the steepest descent direction.
2. Compute the Cauchy step
3. Compute the step beyond the Cauchy step with the **RPCG** algorithm (ignoring symmetry problem)
4. Backtrack along the CG path if needed

Trust-region in dual space

- The preconditioner G_{k-1} that is inherited from previous iteration may **not be symmetric in the current inner product** and may **not be positive-definite in the full dual space** (merely in one direction).
- We need to adapt the strategy in the trust region algorithm.

TR Step calculation with a **flexible (Steihaug-Toint) RPCG algorithm**

1. Check the positive-definiteness along the steepest descent direction.
2. Compute the Cauchy step
3. Compute the step beyond the Cauchy step with the **RPCG** algorithm (ignoring symmetry problem)
4. Backtrack along the CG path if needed

Trust-region in dual space

Flexible trust region algorithm

1. Initialization
2. Compute the step by the **flexible (Steihaug Toint) RPCG algorithm**
3. **Accept the step beyond the Cauchy step** if

$$f(y_k) < f(x_k^C)$$

4. Accept the trial point according to the ratio of achieved to predicted reduction
5. Update the trust region

$\rightarrow$ The global convergence can be proved! (see S. Gurol PhD)

Trust-region in dual space

Flexible trust region algorithm

1. Initialization
2. Compute the step by the **flexible (Steihaug Toint) RPCG algorithm**
3. **Accept the step beyond the Cauchy step** if

$$f(y_k) < f(x_k^C)$$

4. Accept the trial point according to the ratio of achieved to predicted reduction
5. Update the trust region

→ The global convergence can be proved! (see S. Gurol PhD)

→ This approach is similar to the approach that computes the magical step proposed by (Conn, Gould, Toint 2000).

Numerical experiment on heat equation (1/3)

The dynamical model is considered to be the nonlinear heat equation defined by

$$\frac{\delta x}{\delta t} - \frac{\delta^2 x}{\delta u^2} - \frac{\delta^2 x}{\delta v^2} + f[x] = 0 \text{ in } \Omega \times (0, \infty)$$

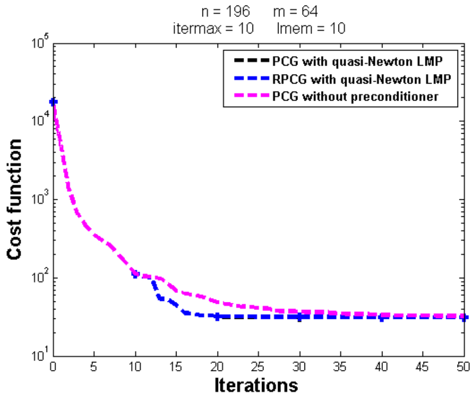
$$x[u, v, t] = 0 \text{ on } \delta\Omega \times (0, \infty)$$

where the temperature variable $x[u, v, t]$ depend on both time t and position given by spatial coordinates u and v. The function $f[x]$ is defined by

$$f[x] = exp[\eta x]$$

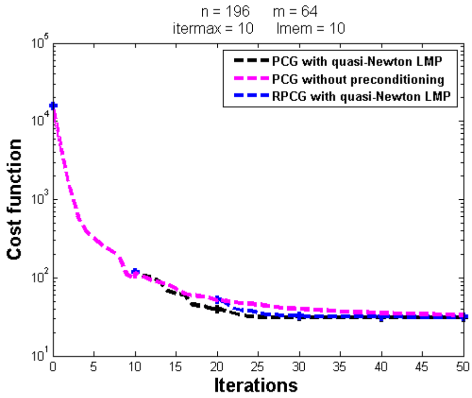
Numerical experiment on heat equation (2/3)

$$f[x] = exp[\eta x] \;\; \eta = 2$$



Numerical experiment on heat equation (3/3)

$$f[x] = exp[\eta x] \quad \eta = 4.2$$



Conclusion

- We developed and implemented a fast and robust preconditioned non-linear solver for large-scale problems
- The solver is implemented in operational systems in Meteorology and Oceanography
- Other techniques are used based on variant of Kalman filters : model reduction, ensemble algorithms

• S. Gratton, P. Toint and J. Tshimanga.
Inexact range-space Krylov solvers for linear systems arising from inverse problems.
SIAM Journal on Matrix Analysis and Applications, 32(3):969-986, 2011

• S. Gratton, P. Laloyaux, A. Sartenaer, and J. Tshimanga.
A reduced and limited-memory preconditioned approach for the 4d-var data-assimilation problem. Q. J. Roy. Meteor. Soc., 137:452–466, 2011.

• S. Gratton and J. Tshimanga.
An observation-space formulation of variational assimilation using a restricted preconditioned conjugate gradient algorithm. Q. J. Roy. Meterol. Soc., 135:1573–1585, 2009.

• J. Tshimanga, S. Gratton, A.T. Weaver, and A. Sartenaer.
Limited-memory preconditioners with application to incremental four-dimensional variational data assimilation. Q. J. Roy. Meterol. Soc., 134:753–771, 2008.

• S. Gratton, S. Gurol, Ph.Toint. *Preconditioning and Globalizing Conjugate Gradients in Dual Space for Quadratically Penalized Nonlinear-Least Squares Problems.* Computational Optimization and Applications, 54(1), pp. 1-25, 2013.

• S. Gratton, J. Tshimanga and Ph. L. Toint. *Conjugate-gradients versus multigrid solvers for diffusion-based correlation models in data assimilation.* Quarterly Journal of the Royal Meteorological Society, to appear, 2013.

• S. Gratton and L. N. Vicente. *A surrogate management framework using rigorous trust-region steps.* Optimization Methods and Software to appear, 2013.