Numerical Methods for PDE-Constrained Optimization

Richard H. Byrd [1] *Frank E. Curtis* [2] Jorge Nocedal [2]

[1]University of Colorado at Boulder

[2]Northwestern University

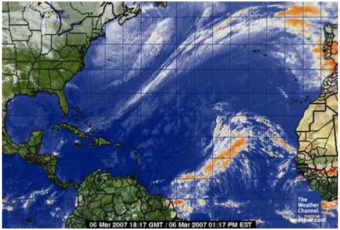
Courant Institute of Mathematical Sciences, 2007

Outline

Outline

Data assimilation in weather forecasting

- Goal: up-to-date global weather forecast for the next 7 to 10 days [1]



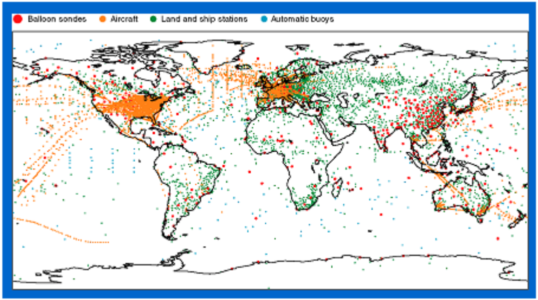
- If an entire *initial state* of the atmosphere (temperatures, pressures, wind patterns, humidities) were known at a certain point in time, then an accurate forecast could be obtained by integrating atmospheric model equations forward in time
- Flow described by Navier-Stokes and further sophistications of atmospheric physics and dynamics (none of which will be discussed here)

[1] (Fisher, Nocedal, Trémolet, and Wright, 2007)

In reality: partial information

Limited amount of data (satellites, buoys, planes, ground-based sensors)

- Each observation is subject to error
- Nonuniformly distributed around the globe (satellite paths, densely-populated areas)

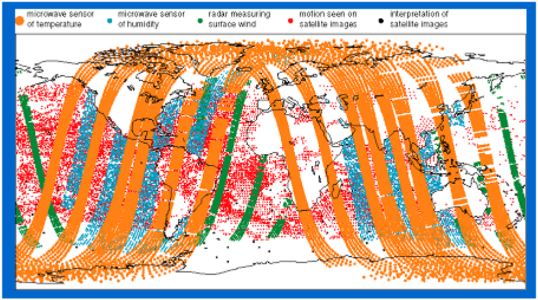


(Graphics courtesy Yannick Trémolet)

In reality: partial information

Limited amount of data (satellites, buoys, planes, ground-based sensors)

- Each observation is subject to error
- Nonuniformly distributed around the globe (satellite paths, densely-populated areas)



(Graphics courtesy Yannick Trémolet)

Data assimilation: defining the unknowns

Currently in operational use at the European Centre for Medium-Range Weather Forecasts (ECMWF)

- We want values for an initial state, call it x^0
- For a given x^0, we could integrate our atmospheric models forward to forecast the state of the atmosphere at N time points

  $$x^i = \mathcal{M}(x^{i-1}), \; i = 1, \ldots, N$$

  (x^i: state of the atmosphere at time i)
- Observe the atmosphere at these N time points

  $$y^1, \ldots, y^N$$

  (y^i: observed state at time i)
- Let x^b (background state) be values at initial time point obtained from previous forecast — carry over old information

Data assimilation as an optimization problem

Define the difference

$$f(x^0) = \begin{bmatrix} x^0 - x^b \\ x^1 - y^1 \\ \vdots \\ x^N - y^N \end{bmatrix}$$

and choose x^0 as the initial state "most likely" to have given the observed data

$$\min_{x^0} \tfrac{1}{2}\|f(x^0)\|^2_{R^{-1}} = \tfrac{1}{2}(x^0 - x^b)^T (R^b)^{-1}(x^0 - x^b) + \tfrac{1}{2}\sum_{i=0}^{N}(x^i - y^i)^T (R^i)^{-1}(x^i - y^i)$$

- $\frac{1}{2}\|f(x^0)\|^2_{R^{-1}}$: distance measure between observed and expected behavior
- $R = (R^b, R^1, \ldots, R^N)$: background and observation error covariance matrices (choice of these values is a separate, but important issue)
- In current forecasts, x^0 contains approximately 3×10^8 unknowns

Outline

Computational issues

Data assimilation optimization problem:

$$\min_{x^0} \tfrac{1}{2}\|f(x^0)\|^2_{R^{-1}} = \tfrac{1}{2}(x^0 - x^b)^T (R^b)^{-1} (x^0 - x^b) + \tfrac{1}{2}\sum_{i=0}^{N}(x^i - y^i)^T (R^i)^{-1}(x^i - y^i)$$

Difficulties include:

- problem is very large ($|x^0| \approx 3 \times 10^8$)
- objective is nonconvex (nonlinear operators $\mathcal{M}^i$)
- exact derivative information not available
- solutions needed in real-time

Current algorithm: nonlinear elimination

Given a guess of the initial state x^0

1. Apply operators $\mathcal{M}^i$ to compute the expected state at the N time points

$$x^i = \mathcal{M}(x^{i-1}),\ i = 1, \ldots, N$$

   and evaluate the objective $\frac{1}{2}\|f(x^0)\|^2_{R^{-1}}$

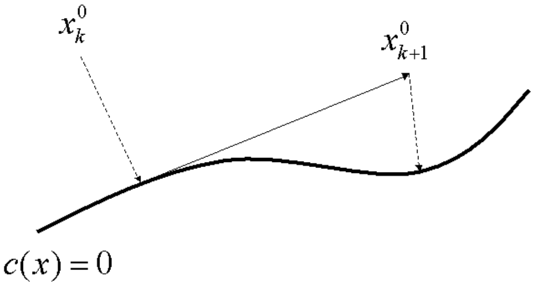
2. Compute a step d toward an improved solution
   a. ... derive sensitivities of $x^1, \ldots, x^N$ with respect to x^0 to form

$$J(x^0): \quad \text{Jacobian of } f(x^0)$$

   (Note: can only be done inexactly)
   b. ... solve the quadratic problem

$$\min_d \ \frac{1}{2}\|f(x^0) + J(x^0)d\|^2_{R^{-1}}$$

Illustration of solution process



Recent advances: multilevel schemes

We are interested in solving the subproblem

$$\min_{d} \; q(d) = \tfrac{1}{2}\|f(x^0) + J(x^0)d\|_{R^{-1}}^2$$

- Conjugate Gradients or Lanczos method is applicable, but not at high resolutions
- Define a restriction operator S such that

$$\hat{x}^i = Sx^i, \; i = 0, \ldots, N$$

- Solve

$$\min_{\hat{d}} \; \hat{q}(\hat{d})$$

  where $\hat{q}$ is a lower-resolution analog of q
- Use a prolongation operator S^+ to map the computed step into the higher-resolution space

$$x^0 \leftarrow x^0 + S^+\hat{d}$$

Future study: "weak constraints"

- Previous formulation assumes that model errors can be neglected
- However, we can lift this questionable assumption by imposing "weak constraints"

$$x^i \approx \mathcal{M}(x^{i-1}),\ i = 1, \ldots, N$$

- These "approximate" equalities can be imposed by creating a penalty term in the objective

$$\min \tfrac{1}{2}(x^0 - x^b)^T (R^b)^{-1}(x^0 - x^b) + \tfrac{1}{2}\sum_{i=0}^{N}(x^i - y^i)^T (R^i)^{-1}(x^i - y^i)$$
$$+ \tfrac{1}{2}\sum_{i=1}^{N}(x^i - \mathcal{M}(x^{i-1}))^T (Q^i)^{-1}(x^i - \mathcal{M}(x^{i-1}))$$

  where Q^i are model error covariances
- Note: state vectors $x^1, \ldots, x^N$ become unknowns; problem dimension can increase from 3×10^8 to as large as 7×10^9!

Future study: "weak constraints"

The "weak constraint" formulation

$$\min \ \tfrac{1}{2}(x^0 - x^b)^T (R^b)^{-1}(x^0 - x^b) + \tfrac{1}{2}\textstyle\sum_{i=0}^{N}(x^i - y^i)^T (R^i)^{-1}(x^i - y^i)$$
$$+ \tfrac{1}{2}\textstyle\sum_{i=1}^{N}(x^i - \mathcal{M}(x^{i-1}))^T (Q^i)^{-1}(x^i - \mathcal{M}(x^{i-1}))$$

can be seen as a step toward constrained optimization

$$\min_{x^0,\ldots,x^N} \ \tfrac{1}{2}(x^0 - x^b)^T (R^b)^{-1}(x^0 - x^b) + \tfrac{1}{2}\textstyle\sum_{i=0}^{N}(x^i - y^i)^T (R^i)^{-1}(x^i - y^i)$$
$$\text{s.t. } x^i = \mathcal{M}(x^{i-1}), \ i = 1, \ldots, N$$

(Note: in solving the constrained formulation, constraints are satisfied only in the limit)

Challenges in PDE-Constrained Optimization

Optimization Method	Multilevel Schemes
Nonlinear elimination Composite-step methods Full space methods	Compute steps at low resolution Optimize at low resolution
DO versus *OD*	Inexactness
Discretize-Optimize Optimize-Discretize	Efficiency Robustness

Overview of important *optimization* questions

Challenges in PDE-Constrained Optimization

Optimization Method	Multilevel Schemes
Nonlinear elimination	Compute steps at low resolution
Composite-step methods	Optimize at low resolution
Full space methods	

DO versus *OD*	Inexactness
Discretize-Optimize	Efficiency
Optimize-Discretize	Robustness

Methods arising in the data assimilation problem

Challenges in PDE-Constrained Optimization

Optimization Method	Multilevel Schemes
Nonlinear elimination	Compute steps at low resolution
Composite-step methods	Optimize at low resolution
Full space methods	

DO versus *OD*	Inexactness
Discretize-Optimize	Efficiency
Optimize-Discretize	Robustness

Our contribution to the field...

Computational savings with inexact methods

Inexact step computations are

- … effective for unconstrained optimization [2]

$$\min_{x \in \mathbb{R}^n} f(x)$$

- … effective for systems of nonlinear equations [3]

$$F(x) = 0 \qquad \text{or} \qquad \min_{x \in \mathbb{R}^n} \|F(x)\|$$

- … *necessary* for many PDE-constrained problems

[2](Steihaug, 1983)
[3](Dembo, Eisenstat, Steihaug, 1982)

Outline

Equality Constrained Optimization

Minimize an objective subject to mathematical equalities

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$\text{s.t. } c(x) = 0$$

(where the constraints $c(x) = 0$ contain the PDE)

Characterizing Optimal Solutions

Defining

$$\mathcal{L}(x,\lambda) = f(x) + \lambda^T c(x)$$

and then the derivatives

$$\begin{aligned} g(x) &= \nabla f(x) \\ \text{and } A(x) &= [\nabla c^i(x)], \end{aligned}$$

we have the first order optimality conditions

$$\begin{bmatrix} g(x) + A(x)^T \lambda \\ c(x) \end{bmatrix} = 0$$

Method of choice: Sequential Quadratic Programming (SQP)

At a given iterate x_k, formulate the quadratic subproblem

$$\min_{d\in\mathbb{R}^n} f_k + g_k^T d + \frac{1}{2} d^T W_k d$$
$$\text{s.t. } c_k + A_k d = 0$$

for some symmetric *positive definite* W_k $(\approx \nabla^2_{xx}\mathcal{L}_k)$ to compute a step, or, equivalently, solve the primal-dual system

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

Measure progress toward a solution with the "merit function"

$$\phi_\pi(x) = f(x) + \pi\|c(x)\|, \qquad \pi > 0$$

SQP:

for $k = 1, 2, \dots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

(It works!)

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

SQP:

for $k = 1, 2, \ldots$

Compute step:

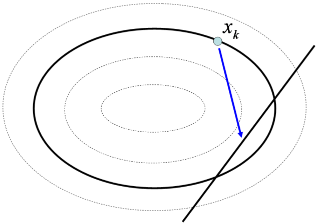
$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor



SQP:

for $k = 1, 2, \dots$

Compute step:

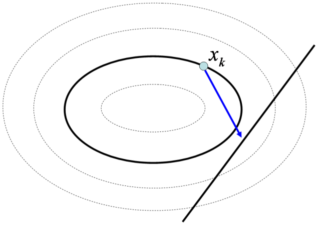
$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor



SQP:

for $k = 1, 2, \ldots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

- The step is good; $D\phi_{\pi_k}(d_k) < 0$
- Algorithm is well-defined; $\alpha_k > 0$
- π will stabilize
- reducing ϕ_π drives search toward solution of optimization problem

Outline

Inexactness is necessary

- An SQP algorithm requires the exact solution of the system

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$

  However, for many PDE constrained problems we cannot form or factor this iteration matrix

- Alternatively, we can consider the application of an iterative solver which, during each *inner iteration*, yields

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

  with residuals (ρ_k, r_k)

- Iterative methods only require mechanisms for computing products with W_k and A_k and its transpose, so the algorithm is "matrix-free"

"Inexact" SQP:

Idea: terminate when $\|(\rho_k, r_k)\|$ is "small"

for $k = 1, 2, \ldots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

"Inexact" SQP:

Idea: terminate when $\|(\rho_k, r_k)\|$ is "small"

for $k = 1, 2, \ldots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = -\begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

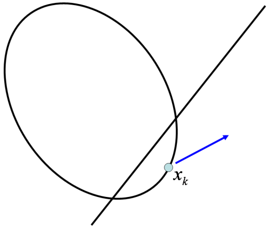
Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

For *any* level of inexactness:



"Inexact" SQP:

Idea: terminate when $\|(\rho_k, r_k)\|$ is "small"

for $k = 1, 2, \ldots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

For *any* level of inexactness:

- Step may be an ascent direction for ϕ_π
- Penalty parameter may tend to ∞

"Inexact" SQP:

We don't know when to terminate the iterative solver

for $k = 1, 2, \ldots$

Compute step:

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

Set $\pi_k \geq \pi_{k-1}$

Line search:

$$\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$$

endfor

For *any* level of inexactness:

- Step may be an ascent direction for ϕ_π
- Penalty parameter may tend to ∞

Main idea: inexactness based on models

- Modern optimization methods work with *models*
- For the merit function

$$\phi_\pi(x) = f(x) + \pi \|c(x)\|,$$

define a model about x_k:

$$m_\pi(d) = f_k + g_k^T d + \tfrac{1}{2} d^T W_k d + \pi \|c_k + A_k d\|,$$

- For a given d_k, we can estimate the reduction in ϕ_π by evaluating

$$\begin{aligned} mred_\pi(d_k) = &\ m_\pi(0) - m_\pi(d_k) \\ = &\ -g_k^T d_k - \tfrac{1}{2} d_k^T W_k d_k + \pi(\|c_k\| - \|r_k\|) \end{aligned}$$

(recall $r_k = c_k + A_k d_k$)

Termination Tests

We may terminate the iteration on

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

if the component d_k:

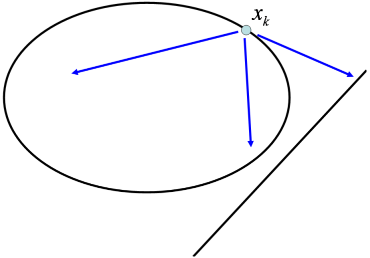
Termination Tests

We may terminate the iteration on

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

if the component d_k: (1) yields a sufficiently large reduction in the model of the merit function for the most recent penalty parameter π_{k-1}; i.e.

$$mred_{\pi_{k-1}}(d_k) \geq \sigma \pi_{k-1} \|c_k\|$$



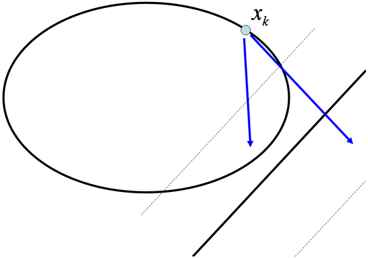
Termination Tests

We may terminate the iteration on

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

if the component d_k: (2) yields a reduction in the linear model of the constraints; i.e., for $\pi_k > \pi_{k-1}$

$$mred_{\pi_k}(d_k) \geq \sigma \pi_k \|c_k\|$$



An Inexact SQP Algorithm (Byrd, Curtis, and Nocedal, 2007)

Given parameters $0 < \epsilon, \tau, \sigma, \eta < 1$ and $0 < \beta$

Initialize x_0, λ_0, and $\pi_{-1} > 0$

for $k = 0, 1, 2, \ldots,$ until convergence

Set $\pi_k = \pi_{k-1}$

Iteratively solve the system

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

Terminate the *inner iterations* when

$$\begin{array}{rcl} mred_{\pi_k}(d_k) & \geq & \sigma \pi_k \|c_k\| \\ \|\rho_k\| & \leq & \max\{\beta\|c_k\|, \epsilon\|g_k + A_k^T \lambda_k\|\} \end{array} \quad \text{or} \quad \begin{array}{rcl} \|r_k\| & \leq & \epsilon\|c_k\| \\ \|\rho_k\| & \leq & \beta\|c_k\| \\ \pi_k & \geq & \frac{g_k^T d_k + \frac{1}{2} d_k^T W_k d_k}{(1-\tau)(\|c_k\| - \|r_k\|)} \end{array}$$

Perform line search $\phi_{\pi_k}(x_k + \alpha_k d_k) \leq \phi_{\pi_k}(x_k) + \eta \alpha_k D\phi_{\pi_k}(d_k)$

Set $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k(d_k, \delta_k)$

endfor

Summary: Our contribution

- SQP is the algorithm of choice for large-scale constrained optimization
- Introducing inexactness is a difficult issue and naïve approaches can fail to ensure convergence
- We present two simple termination tests for the iterative solver, where the level of inexactness is based on models of a merit function
- The algorithm is globally convergent
- Numerical experiments are considered next...

Outline

Question: Is the algorithm robust?

44 problems from standard optimization test sets (CUTEr and COPS)

- Algorithm A: inexactness based on

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} \right\|, \quad 0 < \kappa < 1$$

- Algorithm B: inexactness based on model reductions
  (Note: can have $\kappa \geq 1$!)

Algorithm	Alg. A			Alg. B
κ	2^{-1}	2^{-5}	2^{-10}	—
% Solved	45%	80%	86%	100%

Outline

Question: Is the algorithm practical?

2 model inverse problems in PDE-constrained optimization[4]

$$\min_{(y,z)\in\mathbb{R}^n} \tfrac{1}{2}\|Qz - d\|^2 + \gamma R(y - y_{ref})$$
$$\text{s.t. } \mathcal{A}(y)z = q$$

where

$$\begin{aligned} \text{objective} &= \text{ data fitting } + \text{ regularization} \\ \text{constraints} &= \text{ PDE} \end{aligned}$$

Example applications:

- Elliptic PDE — groundwater modeling, DC resistivity
- Parabolic PDE — optical tomography, electromagnetic imaging

[4](Haber and Hanson, 2007)

Question: Is the algorithm practical?

First problem ($n = 8192, t = 4096$)

- Elliptic PDE (groundwater modeling, DC resistivity)
- . . . *many* details . . .
- Solution obtained in 9 iterations
  Sample output:

$$\left(\text{Note: } \kappa_k = \|(\rho_k, r_k)\| / \|(g_k + A_k^T \lambda_k, c_k)\|\right)$$

k	$\|KKT\|$	Inner Iter.	κ_k	T. Test
3	5.22×10^{-1}	2	9.66×10^{-1}	new π
4	1.11×10^{-1}	10	9.39×10^{-1}	curr. π
5	1.04×10^{-1}	1	9.60×10^{-1}	new π

- Overall, average $\kappa_k \approx 5.40 \times 10^{-1}$

Question: Is the algorithm practical?

Second problem ($n = 69632, t = 65536$)

- Parabolic PDE (optical tomography, electromagnetic imaging)
- . . . *many* details . . .
- Solution obtained in 11 iterations
  Sample output:

$$\left(\text{Note: } \kappa_k = \|(\rho_k, r_k)\| / \|(g_k + A_k^T \lambda_k, c_k)\|\right)$$

k	$\|KKT\|$	Inner Iter.	κ_k	T. Test
4	1.63×10^{-1}	1	9.50×10^{-1}	curr. π
5	1.55×10^{-1}	2	9.55×10^{-1}	curr. π
6	1.48×10^{-1}	3	5.65×10^{-1}	new π

- Overall, average $\kappa_k \approx 4.49 \times 10^{-1}$

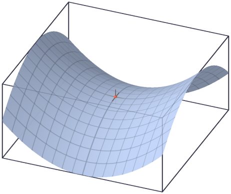
Outline

Convexity assumption

Step computation procedure

$$\min_{d \in \mathbb{R}^n} \; f_k + g_k^T d + \tfrac{1}{2} d^T W_k d \quad \text{s.t.} \; c_k + A_k d = 0$$

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix}$$



Nonconvex optimization

- If W_k is not convex in the null space of the constraints
  - No guarantee of descent
  - Step may be unbounded
- Without a factorization of the primal-dual matrix

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix}$$

  we may not know if the problem is convex or not
- We could always set W_k to a simple positive definite matrix, but then we may be distorting second order information (if available)

First step: recognizing good steps

- If the problem is convex, then a solution to

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} \tag{1}$$

  is a good step
- If the problem is nonconvex, then a solution to (1) may still be a good step
- Idea: characterize d_k based on properties of the decomposition

$$d_k = u_k + v_k \qquad (\text{Note: } \|d_k\|^2 = \|u_k\|^2 + \|v_k\|^2)$$

  where $A_k u_k = 0$ and v_k lies in $range(A_k^T)$

Central claim

A step is good if either termination test for the inexact SQP algorithm is satisfied and for some small $\theta > 0$ we have

$$(a)\ \theta\|u_k\| \leq \|v_k\| \quad \text{or} \quad (b)\ d_k^T W_k d_k \geq \theta\|u_k\|^2$$

Notice that

(a) implies that the step d_k is sufficiently parallel to v_k

(b) implies that the curvature is sufficiently positive along d_k

Estimating properties of the step

Observing

$$\|v_k\| \geq \|A_k v_k\|/\|A_k\| = \|A_k d_k\|/\|A_k\|,$$

we find

$$\theta\|d_k\| \leq \|A_k d_k\|/\|A_k\| \quad \Rightarrow \quad \theta\|u_k\| \leq \|v_k\|$$

and

$$d_k^T W_k d_k \geq \theta(\|d_k\|^2 - \|A_k d_k\|^2/\|A_k\|^2) \quad \Rightarrow \quad d_k^T W_k d_k \geq \theta\|u_k\|^2$$

Proposed Algorithm

Apply an iterative solver to the primal-dual system

- Stop if a termination test is satisfied and
  - ... step is sufficiently parallel to v_k

$$\theta \|d_k\| \leq \|A_k d_k\| / \|A_k\|$$

  - ... or curvature is sufficiently positive

$$d_k^T W_k d_k \geq \theta \left(\|d_k\|^2 - \|A_k d_k\|^2 / \|A_k\|^2 \right)$$

- If $\theta \|d_k\| > \|A_k d_k\| / \|A_k\|$, then set $W_k \leftarrow \tilde{W}_k$ to satisfy

$$\theta \left(\|d_k\|^2 - \|A_k d_k\|^2 / \|A_k\|^2 \right) \leq d_k^T \tilde{W}_k d_k$$

  and iterate on perturbed system

Conclusion

We have:

1. Described an important and challenging sample problem in PDE-constrained optimization
2. Described a globally convergent algorithm in the framework of a powerful algorithm: sequential quadratic programming
3. The algorithm has shown to be robust and efficient for realistic applications
4. Described some ideas for extending the approach to nonconvex problems