Department Review:

Advancing the Research and Integration of Invasive Optimization Technology

Roscoe A. Bartlett

Department 1411: Optimization and Uncertainty Estimation

Sandia National Laboratories

SAND2006-3623P

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

Sandia National Laboratories

Overview of Roscoe Bartlett (1411)

Highlights for PMF year 2005-2006

- Continuing as lead for the Thyra effort (vertical interoperability of numerical algorithms)
  - Collaborators from 1411, 1414, 1416, 1433, 1437, 1514, 1543, and 8962
  - New: Linear solver and preconditioner interfaces and integration
  - New: Nonlinear Model Evaluator for steady-state and transient analysis, design, and UQ
  - New: External impact: SciDAC (SciOPS, TOPS II)
- MOOCHO optimization algorithms/software
  - New: Integration into Trilinos as a proper Trilinos package (to be released with Trilinos 7.0)
  - New: Development of minimally invasive optimization algorithms
- Trilinos: Growth, scalability, capability, design, programming consultation & support
- Publications/talks/reviews
  - Journal articles: submitted = 2, accepted = 1
  - Conference talks = 1, conference papers (submitted = 2, accepted = 1)
  - SAND reports = 1, internal Sandia talks (TUG = 4, SESS = 1, other = 1)
  - Reviewed 3 Journal papers
- Misc:
  - Nominated for an individual ERA award by Long (8962), Heroux (1414), and van Bloemen Waanders (1411)

Publications and Presentations

Bartlett, Roscoe A., Bart G. van Bloemen Waanders, Martin Berggren, "Hybrid Differentiation Strategies for Simulation and Analysis of Applications in C++," Journal Article, *ACM TOMS*, Submitted May 2006.

Bartlett, Roscoe A, Matthias Heinkenschloss, Denis Ridzal, Bart G van Bloemen Waanders, "Domain Decomposition Methods for Advection Dominated Linear-Quadratic Elliptic Optimal Control Problems," Journal Article, *Computer Methods in Applied Mechanics and Engineering*, Accepted/Published May 2006

Bartlett, Roscoe A., David M. Gay, Eric T. Phipps, "Automatic Differentiation of C++ Codes for Large-Scale Scientific Computing," Conference Paper, Third International Workshop on Automatic Differentiation: *Tools and Applications at ICCS 2006*, February 2006.

Bartlett, Roscoe A., Kevin R. Long, Bart G. van Bloemen Waanders, "A scalable optimization interface for numerical simulation applied to the next generation supercomputer," *Conference Paper*, Supercomputing 2006, April 2006.

Bartlett, Roscoe A., “Object-Oriented Generic Programming for Abstract Numerical Algorithms and Interoperability via Thyra”, *SIAM Parallel Processing Conference*, Presentation, February 2006

van Bloemen Waanders (Editor), Bart , Roscoe A. Bartlett … , “Algorithm and Simulation Development in Support of Response Strategies for Contamination Events in Air and Water Systems”, *SAND Report SAND2006-0074*, January 2006

Ghattas, Omar, … Roscoe Bartlett …, “SciDAC Institute for Optimization of Petascale Simulations (SciOPS)”, SciDAC Proposal, March 2006

Bartlett, Roscoe A. “Teuchos::RefCountPtr : An Introduction to the Trilinos Smart Reference-Counted Pointer Class for (Almost) Automatic Dynamic Memory Management in C++”, *Trilinos Software Engineering Seminar Series*, August 2005

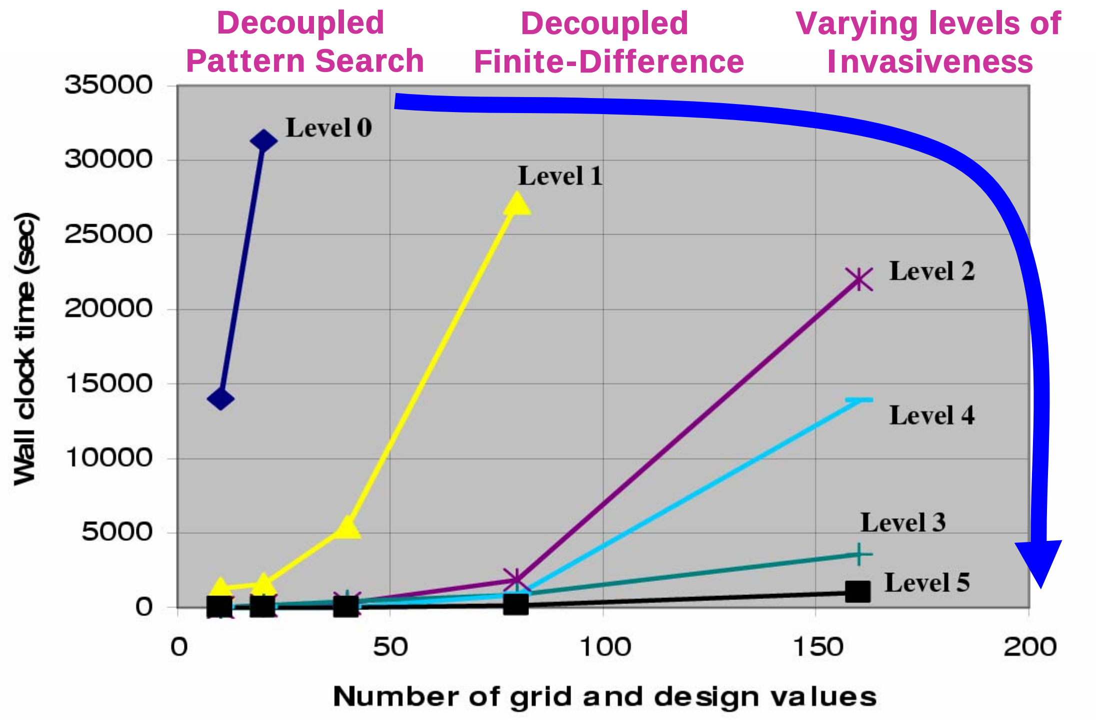
Bartlett, Roscoe A., “An Overview of the Thyra Interoperability Effort : Current Status and Future Plans”, *Trilinos User’s Group Meeting*, Presentation, November 2005

Bartlett, Roscoe A., “Thyra from a Developer's Perspective”, *Trilinos User’s Group Meeting*, Presentation, November 2005

Pawlowski, Roger, “The Trilinos Export Makefile System for Portable and Scalable Dependency Tracking”, *Trilinos User’s Group Meeting*, November 2005

Bartlett, Roscoe A., “An Overview of the Thyra/EpetraExt ModelEvaluator Software”, *Sandia National Laboratories*, Internal Presentation, January 2006

Motivation for Invasive Gradient-Based SAND Optimization



Increasing Levels of Coupling and Derivative and Solve Capabilities

Large Scale Non-Linear Programming for PDE Constrained Optimization, van Bloemen Waanders, B., Bartlett, R., Long, K., Boggs, P., and Salinger, A. Sandia Technical Report SAND2002-3198, October 2002

Key Point

For many/some optimization problems, intrusive optimization methods can be much more computationally efficient and more robust

But:

- It is hard to get our "foot in the door" with production codes
- It is hard to keep a "door stop" in place once we are in ... Because ...

Some Challenges to Incorporation of Invasive Optimization

Where I am Involved

- Lack of Software Infrastructure
  - Linear algebra and linear solvers not supporting optimization requirements
  - Application structure not flexible (i.e. only supports a narrow mode to solve the forward problem)

Thyra

MOOCHO

- Lack of software maintenance
  - Optimization support is not tightly integrated with forward solve code and is not maintained over time.

Model Evaluator

- Lack of derivative support
  - Lack of model smoothness
  - No optimization variables derivatives
  - Lack of transient derivatives

Automatic Differentiation (AD)

Rythmos

Key Point

We need a strategy to reduce the threshold for getting invasive optimization into codes and for keeping the capability once it is there => Software and Algorithms

Overview of Nonlinear Model Evaluator Interface

Approach: Develop a single, scalable interface to address many different types of numerical problems

- (Some) Input arguments:
  - State and differential state: $x \in \mathcal{X}$ and $\dot{x} = \frac{dx}{dt} \in \mathcal{X}$
  - Parameter sub-vectors: $p_l \in \mathcal{P}_l$ for $l = 1 \ldots N_p$
  - Time (differential): $t \in \mathbf{R}$

- (Some) Output functions:
  - State function: $(\dot{x}, x, \{p_l\}, t) \Rightarrow f \in \mathcal{F}$
  - Auxiliary response functions: $(\dot{x}, x, \{p_l\}, t) \Rightarrow g_j \in \mathcal{G}_j$, for $j = 1 \ldots N_g$
  - State/state derivative operator: $(\dot{x}, x, \{p_l\}, t) \Rightarrow W = \alpha \dfrac{\partial f}{\partial \dot{x}} + \beta \dfrac{\partial f}{\partial x}$

Key Points

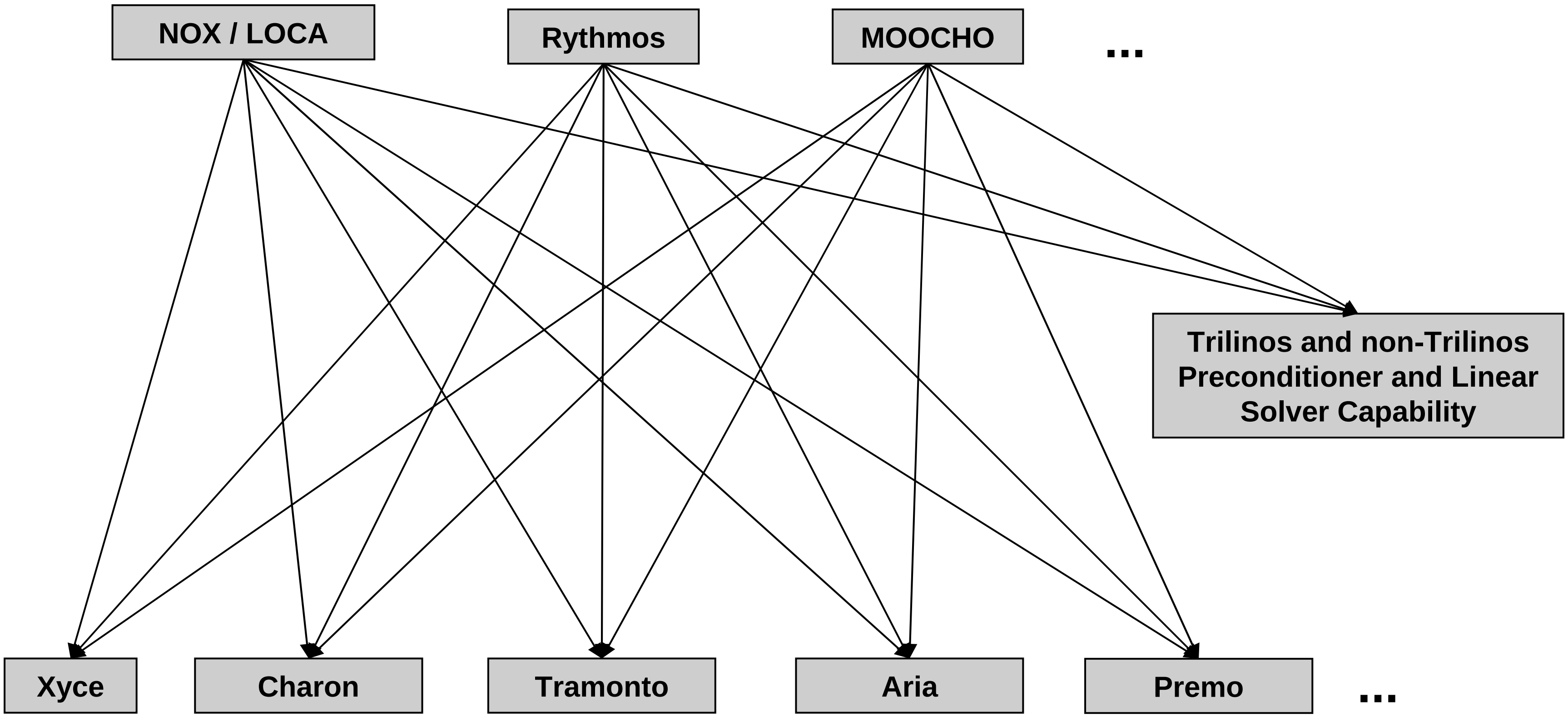
- Flexible/extendable specification of model inputs outputs
- Address a large number steady-state and transient numerical problems and applications
- Designed for augmentation!

Sandia National Laboratories

Some Examples of Supported Nonlinear Problem Types

Nonlinear equations:	Solve $f(x) = 0$ for $x \in \mathbf{R}^n$
Stability analysis:	For $f(x,p) = 0$ find space $p \in \mathcal{P}$ such that $\frac{\partial f}{\partial x}$ is singular
Explicit ODEs:	Solve $\dot{x} = f(x,t) = 0, t \in [0,T]$, $x(0) = x_0$, for $x(t) \in \mathbf{R}^n, t \in [0,T]$
DAEs/Implicit ODEs:	Solve $f(\dot{x}(t), x(t), t) = 0, t \in [0,T]$, $x(0) = x_0$, $\dot{x}(0) = x'_0$ for $x(t) \in \mathbf{R}^n, t \in [0,T]$
Explicit ODE Forward Sensitivities:	Find $\frac{\partial x}{\partial p}(t)$ such that: $\dot{x} = f(x,p,t) = 0, t \in [0,T]$, $x(0) = x_0$, for $x(t) \in \mathbf{R}^n, t \in [0,T]$
DAE/Implicit ODE Forward Sensitivities:	Find $\frac{\partial x}{\partial p}(t)$ such that: $f(\dot{x}(t), x(t), p, t) = 0, t \in [0,T]$, $x(0) = x_0$, $\dot{x}(0) = x'_0$, for $x(t) \in \mathbf{R}^n, t \in [0,T]$
Unconstrained Optimization:	Find $p \in \mathbf{R}^m$ that minimizes $g(p)$
Constrained Optimization:	Find $x \in \mathbf{R}^n$ and $p \in \mathbf{R}^m$ that: minimizes $g(x,p)$ such that $f(x,p) = 0$
ODE Constrained Optimization:	Find $x(t) \in \mathbf{R}^n$ in $t \in [0,T]$ and $p \in \mathbf{R}^m$ that: minimizes $\int_0^T g(x(t),p)$ such that $\dot{x} = f(x(t),p,t) = 0$, on $t \in [0,T]$ where $x(0) = x_0$

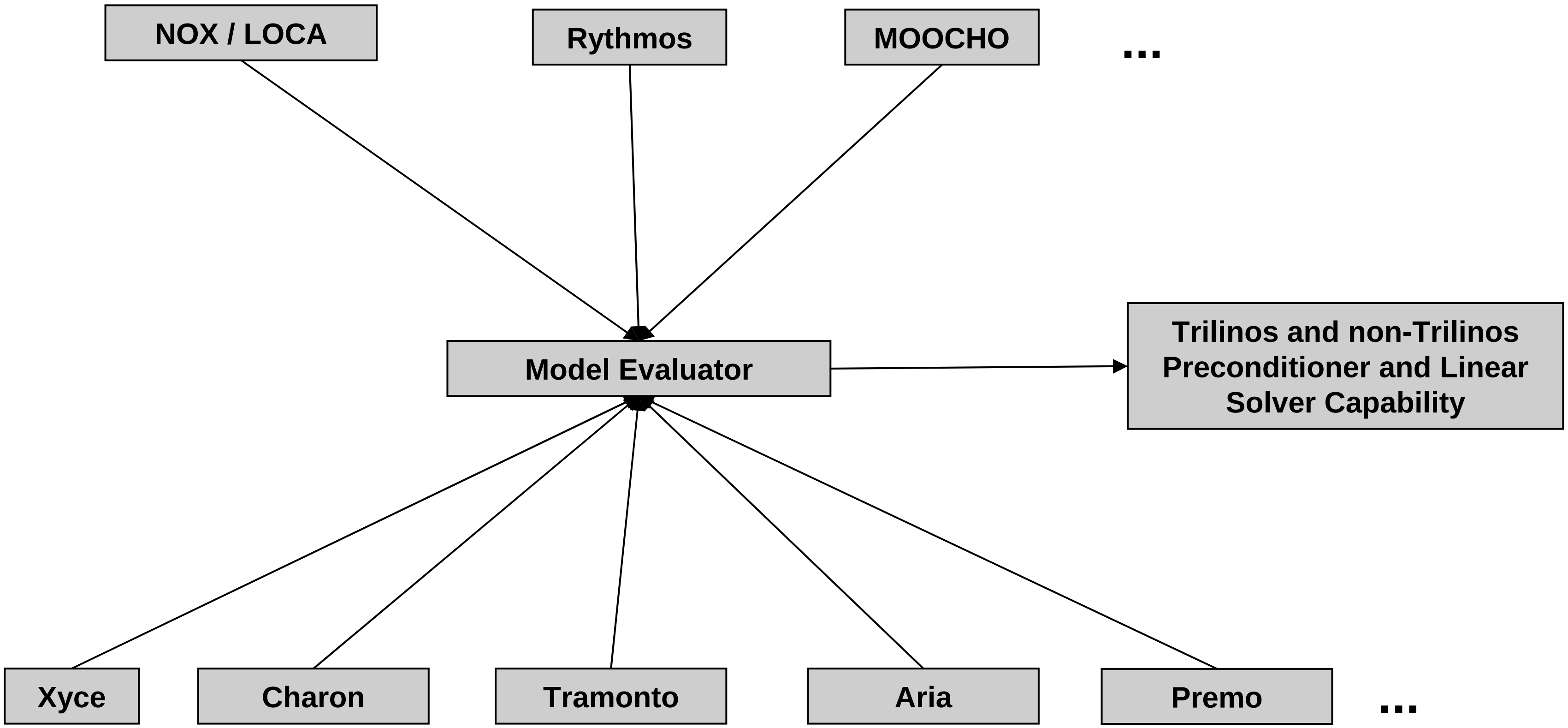
Nonlinear Algorithms and Applications : Everyone for Themselves?



Key Point

- BAD

Nonlinear Algorithms and Applications : Thyra & Model Evaluator!



Key Points

- Avoid duplication of efforts
- Provide more uniform interface for applications and users
- Once on algorithm is interface to an application, others can quickly follow
- Research ↑ Application/user support ↓

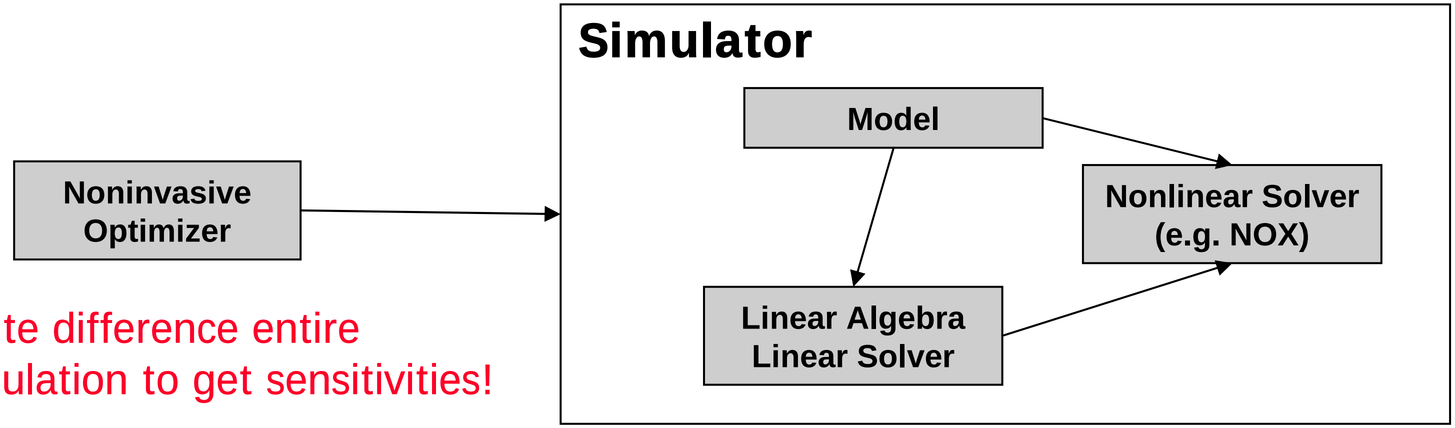
Impact of the Nonlinear Model Evaluator

- Incorporation into simulation codes
  - Sundance: Symbolic simulator (Long(8962)) => SC06 Paper with Red Storm
  - Charon: QASPR project (Hoekstra(1437),...) => Param. Est. to exper. data
  - Tramonto: Decontamination LDRD (vBW(1411),...)
  - ??? SIERRA solution control (Notz(1514), Baur(1543), Hooper(1416)) => Multi-physics
  - ...
- Incorporation into numerical algorithms
  - MOOCHO: Simulation-constrained optimization (Bartlett(1411))
  - Rythmos: Time integration and sensitivity methods (Coffey(1414))
  - NOX: Nonlinear equation solvers (Pawlowski(1416))
  - LOCA: Library of continuation algorithms (Salinger(1416), Phipps(1416))
  - ...
- Connection with other SNL projects
  - 4D CSRF, Transient to steady-state (Salinger(1416))
  - Multi-physics LDRD (Hooper(1416), Pawlowski(1416))
  - ...
- External impact
  - SciDAC-2 SciOPS proposal (Ghattas et. al.) => Bartlett: "Chief opt. software architect"
  - SciDAC-2 TOPS II proposal (Keyes et. al.) =>Thyra a multi-institution standard!

Sandia National Laboratories

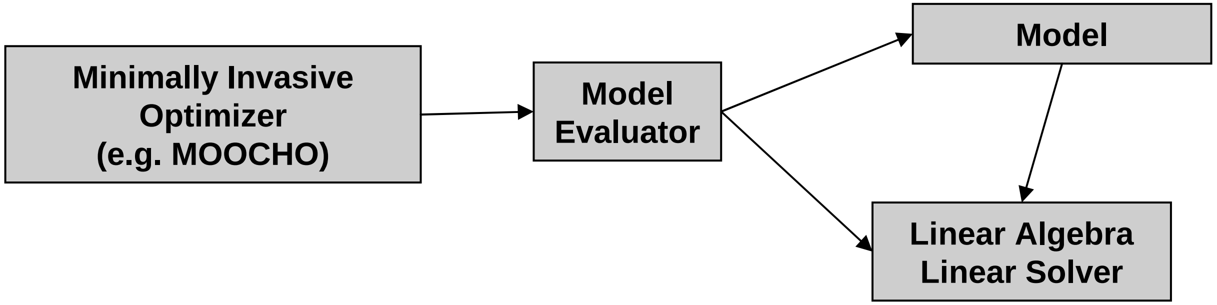
Minimally Invasive Gradient-Based Optimization

Decoupled Optimization: Assume there is optimization capability in the "Simulator"



Finite difference entire simulation to get sensitivities!

Coupled Optimization: Simulator broken up and some pieces are given over to optimizer



Question: What is the minimum that the "Model" and "Linear Solver" have to provide to allow for invasive optimization?

Minimally Invasive Direct Sensitivity MOOCHO

Basic Simulation-Constrained Optimization Problem

Find $x \in \mathbf{R}^n$ and $p \in \mathbf{R}^m$ that:

minimize $g(x,p)$
such that
$f(x,p) = 0$

Defines the state simulator and direct sensitivities

$$p \rightarrow x(p)$$

$$\frac{\partial x}{\partial p} = -\frac{\partial f}{\partial x}^{-1} \frac{\partial f}{\partial p}$$

Reduced Obj. Function

$$p \rightarrow \hat{g}(p)$$

Minimal Requirements for decoupled Newton simulation-constrained optimization

- Residual Eval: $(x,p) \rightarrow f$
- Jacobian Eval: $(x,p) \rightarrow \frac{\partial f}{\partial x}$
- Objective Eval: $(x,p) \rightarrow g$

Linear Solver

State solve with NOX/LOCA

Decoupled Opt.

Minimally Invasive Direct Sensitivity MOOCHO

Derivatives desired but not required

- Residual opt. deriv: $(x,p) \rightarrow \frac{\partial f}{\partial p}$
- Objective state deriv: $(x,p) \rightarrow \frac{\partial g}{\partial x}$
- Objective opt. deriv: $(x,p) \rightarrow \frac{\partial g}{\partial p}$

Approximate using O(n_p) directional finite differences!

$$\frac{\partial f}{\partial p_i} \approx \frac{f(x, p + \delta e_i) - f(x,p)}{\delta}$$

$$\frac{\partial \hat{g}}{\partial p_i} \approx \frac{g\left(x + \delta \frac{\partial x}{\partial p_i}, p + \delta e_i\right) - g(x,p)}{\delta}$$

Sandia National Laboratories

Scalable Optimization Test Problem

Example: Parallel, Finite-Element, 2D, Diffusion + Reaction (GL) Model

$$\begin{array}{lll} \text{min} & \frac{1}{2}\int_{\Omega}(x(y)-x^*(y))^2 dy & \\ \text{s.t.} & \nabla^2 x + \alpha(x - x^3) = r(y) & y \in \Omega \\ & \frac{\partial x(y)}{\partial n} = q(p,y) & y \in \partial\Omega \end{array} \Rightarrow \begin{array}{ll} \text{min} & g(x,p) \\ \text{s.t.} & f(x,p) = 0 \end{array}$$

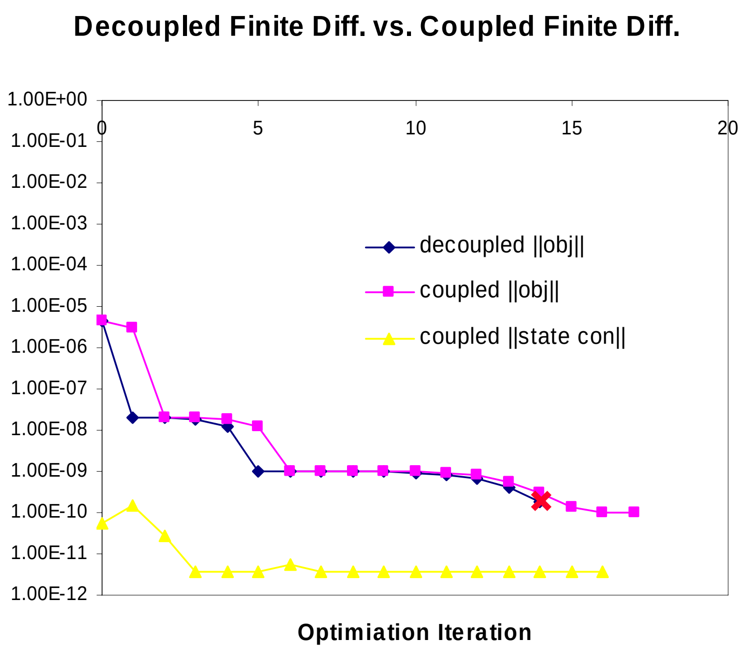
- State PDE: Scalar Ginzburg-Landau equations (based on Denis Ridzal's (1414) code)
- Discretization:
  - Second-oder FE on triangles
  - n_x = 110,011 state variables and equations
- Optimization variables:
  - Sine series basis
  - n_p = 6 optimization variables
  - Note: df/dp is constant in this problem!!!
- Iterative Linear Solver : ILU (Ifpack), (GMRES) AztecOO

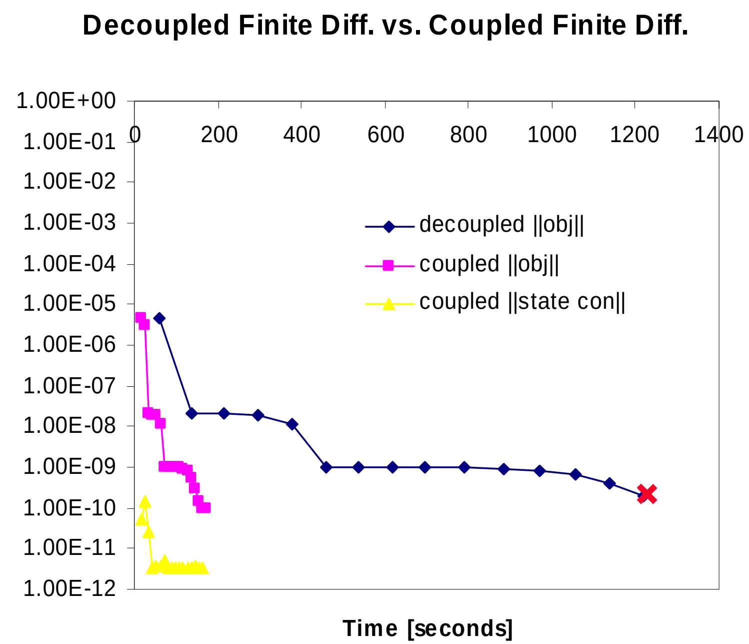
Key Points

- Simple physics but leads to very nonlinear state equations
- Inverse optimization problem is very ill posed in many instances

Sandia National Laboratories

Results: Decoupled vs. Coupled, Finite Differences

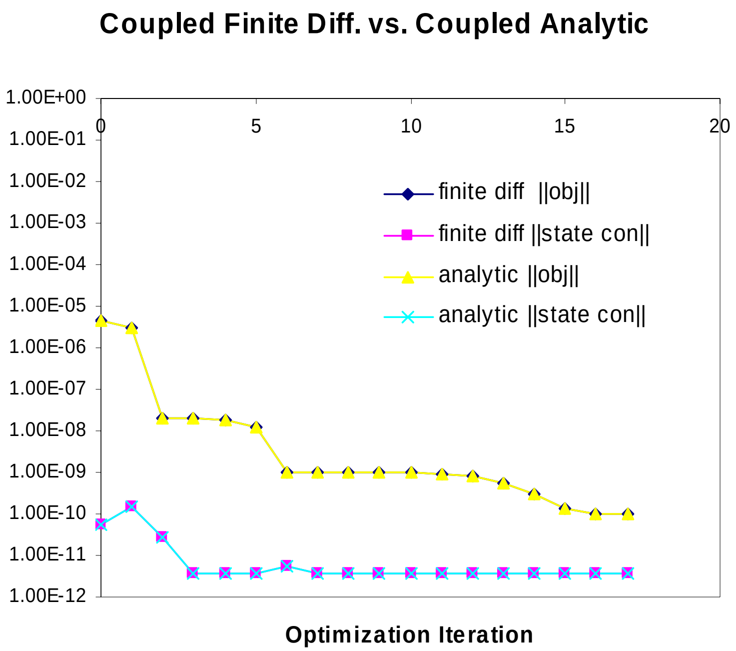


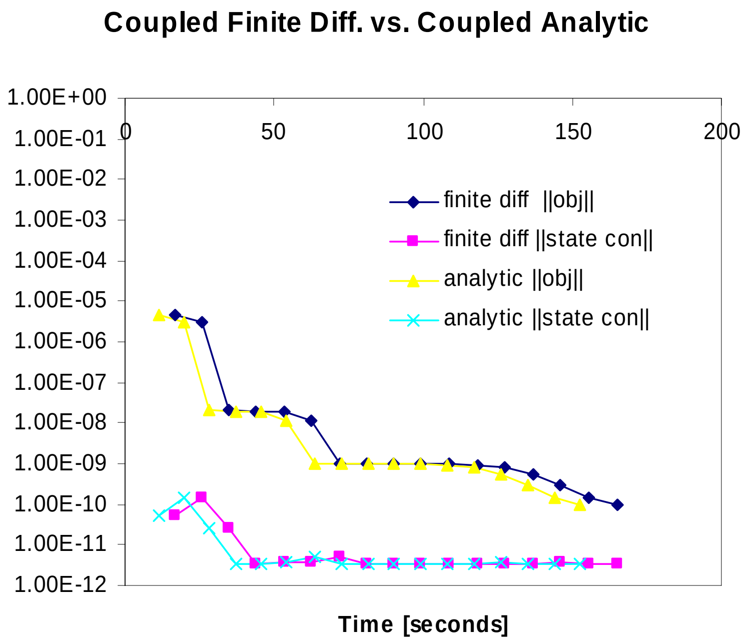


Key Points

- Finite differencing the underlying functions is much more efficient than finite differencing entire simulation!
- Finite differencing the underlying functions is more accurate!
- Coupled approach requires (almost) no extra application functionality!

Results: Coupled Finite Diff. vs. Coupled Analytic





Key Points

- Analytic derivatives are usually not faster
- Analytic derivatives often much more accurate

MOOCHO / Model Evaluator : Future Work

– Algorithmic Research

- Faster multi-RHS solves for direct sensitivities => Cheater sensitivity computations?

$$\begin{bmatrix} \Delta x & \frac{\partial x}{\partial p} \end{bmatrix} = -\frac{\partial f}{\partial x}^{-1} \begin{bmatrix} f & \frac{\partial f}{\partial p} \end{bmatrix}$$

- Inexact iterative linear solves => Cheaper and more robust?
  - Infinite dimensional nature? (i.e. non-Euclidean scalar products)
    - Difficult for optimization algorithms. (i.e. one norms!)
  - Local and global convergence
- Finite difference Newton Handling of inexact Jacobians => More accurate sensitivities
- Contracted projected subspace algorithm => Cheaper sensitivity computations?
- Second derivatives (adjoints+) => Better convergence rates, more robust

– Internal Impact

- Charon: QASPR project (Hoekstra(1437),...) => Param. Est. to exper. data
- Tramonto: Decontamination LDRD (vBW(1411),...)
- ...

– External Impact

- Release Thyra::ModelEvaluator and MOOCHO in Trilinos 7.0
- Journal Paper (in progress)