An Inexact Newton Method for Nonlinear Constrained Optimization

Frank E. Curtis

New York University

Numerical Analysis Seminar, January 23, 2009

Outline

Motivation and background

Algorithm development and theoretical results

Experimental results

Conclusion and final remarks

Outline

Very large-scale optimization

- Consider a constrained optimization problem (NLP) of the form

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \ & f(x) \\ \text{s.t. } & c_{\mathcal{E}}(x) = 0 \\ & c_{\mathcal{I}}(x) \geq 0 \end{aligned}$$

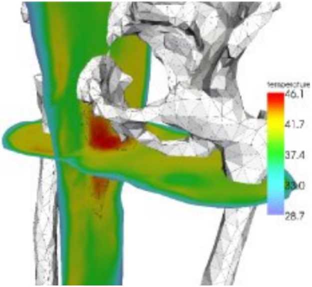
  where $f : \mathbb{R}^n \to \mathbb{R}$, $c_{\mathcal{E}} : \mathbb{R}^n \to \mathbb{R}^p$ and $c_{\mathcal{I}} : \mathbb{R}^n \to \mathbb{R}^q$ are smooth functions

- We are interested in problems for which the best contemporary methods, i.e.,
  - penalty methods
  - sequential quadratic programming
  - interior-point methods

  cannot be employed due to problem size

Hyperthermia treatment planning

Regional hyperthermia is a cancer therapy that aims at heating large and deeply seated tumors by means of radio wave adsorption, which results in the killing of tumor cells and makes them more susceptible to accompanying radio or chemotherapy.



See http://www.youtube.com/watch?v=jF-nm8fi3oo

Hyperthermia treatment planning as a NLP

The problem can be formulated as

$$\min_{y,u} \int_\Omega y(x) - y_t(x)dx \quad \text{where} \quad y_t(x) = \begin{cases} 37 & \text{in } \Omega\backslash\Omega_0 \\ 43 & \text{in } \Omega_0 \end{cases}$$

subject to the bio-heat transfer equation[1]

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

and the bound constraints

$$\begin{aligned} 37.0 \le y(x) \le 37.5, &\quad \text{on } \partial\Omega \\ 41.0 \le y(x) \le 45.0, &\quad \text{in } \Omega_0 \end{aligned}$$

where Ω_0 is the tumor domain

[1] Pennes (1948)

Newton's method

- An approach that sounds ideal for, say, the equality constrained problem

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{s.t. } c(x) = 0 \tag{1.1}$$

would be a Newton method

- The Lagrangian for (1.1) is given by

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

so if f and c are differentiable, the first-order optimality conditions are

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

(A single system of nonlinear equalities.)

Inexact Newton methods

- A Newton method for the nonlinear system of equations

$$\mathcal{F}(x) = 0$$

has the iteration

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

- Applying an iterative linear system solver to this system yields

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

and if progress is judged by the *merit function*

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

then the (inner) iteration may be terminated as soon as

$$\|r_k\| \le \kappa_k\|\mathcal{F}(x_k)\|$$

(for superlinear convergence, choose $\kappa_k \to 0$)[2]

[2] Dembo, Eisenstat, Steihaug (1982)

A naïve Newton method for NLP

- Consider the problem

$$\min \; f(x) = x_1 + x_2, \quad \text{s.t. } c(x) = x_1^2 + x_2^2 - 1 = 0$$

that has the first-order optimality conditions

$$\mathcal{F}(x,\lambda) = \begin{bmatrix} 1 + 2x_1\lambda \\ 1 + 2x_2\lambda \\ x_1^2 + x_2^2 - 1 \end{bmatrix} = 0$$

- A Newton method applied to this problem yields

k	$\frac{1}{2}\|\mathcal{F}(x_k,\lambda_k)\|^2$
0	+3.5358e+00
1	+2.9081e-02
2	+4.8884e-04
3	+7.9028e-08
4	+2.1235e-15

A naïve Newton method for NLP

- Consider the problem

$$\min\ f(x) = x_1 + x_2, \quad \text{s.t. } c(x) = x_1^2 + x_2^2 - 1 = 0$$

that has the first-order optimality conditions

$$\mathcal{F}(x, \lambda) = \begin{bmatrix} 1 + 2x_1\lambda \\ 1 + 2x_2\lambda \\ x_1^2 + x_2^2 - 1 \end{bmatrix} = 0$$

- A Newton method applied to this problem yields

k	$\frac{1}{2}\|\mathcal{F}(x_k, \lambda_k)\|^2$
0	+3.5358e+00
1	+2.9081e-02
2	+4.8884e-04
3	+7.9028e-08
4	+2.1235e-15

k	$f(x_k)$	$\|c(x_k)\|$
0	+1.3660e+00	+1.1102e-16
1	+1.3995e+00	+8.3734e-03
2	+1.4358e+00	+3.0890e-02
3	+1.4143e+00	+2.4321e-04
4	+1.4142e+00	+1.7258e-08

Our goal

- Our goal is to design an inexact Newton method for constrained optimization
- Such a framework is important so the user will be able to balance the computational costs between outer (optimization) and inner (step computation) iterations for their own applications
- The method must always remember that an optimization problem is being solved
- Many of the usual techniques in Newton-like methods for optimization for handling
  - non-convex objective and constraint functions
  - (near) rank deficiency of the constraint Jacobian
  - inequality constraints

  are not applicable here! (We need to get creative.)

Outline

A merit function for optimization

- The failure in applying a standard (inexact) Newton method to an optimization problem is due to the choice of merit function! That is,

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

  is inappropriate for purposes of optimization

- Our method will be centered around the merit function

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

  (known as an exact penalty function)

Algorithm 0: Newton method for optimization

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

- Evaluate $f(x_k)$, $\nabla f(x_k)$, $c(x_k)$, $\nabla c(x_k)$, and $\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)$
- Solve the primal-dual equations

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix}$$

- Increase the penalty parameter, if necessary, so that $D\phi_k(d_k; \pi_k) \ll 0$
- Perform a line search to find $\alpha_k \in (0, 1]$ satisfying the Armijo condition

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

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

Newton methods and sequential quadratic programming

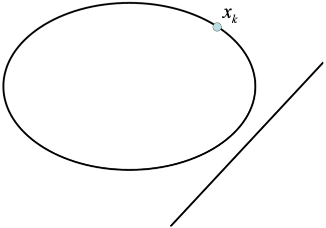
The primal-dual system

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x,\lambda) & \nabla c(x) \\ \nabla c(x)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ \delta \end{bmatrix} = - \begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix}$$

is equivalent to the SQP subproblem

$$\min_{d \in \mathbb{R}^n} \; f(x) + \nabla f(x)^T d + \tfrac{1}{2} d^T (\nabla^2_{xx}\mathcal{L}(x,\lambda)) d$$
$$\text{s.t. } c(x) + \nabla c(x)^T d = 0$$

if $\nabla^2_{xx}\mathcal{L}(x,\lambda)$ *is positive definite on the null space of* $\nabla c(x)^T$



Minimizing an exact penalty function

Consider the exact penalty function for

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

for different values of the penalty parameter π

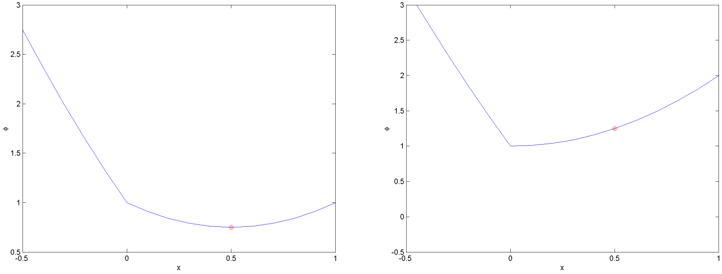


Figure: $\pi = 1$ Figure: $\pi = 2$

Convergence of Algorithm 0

Assumption

The sequence $\{(x_k, \lambda_k)\}$ *is contained in a convex set* Ω *and*

- f, c, *and their first derivatives are bounded and Lipschitz continuous on* Ω
- *(Regularity)* $\nabla c(x_k)^T$ *has full row rank with smallest singular value bounded below by a positive constant*
- *(Convexity)* $u^T(\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k))u \geq \mu\|u\|^2$ *for* $\mu > 0$ *for all* $u \in \mathbb{R}^n$ *satisfying* $u \neq 0$ *and* $\nabla c(x_k)^T u = 0$

Theorem

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

- An inexact solution to the primal-dual equations; i.e.,

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x,\lambda) & \nabla c(x) \\ \nabla c(x)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ \delta \end{bmatrix} = -\begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix} + \begin{bmatrix} \rho \\ r \end{bmatrix}$$

  may yield $D\phi_k(d_k;\pi_k) > 0$ for any $\pi_k \geq \pi_{k-1}$, even if for some $\kappa \in (0,1)$ we have

$$\left\| \begin{bmatrix} \rho \\ r \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix} \right\|$$

  (the standard condition required in inexact Newton methods)

- *It is the merit function that tells us what steps are appropriate, and so it should be what tells us which inexact solutions to the Newton (primal-dual) system are acceptable search directions*

Model reductions

- Modern optimization techniques focus on models of the problem functions
- We apply an iterative solver to the primal-dual system

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x,\lambda) & \nabla c(x) \\ \nabla c(x)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ \delta \end{bmatrix} = - \begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix}$$

(since an exact solution will produce an acceptable search direction)

- A search direction is deemed acceptable if the improvement in the model

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

is sufficiently large; i.e., if we have

$$\begin{aligned} \Delta m(d;\pi) \triangleq\ & m(0;\pi) - m(d;\pi) \\ =\ & -\nabla f(x)^T d + \pi(\|c(x)\| - \|c(x) + \nabla c(x)^T d\|) \gg 0 \end{aligned}$$

Termination test 1

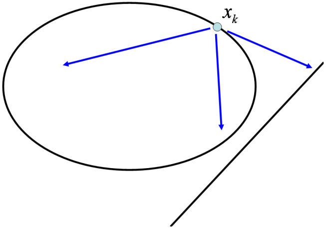
An inexact solution (d, δ) to the primal-dual system is acceptable if

$$\left\| \begin{bmatrix} \rho \\ r \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} \nabla f(x) + \nabla c(x)\lambda \\ c(x) \end{bmatrix} \right\|$$

for some $\kappa \in (0, 1)$ and if

$$\Delta m(d;\pi) \geq \max\{\tfrac{1}{2} d^T(\nabla^2_{xx}\mathcal{L}(x,\lambda))d, 0\} + \sigma\pi \max\{\|c(x)\|, \|r\| - \|c(x)\|\}$$

for some $\sigma \in (0, 1)$

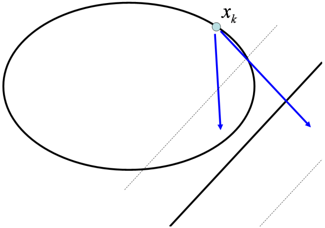


Termination test 2

An inexact solution to the primal-dual system is acceptable if

$$\begin{aligned} \|\rho\| &\leq \beta\|c(x)\| \\ \|r\| &\leq \epsilon\|c(x)\| \end{aligned}$$

for some $\beta > 0$ and $\epsilon \in (0, 1)$



Increasing the penalty parameter can then yield

$$\Delta m(d;\pi) \geq \max\{\tfrac{1}{2}d^T(\nabla^2_{xx}\mathcal{L}(x,\lambda))d, 0\} + \sigma\pi\|c(x)\|$$

Algorithm 1: Inexact Newton for optimization

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

- Evaluate $f(x_k)$, $\nabla f(x_k)$, $c(x_k)$, $\nabla c(x_k)$, and $\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)$
- Iteratively solve the primal-dual equations

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ c(x_k) \end{bmatrix}$$

  until termination test 1 or 2 is satisfied

- If only termination test 2 is satisfied, increase the penalty parameter so

$$\pi_k \geq \frac{\nabla f(x_k)^T d_k + \max\{\frac{1}{2} d_k^T (\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)) d_k, 0\}}{(1 - \tau)(\|c(x_k)\| - \|r_k\|)}$$

- Perform a line search to find $\alpha_k \in (0, 1]$ satisfying the sufficient decrease condition

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

- 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 *and*

- f, c, *and their first derivatives are bounded and Lipschitz continuous on* Ω
- *(Regularity)* $\nabla c(x_k)^T$ *has full row rank with smallest singular value bounded below by a positive constant*
- *(Convexity)* $u^T(\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k))u \geq \mu\|u\|^2$ *for* $\mu > 0$ *for all* $u \in \mathbb{R}^n$ *satisfying* $u \neq 0$ *and* $\nabla c(x_k)^T u = 0$

Theorem

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

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

Take-away idea #1

In a (inexact) Newton method for optimization, global convergence depends on the use of a *merit function* appropriate for optimization. *This function should dictate which inexact solutions to the primal-dual system are acceptable search directions.*

Handling nonconvexity and rank deficiency

- There are two assumptions we aim to drop:
  - *(Regularity)* $\nabla c(x_k)^T$ *has full row rank with smallest singular value bounded below by a positive constant*
  - *(Convexity)* $u^T(\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k))u \geq \mu\|u\|^2$ *for* $\mu > 0$ *for all* $u \in \mathbb{R}^n$ *satisfying* $u \neq 0$ *and* $\nabla c(x_k)^T u = 0$
- If the constraints are ill-conditioned, Algorithm 1 may compute long unproductive search directions
- If the problem is non-convex, Algorithm 1 can easily converge to a maximizer or saddle point
- There is also a danger in each case that the algorithm may stall during a given iteration since an acceptable search direction cannot be computed

No factorizations means no clue

- A significant challenge in overcoming these obstacles is that we have no way of recognizing when the problem is non-convex or ill-conditioned, *since we do not form or factor the primal-dual matrix*

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k,\lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix}$$

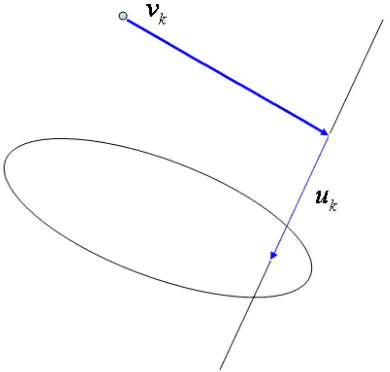
- Common practice is to perturb the matrix to be

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k,\lambda_k) + \xi_1 I & \nabla c(x_k) \\ \nabla c(x_k)^T & -\xi_2 I \end{bmatrix}$$

  where ξ_1 *convexifies* the model and ξ_2 *regularizes* the constraints, but arbitrary choices of these parameters can have terrible consequences on the behavior of the algorithm

Decomposing the step into two components

We guide the step computation by decomposing the primal search direction into a *normal* component (toward satisfying the constraints) and a *tangential* component (toward optimality)



Normal component computation

- We define the trust region subproblem

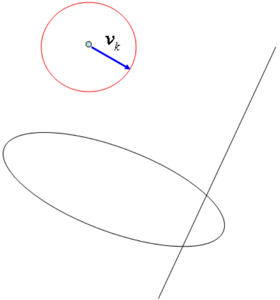
$$\begin{aligned} \min\ & \tfrac{1}{2}\|c(x_k) + \nabla c(x_k)^T v\|^2 \\ \text{s.t. } & \|v\| \leq \omega \|(\nabla c(x_k))c(x_k)\| \end{aligned}$$

for some $\omega > 0$

- An (approximate) solution v_k satisfying the Cauchy decrease condition

$$\begin{aligned} &\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T v_k\| \\ &\qquad \geq \epsilon_v (\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T \tilde{v}_k\|) \end{aligned}$$

for $\epsilon_v \in (0, 1)$, where $\tilde{v}_k = -(\nabla c(x_k))c(x_k)$ is the direction of steepest descent, can be computed without any matrix factorizations (conjugate gradient method, inexact dogleg method)



Normal component computation

- We define the trust region subproblem

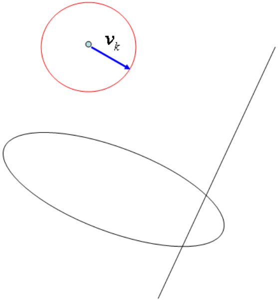
$$\begin{aligned} &\min \ \tfrac{1}{2}\|c(x_k) + \nabla c(x_k)^T v\|^2 \\ &\text{s.t. } \|v\| \leq \omega \|(\nabla c(x_k))c(x_k)\| \end{aligned}$$

  for some $\omega > 0$

- An (approximate) solution v_k satisfying the Cauchy decrease condition

$$\begin{aligned} &\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T v_k\| \\ &\qquad \geq \epsilon_v(\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T \tilde{v}_k\|) \end{aligned}$$

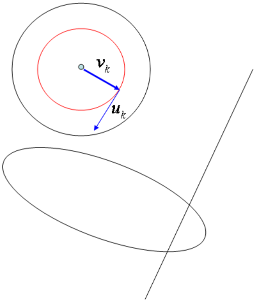
  for $\epsilon_v \in (0,1)$, where $\tilde{v}_k = -(\nabla c(x_k))c(x_k)$ is the direction of steepest descent, can be computed without any matrix factorizations (conjugate gradient method, inexact dogleg method)



Tangential component computation (idea #1)

- Standard practice is to then consider the trust region subproblem

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



- Note that an exact solution would yield $\nabla c(x_k)^T d_k = \nabla c(x_k)^T v_k$
- An iterative procedure for solving this problem (e.g., a projected conjugate gradient method), however, requires repeated (inexact) projections of vectors onto the null space of $\nabla c(x_k)^T$ (to maintain

$$\nabla c(x_k)^T u \approx 0$$

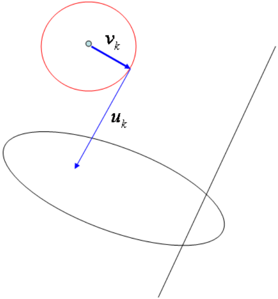
while respecting the trust region constraint)

Tangential component computation

- Instead, we formulate the primal-dual system

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k,\lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} u_k \\ \delta_k \end{bmatrix} = -\begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k + \nabla^2_{xx}\mathcal{L}(x_k,\lambda_k)v_k \\ 0 \end{bmatrix}$$

- Our ideas for assessing search directions based on reductions in a model of an exact penalty function carry over to this perturbed system



Handling non-convexity

- The only remaining component is a mechanism for handling non-convexity
- We follow the idea of convexifying the Hessian matrix as in

$$\begin{bmatrix} \nabla_{xx}^2 \mathcal{L}(x_k, \lambda_k) + \xi_1 I & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix}$$

  by monitoring properties of the computed trial search directions
- Hessian modification strategy: During the iterative step computation, modify the Hessian matrix by increasing its smallest eigenvalue whenever an approximate solution satisfies

$$\begin{aligned} \|u_k\| &> \psi \|v_k\| \\ \tfrac{1}{2} u_k^T (\nabla_{xx}^2 \mathcal{L}(x_k, \lambda_k)) u_k &< \theta \|u_k\|^2 \end{aligned}$$

  for $\psi, \theta > 0$

Inexact Newton Algorithm 2

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

- Evaluate $f(x_k)$, $\nabla f(x_k)$, $c(x_k)$, $\nabla c(x_k)$, and $\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)$
- Compute an approximate solution v_k to the trust region subproblem

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

  satisfying the normal component condition
- Iteratively solve the primal-dual equations

$$\begin{bmatrix} \nabla^2_{xx}\mathcal{L}(x_k, \lambda_k) & \nabla c(x_k) \\ \nabla c(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = -\begin{bmatrix} \nabla f(x_k) + \nabla c(x_k)\lambda_k \\ -\nabla c(x_k)^T v_k \end{bmatrix}$$

  until termination test 1 or 2 is satisfied, modifying $\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k)$ based on the Hessian modification strategy
- If only termination test 2 is satisfied, increase the penalty parameter so

$$\pi_k \ge \frac{\nabla f(x_k)^T d_k + \max\{\frac{1}{2}u_k^T(\nabla^2_{xx}\mathcal{L}(x_k, \lambda_k))u_k, \theta\|u_k\|^2\}}{(1-\tau)(\|c(x_k)\| - \|c(x_k) + \nabla c(x_k)^T d_k\|)}$$

- Perform a line search to find $\alpha_k \in (0, 1]$ satisfying the sufficient decrease condition

$$\phi(x_k + \alpha_k d_k; \pi_k) \le \phi(x_k; \pi_k) - \eta\alpha_k \Delta m(d_k; \pi_k)$$

- 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 *on 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 $\{\pi_k\}$ *is bounded, then*

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

Take-away ideas #2 and #3

In an inexact Newton method for optimization...

- *ill-conditioning* can be handled by regularizing the constraint model with a trust region. *However, a trust region for the tangential component may result in an expensive algorithm. The method we propose avoids these costs and emulates a true inexact Newton method.*
- *non-convexity* can be handled by monitoring properties of the iterative solver iterates and modifying the Hessian only when it appears that a sufficient model reduction may not be obtained. *This process may not (and need not) result in a convex model.*

Handling inequalities

- The most attractive class of algorithms for our purposes are interior point methods
- A class of interior point methods that satisfying

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

  may fail to converge from remote starting points[3]
- Fortunately, the trust region subproblem we use to regularize the constraints also saves us from this type of failure!

[3] (Wächter, Biegler (2000)

Algorithm 2 (Interior-point version)

- We apply Algorithm 2 to the logarithmic-barrier subproblem

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

  for $\mu \to 0$

- We *scale* quantities related to the slack variables d_k^s so that the step computation subproblems still have the form

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

  and

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

  where the Hessian H_k and constraint Jacobian A_k have the same properties as before

- We 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)

Under the same assumption as before, our convergence theorem for Algorithm 2 still holds for the barrier subproblem for a given μ

Theorem

(Curtis, Schenk, Wächter (2009)) 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 exists Lagrange multipliers* $\bar{\lambda}$ *such that the first-order optimality conditions of the nonlinear program are satisfied*

Take-away ideas #4

In an inexact Newton method for optimization...

- *inequality constraints* can be easily incorporated into Algorithm 2 *if quantities related to the slack variables are scaled appropriately*

Outline

Motivation and background

Algorithm development and theoretical results

Experimental results

Conclusion and final remarks

Implementation details

- We implemented our interior-point version of Algorithm 2 in the `IPOPT` software package and used `PARDISO` for the iterative linear system solves
- The normal and tangential (primal-dual) step computation was performed with the symmetric quasi-minimum residual method (SQMR)
- We solved two model PDE-constrained problems on the three-dimensional grid $\Omega = [0,1] \times [0,1] \times [0,1]$, using an equidistant Cartesian grid with N grid points in each spatial direction and a standard 7-point stencil for discretizing the operators

Boundary control problem

Let $u(x)$ be defined on $\partial\Omega$ and solve

$$\begin{aligned}
\min\ & \frac{1}{2}\int_{\Omega}(y(x)-y_t(x))^2 dx \\
\text{s.t.}\ & -\nabla\cdot(e^{y(x)}\cdot\nabla y(x)) = 20, \quad \text{in } \Omega \\
& y(x) = u(x), \quad \text{on } \partial\Omega \\
& 2.5 \le u(x) \le 3.5, \quad \text{on } \partial\Omega
\end{aligned}$$

where

$$y_t(x) = 3 + 10x_1(x_1-1)x_2(x_2-1)\sin(2\pi x_3)$$

N	n	p	q	# nnz	f^*	# iter	CPU sec
20	8000	5832	4336	95561	1.3368e-2	12	33.4
30	27000	21952	10096	339871	1.3039e-2	12	139.4
40	64000	54872	18256	827181	1.2924e-2	12	406.0
50	125000	110592	28816	1641491	1.2871e-2	12	935.6
60	216000	195112	41776	2866801	1.2843e-2	13	1987.2
(direct) 40	64000	54872	18256	827181	1.2924e-2	10	3196.3

(Simplified) Hyperthermia Treatment Planning

Let $u_j = a_j e^{i\phi_j}$ be a complex vector of amplitudes $a \in \mathbb{R}^{10}$ and phases $\phi \in \mathbb{R}^{10}$ of 10 antennas, let $M_{ij}(x) =< E_i(x), E_j(x) >$ with $E_j = \sin(jx_1x_2x_3\pi)$, and let the tumor be defined by $\Omega_0 = [3/8, 5/8]^3$, and solve

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

where

$$y_t(x) = \begin{cases} 37 & \text{in } \Omega \backslash \Omega_0 \\ 43 & \text{in } \Omega_0 \end{cases}$$

N	n	p	q	# nnz	f^*	# iter	CPU sec
10	1020	512	1070	20701	2.3037	40	15.0
20	8020	5832	4626	212411	2.3619	62	564.7
30	27020	21952	10822	779121	2.3843	146	4716.5
40	64020	54872	20958	1924831	2.6460	83	9579.7
(direct) 30	27020	21952	10822	779121	2.3719	91	10952.4

(Simplified) Hyperthermia Treatment Planning

An example solution ($N = 40$)

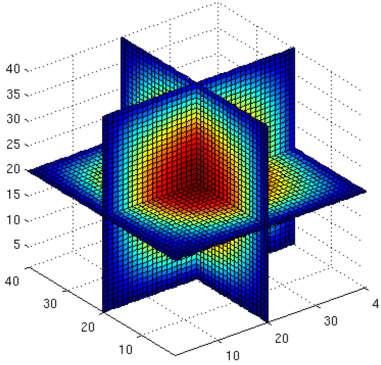


Figure: $y(x)$

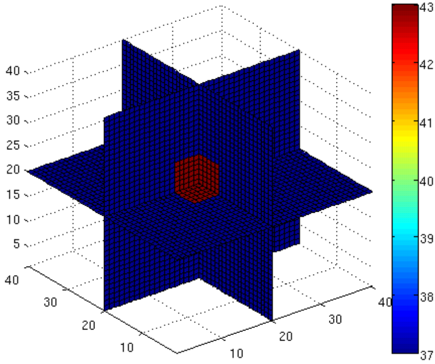


Figure: $y_t(x)$

Numerical experiments (coming soon)

Further numerical experimentation is now underway with Andreas Wächter (IBM) and Olaf Schenk (U. of Basel)

- Hyperthermia treatment planning with real patient geometry (with Matthias Christen, U. of Basel)
- Geophysical modeling applications (with Johannes Huber, U. of Basel)
- Image registration (with Stefan Heldmann, U. of Lübeck)

Outline

Conclusion and final remarks

- We have presented an inexact Newton method for constrained optimization
- The method hinges on a merit function appropriate for optimization, and the iterative linear system solves focus on a model of this function to decide when to terminate
- We have extended the basic algorithm to solve non-convex and ill-conditioned problems, and to solve problems with inequality constraints
- The algorithm is globally convergent to first-order optimal points or infeasible stationary points
- Numerical experiments are promising so far, and further testing on real-world problems is underway