<u>A</u>djoint <u>C</u>ompiler <u>T</u>echnology and <u>S</u>tandards

Uwe Naumann[1], Patrick Heimbach, Chris Hill, Carl Wunsch[2]

One of the most powerful available tools for oceanographic state estimation (data assimilation), for understanding the sensitivity of models and ocean to perturbations in parameters and forcing, and experiment design, is the so-called adjoint of a general circulation model. A major element of the ECCO Consortium approach to determining the global ocean circulation is the adjoint to the MIT model [1, 2]. In the past, adjoints have been hand-coded–a long and difficult process. Giering and Kaminski (1997) produced a translator able to go from the forward model Fortran code to a useful adjoint. The mathematical idea that was used is referred to as Automatic Differentiation (AD). Because adjoints are so generically useful for models in almost any scientific discipline, both for estimation and sensitivity studies, we have formed a consortium to produce a new easier to use, versatile, open-source adjoint tool. This tool, which should be available in preliminary form in about one year will support both Fortran and C-codes.

Given a **forward Model**

$$F : \mathbb{R}^n \to \mathbb{R}^m, \quad y = F(x),$$

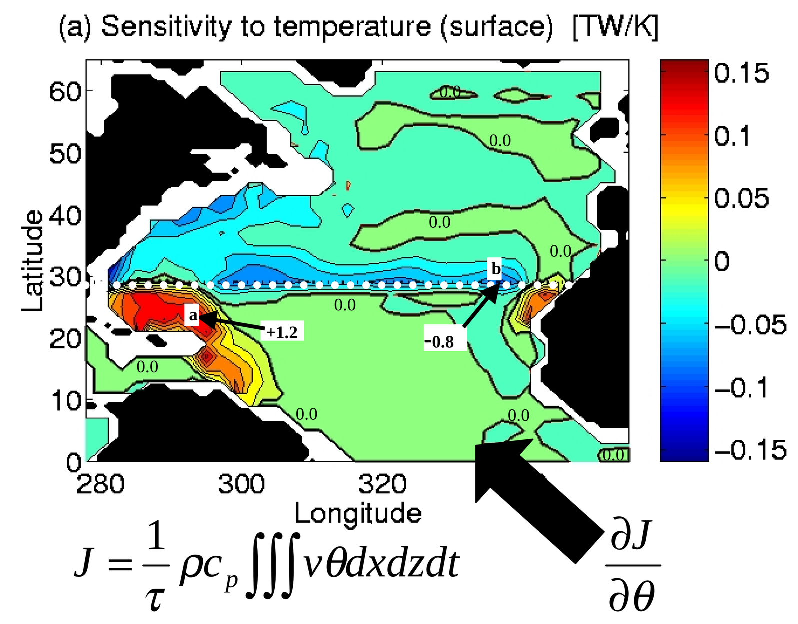
adjoint compilers can produce either a **tangent-linear model**

$$dF : \mathbb{R}^{2n} \to \mathbb{R}^m, \quad dy = dF(x, dx) = F'(x) \cdot dx,$$

or an **adjoint model**

$$aF : \mathbb{R}^{n+m} \to \mathbb{R}^n, ax = aF(x, ay) = (F')^T \cdot ay.$$

The tangent-linear model enables an efficient computation of the sensitivities of the model outputs y wrt. a given parameter x_i. The sensitivities of a given model output y_j wrt. all parameters x can be computed efficiently using the adjoint model. Furthermore, the gradient of an objective wrt. to all model parameter can be obtained efficiently. Hessian matrices can be computed by the tangent-linear of the adjoint model.



The sensitivity, $\frac{\partial J}{\partial \theta}$, of the North Atlantic heat transport at 24^oN, J, to changes in temperature, θ, at the ocean surface. These were obtained using the AD generated adjoint of the global MIT ocean circulation model. At point "a" a persistent change in θ of $+1^o$ will produce a $1.2.10^{12}$ Watt increase in annual mean poleward heat transport. The same change at point "b" produces a $0.8.10^{12}$ Watt decrease [3].

[1] Stammer, D., C. Wunsch, I. Fukumori, and J. Marshall, *State estimation in modern oceanographic research*, EOS, 83(27), 289&294-295, 2002a.

[2] Stammer, D., C. Wunsch, R. Giering, C. Eckert, P. Heimbach, C. Hill, J. Marotzke, J. Marshall, *The global ocean state during 1992-1997, estimated from ocean observations and a general circulation model. Part I. Methodology and estimated state.* J. Geophys. Res., DOI: 10.1029/2001JC000888, 2002b.

[3] Marotzke, J. R. Giering, K. Q. Zhang, D. Stammer, C. Hill, T. Lee, *Construction of the adjoint MIT ocean general circulation model and application to Atlantic heat transport sensitivity*, J Geophys. Res., 104, 29,529-29,548, 1999.

Basic Idea and an Example

Automatic Differentiation [4]

- **decomposition** into elementary assignments

$$v_j = \varphi_j(v_k)_{k \prec j}, \; j = 1, \ldots, q$$

- **augmentation** with local partial derivatives

$$c_{ji} = \frac{\partial \varphi_j}{\partial v_i}(v_k)_{k \prec j}, \; i \prec j, j = 1, \ldots, q$$

- chain rule in **forward mode**

$$dv_j = \sum_{k \prec j} c_{jk} \cdot dv_k, \; j = 1, \ldots, q$$

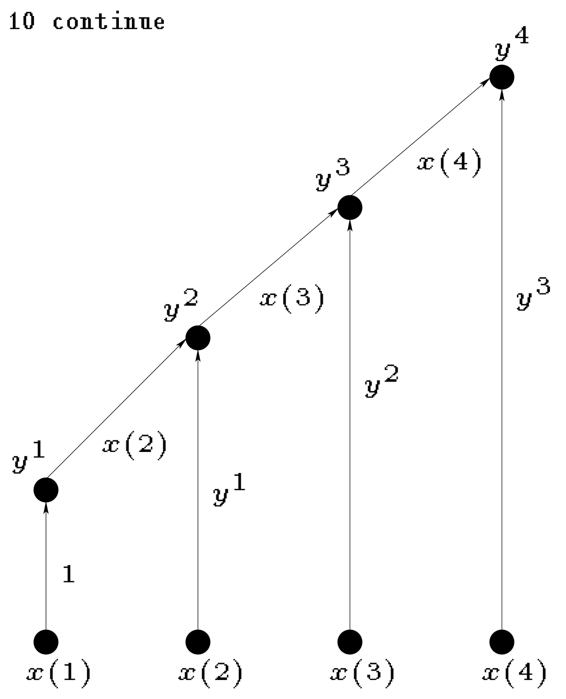
- chain rule in **reverse mode**

$$av_k = \sum_{j:k \prec j} c_{jk} \cdot av_j, \; k = p, \ldots, 1 - n$$

where $i \prec j$ means that v_i appears on the right-hand side of the elementary assignment that computes v_j.

Forward Model

```
  do 10 i=1,n
    if (i.eq.1) then
      y=x(i)
    else
      y=y*x(i)
    endif
10 continue
```



Adjoint Model

```
  do 10 i=1,n
    if (i.eq.1) then
      y=x(i)
      PUSH(1)
    else
      PUSH(y)
      y=y*x(i)
      PUSH(2)
    endif
10 continue

  do 20 i=n,1,-1
    POP(branch)
    if (branch.eq.1) then
      ax(i)=ay
    else
      POP(y)
      ax(i)=ax(i)+ay*y
      ay=ay*x(i)
    endif
20 continue
```

ax $= F'(x)^T \cdot$ ay $\in \mathbb{R}^n$ can be computed by a single (since **y** $\in \mathbb{R}$) run of the adjoint with ay $= 1$. Instances of **y** must be stored or recomputed since they get overwritten during the evaluation of the forward model. The control flow of the forward model must be reversed.

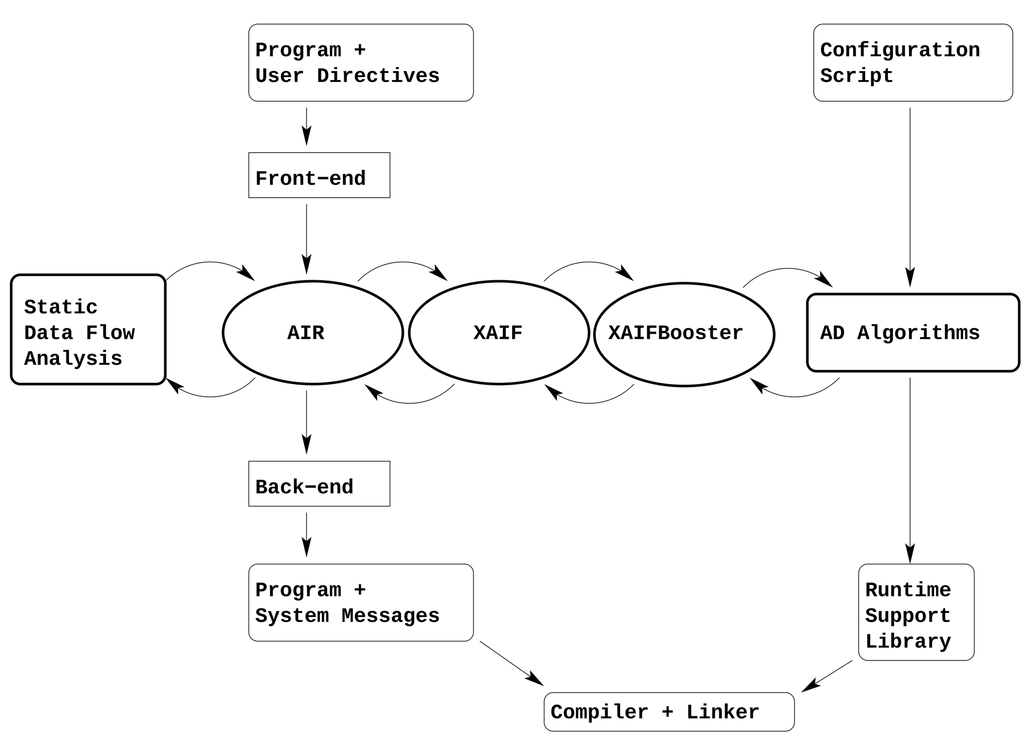
[4] G. Corliss, C. Faure, A. Griewank, L. Hascoet, U. Naumann eds., *Automatic Differentiation of Algorithms – From Simulation to Optimization*, Springer, 2002.

NSF-ITR Project "ACTS"

. . . a three year (Oct 2002 – Sep 2005) collaborative adjoint compiler technology research and development project between MIT., Rice University, and the University of Chicago / Argonne National Laboratory

Main Objectives:

- Language-independent platform for development of adjoint compilers (XAIF)
- Fortran 90 and C/C++ tools based on XAIF
- Efficient program reversal through automatic checkpointing
- Static and dynamic data-flow analysis
- Efficient handling of MPI



See also:

http://www.autodiff.org/ACTS

Interested in becoming a beta-tester of the compiler's prototypes?

naumann@mcs.anl.gov

[1] Mathematics and Computer Science Division, Argonne National Laboratory

[2] all: Department of Earth, Atmospheric and Planetary Sciences, MIT

Gnd(1.1)