Semi-automatic transition from simulation to one-shot optimization with equality constraints

Lisa Kusch, Tim Albring, Andrea Walther, Nicolas Gauger

Chair for Scientific Computing, TU Kaiserslautern, www.scicomp.uni-kl.de
Institute of Mathematics, Universität Paderborn

September 15, 2016
AD 2016
Outline

1. The One-Shot Approach with Additional Constraints
2. AD-Based Discrete Adjoint in SU2
3. Multi-Objective Optimization
4. Application and Results
   - Application to Airfoil Design
   - Multi-Objective Optimization with Single Constraint
   - Multi-Objective Optimization with Multiple Constraints
   - Second-Order Adjoints
5. Summary and Outlook
Problem Definition

PDE-constrained optimization problem

\[
\begin{align*}
\min_{u,y} & \quad f(y,u) \\
\text{s.t.} & \quad c(y,u) = 0, \\
& \quad h(y,u) = 0
\end{align*}
\]

\[
\Rightarrow \text{fixed point iteration} \quad ||G|| \leq \rho < 1
\]

\[
\min_{u,y} \quad f(y,u) \quad \text{s.t.} \quad G(y,u) = y, \\
& \quad h(y,u) = 0
\]

- objective function \( f \in \mathbb{R} \), state \( y \in Y \subset \mathbb{R}^n \), design \( u \in U \subset \mathbb{R}^m \)
- state equation \( c : Y \times U \to Y \), constraints \( h : Y \times U \to V \subset \mathbb{R}^p \)
  (notation: \( \hat{G} = (G(y,u) - y, h(y,u))^T \))

⇒ one-shot approach using the AD-based discrete adjoint
  [Hamdi, Griewank(2010)],[Walther, Gauger, Kusch, Richert(2016)]

\[
L(\tilde{y}, y, u) = f(y,u) + (G(y,u) - y)^T \tilde{y} + h^T \mu = N(y, \tilde{y}, \mu, u) - y^T \tilde{y}
\]
### The One-Shot Approach with Additional Constraints

#### KKT point (discrete)

<table>
<thead>
<tr>
<th>Expression</th>
<th>Simplified</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y^<em>$ $=$ $N_y(y^</em>, \bar{y}^<em>, \mu, u^</em>)^T$ $=$ $G(y^*, u)$</td>
<td></td>
</tr>
<tr>
<td>$\bar{y}^<em>$ $=$ $N_y(y^</em>, \bar{y}^<em>, \mu, u^</em>)^T$ $=$ $f_y(y^<em>, u^</em>)^T + G_y(y^<em>, u^</em>)^T \bar{y}^*$</td>
<td></td>
</tr>
<tr>
<td>0 $=$ $N_u(y^<em>, \bar{y}^</em>, \mu, u^<em>)^T$ $=$ $f_u(y^</em>, u^<em>)^T + G_u(y^</em>, u^<em>)^T \bar{y}^</em>$</td>
<td></td>
</tr>
<tr>
<td>0 $=$ $N_\mu(y^<em>, \bar{y}^</em>, \mu, u^<em>)^T$ $=$ $h(y^</em>, u^*)$</td>
<td></td>
</tr>
</tbody>
</table>

#### One-shot approach with equality constraints

For $k = 1, ...$

- $y_{k+1} = G(y_k, u_k)$
- $\bar{y}_{k+1} = N_y(y_k, \bar{y}_k, \mu_k, u_k)$
- $u_{k+1} = u_k - B_k^{-1} N_u(y_k, \bar{y}_k, \mu_k, u_k)$
- $\mu_{k+1} = \mu_k - \tilde{B}_k^{-1} h(y_k, u_k)$
Convergence Properties

doubly augmented Lagrangian

\[ L^a(y, \bar{y}, \mu, u) = \frac{\alpha}{2} ||\hat{G}(y, u)||^2 + \frac{\beta}{2} ||N_y(y, \bar{y}, \mu, u)^T - \bar{y}||^2 + N - \bar{y}^T y \]

correspondence and descent condition

\[ \sqrt{\alpha\beta(1 - \rho)} > 1 + \frac{\beta}{2} ||N_{yy}|| \] (1)

and \( \alpha \geq \|2(\hat{G}_u^T)^1\cdots P(\hat{G}_y^T)^+ (N_{yu})_1\cdots p - (N_{uu})_1\cdots p\|/\|\hat{G}_u\|_1\cdots p\| \)

\( \Rightarrow \) exact penalty function, descent for all large symmetric pos. def. \( B, \tilde{B} \)

choice of \( B \)

\[ B \approx \nabla^2_{uu} L^a(y, \bar{y}, u) \Rightarrow B = \alpha \hat{G}_u^T \hat{G}_u + \beta N_{yu}^T N_{yu} + N_{uu} \]
Implementation Details

BFGS Update: \( B \Delta u \approx \nabla_u L^a(y, \tilde{y}, \mu, u + \Delta u) - \nabla_u L^a(y, \tilde{y}, \mu, u) =: r \)

calculate \( \nabla_u L^a = \alpha \Delta y_k^T G_u + \alpha h_u^T h + \beta \Delta \tilde{y}_k^T N_{yu} + N_u \)

- \( N_u \) and \( \Delta y_k^T G_u + h_u^T h \) can be obtained with reverse mode of AD from \( \tilde{u} \)
  
  \[
  y = G(y, u) \\
  v = f(y, u) \\
  \tilde{u} = f_u(y, u)^T \tilde{v} + G_u(y, u)^T \tilde{y} + h_u(y, u)^T \tilde{w} \\
  w = h(y, u) \\
  \]

- finite difference over reverse / tangent over reverse

\[
\Delta \tilde{y}_k^T N_{yu} \approx \frac{1}{h} (N_u(y_k + h \Delta \tilde{y}_k, \tilde{y}_k, \mu_k, u_k) - N_u(y_k, \tilde{y}_k, \mu_k, u_k)) \quad (2)
\]

- curvature condition: set \( B_k = I \) if \( r_k^T \Delta u_k \leq 0 \)
Open-source multiphysics suite SU2

- Under active development (Stanford University, TU Kaiserslautern, TU Delft,...)
- FV discretization of (U)RANS equations and turbulence models using highly modular C++ code-structure, additional PDE solvers
- Multi-grid and local time-stepping methods
- Several space and time discretizations, MPI, python scripts
- optimization environment based on continuous or discrete adjoint, integrated AD support [Albring, Sagebaum, Gauger(2015)]
Discrete-Adjoint Solver in SU2

- operator overloading (frequent extensions/improvements)
- shape optimization → mesh deformation as additional constraint in the optimization problem $M(u) = x$
- fixed-point form $\bar{y}_{k+1} = N_y(y_k, \bar{y}_k, \mu_k, u_k)^T$

CoDiPack

- use of expression templates
- different tape implementations

Performance SU2 (NASA Common Research Model)

- Runtime Factor: 1.2
- Memory Factor: 7.2
Multi-Objective Optimization Problem

\[
\begin{align*}
\min_{y, u} & \quad F(y, u) \\
\text{s.t.} & \quad c(y, u) = 0, \\
& \quad h(y, u) \geq 0
\end{align*}
\]

(3)

(4)

**Pareto dominance:**

\( \bar{x} \) dominates \( x \) if

\[
\forall i \in \{1, \ldots, k\} : \quad f_i(\bar{x}) \leq f_i(x) \\
\exists j \in \{1, \ldots, k\} : \quad f_j(\bar{x}) < f_j(x)
\]
Methods for Multi-Objective Optimization

- common: weighted sum, evolutionary algorithms
- \( \varepsilon \)-constraint method
  [Marglin(1967)]

\[
\begin{align*}
\min_{y,u} & \quad f_{s_j}(y,u) \\
\text{s.t.} & \quad c(y,u) = 0, \quad (5) \\
& \quad h(y,u) = 0, \\
& \quad f_i(x) \leq f_i^{(j)} \\
& \quad \forall \ i \in \{1, ..., k\} \quad i \neq s_j
\end{align*}
\]

- connected: equality constraints
Application (Airfoil Design)

- aerodynamic shape optimization of NACA 0012 airfoil
- objectives: drag coefficient $c_d$, lift coefficient $c_l$
- bounded design variables: 38 Hicks-Henne bump functions (bounds: [-0.005,0.005])
- constraints: moment, thickness
- steady Euler equations, transonic flow (Mach 0.8 and AOA 1.25)
Result: Lift Constraint

preconditioner $\tilde{B} = 20$, $\mu_0 = 0$, target lift: 0.4

backtracking line search, trial step size: 1, $\alpha = 20$, $\beta = 2$
Resulting Pareto-optimal Front

![Graph showing the relationship between lift coefficient and drag coefficient. The graph displays several points marked with an 'x', indicating the Pareto-optimal front.]

Kusch, Albring, Walther, Gauger

From simulation to one-shot optimization with constraints
Result: Lift, Moment, Thickness Constraint

target lift: 0.3, moment: 0.0, thickness: 0.12
preconditioner diag(20, 20, 100), $\mu_{i,0} = 0$
Resulting Pareto-optimal Front

![Graph showing Pareto-optimal front with lift and drag coefficients.](image)

- Kusch, Albring, Walther, Gauger
- From simulation to one-shot optimization with constraints
Using Second-Order Adjoints

Tangent over Reverse: \( \Delta \bar{y}_k^T N_{yu} \)

\[
\hat{u}^T = (\bar{y}^T G_{uy} + \bar{v}^T f_{yu} + \bar{w}^T h_{yu}) \dot{y} + \text{other terms}
\]

(6)

\[\Rightarrow \text{Set } \bar{v} = 1, \bar{w} = \mu, \bar{y} = \bar{y}, \dot{y} = \Delta \bar{y}, \text{ Read from } \hat{u} \text{ (Runtime Factor: 1.6)}\]

**DataType:**  
ActiveReal<RealForward, ChunkTape<RealForward, int>>

**GetMixedDerivative:**  
tape.getGradient(x[adjPos++]).getGradient()

---

![Graph](image)

**Objective:**  
- Second order
- Finite differences

---

**Kusch, Albring, Walther, Gauger**  
From simulation to one-shot optimization with constraints
Summary and Outlook

Summary
Semi-automatic transition from simulation to optimization involving

- AD-based discrete adjoint in SU2
- constrained one-shot method and
- deterministic multi-objective optimization

Outlook

- different application in SU2 (multi-disciplinary)
- investigations on preconditioner for constraints

Thank you for your attention!
References

Hamdi, A. and Griewank, A.
Properties of an augmented lagrangian for design optimization.

Walther, A., Gauger, N., Kusch, L., and Richert, N.
On an extension of one-shot methods to incorporate additional constraints.
*Optimization Methods and Software* (2016).

Albring, T., Sagebaum, M., and Gauger, N.
Development of a consistent discrete adjoint solver in an evolving aerodynamic design framework.

Marglin, S. A.
*Public investment criteria.*