Optimization methods for variationnal data assimilation

Serge **Gratton** (CERFACS)
joint work with
M. Arioli (RAL),
and
A. Sartenaer (FUNDP), J. Tshimanga (FUNDP) and Ph.L.Toint (FUNDP)

Optimization and Engineering

Leuven, May 24, 2006

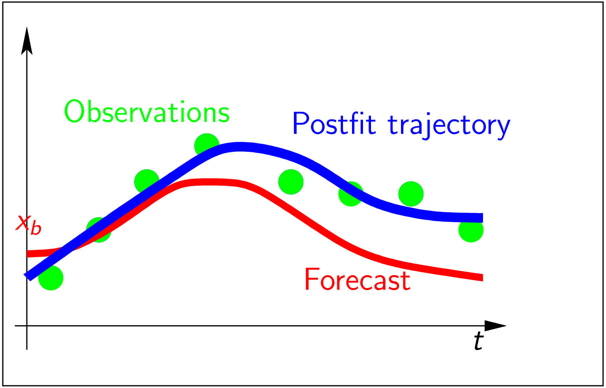
Data assimilation : some dates

- 1950. Beginning of Numerical forecast.
- 1960. First operational models (barotropic models).
- 1963. Gandins method (kgriging).
- Optimal interpolation.
- 1980. 4D VAR equations.

4D VAR approach

- Find the initial state of a dynamical system.
- Use observations of the system.
- Perform forecasts.

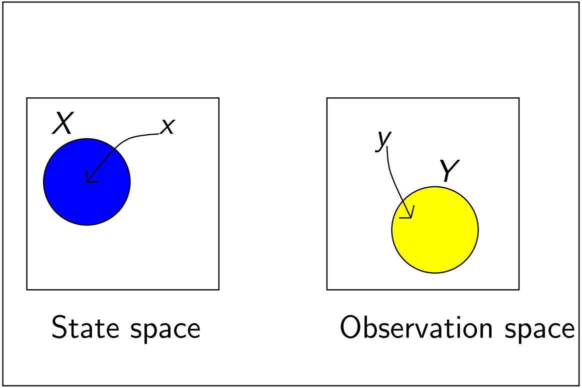
4D VAR approach



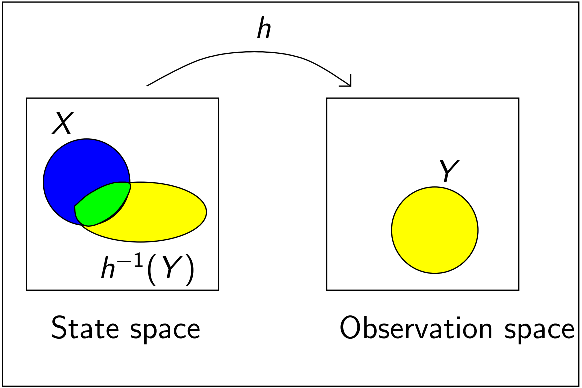
Statistical model

- A priori knowledge on values of $\mathbf{x}$.
- Observations : y_i
- Observation model $y_i = h_i(x)$.

Estimation from set theory

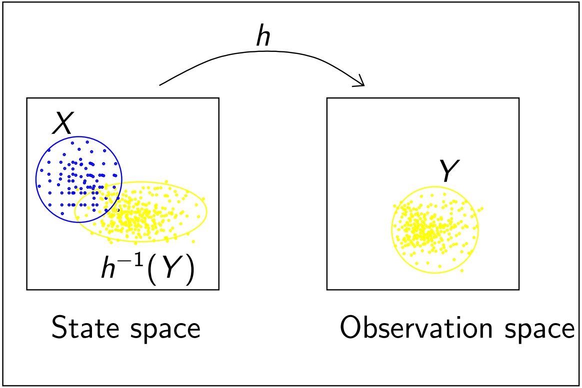


Estimation from set theory



Model $y = h(x)$, solution in $X \cap h^{-1}(Y)$.

Statistical point of view



Estimation using observations

- If $\mathbf{x}$ and $\mathbf{y}$ (random vectors) are independent : little (if any) can be said from $\mathbf{x}$ if some values are known from $\mathbf{y}$
- General question : if $\mathbf{y}$ assumes some values in an experiment, what can be guessed about values taken by $\mathbf{x}$?
- Problem : find a good estimate $\hat{\mathbf{x}} = g(\mathbf{y})$ of $\mathbf{x}$.
- The minimum variance estimator solves

$$\min_{g(.)(measurable)} \mathrm{E}[\mathbf{x} - g(\mathbf{y})][\mathbf{x} - g(\mathbf{y})]^T.$$

Parameter estimation

- Linear dependence $\mathbf{y} = H\mathbf{x} + \mathbf{e}$.
- The random variables $\mathbf{x}$ and $\mathbf{e}$ are uncorrelated.
- $E[\mathbf{x}] = x_b$, $E[\mathbf{e}] = 0$.
- $E[(\mathbf{x} - x_b)(\mathbf{x} - \mathbf{x_b})^T] = B$, $E[\mathbf{e}\mathbf{e}^T] = R$.
- $g(K) = x_b + K(\mathbf{y} - Hx_b)$.
- The minimum variance estimator of $\mathbf{x}$ is

$$\hat{\mathbf{x}} = x_b + (B^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} (\mathbf{y} - Hx_b)$$

"Equivalence" with the deterministic problem

$$\min_{\hat{x}} J(\hat{x}) = \frac{1}{2}||\hat{x} - x_b||^2_{B^{-1}} + \frac{1}{2}||H\hat{x} - y||^2_{R^{-1}}$$

Inclusion of time : statistical model

- Observations at t_j : $\mathcal{H}_j(\mathcal{M}_j(x)) = y_j + e_j$
- Initial state $x = x(t_0) = x_b + \epsilon$.
- No model error : $x(t_j) = \mathcal{M}_j(x)$.

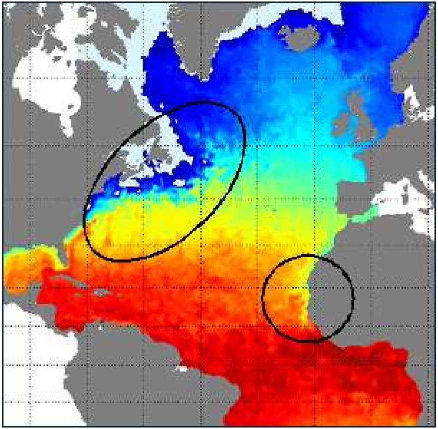
Optimization methods for Incr. 4DVAR

Problem formulation: nonlinear optimization problem

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

- Regularized problem
- Large problems : $x \in \mathbb{R}^{10^6}$, $y_j \in \mathbb{R}^{10^5}$ [ORCAVAR, Weaver].
- The observations y_j are noisy.
- Effective solution strategy Incremental 4DVAR : use the inexact/truncated Gauss-Newton algorithm on $J(x) = \frac{1}{2}F(x)^T F(x) = \frac{1}{2}||F(x)||_2^2$.

Example : sea temperature



Gulf stream and Upwellings on the African coast
Use of satellite altimetry, model 1/3 degree

Incremental 4DVAR algorithm

For $k = 0$ DO

Solve for s_k the LLSP $\min_s \|F'(x_k)s + F(x_k)\|_2$ by the conjugate gradients method (CG, e.g. CONGRAD)

Set $x_{k+1} = x_k + s_k$

Fixed point convergence theory for
$x_{k+1} = x_k - F(x_k)'^{\dagger} F(x_k) = G(x_k)$:

- F is twice continuously differentiable in neighborhood of $x_\star$.
- $F'(x_\star)^T F(x_\star) = 0$.
- $F'(x_\star)$ has full column rank.

$\{x_k\}$ converges locally to $x_\star$ if $\sigma = \rho(G'(x_\star)) < 1$.
Note : σ is a geometrical quantity that is invariant under change of variables.

Outline

- **Convergence condition and truncation (M. Arioli)**
- Preconditioning (A. Sartenaer and J. Tshimanga)
- Use of the underlying PDE structure (A. Sartenaer and Ph.L. Toint)
- Conclusions and perspectives

CG truncation

- Solving $F'(x_k)^T F'(x_k) s_k = F'(x_k)^T F(x_k)$ (denoted ${F'_k}^T F'_k s_k = {F'_k}^T F_k$) exactly is very expensive for large systems.
- For $\eta_k > 0$, stop the CG method when $\delta(s_k) \leq \eta_k \ \delta(s_0)$ i.e. the stopping criterion is satisfied.
  - $\delta_{Res}(s) = \|{F'_k}^T F'_k s - {F'_k}^T F_k\|_2$
  - $\delta_{EN}(s) = \|s - {F'_k}^{\dagger} F_k\|_{{F'_k}^T F'_k}$ (see [Strakos, Tichy, 2005],[Arioli, 2004]).

  $\{x_k\}$ converges locally to $x_\star$ if $\eta_k \leq \eta_{\max}$ and $\sigma + \eta_{\max}(1+\sigma) < 1$ (See [Dennis, Steihaug, 1986] for δ_{Res})
- Why δ_{EN} ? :
  - CG converges monotonically in the energy norm.
  - Case of noisy problems.

Energy norm of the error for linear least-squares problems

- Linear case $As = b + \epsilon$, $\epsilon \sim \mathcal{N}(0, I)$ (or after linearization, $F_k' s_k = -F_k + \epsilon$)
- Maximum Likelihood estimate: s_k minimizing $\|F_k' s + F_k\|_2$
- Backward error problem

$$\eta(s) = \min\{\|\Delta F_k\|_2 \text{ s.t. } s \text{ solves } \min_u \|F_k' u + (F_k + \Delta F_k)\|_2\}$$

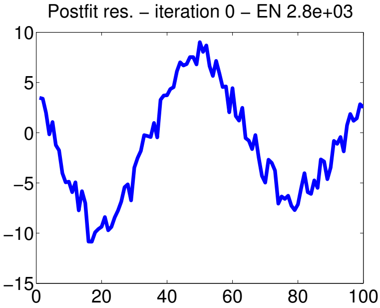
- Closed solution $\eta(s_k) = \delta_{EN}(s_k) = \|s_k - F_k'^{\dagger} F_k\|_{F_k'^T F_k'}$
- Want to have $\delta_{EN}^2(s)$ below the noise level $\|\epsilon\|_2^2$.
- $\|\epsilon\|_2^2$ follows a χ squared distribution, with m dof.

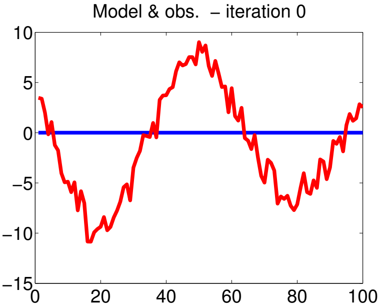
Numerical experiment with the energy norm

- Linear case $As = b + \epsilon$, $A \in \mathbb{R}^{m \times n}$, $m = 100$, $n = 10$
- Two test-cases best discrete least-squares approximation of a function
  - as linear combination of $t \mapsto sin(i\, t)$ (Well-cond. case),
  - as linear combination of $t \mapsto t^i$ (Ill-cond. case),
- where the t_i's are equally spaced between in $[1 \quad 2]$, the exact solution being $(1, 2 \ldots, 10)^T$.
- ϵ is a Gaussian random vector $\mathcal{N}(0, I_n)$.
- We plot the residual $b - As$ for each CG iterate s and compute $\delta^2_{EN}(s)$
- The probability that a sample of $\epsilon^T \epsilon$ is below 50.0 is very weak ($< 0.1\%$)

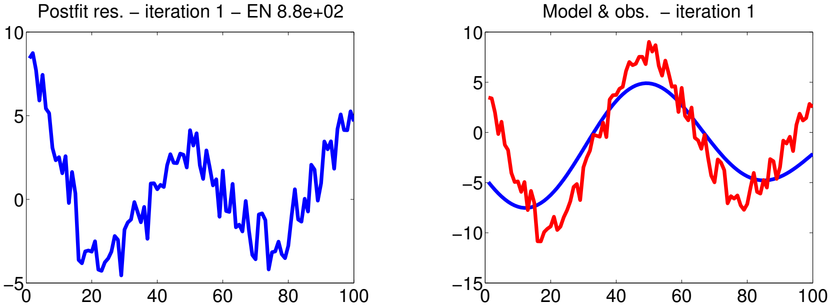
Well-conditioned problem

Left : Residual $b - As$. Right : Observations b (red) and As (blue)

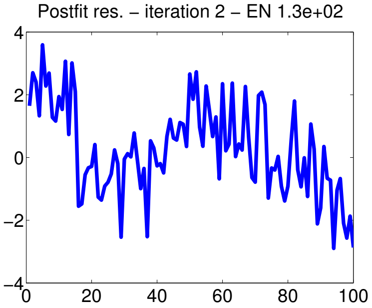


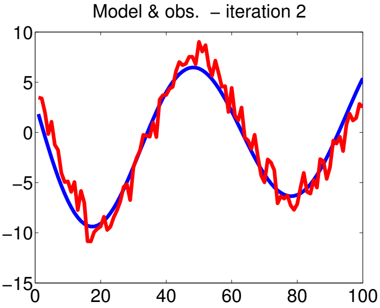


Well-conditioned problem

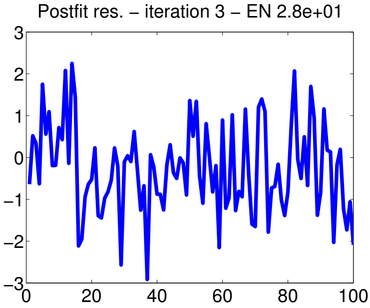


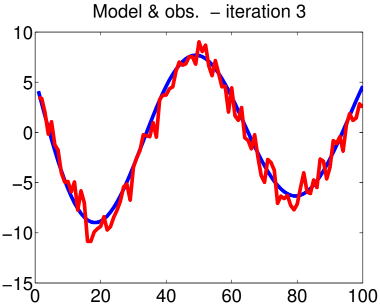
Well-conditioned problem



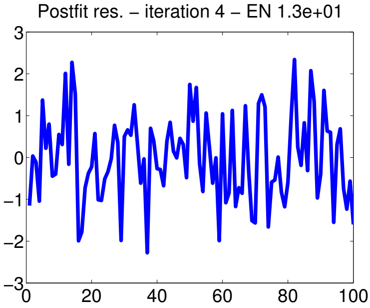


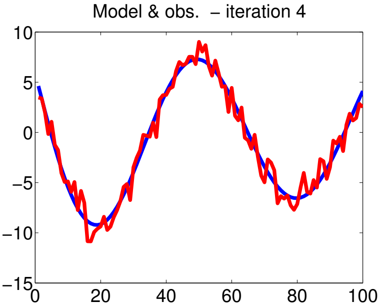
Well-conditioned problem



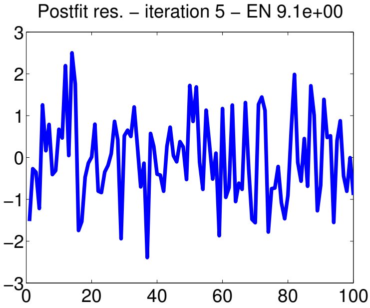


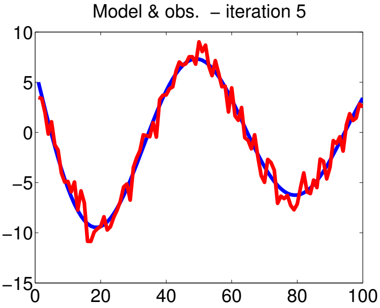
Well-conditioned problem



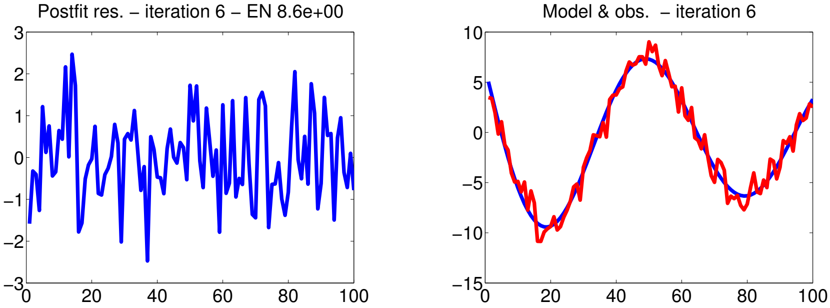


Well-conditioned problem

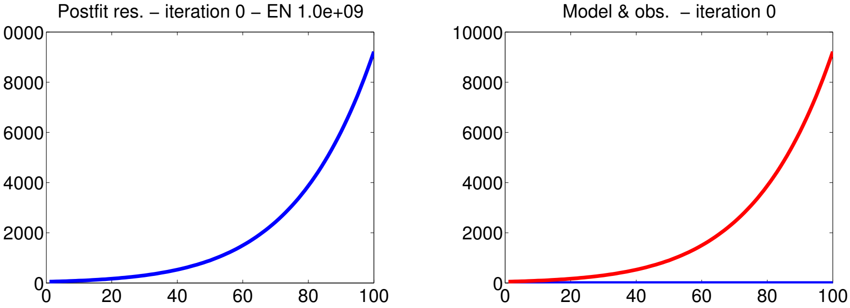




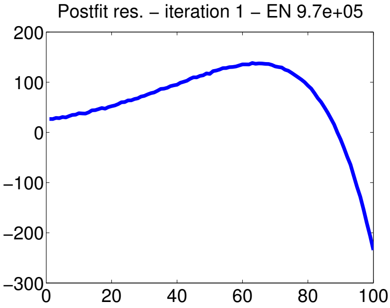
Well-conditioned problem

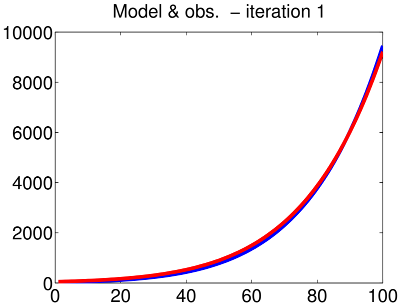


Ill-conditioned problem

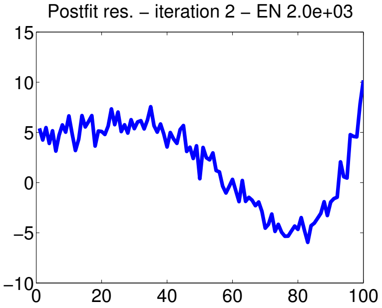


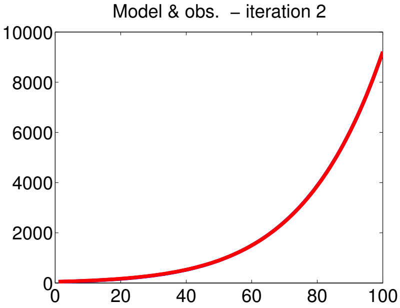
Ill-conditioned problem



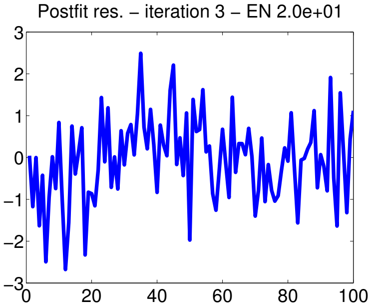


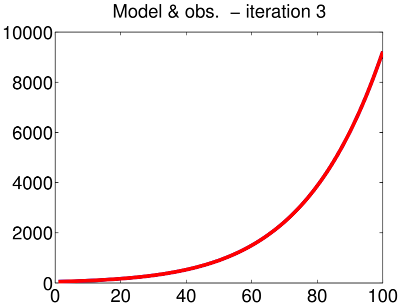
Ill-conditioned problem



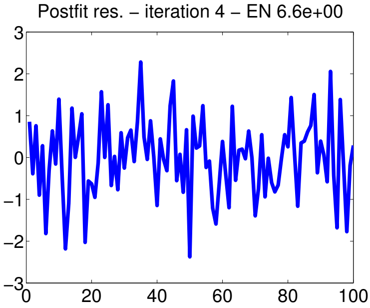


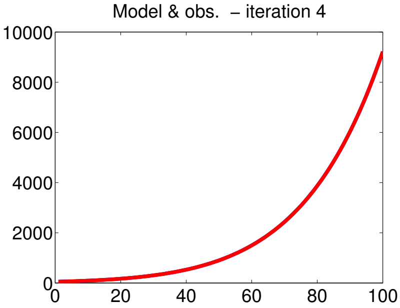
Ill-conditioned problem



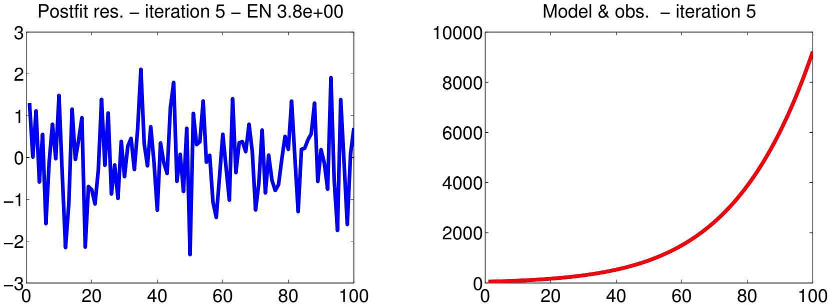


Ill-conditioned problem

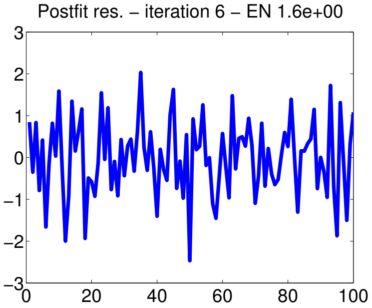


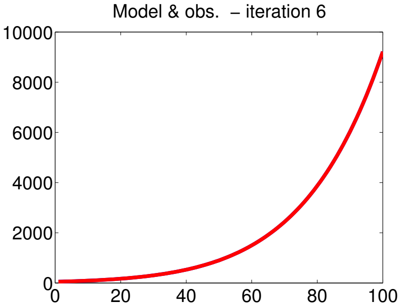


Ill-conditioned problem



Ill-conditioned problem





Conclusion

- Stopping criterion based on the energy norm of the error.
- Natural when CG is used.
- Interesting properties for noisy problems.
- More test needed ...

Outline

- Convergence condition and truncation (M. Arioli)
- **Preconditioning (A. Sartenaer and J. Tshimanga)**
- Use of the underlying PDE structure (A. Sartenaer and Ph.L. Toint)
- Conclusions and perspectives

A sequence of linear least-squares problems

- Originally developped for SPD linear systems with multiple right-hand sides (RHS).
- Solve systems $Ax = b_1$, $Ax = b_2$, ..., $Ax = b_r$ with RHS in sequence, by iterative methods: Conjugate Gradient (CG) or variants.
- Precondition the CG using information obtained when solving the previous system.
- Extension of the idea to nonlinear process such as Gauss-Newton method. The matrix of the normal equations varies along the iterations.

The CG algorithm (A is spd and large !)

- CG is an iterative method for solving

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}x^T Ax - b^T x \qquad or \qquad Ax = b$$

- Iterations: Given $x_0 \in \mathbb{R}^n$; $\quad A \in \mathbb{R}^{n\times n}$; $\quad b \in \mathbb{R}^n$
  - Set $r_0 \leftarrow Ax_0 - b_0$; $\quad p_0 \leftarrow -r_0$; $\quad i \leftarrow 0$
  - Loop on i

$$\begin{array}{lcl} \alpha_i & \leftarrow & (r_i^T r_i)/(p_i^T Ap_i) \\ x_{i+1} & \leftarrow & x_i + \alpha_i p_i \\ r_{i+1} & \leftarrow & r_i + \alpha_i Ap_i \\ \beta_{i+1} & \leftarrow & (r_{i+1}^T r_{i+1})/(r_i^T r_i) \\ p_{i+1} & \leftarrow & -r_{i+1} + \beta_{i+1} p_i \end{array}$$

- r_i are residuals; p_i are descent directions; $\alpha_i p_i$ are steps.

The CG properties (in exact arithmetic !)

- Orthogonality of the residuals: $r_i^T r_j = 0$ if $i \neq j$.
- A-conjugacy of the directions: $p_i^T A p_j = 0$ if $i \neq j$.
- The distance of the iterate x_i to the solution x^* is related to the condition number of A, denoted by $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$ (≥ 1):

$$||x_i - x^*||_A \leq \eta_i \; ||x_0 - x^*||_A \quad \text{with} \quad \eta_i = 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^i$$

  $\Rightarrow$ The smaller $cond(A) \equiv \kappa$ is, the faster the convergence.
- Exact solution found exactly in r iterations, where $r \leq n$ is the number of distinct eigenvalues of $A \in \mathbb{R}^{n \times n}$.
  $\Rightarrow$ The more clustered the eigenvalues are, the faster the convergence.

Why to precondition ?

- Transform $Ax = b$ in an equivalent system having a more favorable eigenvalues distribution.
- Use a preconditioning matrix H (which must be cheap to apply).
- Ideas to design preconditioner H:
  - H approximates A^{-1}.
  - $cond(HA) < cond(A)$.
  - HA has eigenvalues more clustered than those of A.
- Note: when a preconditioning is used, residuals are:
  - Orthogonal if H is factored in LL^T.
  - Conjugate w.r.t. H if H is not factored.

Preconditioning techniques considered (I)

- We consider techniques to precondition or improve an existing preconditioner (second level preconditioning) :
  - Solve $Ax = b_1$ and extract information $info_1$.
  - Use $info_1$ to solve $Ax = b_2$ and extract information $info_2$.
  - Use $info_2$ (and possibly $info_1$) to solve $Ax = b_3$ and ...
  - ...
- $Info_k$ will be:
  - residuals;
  - descent directions;
  - steps;
  - or other vectors such as eigenvectors of A ...

Preconditioning techniques considered (II)

We study and compare two approaches:

- Deflation [Frank, Vuik, 2001].
- Limited Memory Preconditioners (LMP): Preconditioners based on a set of A-conjugate directions.
  - Generalization of known preconditioners: spectral [Fisher, 1998], L-BFGS [Nocedal, Morales, 2000], warm start [Gilbert, Lemaréchal, 1989].

We cover:

- Theoretical properties.
- Numerical experiments (data assimilation).

Deflation Techniques

- Given $W \in \mathbb{R}^{n\times k}$ $(k \ll n)$ formed with appropriate information obtained when solving the previous system.
- Consider the oblique projector $P = I - AW(W^TAW)^{-1}W^T$.
- Split the solution vector as follows $x^* = \underbrace{(I - P^T)x^*}_{direct} + \underbrace{P^Tx^*}_{iterative}$.
- Compute $(I - P^T)x^*$ with a direct method.
- Compute P^Tx^* with an iterative method.

Some Properties (Deflation)

- Computation of $(I - P^T)x^*$:
  - $(I - P^T)x^* = W(W^TAW)^{-1}W^TAx^* = W(W^TAW)^{-1}W^Tb.$
- Computation of P^Tx^*:
  - Any solution of the compatible singular system $PAy = Pb$ satisfies $P^Tx^* = P^Ty$.
  - Note: $PA = (PA)^T$ and $cond(PA) \leq cond(A)$.
  - Use CG with $y_0 = 0$ to solve $PAy = Pb$ and compute P^Ty.

Limited Memory Preconditioners (LMP)

- General form:

$$H_{k+1} = [I - \sum_{i=0}^{k} \frac{Aw_i w_i^T}{w_i^T A w_i}]^T [I - \sum_{i=0}^{k} \frac{Aw_i w_i^T}{w_i^T A w_i}] + \sum_{i=0}^{k} \frac{w_i w_i^T}{w_i^T A w_i},$$

with $w_i^T A w_j \begin{cases} = 0 & \text{if } i \neq j \\ > 0 & \text{if } i = j \end{cases}$

- Particular forms
  - The w_i's are the descent directions obtained from CG: $w_i = p_i$ $\Rightarrow$ L-BFGS preconditioner.
  - The w_i's are eigenvectors of A: $w_i = v_i$ $\Rightarrow$ spectral preconditioner.

Spectral Properties for LMP (I)

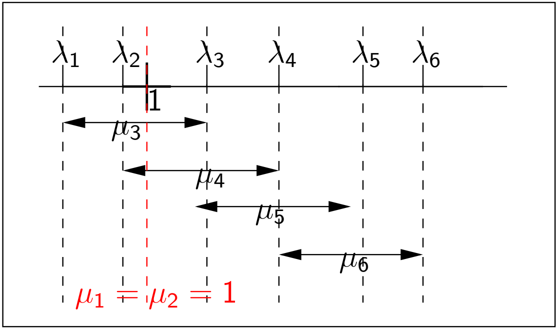
- Theorem : the spectrum $\mu_1, \ldots, \mu_n$ of the preconditioned matrix $H_{k+1}A$ satisfies:

$$\begin{cases} \mu_j = 1, & \text{for } j = 1, \ldots, k \\ \lambda_{j-k}(A) \leq \mu_j \leq \lambda_j(A), & \text{for } j = k+1, \ldots, n, \end{cases}$$

where $\lambda_j(A)$ is the $j-$th eigenvalue of A (increasing order assumed).

- Note: the matrix A to precondition is the same (only the RHS changes).

Spectral properties for LMP (II)



- Eigenvalues translated to 1.
- The rest of the spectrum is not expanded compared to the spectrum of A.

Existence of a factored form for the LMP (not the Cholesky factor !)

- L-BFGS:
  - A possible factored form is $H_{k+1} = L_{k+1} L_{k+1}^T$ where:

$$L_{k+1} = \prod_{i=0}^{k} \left(I - \frac{s_i {y_i}^T}{{y_i}^T s_i} + \frac{s_i}{\sqrt{{y_i}^T s_i}} \frac{{r_i}^T}{\|r_i\|} \right),$$

    with $s_i = x_{i+1} - x_i$ and $y_i = r_{i+1} - r_i$.
  - Same cost in memory and CPU as the unfactored form.
- Spectral:
  - A possible factored form is $H_{k+1} = L_{k+1}^2$ where:

$$L_{k+1} = I + \sum_{i=1}^{k+1} \left(\frac{1}{\sqrt{\lambda_i}} - 1 \right) \frac{v_i {v_i}^T}{{v_i}^T v_i}.$$

  - Same cost in memory as the unfactored form.

Why looking for a factored form $H = LL^T$?

- With a non factored form, we use CG preconditioned by H.
- With a factored form, we solve $L^T A L u = L^T b$; $x = Lu$.

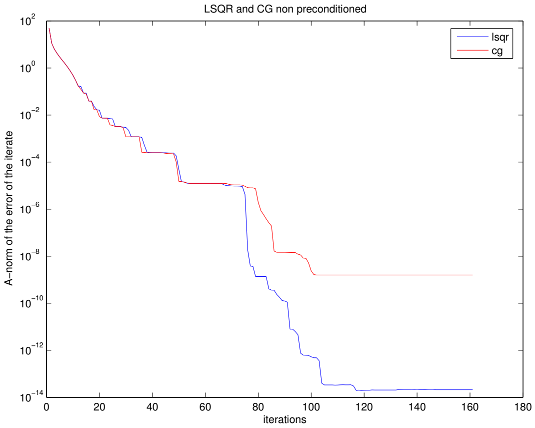
  Advantages:

  - When accumulating preconditioners, symmetry and positiveness are still maintained:

    $$L_1^T A L_1 y_1 = L_1^T b_1, \quad L_2^T (L_1^T A L_1) L_2 y_2 = L_2^T L_1^T b_2, \quad \dots$$

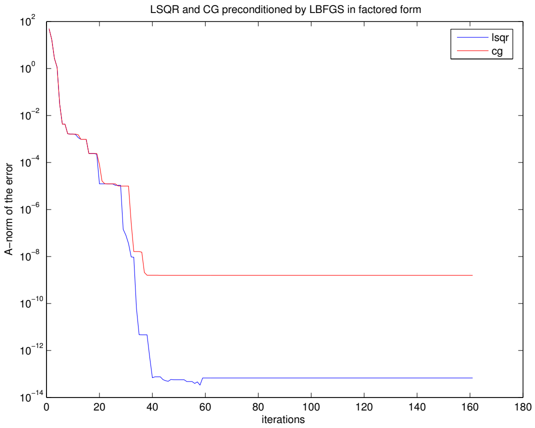
  - Least-squares $min_x \|Ax - b\|$ or $AA^T x = A^T b$:
    LSQR (or CGLS) is more accurate than CG in presence of rounding errors but works with (A, A^T, L, L^T, b) instead of $(A^T A, A^T b, H)$.
  - More appropriate if reorthogonalization of the residuals is used.

Experiments with unpreconditioned LSQR



LSQR is better than CG !

Experiments with LSQR preconditioned with factored L-BFGS

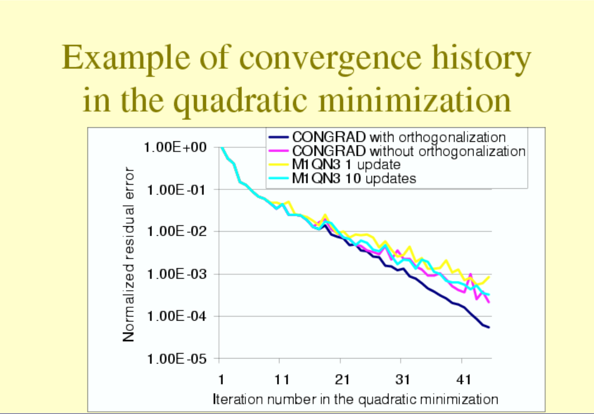


LSQR is again better than CG !

Why to reorthogonalize the residuals ?

- In finite precision, residuals often loose orthogonality (or conjugacy) and theoretical convergence is then slowed down.
- Reorthogonalization of residuals in CG is terribly successful when matrix-vector product is very expensive compared to other computations in CG (see example in the next slide).
- Note: to restore orthogonality or conjugacy, working with $L^T AL$ and the canonical inner-product is better (memory, CPU, error propagation) than working on A preconditioned by H.

Example of reorthogonalization effect : CERFACS data assimilation system (1 000 000 unknowns)



Experiments with a data assimilation problem

Problem formulation: nonlinear least-squares problem

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

- Size of real (operational) problems : $x \in \mathbb{R}^{10^6}$, $y_j \in \mathbb{R}^{10^5}$.
- The observations y_j are noisy.
- Solution strategy : Incremental 4DVAR (i.e. inexact/truncated Gauss-Newton algorithm).

Main ingredients

- Sequence of linear symmetric positive definite systems to solve:
$$A_i^T A_i x = A_i^T b_i$$

- Whose matrix varies.

Experiments description

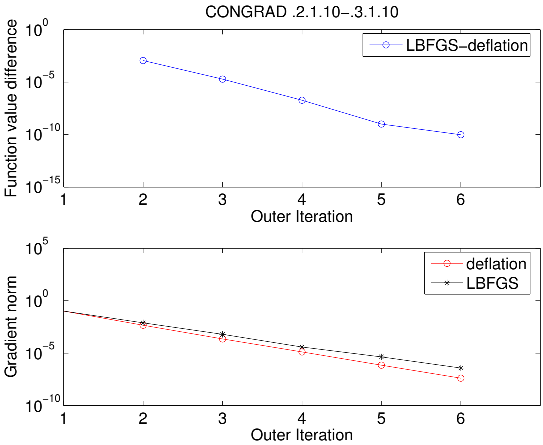
Algorithmic variants tested:

- Use CG to solve the normal equations.
- Compare 3 preconditioning techniques:
  - Deflation technique (using spectral information).
  - Spectral preconditioner (using spectral info. but differently).
  - L-BFGS preconditioner (using descent directions).
- Where spectral information is needed, use Ritz (vectors) as approximations of the eigenvectors.
- Ritz vectors are obtained by mean of a variant of CG: the Lanczos algorithm which combines linear and eigen solvers.

Experiment on a small system (I) [A. Lawless, N. Nichols, 2001, University of Reading]

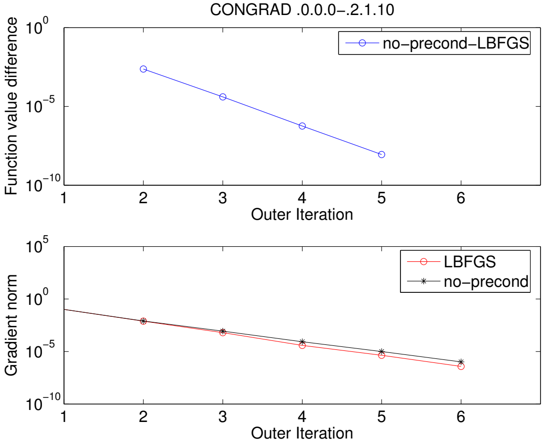
Ranking of the preconditioners using the basic strategies.

Results L-BFGS - Deflation



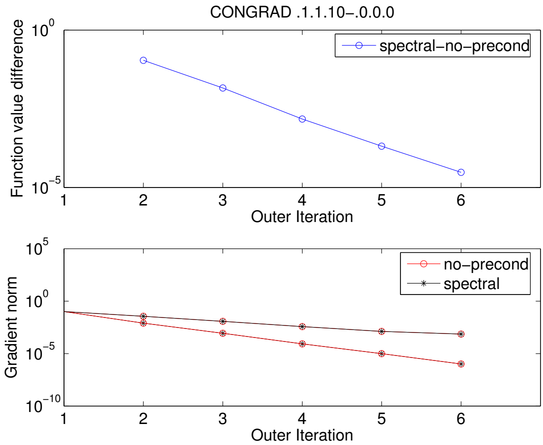
Deflation is better than L-BFGS !

Results Noprecond - L-BFGS



L-BFGS is better than Noprecond !

Results Spectral - Noprecond



Noprecond is better than Spectral !

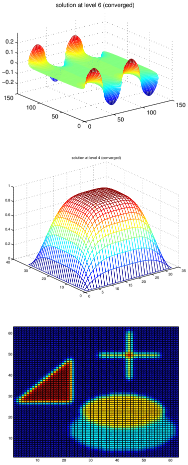
Remarks on our system !

- Spectral preconditioner:
  - Does not work in our case.
- L-BFGS preconditioner:
  - Requires no large changes in the matrix.
  - Based on by-products of CG.
  - More efficient than the spectral preconditioner or than no preconditioner.
- Deflation:
  - Is stable even when the matrix changes.
  - May be expensive ($W^T A W$) in CPU time.
  - More efficient than the other techniques.

Outline

- Convergence condition and truncation (M. Arioli)
- Preconditioning (A. Sartenaer and J. Tshimanga)
- **Use of the underlying PDE structure (A. Sartenaer and Ph.L. Toint)**
- Conclusions and perspectives

Some treated problems



- Minimal surface :

$$\min_u \int\int \sqrt{1 + u_x^2 + u_y^2} dxdy$$

- Quadratic minimization :

$$\min_u u^T A u - 2u^T b \Leftrightarrow Au = b$$

- Image deblurring problem :

$$\min \mathcal{J}(f) = \frac{1}{2}\|Tf - d\|_2^2 + TV(f),$$

where $TV(f)$ is the discretization of

$$\int_0^1 \int_0^1 \left(1 + (\partial_x f)^2 + (\partial_y f)^2\right)^{\frac{1}{2}} dxdy.$$

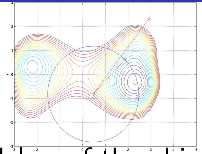
Why multigrid

- Solution based on discretization :
  High accuracy $\Rightarrow$ computational cost
- Use of coarse grids :
  1. to find a good starting point
  2. to solve a subproblem (ex : residual equation)
- Well-known efficient method for solving SPD linear systems resulting of the discretization of a continuous problem
- Multigrid tutorial [W. Briggs, V.E. Henson and S. McCormick, 2000]

Why trust-region methods

- Newton method : local quadratic convergence
- Trust-region methods : Convergence for all starting point (Global convergence)
- Reduces to the Newton method when close enough to the solution $\Rightarrow$ Quadratic convergence
- Overview of convergence results and algorithms [A. Conn, N. Gould and Ph. Toint, 2000]

Trust-region mechanism

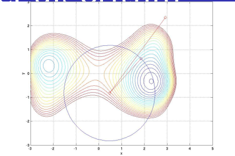


- Definition of a model m of the objective function f
- Definition of a region where the model is supposed to represent well the objective function
- Computation of a step that sufficiently reduces m
- Step acceptance and TR radius Δ update related to the ratio

$$\frac{f(x_{k+1}) - f(x_k)}{m(x_{k+1}) - m(x_k)}$$

  - Refuse the step and shrink the TR when the ratio is smaller than a constant
  - Accept the step and possibly enlarge the TR when the ratio is large enough

Trust-region multilevel algorithm



- The trust-region model is possibly based on the coarse grid problem.
- Algorithm using same ingredient as traditional multigrid: smoothing, prolongation (P), restriction (R),...
- Correction of the models similar to that in the FAS multigrid method is needed.
- Global convergence to first order and weakly second order critical points is proved [Gratton, Sartenaer, Toint, 2005] .

2-level description

$$
\begin{tikzcd}
x_{h,k} \arrow[d, "R"'] & & & x_{h,k+1}+s_h = x_{h,k}+P(x_{H,*}-x_{H,0}) \\
Rx_{h,k}=x_{H,0} \arrow[r] & x_{H,1} & x_{H,2} \arrow[r] & x_{H,3}=x_{H,*} \arrow[u]
\end{tikzcd}
$$

- Fine space h, coarse space H.
- f_h fine function to be minimized. f_H coarse representation.

First order coherence between levels

- The immediate coarse model is defined by $h_H(x_{H,0}+s) = f_H(x_{H,0}+s) + < v, s >$, where $v = R\nabla_x h_h(x_{h,k}) - \nabla_x f_H(x_{H,0})$. Therefore $\nabla_x h_H(x_{H,0}) = R\nabla_x h_h(x_{i+1,k})$
- Linear coherence

$$\begin{aligned} h_h(x_{h,k}+Ps) &= h_h(x_{h,k}) + \\ &\quad < R\nabla_x h_h(x_{h,k}), s > + o(s) \\ h_H(x_{H,0}+s) &= h_H(x_{H,0}) + \\ &\quad < \nabla_x h_H(x_{H,0}), s > + o(s) \end{aligned}$$

- Recursion useful only if $\|R\nabla_x h_h(x_{h,k})\| \geq \kappa \|\nabla h_h(x_{h,k})\|$
- Linear correction: similar to the full approximation scheme !

Model choice

At iterate $x_{h,k}$, if recursion useful (i.e. $\|R\nabla_x h_h(x_{h,k})\| \geq \kappa \|\nabla h_h(x_{h,k})\|$)

- use either a Taylor model $m_k(s) = h_h(x_{h,k}) + < \nabla_x h_h(x_{h,k}), s > + \frac{1}{2} < \nabla_{xx} h_h(x_{h,k}) s, s >$ or the coarse model h_H
- if not useful, use a Taylor model

A possible smoother

Approximatively solve (suff. decrease) the local TR subproblem $\min_{\|s\|_i \le \Delta} Q(s) = \frac{1}{2} < Hs, s > + < g, s >$

- Steihaug-Toint truncated CG
- Exact Moré-Sorensen Method on small dimension spaces
- Take advantage of the good smoothing properties of linear Gauss-Seidel
  - Compute s_0 by minimizing along the largest gradient component
  - Perform some Gauss-Seidel cycles (minimization along coordinate axes) to obtain s_1
  - Take s_1 if $\|s_1\|_i \le \Delta$
  - Else, if s_1 is gradient related ($< g, s_1 > \le \kappa \|s_1\| \|g\|$), backtrack
  - Else minimize $Q(s)$ for $\|s\|_i \le \Delta$ on the path $[0, s_0, s_1]$

Numerical experiments

Comparison of multigrid and mesh refinement on test-problems

Test-problems

- Dirichlet-to-Neumann transfer problem (DN) Minimize

$$\int_0^\pi \left(\partial_y u(x,0) - \phi(x)\right)^2 dx,$$

where u is the solution of the boundary value problem

$$\begin{array}{lcll} \Delta u & = & 0 & \text{in } S, \\ u(x,y) & = & a(x) & \text{on } \Gamma, \\ u(x,y) & = & 0 & \text{on } \partial S \backslash \Gamma. \end{array}$$

and $\phi(x) = \sum_{i=1}^{15} \sin(i\,x) + \sin(40\,x)$.

- 2D Quadratic (check) example (Q2)

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} x^T A x - x^T b,$$

where A is a FD discretization of the Laplacian.

Test-problems

- 3D Quadratic example (Q3) obtained as from a FD of

$$\begin{aligned} -(1+\sin(3\pi x)^2)\Delta u(x,y,z) &= f \quad \text{in } S_3, \\ u(x,y,z) &= 0 \quad \text{on } \partial S_3. \end{aligned}$$

- The minimum surface problem (Surf)

$$\min_v \int_0^1 \int_0^1 \left(1+(\partial_x v)^2+(\partial_y v)^2\right)^{\frac{1}{2}} \, dx\, dy,$$

The oscillatory boundary condition is

$$v_0(x,y) = \begin{cases} f(x), & y=0, \quad 0 \le x \le 1, \\ 0, & x=0, \quad 0 \le y \le 1, \\ f(x), & y=1, \quad 0 \le x \le 1, \\ 0, & x=1, \quad 0 \le y \le 1, \end{cases}$$

where $f(x) = \sin(4\pi x) + \frac{1}{10}\sin(120\pi x)$

Test-problems

- The image deblurring problem (Inv)

$$\min \mathcal{J}(f) \quad \text{where} \quad \mathcal{J}(f) = \frac{1}{2}\| Tf - d\|_2^2 + TV(f),$$

where $TV(f)$ is the discretization of the total variation function

$$\int_0^1 \int_0^1 \left(1 + (\partial_x f)^2 + (\partial_y f)^2\right)^{\frac{1}{2}} \, dx\, dy.$$

- Borzi and Kunish's solid ignition optimal control (Opt):

$$\min_f \mathcal{J}(u(f), f) = \int_{S_2} (u - z)^2 + \frac{\beta}{2} \int_{S_2} (e^u - e^z)^2 + \frac{\nu}{2} \int_{S_2} f^2,$$

where

$$\begin{aligned} -\Delta u + \delta e^u &= f \quad \text{in } S_2, \\ u &= 0 \quad \text{on } \partial S_2. \end{aligned}$$

Test-problems

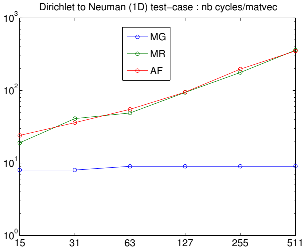
- A nonconvex optimization problem (NC)

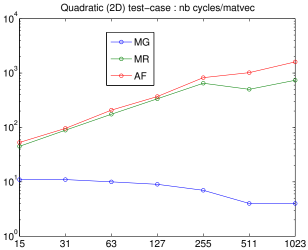
$$\min_{u,\gamma} \mathcal{J}(u,\gamma) = \int_{S_2} (u-u_0)^2 + \int_{S_2} (\gamma-\gamma_0)^2 + \int_{S_2} f^2,$$

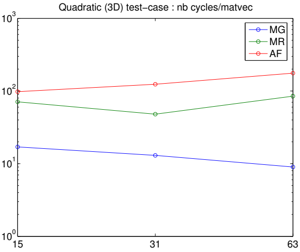
where

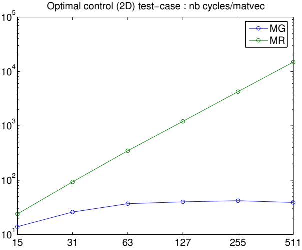
$$\begin{array}{rcll} -\Delta u + \gamma u - f_0 & = & f & \text{in } S_2, \\ u & = & 0 & \text{on } \partial S_2, \end{array}$$

Some treated problems: number of cycles

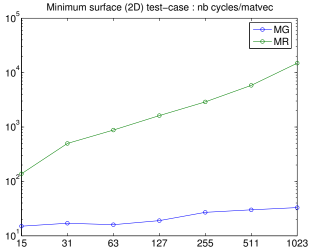


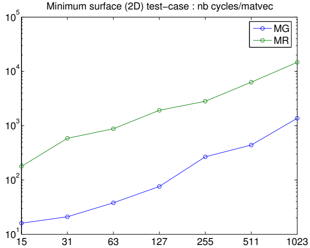


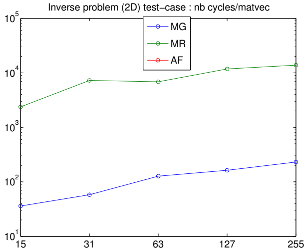


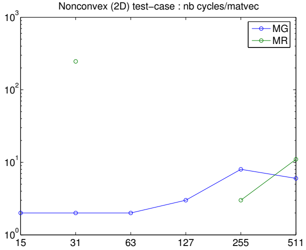


Some treated problems: number of cycles









Results for 4D Var

- Improvements of the incremental 4D VAR
  - Stopping criterion based on the energy norm of the error.
  - Comparison of deflation, spectral and BFGS preconditioner.
  - Preliminary tests show weakness of spectral compared to deflation and L-BFGS in a data assimilation experiment.
- Work on new algorithms
  - Use multigrid techniques and trust-region mechanism.
  - Globally convergent multigrid algorithm for optimization proposed.
  - Encouraging results on academic test-cases are presented : multigrid behaviour of the solution methods.
  - Extension to a Saint Venant system under study.