# Toward Advances in 3 Areas

## *Dept. 1411 Review, May 2006*

**David M. Gay (dmg)**

***Sandia National Laboratories***

**`dmgay@sandia.gov`**

**`http://endo.sandia.gov/~dmgay`**

SAND2006-3620P

**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

# Outline

1. Background
2. Automatic Differentiation
3. Global Optimization
4. AMPL
5. Service

# Background

- Interests include
  - Scientific computing in general
  - Smooth optimization in particular
- Good experience using AD with AMPL
  - Gradient, Hessian computations
    on AMPL expression graphs
- Early experience bounding nonlin. eqn sol'ns
- Now at Sandia ➔ chance to work on large-scale computations.
  - This year, focused on three disparate topics.

# Automatic Differentiation CSRF

- Work with Eric Phipps (1416); dmg = PI
  - With helpful comments from Roscoe Bartlett, Andy Salinger, etc.
- Goal:  promote better use of computing resources via better algorithms for derivatives.
- Work this past year includes
  - More work with Charon (Eric Phipps)
  - Improving, templating RAD (dmg)
  - Paper for ICCS06 (Bartlett, Gay, Phipps)
    - Eric will present in Reading, UK, late May

# Need for derivatives

- **Nonlinear equations: solving $f(x) = 0$**
  - **Newton's method: $x^{k+1} = x^k - f'(x^k)^{-1} f(x^k)$**
- **ODEs, PDEs, DAEs**
- **Optimization**
  - **Minimize $f(x)$ s.t. $L \cdot c(x) \cdot u$**
  - **Find $\beta$ to minimize $\sum_i [y_i - M(x_i, \beta)]^2$**
- **Sensitivities (first order)**
  - **How do small changes in x affect f(x)?**
  - **Useful in Verification and Validation**

***What if we could compute as though we had several Red Storms?***

# Action on a Mesh

- In large-scale computation, values of interest are often computed on elements of a mesh.
- Often a small number of mesh-element functions.
- Absent time stepping,

  Visiting each element once and

  computing its contribution to $f$, $rf$ , $r^2f$ and

  Manually assembling the overall $f$

may be much more efficient than straightforward backward AD.

# *RAD*: Specialized overloading for r*f*

- "Active" variables of type ADvar
  - Simply assigned values initially
  - Can be overwritten
  - Each operation stores values, partials in memory
- Invoke

```
ADcontext::Gradcomp();
```

  when gradient desired.
  - Causes reverse sweep and reclamation of memory
  - For ADvar v,
    - `v.val()` = current value
    - `v.adj()` = adjoint w.r.t. last computed ADvar value
- Block memory allocation for efficiency.

# Efficiency of AD in Charon

Set of N hypothetical chemical species:

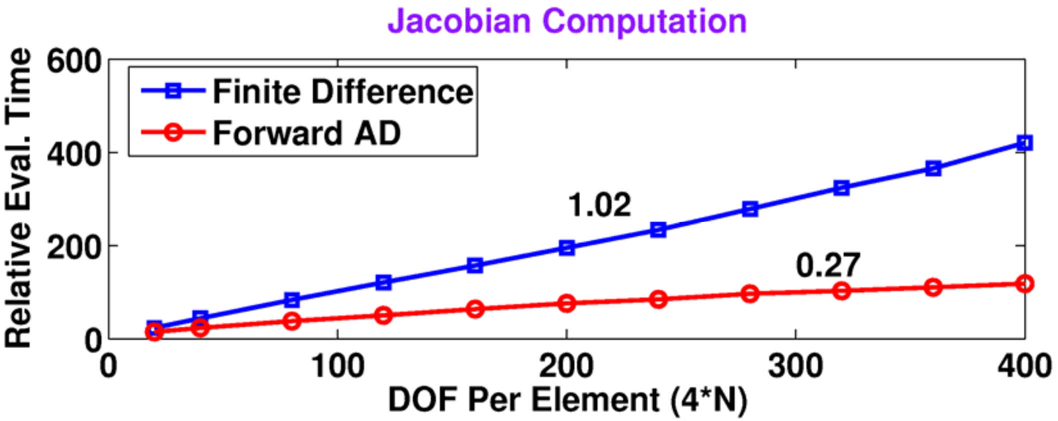
$$2X_j \rightleftharpoons X_{j-1} + X_{j+1}, \quad j = 2, \ldots, N-1$$

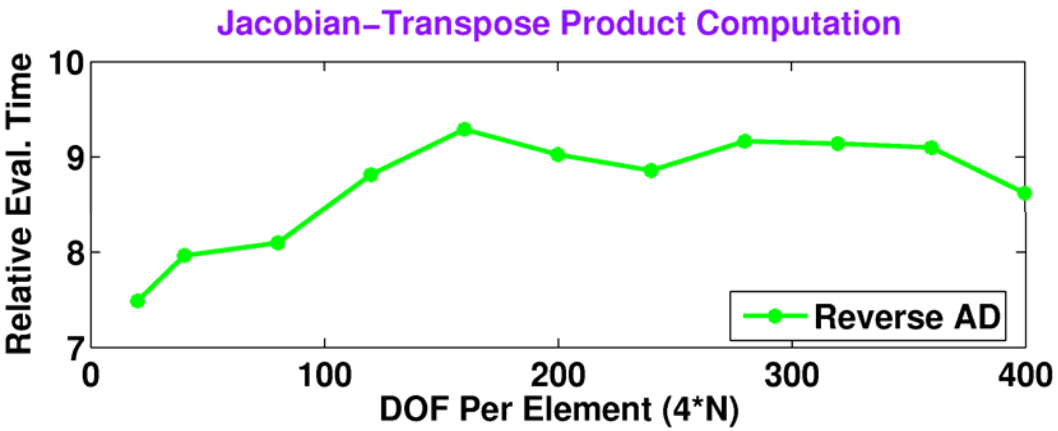
Steady-state mass transfer equations:

$$\nabla^2 Y_j + \mathbf{u} \cdot \nabla Y_j = \dot{\omega}_j, \quad j = 1, \ldots, N-1$$

$$\sum_{j=1}^{N} Y_j = 1$$

Efficiency of the element-level derivative computation




**Automatic Differentiation**

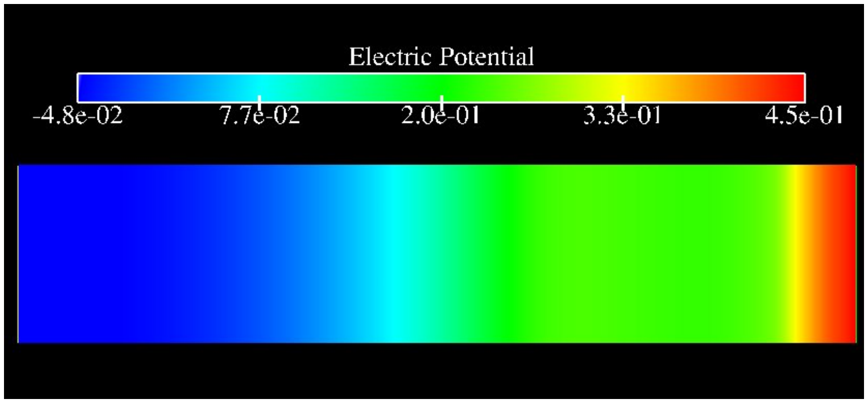
- ✓ **Faster than FD**
- ✓ **Better scalability in number of PDEs**
- ✓ **Analytic derivative**

Results published, presented at ICCS 2006, Reading, UK

Sandia National Laboratories

# Steady-State Sensitivity Analysis of 1n5614 Diode -- 0.5V Bias

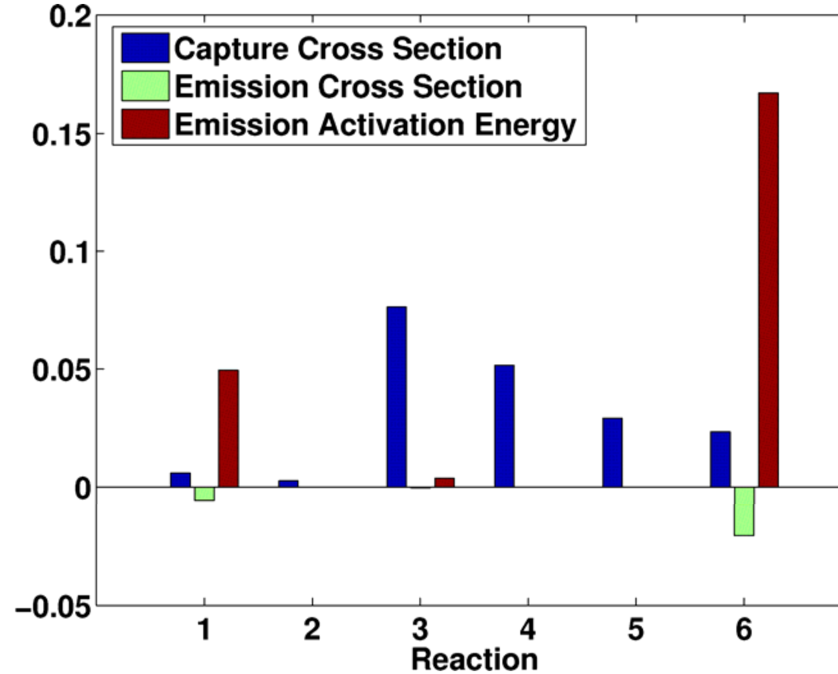
**4 defect species, 2 currents, 18 parameters, 126k unknowns, 2 procs**




$$\frac{\Delta I}{I} \approx \left(\frac{p}{I}\frac{\partial I}{\partial p}\right)\frac{\Delta p}{p}$$

| # | Reaction | Cross Section | Emission Activation Energy |
|---|---|---|---|
| 1 | $VV^- + e^- \leftrightarrow VV^{--}$ | 3.0e-16 | 0.23 |
| 2 | $VV^{--} + h^+ \leftrightarrow VV^-$ | 3.0e-14 | 0.89 |
| 3 | $VV^0 + e^- \leftrightarrow VV^-$ | 3.0e-15 | 0.41 |
| 4 | $VV^- + h^+ \leftrightarrow VV^0$ | 3.0e-14 | 0.71 |
| 5 | $VV^+ + e^- \leftrightarrow VV^0$ | 3.0e-14 | 0.91 |
| 6 | $VV^0 + h^+ \leftrightarrow VV^+$ | 3.0e-15 | 0.21 |

**Scaled Current Sensitivities**




**Adjoint sensitivities > 9 times faster then direct!**

Sandia National Laboratories

# RAD improvements

- More templating (for use with various types)
  - Easier use with constants (integer, floating-point)
  - Interacts successfully with FAD
    - Hessian-vector products
- Means of interacting with 3rd-party functions
  - E.g., in Charon, Chemkin processed by Adifor
- Allow multiple reverse sweeps for several outputs
- More debugging facilities
  - RAD_DEBUG_BLOCKKEEP=n (puts NaNs into n old memory blocks)
  - RAD_AUTO_AD_Const treats unfreed ADvar's as constants

# AD outreach at Sandia

- Presentation at Sandia Software Developer's Conference (dmg, May 2006)
- Interest in more widespread AD introduction

Working on…

- Web site for *Sacado*
- Examples

Goal: facilitate code owner's use of AD

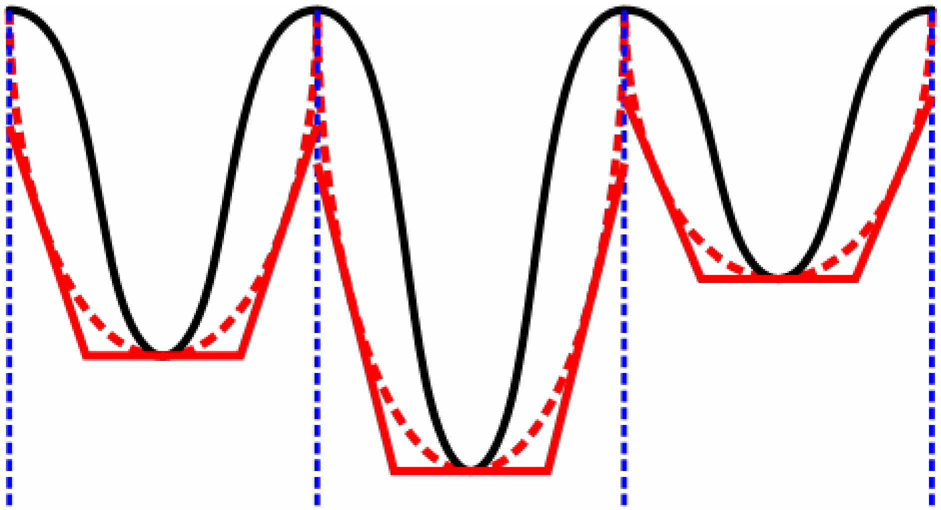
# AD Summary

- Programmatic issue:  better algs for derivatives
- Activities:  develop *Sacado* = Sandia AD tools
- Technical:
  - New RAD package for reverse AD (dmg)
  - Improved FAD package for fwd AD (etphipp)
- Results: good experience in Charon
- Impact:
  - More efficient computation
  - Banish confusion from truncation error
  - Simplify maintenance — save programmer time

# Rigorous Global Optimization

PI: Bill Hart

Application example: NW safing



# FY06 Rigorous Global Optimization

- Bill Hart, PI
- Dave Gay
- Hosted two summer students in 2005
  - Lianjun Jiang (U of CO)
    - interval slopes, other Coconut facilities
  - Pradeep Polisetty (U of S. Carolina)
    - convex underestimators from expression graph
- Adding “gnlp” to Acro
- Rewrote “ampltodag”
  - Indication of integer variables

Sandia National Laboratories

# Global optimization summary

- **Programmatic issue: minimize or maximize a compactly constrained objective**
- **Activities: integration of rigorous bounding and fathoming tests (exclusion, sufficiency) into PICO parallel branch-and-bound framework**
- **Technical: developing *gnlp***
- **Results: working toward first *gnlp* release**
- **Impact:**
  - **NW safing**
  - **Enable asking, "How well could we do?"**
  - **Decision support**

# AMPL extensions

**FY06 MICS support for extensions to stochastic programming (decision support, ISE)**

**AMPL: a language for *mathematical programming*:**

$$\text{minimize } f(x)$$

$$\text{s.t. } L \cdot c(x) \cdot u$$

**With x 2 $R^n$ and c: $R^n \rightarrow R^m$**

**Given algebraically and some $x_i$ discrete.**

Sandia National Laboratories

# AMPL goals/features

- **Easy transcription from math (*avoid mistakes*)**
- **Explicit indexing (*no hidden magic*)**
- **Declare before use (*one-pass reading*)**
- **Separate model, data, commands (*orthogonality*)**
- **Separate solvers (*open solver interface*)**
- **Update entities as needed (*lazy evaluation*)**
- **Builtin math. prog. Stuff (*presolve, red. costs,* …)**
- **Design for large scale nonlinear (*sparsity, algs*.)**
- **Allow extensions via shared libraries**
  - **Imported functions, table handlers**

# Stochastic programming examples

**Recourse problems:**

**Minimize E(today's decision + tomorrows corrections)**

**Probabilistic constraints:**

**Prob(failure) · ε**

**Prob(success) ¸ 1 - ε**

Sandia National Laboratories

# AMPL extension: random variables

```
var v random;
# …
s.t. Constr1: … >= v;
s.t. Constr2: c(v) == 0;
# …
let v := Normal01() + Uniform(-1, 2);
# or
let v := 4.2;
```

***Let the solver do the sampling.***

# Implementation goals

Interface to Gassmann’s MSLip, SMPS.

Interface to Gerd Infanger’s importance-sampling solver.

# AMPL support at SNL

- Overview talk to 1400 last Oct.
- Overview talk to JWAC last Nov.
- Interactions with 6221 re table handler.
- Binaries for AMPL, several solvers in dropzone
  - Re pcx, Craig Lawton (6642) said "Wow, this is going to help me quite a bit."
- Recent ISE model change

```
s.t. C{(i,j,k) in A, (i,j,p) in A, (p,k) in Parent): …
```

to

```
s.t. C{(p,k) in Parent, (i,j,k) in A: (i,j,p) in A}: …
```

reduced instantiation time from 10 hrs to 40 sec.

Sandia National Laboratories

# AMPL extensions summary

- Programmatic issue: stochastic decision support
- Activities: stochastic-programming extensions to AMPL
- Thesis: let the solver do the sampling
- Results: work in progress
- Impact: better design, decision making
  - Another tool for handling uncertainty

# Service

- Math. Programming Society Treasurer
- AMPL support

  R&D and international support at home

  Recently:

    indicator constraints in CPLEX

      (special case of constraint programming)

    converting logical constr’s to algebraic in presolve
- Refereeing, papers for
  - SCAN04 proceedings
  - ACM Trans. Math. Software

# Automatic Differentiation (AD) CSRF Project Funded for FY05-07

- Acquired funding for a new CSRF project on AD with David Gay (PI, 9211)
- Develop AD tools for Sandia DP applications
  - Forward: Extend and "robustify" TFAD
  - Reverse: Investigate efficiency, scalability of operator overloading approaches
  - Taylor: Study Taylor polynomial based algorithms for time integration, uncertainty quantification
- Goals:
  - Make AD a commonplace technology at Sandia
  - Provide derivative technology to foster new algorithm development

Finite element residual equations

$$f(x) = 0$$

$$\frac{\partial f}{\partial x} V$$

$$\left(\frac{\partial f}{\partial x}\right)^T V$$

$$x(t) = \sum_{k=0}^{d} x_k t^k$$

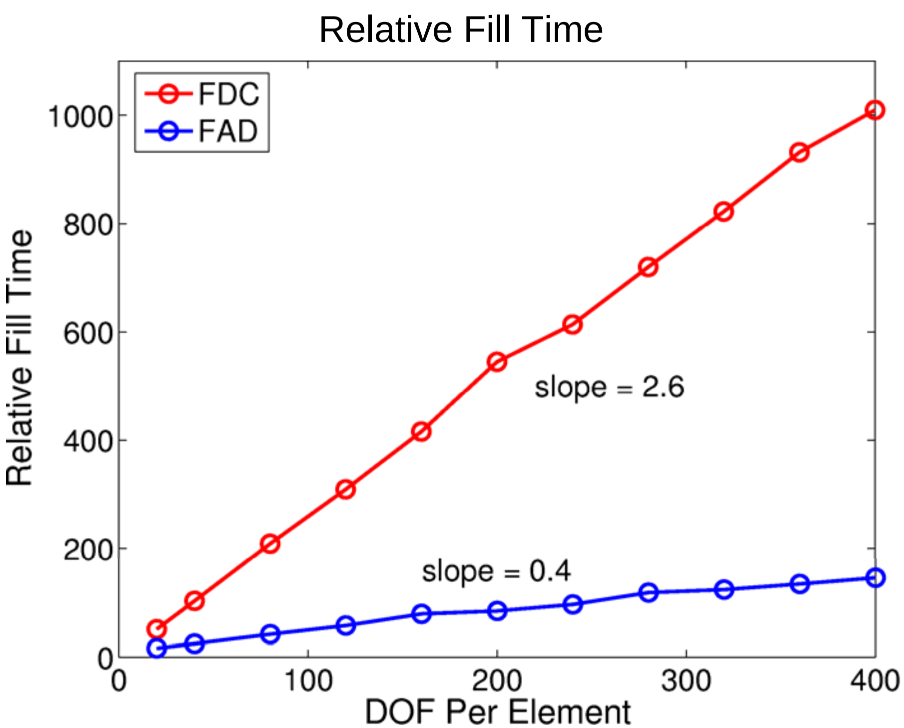
$$f(x(t)) = \sum_{k=0}^{d} f_k t^k + O(t^{d+1})$$

"Analytic derivatives are critical for the success of the Charon/QASPR project." – Rob Hoekstra, Charon Project Lead

Sandia National Laboratories

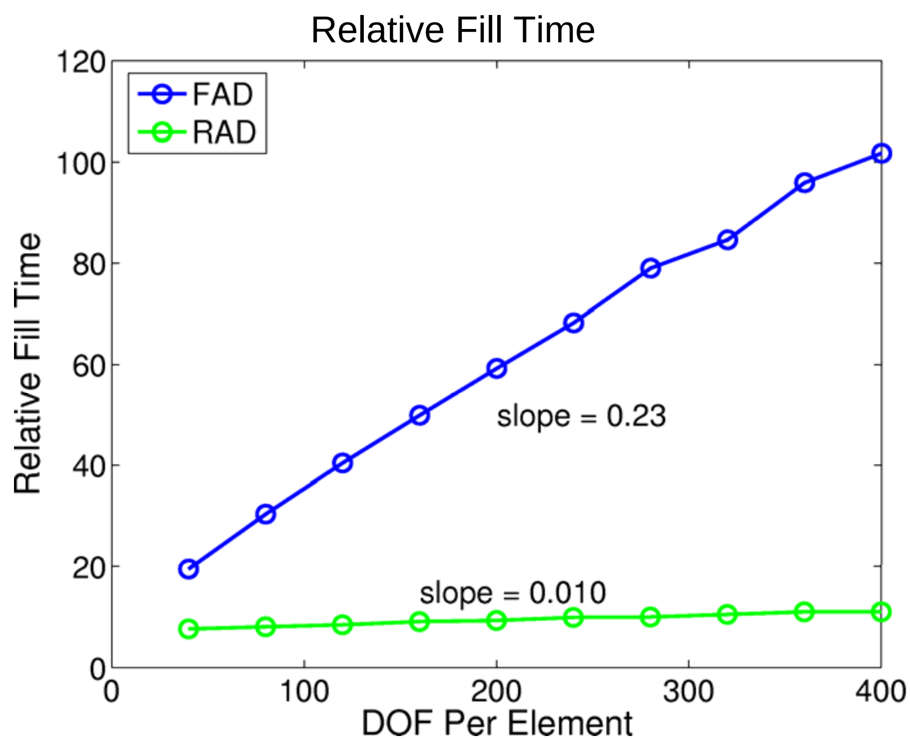
# Forward and Reverse Mode AD in Charon

## Forward Mode Jacobian Using FAD Versus Finite Differencing w/Coloring




FAD has become the standard Jacobian strategy in Charon

## Reverse Mode Jacobian-Transpose Product Versus Forward Mode




Enabling technology for large-scale sensitivity analysis & optimization

Sandia National Laboratories