Semismooth Newton Method for Gradient Constrained Minimization Problem

Serbiniyaz Anyyeva and Karl Kunisch

Institute of Mathematics and Scientific Computing, University of Graz, Austria

October 26, 2011
Problem Statement

Let $\Omega \subset \mathbb{R}^2$ be simply connected bounded Lipschitz domain. The set $K = \{ v \in H^1_0(\Omega) | |\nabla v| \leq 1 \text{ a.e. in } \Omega \}$ is non-empty, convex and closed in $H^1_0(\Omega)$. For a given $f \in H^{-1}(\Omega)$ we treat the variational inequality:

\begin{align*}
\text{Problem} \\
\text{To find a solution } u \in K \text{ such that} \\
\int_{\Omega} \nabla u \nabla (v - u) dx \geq \langle f, v \rangle \quad \forall v \in K.
\end{align*}

(1)

This variational inequality can equivalently be formulated as a gradient constrained minimization problem:

\begin{align*}
\text{Problem} \\
\text{To find a solution } u \in K \text{ such that} \\
J(u) = \min_{v \in K} J(v)
\end{align*}

where

\begin{align*}
J(v) = \frac{1}{2} \int_{\Omega} |\nabla v|^2 dx - \int_{\Omega} fv dx.
\end{align*}

(2)
Regularization

- We replace constraint $|\nabla u| \leq 1$ with the equivalent constraint $|\nabla u|^2 \leq 1$
- The formal Lagrangian for the problem:

$$
\mathcal{L}(v, \lambda) = \frac{1}{2} \int_{\Omega} |\nabla v|^2 \, dx - \int_{\Omega} fv \, dx + \int_{\Omega} \lambda (|\nabla v|^2 - 1) \, dx.
$$

Now if $u^\ast$ denotes a solution (the existence of which we know) then the formal Karush-Kuhn-Tucker (KKT) conditions for a Lagrange multiplier $\lambda^\ast$ is:

$$
\int_{\Omega} (1 + 2\lambda^\ast) \nabla u^\ast \nabla v \, dx = \langle f, v \rangle \quad \forall v \in H^1_0(\Omega)
$$

$$
\lambda^\ast \geq 0, \quad |\nabla u^\ast|^2 - 1 \leq 0, \quad (\lambda^\ast, |\nabla u^\ast|^2 - 1) = 0 \quad (3)
$$

This system has a nonlinear structure and we want to use the Newton method for solving it. Since with the last row it is impossible to apply Newton method we reformulate the optimality system in the following way:

$$
(\nabla u^\ast, \nabla v) + 2(\lambda^\ast, \nabla u^\ast \cdot \nabla v) = \langle f, v \rangle \quad \forall v \in H^1_0(\Omega),
$$

$$
\lambda^\ast = \max(0, \lambda^\ast + c(|\nabla u^\ast|^2 - 1)) \quad (4)
$$

where $c > 0$ is fixed and the max-operation is defined pointwise.
Newton differentiability

**Definition**

The mapping $F : D \subset X \to Z$ is called generalized differentiable (Newton differentiable) on the open subset $U \subset D$ if there exists a family of generalized derivatives $G : U \to L(X, Z)$ such that

$$
\lim_{\|h\| \to 0} \frac{1}{\|h\|} \|F(x + h) - F(x) - G(x + h)h\| = 0,
$$

for every $x \in U$.

For $\delta \in R$ we introduce the following candidate for its generalized derivative of the form:

$$
G_\delta(u)(x) = \begin{cases} 
1 & \text{if } u(x) > 0 \\
\delta & \text{if } u(x) = 0 \\
0 & \text{if } u(x) < 0.
\end{cases}
$$

**Lemma**

The mapping $\max(0, \cdot) : L^q(\Omega) \to L^r(\Omega)$ with $1 \leq r < q \leq \infty$ is Newton differentiable on $L^q$ and $G_\delta$ is a generalized derivative.
Semismooth Newton method

We use the augmented Lagrangian method for solving of the constrained minimization problem: we solve the sequence of unconstrained minimization problem with the objective functional

$$J_\gamma(v) = \frac{1}{2} \int_\Omega |\nabla v|^2 dx - \int_\Omega fvdx + \frac{\gamma}{2} \int_\Omega \max(0, (|\nabla v|^2 - 1))^2 dx.$$ 

to be minimized over the space $H^1_0(\Omega)$.

Further, in order to obtain the Newton differentiability we modify the problem: for $\varepsilon > 0$ sufficiently small we look for $u_{\gamma, \varepsilon} \in H^2(\Omega) \cap H^1_0(\Omega)$ minimizer of the functional

$$J_{\gamma, \varepsilon}(v) = \frac{\varepsilon}{2} (\Delta v)^2 + J_\gamma(v)$$

over the space $H^2(\Omega) \cap H^1_0(\Omega)$. 

Semismooth Newton method

**Problem**

Find $u_{\gamma, \varepsilon} \in H^2(\Omega) \cap H^1_0(\Omega)$ such that

$$
\varepsilon(\Delta u_{\gamma, \varepsilon}, \Delta v) + (\nabla u_{\gamma, \varepsilon}, \nabla v) + (\lambda_{\gamma, \varepsilon}, \nabla u_{\gamma, \varepsilon} \cdot \nabla v) = \langle f, v \rangle \quad \forall v \in H^2(\Omega) \cap H^1_0(\Omega),
$$

$$
\lambda_{\gamma, \varepsilon} = 2\gamma \max(0, |\nabla u_{\gamma, \varepsilon}|^2 - 1)).
$$

For the semismooth Newton method the linearization of the nonlinear operator equation $F(u) = 0$ has the form

$$
DF(u^{(k)}) \delta u = -F(u^{(k)}),
$$

where $DF$ is the Newton derivative of $F$, $\delta u$ is the update for $u$. 
Semismooth Newton method

The use of the method leads to the following variational equation at the Newton iteration:

\[ \varepsilon \int_{\Omega} \Delta u_{\gamma,\varepsilon}^{(k+1)} \Delta v + \int_{\Omega} a^{(k)} \nabla u_{\gamma,\varepsilon}^{(k+1)} \nabla v = \int_{\Omega} g^{(k)} \nabla u_{\gamma,\varepsilon}^{(k)} \nabla v + \int_{\Omega} f v, \]

\[ \forall v \in H^2(\Omega) \cap H^1_0(\Omega) \quad (5) \]

where

\[ a^{(k)} = (1 + 2\gamma \chi^{(k)}_{\mathcal{A}} \cdot (|\nabla u_{\gamma,\varepsilon}^{(k)}|^2 - 1)) I + 4\gamma \chi^{(k)}_{\mathcal{A}} \nabla u_{\gamma,\varepsilon}^{(k)} \otimes \nabla u_{\varepsilon}^{(k)}, \]

with

\[ I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \]

and

\[ g^{(k)} = 4\gamma \chi^{(k)}_{\mathcal{A}} |\nabla u_{\gamma,\varepsilon}^{(k)}|^2. \]

Here the characteristic function

\[ \chi^{(k)}_{\mathcal{A}}(x) = \begin{cases} 1 & \text{if } |\nabla u_{\gamma,\varepsilon}^{(k)}(x)| \geq 1 \\ 0 & \text{if } |\nabla u_{\gamma,\varepsilon}^{(k)}(x)| < 1 \end{cases} \]
Algorithm

Algorithm 1 Semismooth Newton Method

1: $\gamma := \gamma_0$, choose $u^{(0)}_{\gamma,\varepsilon}$, choose $\varepsilon > 0$
2: $u^{(c)} = u^{(0)}_{\gamma,\varepsilon}$
3: while not converged do
4:   $k = 0$
5:   Set $\mathcal{A}_{\gamma,0} = \{x \in \Omega : |\nabla u^{(c)}|^2 > 1\}$
6:   while not converged do
7:     $k = k + 1$
8:     solve (5) for $u^{(k+1)}_{\gamma,\varepsilon}$
9:     $\mathcal{A}_{\gamma,k+1} = \{x \in \Omega : |\nabla u^{(k+1)}_{\gamma,\varepsilon}|^2 > 1\}$
10:    if $\mathcal{A}_{\gamma,k+1} = \mathcal{A}_{\gamma,k}$ then
11:       STOP
12:    end if
13:   end while
14:   $u^{(c)} = u^{(k)}_{\gamma,\varepsilon}$
15:   increase $\gamma$
16: end while
Using of COMSOL Multiphysics

![Diagram of COMSOL Multiphysics interface showing Model Builder and Equation View with a partial weak expression]

- Variables
  - Name: u
  - Shape function: Argyris
  - Unit: Dependent variable

- Weak Expressions
  - Weak expression: 
    
    \[-\varepsilon(u_{xx} + u_{yy})\text{test}(u_{xx} + u_{yy}) - (1 + 2\gamma \text{...rev}(u,1),y)^*\text{test}\]

- Additional components shown in the model are:
  - Model 1 (mod1)
  - Definitions
  - Geometry 1
  - Materials
  - PDE (w)
    - Weak Form PDE 1
      - Equation View
    - Zero Flux
    - Initial Values 1
    - Pointwise Constraint 1
    - Mesh 1
  - Study 1
    - Step 1: Time Discrete
    - Solver Configurations
    - Job Configurations
  - Results

Serbiniz A. A., and K. Kunisch (Institute of Mathematics and Scientific Computing, University of Graz, Austria)
Using of COMSOL Multiphysics
Using of COMSOL Multiphysics
Using of COMSOL Multiphysics
Using of COMSOL Multiphysics

Parameter names: gamma
Parameter values: \(10^{\text{range}(0,1,3)}\)

Load parameter values: Browse... Read File

Stop condition:

Results While Solving

Error

Parameters | Error
-------------|---------
"gamma","1" | null
"gamma","10" | null
"gamma","100" | null
"gamma","1000" | null
Numerical results: Example 1

\[
\min_{v \in K} \left[ \frac{1}{2} \int_{\Omega} |\nabla v|^2 dx - d \int_{\Omega} v(x) dx \right]
\]

\[K = \{ v \in H_0^1(\Omega) : |\nabla v| \leq 1 \text{ a.e. in } \Omega \}\]

with \( d = 5 \) and

\[\Omega = \{ x \in \mathbb{R}^2 | x = (x_1, x_2), x_1^2 + x_2^2 < 1 \}\].

- The continuation method is initialized by zero.
- The stopping condition for the Newton iterations is \( \|u^{(k+1)}\| - \|u^{(k)}\| < 10^{-10} \), where \( \|u\| = (\int_{\Omega} |\Delta u|^2)^{\frac{1}{2}} \).
Numerical results: Example 1

Figure: Convergence results for the triangulation mesh with 1902 triangles and 8873 DOF ($u_h$ computed solution); $\varepsilon = 0.0001$; time estimation for Intel(R) Core(TM) i3 CPU 2.27 GHz
Numerical results: Example 1

To check the convergence rate of Newton iterations:

- first we increase the number of time-discrete levels up to maximal number of iterations and add the same number of nodes

Solver → Other → Store Solution

- for each iteration except the last we compute the norm value:

$$\| u^{(k)} - u^{(M)} \| = \left( \int_{\Omega} |\Delta (u^{(k)} - u^{(M)})|^2 \right)^{\frac{1}{2}},$$

where $u^{(M)}$ is the last iteration for the current value of $\gamma$. This we can do by

Results → Derived values → Surface Integration.

Figure: Superlinear convergence of Newton iterations (on a log-scale) with $\varepsilon = 0.0001$; mesh with 1902 triangles.
Numerical results : Example 1

To get the approximate rate of convergence we used the maximum element size in the meshes:

\[
\text{convergence rate} = \log \frac{h_{1,\text{max}}}{h_{2,\text{max}}} \frac{\text{error}_1}{\text{error}_2}
\]

Table: Tests with various meshes; last column: convergence in \(H^1_0(\Omega)\)-seminorm

| # of triangles | # of DOF | # of iter. | \(|u_h - u^*_h|\) | conv. rate |
|----------------|---------|------------|-----------------|------------|
| 546            | 2631    | 13         | 0.0336          |            |
| 936            | 4428    | 13         | 0.0286          | 0.7        |
| 1902           | 8873    | 11         | 0.0157          | 1.7        |
| 6530           | 29951   | 11         | 0.0067           | 1.4        |
| 24924          | 113270  | 12         | 0.0026           | 1.4        |
Numerical results: Example 2

We choose rectangular domain
$\Omega = \{ x \in \mathbb{R}^2 \mid x = (x_1, x_2), -1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1 \}$ and

$$f(x, y) = \begin{cases} 
10 \cos(2((y - 1)^2 + x^2 - 1)) & \text{if } x^2 + (y - 1)^2 < 1 \\
0 & \text{elsewhere}
\end{cases}$$

Figure: The gradient magnitude of the solution obtained on the mesh with 268 triangles and with final $\gamma = 10^3$. 

(a) $\epsilon = 0$ \hspace{2cm} (b) $\epsilon = 10^{-4}$