An Inexact Newton Method for Large-Scale Nonlinear Optimization

Frank E. Curtis, Lehigh University

including joint work with
Richard H. Byrd, University of Colorado at Boulder
Jorge Nocedal, Northwestern University
Olaf Schenk, University of Basel
Andreas Wächter, IBM Research

Department of Mathematics and Statistics Seminar
University of Guelph

November 12, 2010

Outline

Outline

PDE-Constrained Optimization

An Inexact Newton Method

Numerical Results

Summary and Future Work

Large-scale constrained optimization

Consider large-scale problems of the form

$$\begin{aligned} \min\ & f(x) \\ \text{s.t. } & c^{\mathcal{E}}(x) = 0 \\ & c^{\mathcal{I}}(x) \leq 0. \end{aligned}$$

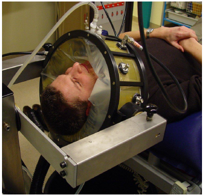
For example, an active area of research:

- True problem of interest is infinite-dimensional;
- Equality constraints include a discretized PDE;
- Often, $x = (u, y)$ is composed of controls u and states y.

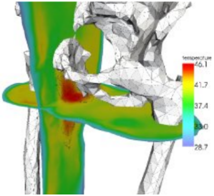
(For the most part, I focus on only having equality constraints.)

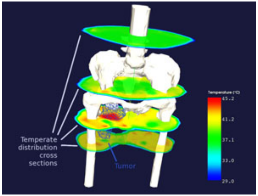
Motivating example 1: Hyperthermia treatment

- Regional hyperthermia is a cancer therapy that aims at heating large and deeply seated tumors by means of radio wave adsorption.

- Computer modeling and numerical optimization can be used to plan the therapy to heat the tumor while minimizing damage to nearby cells.





Hyperthermia treatment as an optimization problem

Optimization problem is to

$$\min_{y,u} \int_{\Omega} (y - y_t)^2 dV \quad \text{where} \quad y_t = \begin{cases} 37 & \text{in } \Omega \backslash \Omega_{tumor} \\ 43 & \text{in } \Omega_{tumor} \end{cases}$$

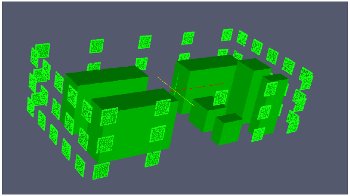
subject to the bio-heat transfer equation (Pennes (1948))

$$- \underbrace{\nabla \cdot (\kappa \nabla y)}_{\text{thermal conductivity}} + \underbrace{\omega(y)\pi(y - y_b)}_{\text{effects of blood flow}} = \underbrace{\frac{\sigma}{2} \left| \sum_i u_i E_i \right|^2}_{\text{electromagnetic field}} \quad \text{in } \Omega$$

and appropriate boundary conditions.

Motivating example 2: Server room cooling

- Heat generating equipment in a server room must be cooled.



- Numerical optimization can be used to help place and control air conditioners to satisfying cooling demands while minimizing costs.

Choosing the search space

Suppose we have distinct controls u and states y:

$$\boxed{\min_{u,y} \; f(u,y) \text{ s.t. } c(u,y) = 0} \Leftrightarrow \boxed{\min_{u} \; f(u,y(u))}$$

Numerical methods generally fall under one of two categories:

- Full-space methods

$$\boxed{\begin{bmatrix} \nabla_u f(u,y) + \langle \nabla_u c(u,y), \lambda \rangle \\ \nabla_y f(u,y) + \langle \nabla_y c(u,y), \lambda \rangle \\ c(u,y) \end{bmatrix} = 0}$$

- Reduced-space methods

$$\boxed{\nabla_u f(u,y) + \langle \nabla_u y(u), \nabla_y f(u,y) \rangle = 0}$$

The latter is often used, but there are benefits in the former.

Choosing when/what to discretize

PDE-constrained optimizers use the phrases:

- Discretize-then-optimize
- Optimize-then-discretize

I prefer:

- Discretize the optimization problem

$$\boxed{\begin{aligned} &\min\ f(x) \\ &\text{s.t. } c(x) = 0 \end{aligned}} \Rightarrow \boxed{\begin{aligned} &\min\ f_h(x) \\ &\text{s.t. } c_h(x) = 0 \end{aligned}}$$

- Discretize the optimality conditions

$$\boxed{\begin{bmatrix} \nabla f(x) + \langle \nabla c(x), \lambda \rangle \\ c(x) \end{bmatrix} = 0} \Rightarrow \boxed{\begin{bmatrix} (\nabla f(x) + \langle \nabla c(x), \lambda \rangle)_h \\ c_h(x) \end{bmatrix} = 0}$$

- Discretize the search direction computation

Computational challenges

We propose a numerical method for large-scale optimization.

- We assume that a full space method is beneficial.
- We assume that we have discretized the optimization problem.

There are numerous computational challenges in such a context.

- Need to avoid storage/factoring of derivatives matrices.
- Need to use iterative in place of direct linear algebra methods.
- Need to control inexactness in computations.
- Need to ensure global convergence.
- Need to handle ill-conditioning, nonconvexity, and inequality constraints.

Strengths and weaknesses

Our methods have numerous strengths:

- It can handle ill-conditioned/rank-deficient and nonconvex problems.
- Inexactness is allowed and controlled with implementable conditions.
- Algorithm is globally convergent, can handle control and state constraints.
- Numerical results are encouraging (but much more to do).

However, we aim to have an algorithm for PDE-constrained optimization, but so far:

- We solve for a single discretization.
- We use finite-dimensional norms.
- Our implementation does not exploit structure.
- We need further experimentation on interesting problems.

Outline

Newton methods

Newton's method for nonlinear equations:

$$\boxed{F(x) = 0} \Rightarrow \boxed{F(x_k) + \nabla F(x_k) d_k = 0}$$

Newton's method for (convex) unconstrained optimization:

$$\boxed{\min_x \ f(x)} \Rightarrow \boxed{\nabla f(x) = 0} \Rightarrow \boxed{\nabla f(x_k) + \nabla^2 f(x_k) d_k = 0}$$

In either case, the main computational effort is to solve a linear system of equations:

$$\boxed{\mathcal{F}(x) = 0} \Rightarrow \boxed{\mathcal{F}(x_k) + \nabla \mathcal{F}(x_k) d_k = 0}$$

Merit function

Is solving the Newton system a useful thing to do?

$$\boxed{\mathcal{F}(x) = 0} \Rightarrow \boxed{\mathcal{F}(x_k) + \nabla\mathcal{F}(x_k)d_k = 0}$$

Judging progress by the merit function

$$\phi(x) := \tfrac{1}{2}\|\mathcal{F}(x)\|^2$$

there is nice consistency between d_k and $\phi(x)$:

$$\nabla\phi(x_k)^T d_k = \mathcal{F}(x_k)^T\nabla\mathcal{F}(x_k)d_k = -\|\mathcal{F}(x_k)\|^2 < 0.$$

That is, d_k is a descent direction for ϕ at x_k.

Inexact Newton methods

Suppose we have a large-scale problem, so we only compute

$$\mathcal{F}(x_k) + \nabla\mathcal{F}(x_k)d_k = r_k$$

where (Dembo, Eisenstat, Steihaug (1982))

$$\|r_k\| \leq \kappa\|\mathcal{F}(x_k)\|, \quad \kappa \in (0,1).$$

Judging progress by the merit function

$$\phi(x) \triangleq \tfrac{1}{2}\|\mathcal{F}(x_k)\|^2$$

there is still consistency between d_k and $\phi(x)$:

$$\nabla\phi(x_k)^T d_k = \mathcal{F}(x_k)^T\nabla\mathcal{F}(x_k)d_k = -\|\mathcal{F}(x_k)\|^2 + \mathcal{F}(x_k)^T r_k \leq (\kappa - 1)\|\mathcal{F}(x_k)\|^2 < 0.$$

Nonconvex or constrained optimization

Everything appears to be the same for nonconvex or constrained problems.

- (Note: Any nonaffine equality constraint means you have a nonconvex problem.)

Consider the equality constrained problem

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

The corresponding Lagrangian is

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

so the first-order optimality conditions are

$$\nabla\mathcal{L}(x,\lambda) = \begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix} \triangleq \mathcal{F}(x,\lambda) = 0.$$

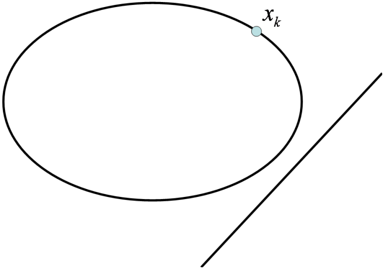
Newton methods and sequential quadratic optimization

If $H(x_k, \lambda_k)$ is positive definite on the null space of $\nabla c(x_k)^T$, then the Newton system

$$\begin{bmatrix} H(x_k,\lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ \delta \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix}$$

is equivalent to the quadratic optimization subproblem

$$\min_{d\in\mathbb{R}^n} \ f(x_k) + \nabla f(x_k)^T d + \tfrac{1}{2} d^T H(x_k,\lambda_k) d$$
$$\text{s.t. } c(x_k) + \nabla c(x_k)^T d = 0.$$



Merit function

The issue (that causes all of our problems later on!) is how to judge progress.

- Simply minimizing

$$\varphi(x,\lambda) = \tfrac{1}{2}\|\mathcal{F}(x,\lambda)\|^2 = \tfrac{1}{2}\left\|\begin{bmatrix}\nabla f(x) + \nabla c(x)\lambda \\ c(x)\end{bmatrix}\right\|^2$$

  is generally inappropriate for constrained optimization.

- Standard practice is to instead use the merit/penalty function

$$\phi(x;\nu) \triangleq f(x) + \nu\|c(x)\|$$

  where ν is a penalty parameter.

Minimizing a penalty function

Consider the penalty function for

$$\min\ (x-1)^2,\ \text{s.t. } x=0 \quad \text{i.e.} \quad \phi(x;\nu)=(x-1)^2+\nu|x|$$

for different values of the penalty parameter ν:

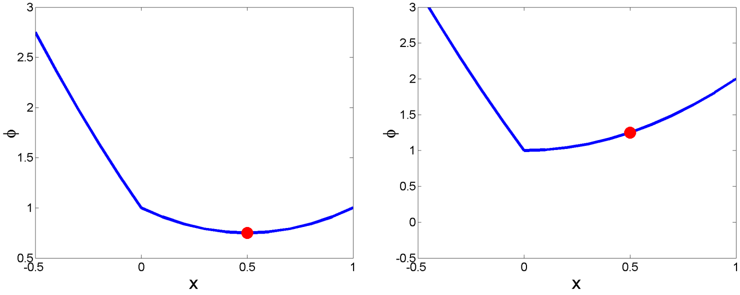


Figure: $\nu = 1$ Figure: $\nu = 2$

Algorithm 0: Newton method for constrained optimization

(Assume the constraints are regular and the Hessians are sufficiently convex)
for $k = 0, 1, 2, \ldots$

- Solve the primal-dual (Newton) equations

$$\begin{bmatrix} H(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix}.$$

- Increase ν, if necessary, so that $\nu_k \geq \|\lambda_k + \delta_k\|$ (yields $D\phi_k(d_k; \nu_k) \ll 0$).
- Backtrack from $\alpha_k \leftarrow 1$ to satisfy the Armijo condition

$$\phi(x_k + \alpha_k d_k; \nu_k) \leq \phi(x_k; \nu_k) + \eta \alpha_k D\phi_k(d_k; \nu_k).$$

- Update iterate $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k(d_k, \delta_k)$.

Convergence of Algorithm 0

Assumption

The sequence $\{(x_k, \lambda_k)\}$ is contained in a convex set Ω over which f, c, and their first derivatives are bounded and Lipschitz continuous. Also,

- *(Regularity) $\nabla c(x_k)^T$ has full row rank with singular values $>$ positive constant;*
- *(Convexity) $u^T H(x_k, \lambda_k) u \geq \mu \|u\|^2$ for $\mu > 0$ for all $u \neq 0$ s.t. $\nabla c(x_k)^T u = 0$.*

Theorem

(Han (1977)) The sequence $\{(x_k, \lambda_k)\}$ yields the limit

$$\lim_{k \to \infty} \left\| \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k) \lambda_k \\ c(x_k) \end{bmatrix} \right\| = 0.$$

Incorporating inexactness

For large-scale problems, we need iterative in place of direct methods, and we need to allow inexactness in our computations.

- Suppose we only compute

$$\begin{bmatrix} H(x_k,\lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}$$

  satisfying

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix} \right\|, \quad \kappa \in (0,1)$$

  as in inexact Newton methods for nonlinear equations.
- Major issue: If κ is not sufficiently small, then d_k may be an ascent direction for our merit function; i.e.,

$$D\phi_k(d_k;\nu_k) > 0 \quad \text{for all } \nu_k \geq \nu_{k-1}.$$

We no longer have nice consistency between our search direction and merit function.

Model reductions

Our first main contribution are a set of implementable conditions that dictate when an inexact solution to the Newton system yields an acceptable search direction so that global convergence can be guaranteed.

Main idea:

- Define the model of $\phi(x;\nu)$:

$$m(d;\nu) \triangleq f(x) + \nabla f(x)^T d + \nu(\|c(x) + \nabla c(x)^T d\|).$$

- d_k is acceptable if

$$\begin{aligned} \Delta m(d_k;\nu_k) &\triangleq m(0;\nu_k) - m(d_k;\nu_k) \\ &= -\nabla f(x_k)^T d_k + \nu_k(\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T d_k\|) \gg 0. \end{aligned}$$

- This ensures $D\phi_k(d_k;\nu_k) \ll 0$ (and more).

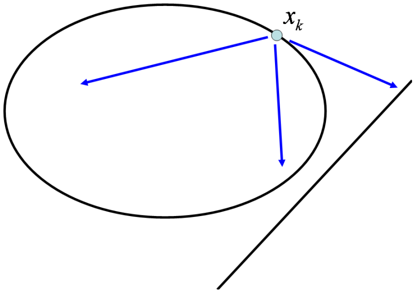
Termination test 1

The search direction (d_k, δ_k) is acceptable if

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix} \right\|, \quad \kappa \in (0,1)$$

and if for $\nu_k = \nu_{k-1}$ and some $\sigma \in (0,1)$ we have

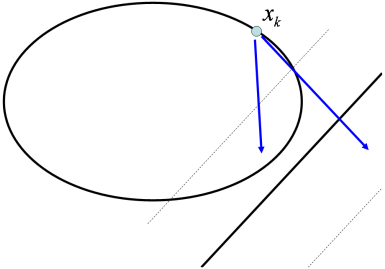
$$\Delta m(d_k; \nu_k) \geq \underbrace{\max\{\tfrac{1}{2} d_k^T H(x_k, \lambda_k) d_k, 0\} + \sigma \nu_k \max\{\|c(x_k)\|, \|r_k\| - \|c(x_k)\|\}}_{\geq 0 \text{ for any } d}$$



Termination test 2

The search direction (d_k, δ_k) is acceptable if

$$\begin{aligned} \|\rho_k\| &\leq \beta\|c(x_k)\|, \quad \beta > 0 \\ \text{and} \quad \|r_k\| &\leq \epsilon\|c(x_k)\|, \quad \epsilon \in (0,1) \end{aligned}$$



Increasing the penalty parameter ν then yields

$$\Delta m(d_k;\nu_k) \geq \underbrace{\max\{\tfrac{1}{2}d_k^T H(x_k,\lambda_k)d_k, 0\} + \sigma\nu_k\|c(x_k)\|}_{\geq 0 \text{ for any } d}$$

Algorithm 1: Inexact Newton for optimization

(Byrd, Curtis, Nocedal (2008))
for $k = 0, 1, 2, \ldots$

- Iteratively solve

$$\begin{bmatrix} H(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix}$$

  until termination test 1 or 2 is satisfied.

- If only termination test 2 is satisfied, increase ν so

$$\nu_k \geq \max\left\{ \nu_{k-1}, \frac{\nabla f(x_k)^T d_k + \max\{\frac{1}{2} d_k^T H(x_k, \lambda_k) d_k, 0\}}{(1-\tau)(\|c(x_k)\| - \|r_k\|)} \right\}.$$

- Backtrack from $\alpha_k \leftarrow 1$ to satisfy

$$\phi(x_k + \alpha_k d_k; \nu_k) \leq \phi(x_k; \nu_k) - \eta \alpha_k \Delta m(d_k; \nu_k).$$

- Update iterate $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k (d_k, \delta_k)$.

Convergence of Algorithm 1

Assumption

The sequence $\{(x_k, \lambda_k)\}$ is contained in a convex set Ω over which f, c, and their first derivatives are bounded and Lipschitz continuous. Also,

- *(Regularity) $\nabla c(x_k)^T$ has full row rank with singular values $>$ positive constant;*
- *(Convexity) $u^T H(x_k, \lambda_k) u \geq \mu \|u\|^2$ for $\mu > 0$ for all $u \neq 0$ s.t. $\nabla c(x_k)^T u = 0$.*

Theorem

(Byrd, Curtis, Nocedal (2008)) The sequence $\{(x_k, \lambda_k)\}$ yields the limit

$$\lim_{k \to \infty} \left\| \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix} \right\| = 0.$$

Handling nonconvexity and rank deficiency

There are two assumptions we aim to drop:

- (Regularity) $\nabla c(x_k)^T$ has full row rank with singular values $>$ positive constant;
- (Convexity) $u^T H(x_k, \lambda_k) u \geq \mu \|u\|^2$ for $\mu > 0$ for all $u \neq 0$ s.t. $\nabla c(x_k)^T u = 0$.

Without them, Algorithm 1 may stall or may not be well-defined.

Our second and third main contributions are extensions to the previous algorithm so that rank deficient and nonconvex problems can also be solved.

No factorizations means no clue

Since we use iterative methods, we do not factor the Newton matrix

$$\begin{bmatrix} H(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix}$$

so we might not know if the problem is nonconvex or ill-conditioned.

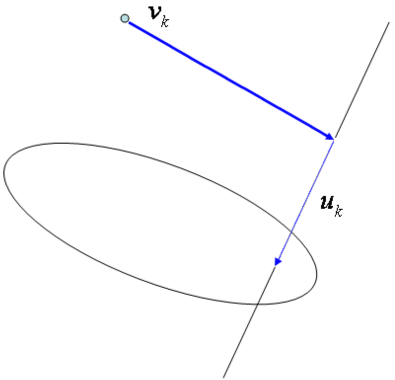
- Common practice is to perturb the matrix to be

  $$\begin{bmatrix} H(x_k, \lambda_k) + \xi_1 I & \nabla c(x_k) \\ \nabla c(x_k)^T & -\xi_2 I \end{bmatrix}$$

  where ξ_1 convexifies the model and ξ_2 regularizes the constraints.
- Poor choices of ξ_1 and ξ_2 can have terrible consequences in the algorithm.

Our approach for global convergence

- Decompose the direction d_k into a normal component (toward the constraints) and a tangential component (toward optimality):



- Without convexity, we do not guarantee a minimizer, but our merit function biases the method to avoid maximizers and saddle points.

Normal component computation

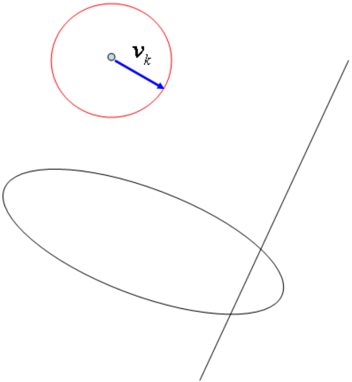
(Approximately) solve, for some $\omega > 0$,

$$\min \tfrac{1}{2}\|c(x_k) + \nabla c(x_k)^T v\|^2$$
$$\text{s.t. } \|v\| \leq \omega\|(\nabla c(x_k))c(x_k)\|$$

We only require Cauchy decrease:

$$\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T v_k\|$$
$$\geq \epsilon_v(\|c(x_k)\| - \|c(x_k) + \alpha \nabla c(x_k)^T \tilde{v}_k\|)$$

for $\epsilon_v \in (0,1)$, where $\tilde{v}_k = -(\nabla c(x_k))c(x_k)$.



Tangential component computation (idea #1)

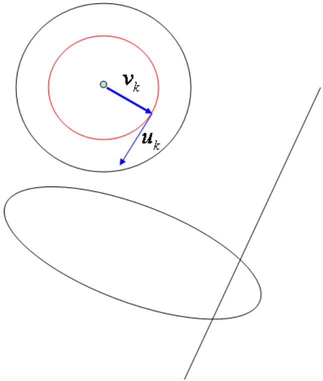
Standard practice is to then (approximately) solve

$$\min\ (\nabla f(x_k) + H(x_k,\lambda_k)v_k)^T u + \tfrac{1}{2} u^T H(x_k,\lambda_k) u$$
$$\text{s.t. } \nabla c(x_k)^T u = 0,\ \|u\| \le \Delta_k.$$

However, maintaining

$$\nabla c(x_k)^T u \approx 0 \quad \text{and} \quad \|u\| \le \Delta_k$$

can be expensive.

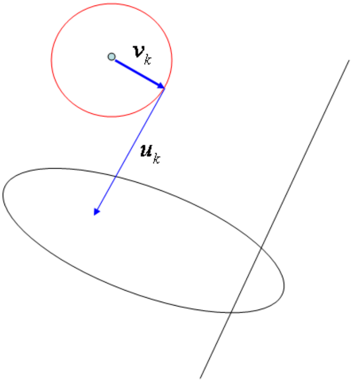


Tangential component computation

Instead, we formulate the primal-dual system

$$\begin{bmatrix} H(x_k,\lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} u_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k + H(x_k,\lambda_k) v_k \\ 0 \end{bmatrix}$$

and apply our ideas from before!



Handling nonconvexity

- Convexify the Hessian as in

$$\begin{bmatrix} H(x_k, \lambda_k) + \xi_1 I & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix}$$

  by monitoring iterates.
- Hessian modification strategy: Increase ξ_1 whenever

$$\begin{aligned} \|u_k\|^2 &> \psi \|v_k\|^2, \quad \psi > 0; \\ \tfrac{1}{2} u_k^T (H(x_k, \lambda_k) + \xi_1 I) u_k &< \theta \|u_k\|^2, \quad \theta > 0. \end{aligned}$$

Inexact Newton Algorithm 2

(Curtis, Nocedal, Wächter (2009))

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

- Approximately solve the following for v_k to obtain Cauchy decrease:

$$\min \ \tfrac{1}{2}\|c(x_k) + \nabla c(x_k)^T v\|^2, \quad \text{s.t. } \|v\| \leq \omega \|(\nabla c(x_k)) c(x_k)\|.$$

- Iteratively solve

$$\begin{bmatrix} H(x_k, \lambda_k) + \xi_1 I & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ -\nabla c(x_k)^T v_k \end{bmatrix}$$

until termination test 1 or 2 is satisfied, increasing ξ_1 as described.

- If only termination test 2 is satisfied, increase ν so

$$\nu_k \geq \max\left\{ \nu_{k-1}, \frac{\nabla f(x_k)^T d_k + \max\{\frac{1}{2} u_k^T (H(x_k, \lambda_k) + \xi_1 I) u_k, \theta \|u_k\|^2\}}{(1-\tau)(\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T d_k\|)} \right\}.$$

- Backtrack from $\alpha_k \leftarrow 1$ to satisfy

$$\phi(x_k + \alpha_k d_k; \nu_k) \leq \phi(x_k; \nu_k) - \eta \alpha_k \Delta m(d_k; \nu_k).$$

- Update iterate $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k (d_k, \delta_k)$.

Convergence of Algorithm 2

Assumption

The sequence $\{(x_k, \lambda_k)\}$ *is contained in a convex set* Ω *over which* f, c, *and their first derivatives are bounded and Lipschitz continuous.*

Theorem

(Curtis, Nocedal, Wächter (2009)) If all limit points of $\{\nabla c(x_k)^T\}$ *have full row rank, then the sequence* $\{(x_k, \lambda_k)\}$ *yields the limit*

$$\lim_{k \to \infty} \left\| \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix} \right\| = 0.$$

Otherwise,

$$\lim_{k \to \infty} \|(\nabla c(x_k)) c(x_k)\| = 0$$

and if $\{\nu_k\}$ *is bounded, then*

$$\lim_{k \to \infty} \|\nabla f(x_k) + \nabla c(x_k)\lambda_k\| = 0.$$

Handling inequalities

- Interior point methods are attractive for large applications.
- Line-search interior point methods that enforce

$$c(x_k) + \nabla c(x_k)^T d_k = 0$$

  may fail to converge globally (Wächter, Biegler (2000)).
- Fortunately, the trust region subproblem we use to regularize the constraints also saves us from this type of failure!

Our fourth main contribution is to extend our techniques to handle inequalities.

Algorithm 2 (Interior-point version)

- Apply Algorithm 2 to the logarithmic-barrier subproblem for $\mu \to 0$:

$$\min \ f(x) - \mu \sum_{i=1}^{q} \ln s^i, \quad \text{s.t. } c_{\mathcal{E}}(x) = 0, \ c_{\mathcal{I}}(x) - s = 0$$

- Define

$$\begin{bmatrix} H(x_k, \lambda_{\mathcal{E},k}, \lambda_{\mathcal{I},k}) & 0 & \nabla c_{\mathcal{E}}(x_k) & \nabla c_{\mathcal{I}}(x_k) \\ 0 & \mu I & 0 & -S_k \\ \nabla c_{\mathcal{E}}(x_k)^T & 0 & 0 & 0 \\ \nabla c_{\mathcal{I}}(x_k)^T & -S_k & 0 & 0 \end{bmatrix} \begin{bmatrix} d_k^x \\ d_k^s \\ \delta_{\mathcal{E},k} \\ \delta_{\mathcal{I},k} \end{bmatrix}$$

so that the iterate update has

$$\begin{bmatrix} x_{k+1} \\ s_{k+1} \end{bmatrix} \leftarrow \begin{bmatrix} x_k \\ s_k \end{bmatrix} + \alpha_k \begin{bmatrix} d_k^x \\ S_k d_k^s \end{bmatrix}.$$

- Incorporate a fraction-to-the-boundary rule in the line search and a slack reset in the algorithm to maintain $s \geq \max\{0, c_{\mathcal{I}}(x)\}$.

Convergence of Algorithm 2 (Interior-point)

Assumption

The sequence $\{(x_k, \lambda_{\mathcal{E},k}, \lambda_{\mathcal{I},k})\}$ is contained in a convex set Ω over which f, $c_{\mathcal{E}}$, $c_{\mathcal{I}}$, and their first derivatives are bounded and Lipschitz continuous.

Theorem

(Curtis, Schenk, Wächter (2009))

- *For a given μ, Algorithm 2 yields the same result as before.*
- *If Algorithm 2 yields a sufficiently accurate solution to the barrier subproblem for each $\{\mu_j\} \to 0$ and if the linear independence constraint qualification (LICQ) holds at a limit point $\bar{x}$ of $\{x_j\}$, then there exist Lagrange multipliers $\bar{\lambda}$ such that the first-order optimality conditions of the nonlinear program are satisfied.*

Outline

Implementation details

- Incorporated in IPOPT software package (Wächter, Laird, Biegler):
  - interior-point algorithm with inexact step computations;
  - flexible penalty function for promoting faster convergence (Curtis, Nocedal);
  - tests on $\sim$ 700 CUTEr problems yields robustness (almost) on par with original IPOPT.
- Linear systems solved with PARDISO (Schenk, Gärtner):
  - includes iterative linear system solvers, e.g., SQMR (Freund);
  - incomplete multilevel factorization with inverse-based pivoting;
  - stabilized by symmetric-weighted matchings.
- Server cooling room example coded w/ libmesh (Kirk, Peterson, Stogner, Carey)

Hyperthermia treatment planning

Let $u_j = a_j e^{i\phi_j}$ and $M_{jk}(x) = \langle E_j(x), E_k(x)\rangle$ where $E_j = \sin(jx_1x_2x_3\pi)$:

$$\min \frac{1}{2}\int_\Omega (y(x) - y_t(x))^2 dx$$
$$\text{s.t.} \begin{cases} -\Delta y(x) - 10(y(x) - 37) = u^* M(x) u & \text{in } \Omega \\ 37.0 \le y(x) \le 37.5 & \text{on } \partial\Omega \\ 42.0 \le y(x) \le 44.0 & \text{in } \Omega_0 \end{cases}$$

Original IPOPT with $N = 32$ requires 408 seconds per iteration.

N	n	p	q	# iter	CPU sec (per iter)
16	4116	2744	2994	68	22.893 (0.3367)
32	32788	27000	13034	51	3055.9 (59.920)

Groundwater modeling

Let $q_i = 100\sin(2\pi x_1)\sin(2\pi x_2)\sin(2\pi x_3)$:

$$\min \frac{1}{2}\int_\Omega (y(x) - y_t(x))^2 dx + \frac{1}{2}\alpha \int_\Omega [\beta(u(x) - u_t(x))^2 + |\nabla(u(x) - u_t(x))|^2] dx$$

$$\text{s.t.} \begin{cases} -\nabla \cdot (e^{u(x)} \cdot \nabla y_i(x)) = q_i(x) & \text{in } \Omega, \quad i = 1, \ldots, 6 \\ \nabla y_i(x) \cdot n = 0 & \text{on } \partial\Omega \\ \int_\Omega y_i(x) dx = 0, & i = 1, \ldots, 6 \\ -1 \le u(x) \le 2 & \text{in } \Omega \end{cases}$$

Original `IPOPT` with $N = 32$ requires 20 hours for the first iteration.

N	n	p	q	# iter	CPU sec (per iter)
16	28672	24576	8192	18	206.416 (11.4676)
32	229376	196608	65536	20	1963.64 (98.1820)
64	1835008	1572864	524288	21	134418. (6400.85)

Server room cooling

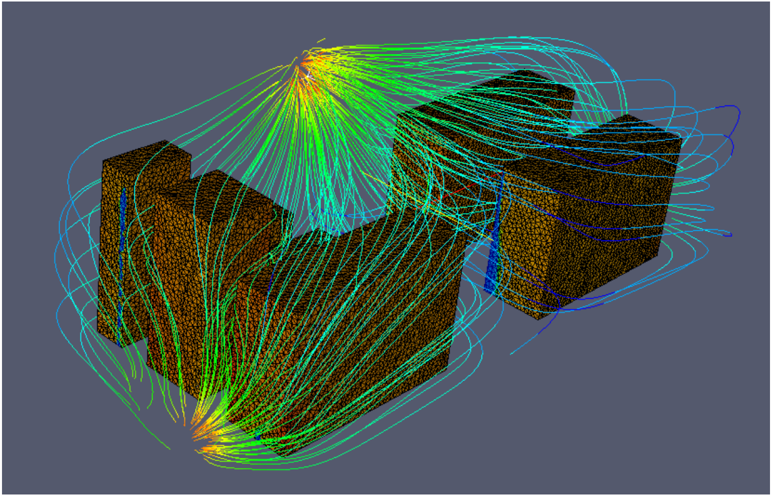
Let $\phi(x)$ be the air flow velocity potential:

$$\min \sum c_i v_{AC_i}$$
$$\text{s.t.} \begin{cases} \nabla\phi(x) = 0 & \text{in } \Omega \\ \partial_n\phi(x) = 0 & \text{on } \partial\Omega_{wall} \\ \partial_n\phi(x) = -v_{AC_i} & \text{on } \partial\Omega_{AC_i} \\ \phi(x) = 0 & \text{in } \Omega_{Exh_j} \\ \|\nabla\phi(x)\|_2^2 \geq v_{min}^2 & \text{on } \partial\Omega_{hot} \\ v_{AC_i} \geq 0 & \end{cases}$$

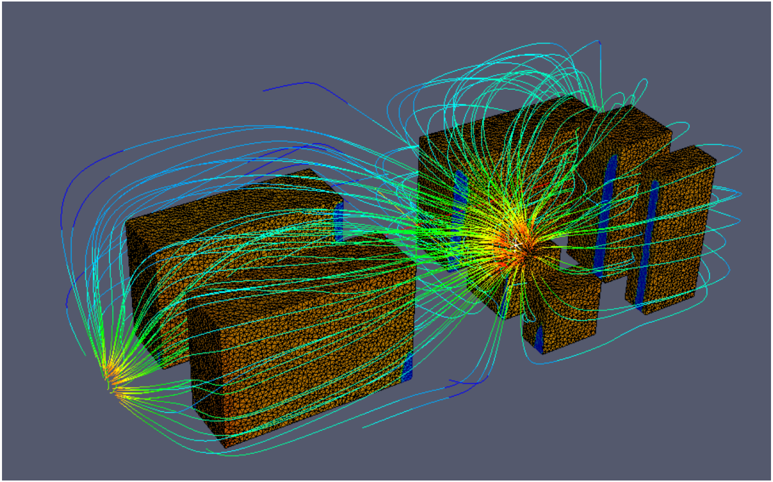
Original `IPOPT` with $h = 0.05$ requires 2390.09 seconds per iteration.

h	n	p	q	# iter	CPU sec (per iter)
0.10	43816	43759	4793	47	1697.47 (36.1164)
0.05	323191	323134	19128	54	28518.4 (528.119)

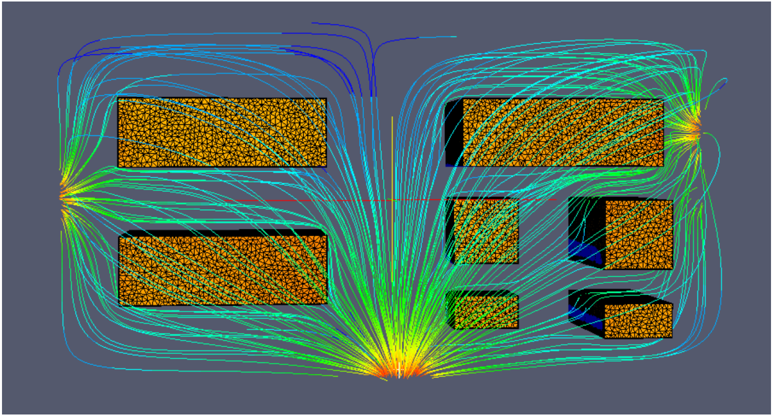
Server room cooling solution



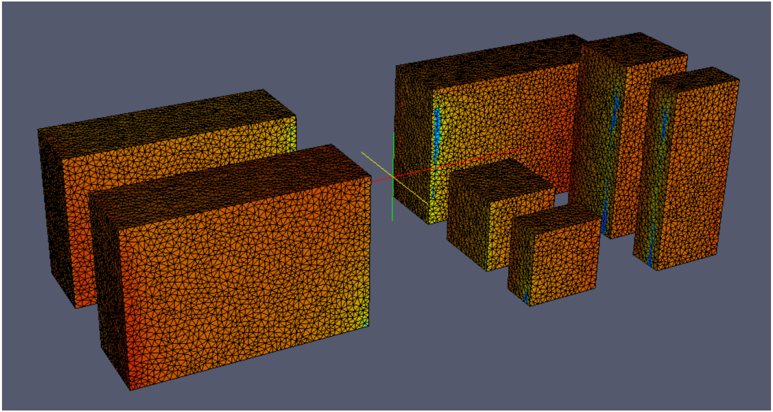
Server room cooling solution



Server room cooling solution



Server room cooling solution (active constraints)



Outline

Summary

We proposed an algorithm for large-scale nonlinear optimization:

- It can handle ill-conditioned/rank-deficient problems.
- It can handle nonconvex problems.
- Inexactness is allowed and controlled with loose conditions.
- The conditions are implementable (in fact, implemented).
- The algorithm is globally convergent.
- It can handle problems with control and state constraints.
- Numerical results are encouraging so far.

Future work and questions

What are we missing (to *really* solve PDE-constrained problems)?

- PDE-specific preconditioners
- Use of appropriate norms
- Mesh refinement, error estimators

What does it take to transform an algorithm for finite-dimensional optimization into one for solving infinite-dimensional problems?

- Can the finite-dimensional solver be a black-box?
- If not, to what extent do the outer and inner algorithms need to be coupled? (Do *all* components of the finite-dimensional solver need to be checked for their effect on the infinite-dimensional problem?)

What interesting problems may be solved with our approach?

References

- "An Inexact SQP Method for Equality Constrained Optimization," R. H. Byrd, F. E. Curtis, and J. Nocedal, *SIAM Journal on Optimization*, Volume 19, Issue 1, pg. 351–369, 2008.
- "An Inexact Newton Method for Nonconvex Equality Constrained Optimization," R. H. Byrd, F. E. Curtis, and J. Nocedal, *Mathematical Programming*, Volume 122, Issue 2, pg. 273–299, 2010.
- "A Matrix-free Algorithm for Equality Constrained Optimization Problems with Rank-Deficient Jacobians," F. E. Curtis, J. Nocedal, and A. Wächter, *SIAM Journal on Optimization*, Volume 20, Issue 3, pg. 1224–1249, 2009.
- "An Interior-Point Algorithm for Large-Scale Nonlinear Optimization with Inexact Step Computations," F. E. Curtis, O. Schenk, and A. Wächter, to appear in *SIAM Journal on Scientific Computing*.