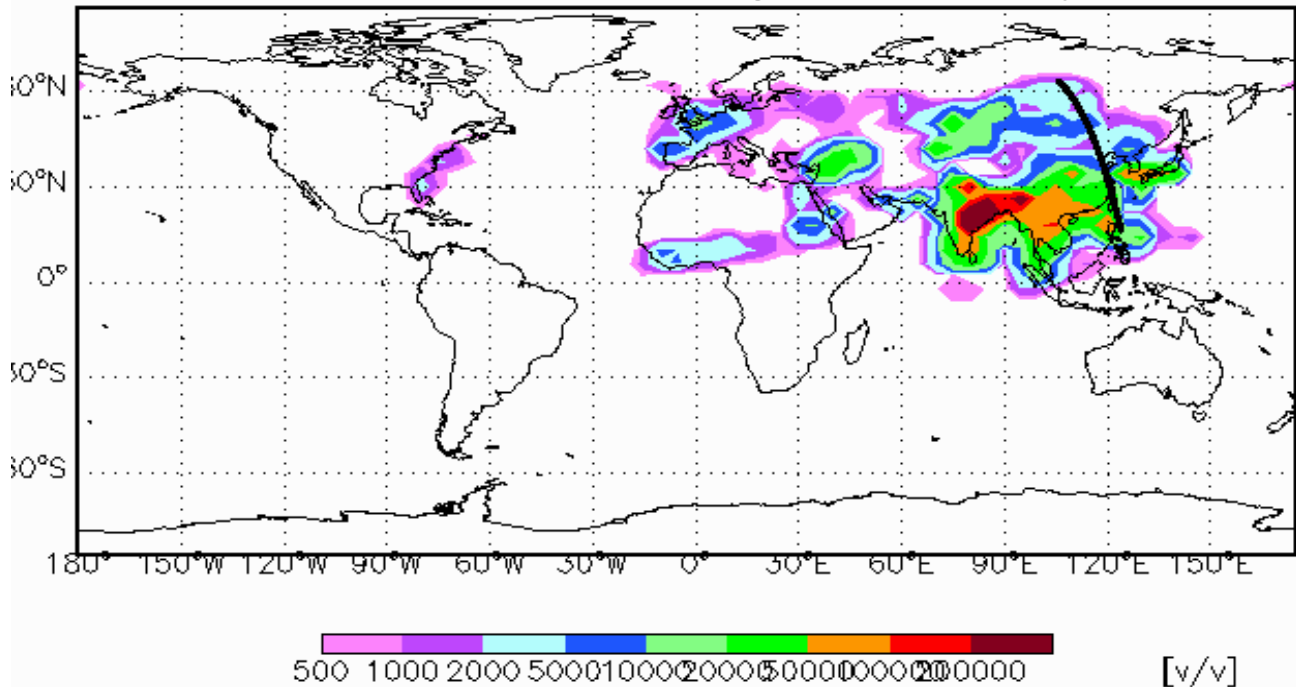


GEOS4 E\_NO 010401 at 00:00 GMT Avg from L=1-30 (0.3-63.1 km)



# GEOS-Chem v7 Adjoint User's Guide

*An adjoint model for GEOS-Chem to perform 4D-Var Data Assimilation and Sensitivity Analysis*

**Kumaresh Singh, Adrian Sandu & Paul Eller**

Department of Computer Science  
Virginia Polytechnic Institute and State University  
Blacksburg, VA 24060, USA

[kumaresh@cs.vt.edu](mailto:kumaresh@cs.vt.edu); [asandu7@cs.vt.edu](mailto:asandu7@cs.vt.edu); [peller@vt.edu](mailto:peller@vt.edu)

**Daven Henze**

Columbia University  
NASA Goddard Institute for Space Studies  
2880 Broadway, New York, NY 10025 USA

[dkh2114@columbia.edu](mailto:dkh2114@columbia.edu)

## Table of Contents

<b>1 Installation</b> .....	<b>4</b>
1.1 Public vs. Beta Version.....	4
1.2 Obtaining Source Code and Run Directory.....	4
1.3 Obtaining Data Directories .....	4
<b>2 Compilation</b> .....	<b>5</b>
2.1 Overview.....	5
2.2 Setting C-Preprocessor Switches.....	5
2.3 Compiling GCv7_ADJ Source Code.....	5
<b>3 Running GC Adjoint Model</b> .....	<b>6</b>
3.1 Mode Selection .....	6
3.2 Forward Mode.....	6
3.3 Sensitivity Mode.....	6
3.4 4D-Var Data Assimilation Mode.....	6
3.5 Finite Difference Test Mode.....	6
<b>4 Code Validation – Finite Difference Test</b> .....	<b>6</b>
4.1 Mode Selection .....	7
4.1.1 Chemistry.....	7
4.1.2 Convection.....	7
4.1.3 Transport.....	8
4.1.4 Emissions/DryDeposition.....	9
<b>5 Sensitivity Test</b> .....	<b>9</b>
5.1 Settings.....	10
5.2 Test Details .....	10

5.3 Sensitivity Results .....	10
<b>6 4D-Var Data Assimilation.....</b>	<b>11</b>
6.1 Experimental Details .....	12
6.2 Settings.....	12
6.3 Results.....	13
<b>7 New Subroutines in GC Adjoint .....</b>	<b>14</b>
7.1 Drivers .....	14
7.2 Subdrivers .....	14
7.3 Transport.....	14
7.4 Convection.....	14
7.5 Chemistry.....	14
7.6 Emissions/DryDeposition .....	14

# 1 Installation

## 1.1 Public vs. Beta Version

The GEOS-Chem v7 adjoint is designed targeting GC user group worldwide to provide them two very important tools: Sensitivity Analysis and Data Assimilation. The adjoint package comes in a similar format as the GEOS-Chem with a source code directory, a run directory and a data directory. Also the coding practice and style guide provided in Chapter 7 of the GC user's manual has been followed so that users find it easier to learn the adjoint basics rather than struggling to understand the code flow.

The beta version of the code has been handed over to other adjoint developers to perform rigorous testing on the code and reduce redundancies if any. The public version will be made available at <http://people.cs.vt.edu/~asandu/CSL/software.php>.

## 1.2 Obtaining Source Code and Run Directory

The source code and run directories could be obtained through the link provided above or simply clicking at: <http://filebox.vt.edu/~kumaresh/links.html>. The current adjoint directories are contained in GNU-zipped TAR file (extension \*.tar.gz). To unzip the file, type:

```
tar zxvf Code.v7-04-10.tar.gz
tar zxvf runs.tar.gz
```

This will create the following directories:

Directory	Description
Code.v7-04-10	Directory containing GEOS-Chem adjoint source code
runs/geos.v7-04-10/run.fullchem.geos4	Contains compilation and run scripts, and other necessary files for compilation and linking purposes

## 1.3 Obtaining Data Directories

The data directories required for the adjoint version are same as for the GEOS-Chem. It can be downloaded from the data server ftp.as.harvard.edu. If one does not have an account on this machine, please contact Harvard University's systems administrator Jack Yatteau.

To download the [4° x 5°](#) data directories, one may use anonymous FTP:

```
ftp ftp.as.harvard.edu
cd pub/geos-chem/data/GEOS_4x5
```

and then download each of the individual subdirectories which contain GEOS-Chem 4°x5° input files. Data directories for other resolutions are also available on this server.

## 2 Compilation

### 2.1 Overview

Similar to GEOS-Chem, the adjoint code can be compiled using the [make](#) utility, which facilitates code maintenance and testing. More information about *make* and Makefile is readily available on the web.

A script to call *make* with the correct Makefile is provided in the Code directory. The script `build` can be used interactively or submitted to a batch queue system. But before compiling, one must choose between a set of available options, as described in the next section.

### 2.2 Setting C-Preprocessor Switches

The proper C-preprocessor switches must be set to tell the compiler which options one would like compiled into the GEOS-Chem executable. These switches are located in header file `define.h`, and are as follows:

Met Fields	
GEOS_3	Use <a href="#">GEOS-3</a> met fields ( <a href="#">48 layer</a> , 1998–2001 )
GEOS_4	Use <a href="#">GEOS-4</a> met fields ( <a href="#">55 layer</a> , hybrid, 1985–2006)

Grids	
GRID4x5	Use <a href="#">4° x 5°</a> grid
GRID2x25	Use <a href="#">2° x 2.5°</a> grid

Compilers	
LINUX_IFORT	Compile for Linux with <a href="#">Intel Fortran Compiler</a> (IFORT 9.x or 10.x)
LINUX_PGI	Compile for Linux with <a href="#">PGI compiler</a>

### 2.3 Compiling GCv7\_ADJ Source Code

The current version of the GC adjoint model comes with Intel Fortran Compiler Makefiles for various analysis modes presented as follows:

Compilers	
Makefile.ifort.orig	For building original GEOS-Chem, Forward mode
Makefile.ifort.fd	For compiling Finite Difference test modules
Makefile.ifort.senst	For compiling Sensitivity Analysis module
Makefile.ifort.4d	For compiling 4D-Var Data Assimilation modules

These selections are made in the `v7-04-10.cmp` file line 21, under the `run` directory. Once the executable `geos` is created, one is ready to run the adjoint with the settings described in the following section.

## 3 Running GC Adjoint Model

### 3.1 Mode Selection

As mentioned in the previous section, there are total four modes of selection provided to a user. However, before running the executable the user will have to take care of the following settings according to the mode selected.

#### 3.1.1 Forward Mode

*input.geos* - Start/End dates, process flags.

*v7-04-10.run* - To be used to run the executable and dump output in a log file.

#### 3.1.2 Sensitivity Mode

*input.geos* - Start/End dates, process flags.

*v7-04-10.run* - To be used to run the executable and dump output in a log file.

#### 3.1.3 4D-Var Data Assimilation Mode

*input.geos* - Start/End dates, process flags.

*v7-04-10.run.4d* - To run the executable and dump output in a log file. [Maximum allowed iteration number (XSTOP) is set here].

*ITER* - Determines which iteration is in progress, must be 0 at the start.

#### 3.1.4 Finite Difference Test Mode

*input.geos* - Start/End dates, process flags.

*v7-04-10.run* - To be used to run the executable and dump output in a log file.

*ITER* - To differentiate between perturbed run, must be 0 at the start.

### 3.2 Data Directories

*run\_directory/* - Outputs from Forward, Finite Difference and 4D-Var modes.

*run\_directory/adjoint* - Output checkpoint files for Finite Difference and Sensitivity Analysis modes.

*run\_directory/opt* - Optimization checkpoint files for 4D-Var Data Assimilation mode.

### 3.3 Other Files and Settings

*createmask.f* - To create observation grid for 4D-Var calculations.

*calcratio.f* - To analyze the ratio of finite difference approximation to the adjoint value.

*clean* - To remove 4D-Var ASCII outputs such as rms value, cost function, etc.

## 4 Code Validation – Finite Difference Test

To obtain consistent data assimilation and sensitivity results, it is necessary to validate all the adjoint science process subroutines. The validation test is designed to compare the adjoint variable with its finite-difference approximation. A detailed description of the test setup is provided as follows.

### Driver Files:

*fd\_driver.f* - Main driver file for finite difference test.  
*subdriver\_fwd\_fd.f* - Forward subdriver routine for finite difference.  
*subdriver\_bwd\_fd.f* - Reverse subdriver routine for finite difference.

Output Files	(use GAMAP to visualize)
<i>run_directory/adjoint/CHEM_CHK_P2.*</i>	Tracer species 1 represents 1-way FD, species 2 represents 2-way FD.
<i>run_directory/adjoint/CHEM_CHK_P3.*</i>	Tracer species 1 represents difference between 2-way FD and adjoint values, species 2 represents difference between 2-way FD and 1-way FD, species 3 represents difference between 1-way FD and adjoint values.
<i>run_directory/fd1_*, fd2_*, adj_*</i>	<i>fd1_*, fd2_*, adj_*</i> contain 1-way FD, 2-way FD and adjoint values respectively to be used as an input for <i>calcratio.f</i> .

## 4.1 Mode Selection

In order to test the overall consistency of the adjoint code, each science process is tested individually. Following are the test setup for each science process. The test results are available in the report submitted to JPL as part of the AIST project.

### 4.1.1 Chemistry

- ➔ In order to perform the **box model testing** for GEOS-Chem chemistry adjoint, we choose a chemical species (say S) with respect to which the derivative of the concentration of another tracer species (say I) has to be calculated, i.e.,  $dC_I/dC_S$ . A forward run of GEOS-Chem is performed for the time period (T) under consideration with an initial perturbation ( $\text{pert}C_{S0}$ ) in S. The concentrations from the final time ( $\text{pert}C_{IT}$ ) are saved for finite difference calculations.
- ➔ A similar forward run is performed but this time with original C and the concentrations are checkpointed every chemistry time step to be used in the adjoint run which follows it. The concentrations for the final time ( $\text{orig}C_{IT}$ ) are again saved.
- ➔ In the adjoint mode, the adjoint tracer array for species I are initialized with unit concentration over the grid points under consideration at the final time (T) integrating backwards to the initial time. At T=0, the adjoint tracer concentrations for species S ( $\text{adj}C_{S0}$ ) are compared with their finite difference approximations in the following manner:

$$\text{adj}C_{S0} \approx (\text{pert}C_{IT} - \text{orig}C_{IT}) / \text{pert}C_{S0}$$

### 4.1.2 Convection

Geos-3 and Geos-4 convection subroutines are quite different and their adjoints have been handled separately. The adjoints are discrete and are generated using TAMC. The results presented below show a good conformation of adjoints with their finite difference approximations. In addition, the execution times for geos-3 and geos-4 convection adjoints are similar to their forward equivalents.

- ➔ In order to perform the **column testing** for GEOS-Chem convection adjoint, we choose a chemical species (say S), a layer (say L) with respect to which the derivative of the concentration of the same species at a higher layer (say H) has to be calculated, i.e.,  $dC_{SH}/dC_{SL}$ . A forward run of GEOS-Chem is performed for the time period (T) under consideration with an initial perturbation ( $pertC_{L0}$ ) in S at layer L only. The concentrations from the final time ( $pertC_{HT}$ ) are saved for finite difference calculations.
- ➔ A similar forward run is performed but this time with original C and the concentrations are checkpointed every convection time step to be used in the adjoint run which follows it. The concentrations for the final time ( $origC_{HT}$ ) are again saved.
- ➔ In the adjoint mode, the adjoint tracer array for species S are initialized with unit concentration over layer H at the final time (T) convecting backwards to the initial time. At T=0, the adjoint tracer concentrations for specie S ( $adjC_{L0}$ ) are compared with their finite difference approximations in the following manner:

$$adjC_{L0} \approx ( pertC_{HT} - origC_{HT} ) / pertC_{L0}$$

### 4.1.3 Transport

Geos-3 and Geos-4 use TPCORE and TPCORE\_FVDAS subroutines respectively for transportation. The adjoints of these subroutines are continuous and are achieved by reversing the wind fields. The results presented below show a good conformation of adjoints with their finite difference approximations considering the fact that continuous adjoints are used.

- ➔ In order to **test** the GEOS-Chem transportation adjoint, we choose a chemical species (say S), a grid point (say g) with respect to which the derivative of the same species at a different grid point (say h) has to be calculated, i.e.,  $dC_h/dC_g$ . For verification purposes, we take an ensemble of such sample grid points (day  $G=\{g_1, g_2, \dots, g_n\}$ ) and perform the finite different test for each of those.
- ➔ A forward run of GEOS-Chem is performed for the time period (T) under consideration, with an initial perturbation ( $pertC_{g0}$ ) in S at grid point g only. The concentrations from the final time ( $pertC_{hT}$ ) are saved for finite difference calculations.
- ➔ A similar forward run is performed but this time with original C and the pressure is checkpointed every transportation time step to be used in the adjoint run which follows it. The concentrations for the final time ( $origC_{hT}$ ) are again saved.
- ➔ In the adjoint mode, the adjoint tracer array for specie S are initialized with unit concentration over grid point h only at the final time (T) transporting backwards to the initial time. At T=0, the adjoint tracer concentrations for specie S ( $adjC_{g0}$ ) are compared with their finite difference approximations in the following manner:

$$adjC_{g0} \approx ( pertC_{hT} - origC_{hT} ) / pertC_{g0}$$



#### 4.1.4 Emissions/DryDeposition

In GEOS-Chem emission and drydeposition is handled through chemistry via fake equations. The rates for these processes are calculated separately and then attached to the chemistry reaction rates. The adjoints of these subroutines are scaled and calculated using the adjoint integrator which provides adjoint with respect to the rates which is then multiplied with the individual rates and is accumulated over time. The results presented below show a good conformation of adjoints with their finite difference approximations considering the fact that continuous adjoints are used.

- ➔ In order to **test** the GEOS-Chem emissions adjoint, we choose an emissions species (say  $S$ ), with respect to which the derivative of a chemical species ( $P$ ) at the same grid point (say  $h$ ) has to be calculated, i.e.,  $dP_h/dS_h$ . For verification purposes, we choose all the grid points and perform the finite different test at all these points.
- ➔ A forward run of GEOS-Chem is performed for the time period ( $T$ ) under consideration, with a perturbation ( $\text{pert}\alpha_0$ ) in the emission rates REMIS for species  $S$ , at all grid points, for all times  $\{0,1,\dots,T-1\}$ . The concentrations from the final time ( $\text{pert}P_T$ ) are saved for finite difference calculations.
- ➔ A similar forward run is performed but this time with original emission rates. The concentrations for the final time ( $\text{orig}P_T$ ) are again saved.
- ➔ In the adjoint mode, the adjoint tracer array for species  $P$  are initialized with unit concentration over all the grid points at the final time ( $T$ ) integrating backwards to the initial time. At  $T=0$ , the adjoint tracer concentrations for species  $S$  ( $\text{adj}S_0$ ) are compared with their finite difference approximations in the following manner:

$$\text{adj}S_0 \approx (\text{pert}P_T - \text{orig}P_T) / \text{pert}\alpha_0$$

## 5 Sensitivity Test

Sensitivity analysis is an approach to quantify the changes in output due to different sources of variation. Since adjoint calculations are receptor based, one can calculate the sensitivity of an output with respect to large number of parameters.

In order to perform adjoint sensitivity tests on the validated GEOS-Chem adjoint model, it is required to perform one forward and one reverse mode of simulation. In the forward run, the concentration values and air densities are checkpointed and in the reverse mode they are utilized for adjoint calculations. A receptor (cost-function) measuring certain species (say  $S$ ) is defined at a given location at the final time in the reverse mode. At  $T=0$  (end of the simulation), we obtain sensitivities of  $S$  with respect to the tracers and emission species.

## 5.1 Settings

### Driver Files:

- senst\_driver.f* - Main driver file for sensitivity analysis.
- subdriver\_fwd\_senst.f* - Forward subdriver routine for sensitivity analysis.
- subdriver\_bwd\_senst.f* - Reverse subdriver routine for sensitivity analysis.
- read\_sciao3\_mod.f* - Subroutine to initialize tracer adjoint, current setup for TES trajectory.

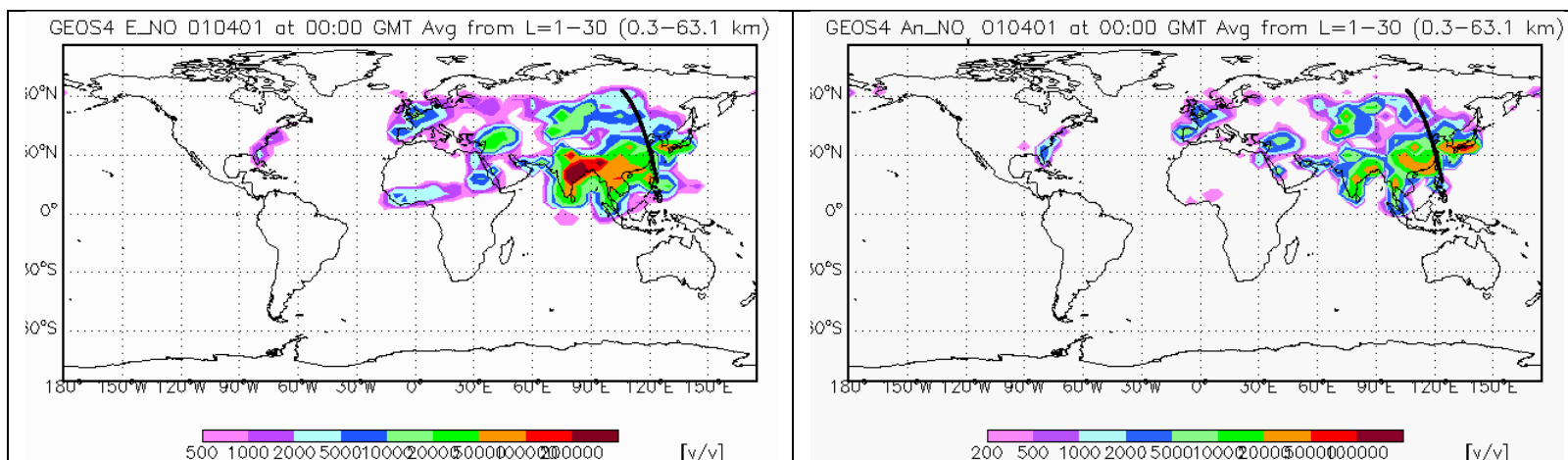
Output Files	(use GAMAP to visualize)
<i>run_directory/adjoint/ADJ.*</i>	Tracer species 1-43 represent sensitivity of receptor with respect to GEOS-Chem tracers 1-43.
<i>run_directory/adjoint/CHEM_CHK_P2.*</i>	Tracer species 1-12 represent sensitivity of receptor with respect to total emissions of 12 GC emission species, species 13-35 with respect to total dry deposition of 23 GC ddep species
<i>run_directory/adjoint/CHEM_CHK_P3.*</i>	Tracer species 1-35 represent sensitivity of receptor with respect to individual source emissions

## 5.2 Test Details

Sensitivity analysis test requires one forward and one backward run. The forward run is with the original concentration field without any perturbation. In the backward run, we initialize the tracer adjoint with the derivative of the cost function under consideration. Once the tracer adjoint is initialized, the adjoint values can be tracked every dynamic time step for *transport*, *chemistry* and *convection* and at the end for the *emissions/dry deposition*.

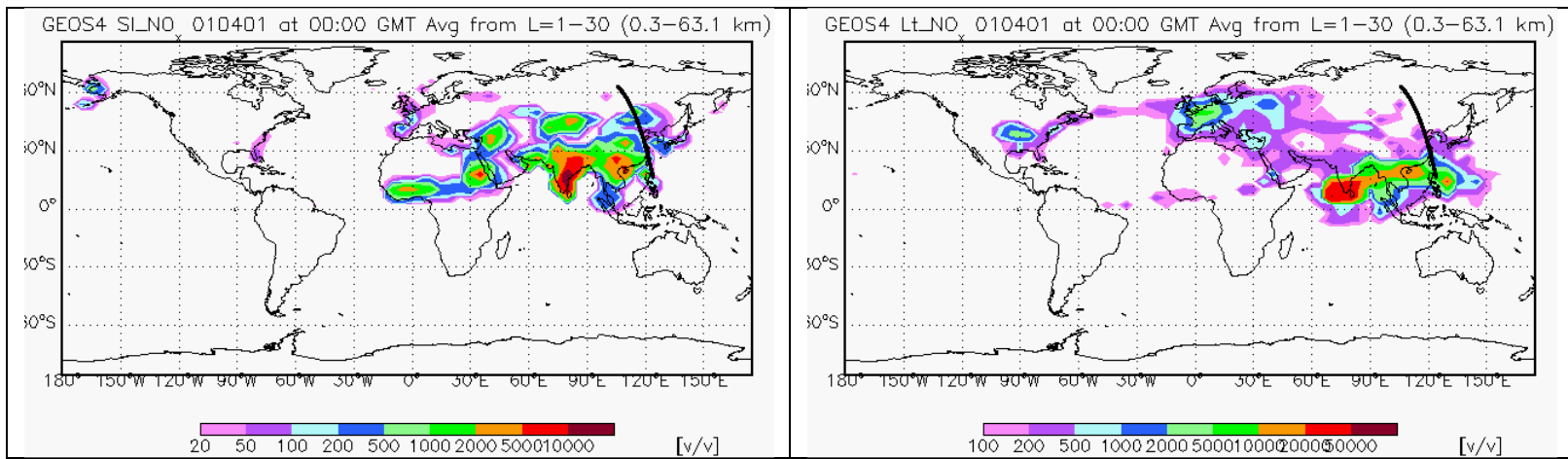
## 5.3 Sensitivity Results

The following emission plots are generated from 48 hours of adjoint run 010401:000000 to 010403:000000, all layers. The tracer adjoint is initialized by unit value at the TES trajectory and then the emission adjoint values on April 1<sup>st</sup>, 2001 are plotted.



**Fig 1(a)** Ox Adjoint with respect to Total NOx emissions.

**Fig 1(b)** Ox Adjoint with respect to Anthro NOx Emissions.



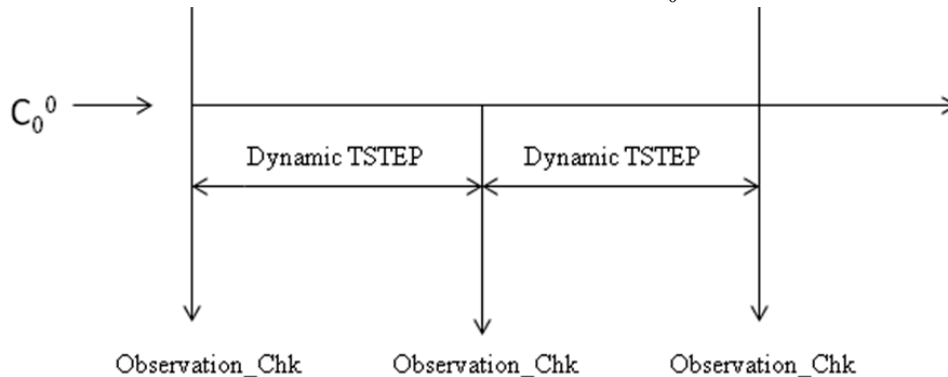
**Fig 1(c)** Ox Adjoint with respect to Soil NOx NOx emissions.

**Fig 1(d)** Ox Adjoint with respect to Lightning emissions.

## 6 4D-Var Data Assimilation

4D-Var data assimilation allows the optimal combination of three sources of information: an a priori (background) estimate of the state of the atmosphere, knowledge about the physical and chemical processes that govern the evolution of pollutant fields as captured in the chemistry transport model (CTM), and observations of some of the state variables.

In order to carry out data assimilation experiment on GEOS-Chem, an interface has been developed to integrate the science processes with the optimization routine L-BFGS. An observation grid is extracted by performing a forward GC run on the initial concentration field  $c_0^0$ .



A perturbation is then introduced in  $c_0^0$  to produce  $c_p^0$ .

$$c_p^0 = c_0^0 + \text{perturbation}$$

This perturbed concentration is then transferred to the optimization subroutine in order to obtain the best estimate  $c_{op}^0$  of the original concentration  $c_0^0$  after several iterations.

At iteration 0,

$$x_0 = c_p^0$$

At each subsequent iteration  $k$  ( $k \geq 1$ ),

$$\begin{aligned} x_{k+1} &\leftarrow \text{L-BFGS}(x_k, f, g) \\ c_{op}^0 &\leftarrow x_{k+1} \\ (f, g) &\leftarrow \text{reverse\_mode}(c_{op}^0, \text{Observation\_Chk}) \end{aligned}$$

where,  $f$  is the cost function and  $g$  is the gradient of the cost function.

## 6.1 Experimental Details

In our test case the cost function and its gradient are being defined as:

$$f = (1/2) \sum (c_{op}^{k,m} - c_{obs}^{k,m})^T (c_{op}^{k,m} - c_{obs}^{k,m})$$

and

$$g = \sum (c_{op}^{k,m} - c_{obs}^{k,m})$$

where  $k = 1, 2, \dots, 2 * T_{final}$  is the total number of science process runs in the forward mode and  $m$  is a 4-tuple observation grid. The observation grid specifications and perturbation amounts used are presented as follows:

Observation Grid				Perturbation amount
Columns	Rows	Layer	Species	$c_p^0 = c_0^0 * (1 + \text{eps}),$ $\text{eps} = 0.1$
x1=1,x2=IIPAR	x1=1,x2=JJPAP	1:LLPAR	Ox	

## 6.2 Settings

### Important Files:

- 4dvar\_driver.f* - Main driver file for 4D-Var data assimilation.
- subdriver\_fwd\_4d.f* - Forward subdriver routine for data assimilation.
- subdriver\_bwd\_4d.f* - Reverse subdriver routine for data assimilation.
- calc\_obsgrad.f* - To calculate observation cost function and update tracer adjoints.
- calc\_bggrad.f* - To calculate background cost function and update tracer adjoints.

The subroutine `calc_obsgrad()` is called to initialize the tracer adjoint at the final time and every dynamic time step to update it. The `calc_bggrad()` is called at  $t=0$  to update the adjoint values at the initial time.

Output Files	(use GAMAP to visualize)
<i>run_directory/opt/ORIG_CHK.*</i> , <i>PERT_CHK.*</i> <i>OPTZ_CHK.*</i>	<i>ORIG_CHK*</i> and <i>PERT_CHK*</i> represent the original and perturbed concentrations at the initial time. <i>OPTZ_CHK*</i> represents the best estimate at each iteration.
<i>run_directory/opt/DIFFPERT_CHK.*</i> , <i>DIFFOPTZ_CHK.*</i>	Represent the difference between perturbed and original at the start, and difference between optimized and original at each iteration respectively.
<i>run_directory/cost_func*</i> , <i>rms_value*</i> ,	<i>Cost_func*</i> stores total cost function for each iteration,

<i>model_runs*</i> , <i>obs_cost*</i> , <i>bg_cost*</i>	<i>rms_value*</i> stores root mean square value for each iteration, <i>model_runs*</i> stores the total number of model runs used by l-bfgs subroutine, <i>obs_cost*</i> stores cost function value due to observation difference, <i>bg_cost*</i> stores cost function due to background difference.
---	---

### 6.3 Results

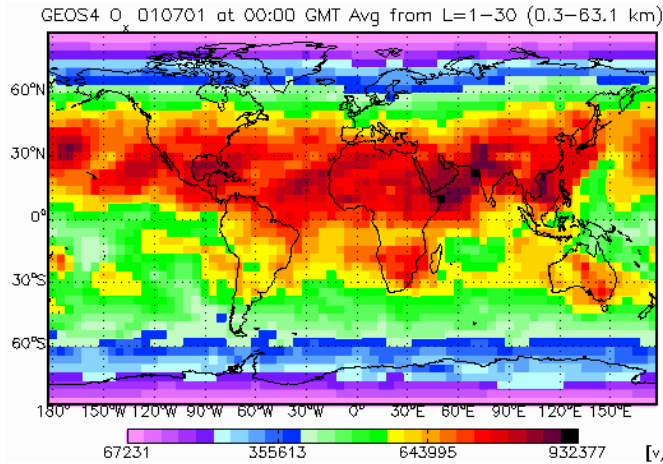
To validate the developed test-bed, two sets of plots are being generated; one for the difference  $c_p^0 - c_0^0$  and the second for  $c_{op}^0 - c_0^0$ . The idea is to illustrate the fact that data assimilated concentration fields are the best estimates of the original concentration fields. Figure 2(a) and 2(b) reflect that  $c_{op}^0$  is quite close to  $c_0^0$ . A cost-function and root-mean square vs. model runs plot, Figure 2(c) and 2(d), are also provided for further validations. This simulation is performed for geos-4 with chemistry, transport and convection, for the month of July 2001, for 12hrs, starting from 000000 hrs.

### Conclusion

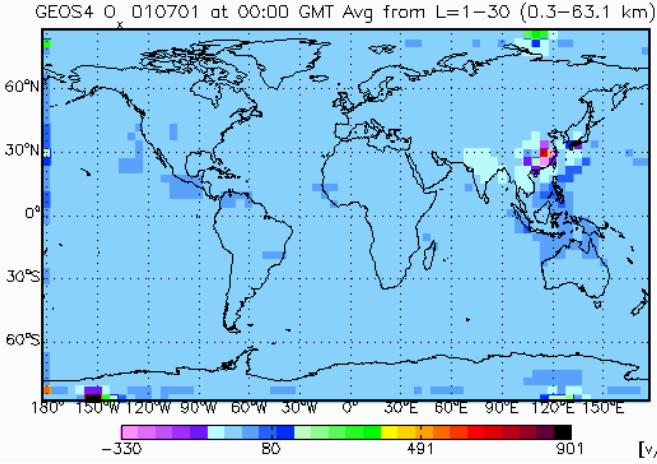
Figure 2(a) and 2(b) show a good recovery of the original concentration field from the perturbed field. Figure 2(c) and 2(d) on the other hand reflect that there is a significant cost function decrease with model runs which means that the solution converges to the original concentration field at the grid points. However, to illustrate an overall convergence of the solution, we calculate the root-mean square (RMS) value given by:

$$RMS = \| c_{op}^0 - c_0^0 \| / \| c_0^0 \|$$

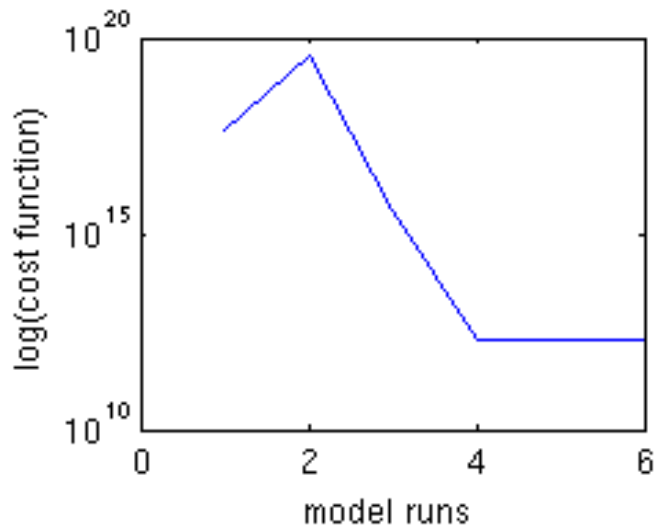
The decrease in this RMS value with model runs signifies that there is an overall convergence of the solution.



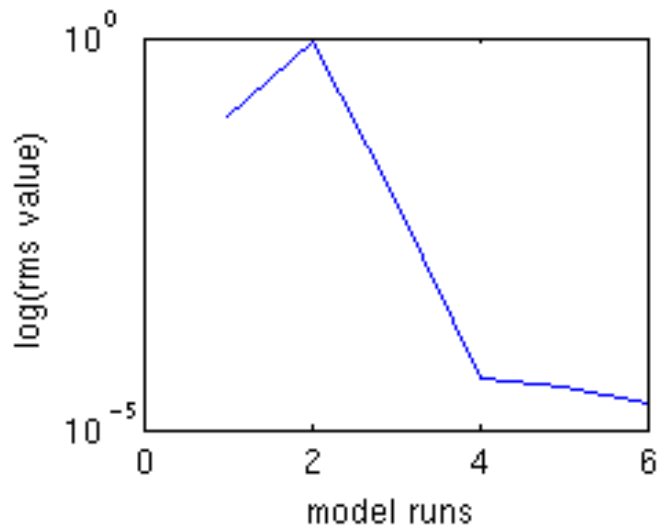
**Fig 2(a)** Difference between perturbed and reference concentration ( $c_p^0 - c_0^0$ ).



**Fig 2(b)** Difference between optimized and reference concentration ( $c_{op}^0 - c_0^0$ ).



**Fig 2(c)** Log(Cost-function) vs. model runs.



**Fig 2(d)** Log(root-mean square value) vs. model runs.

## 7 New Subroutines in GC Adjoint

Listed below are the new or modified subroutines brought into the GCv7\_ADJ package as compared to the original GEOS-Chem package.

### 7.1 Drivers

### 7.2 Subdrivers

### 7.3 Transport

### 7.4 Convection

### 7.5 Chemistry

### 7.6 Emissions/DryDeposition