High performance distributed data reduction and analysis with the netCDF Operators (NCO)

Charlie Zender[1] and Daniel Wang[2]
[1]Department of Earth System Science
[2]Department of Electrical Engineering and Computer Science
University of California, Irvine

Thanks to:
H. Butowsky (UCI), S. Jenks (UCI), H. Mangalam (UCI),
R. Peterson (U. Alaska), R. Rew (Unidata)

Presented to:
American Meteorological Society 2007 Annual Meeting, San Antonio TX
(Web: http://dust.ess.uci.edu/smn/smn_nco_ams_200701.pdf)

1. IPCC Data Reduction Prototype Problem

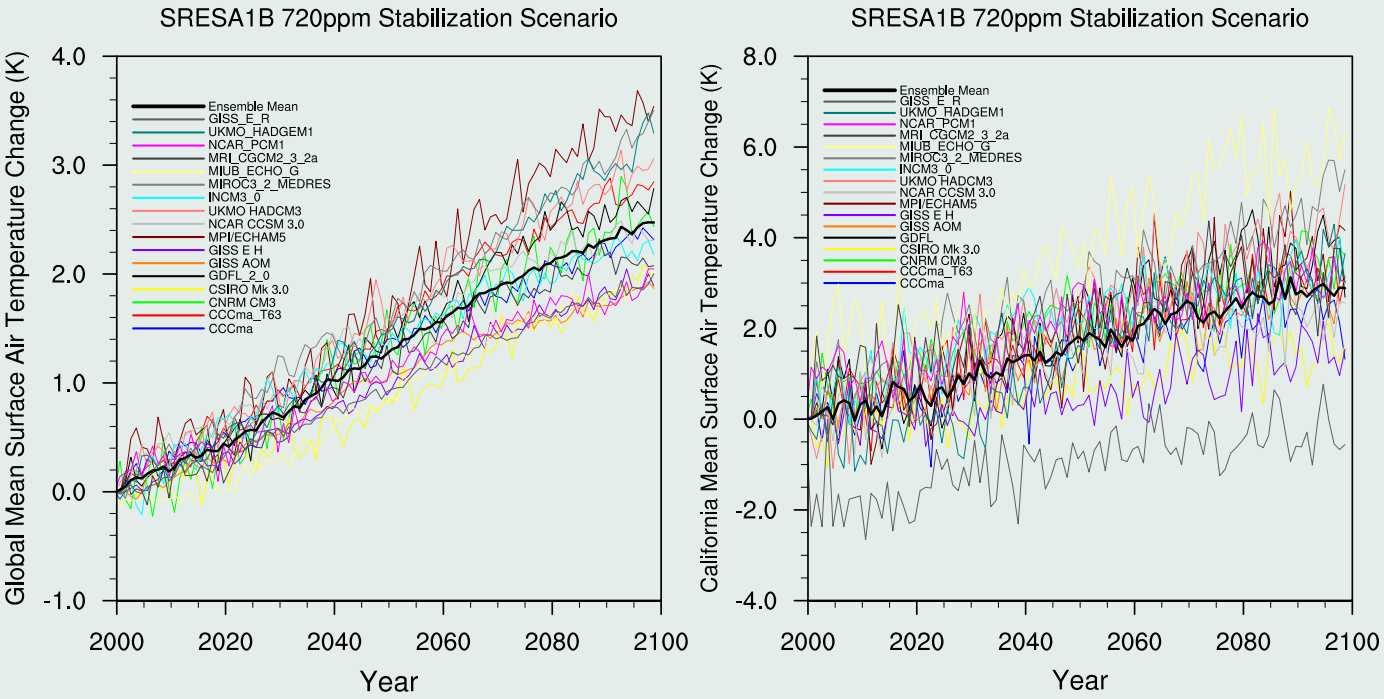

Figure 1: Predicted Global (left) and California (right) annual-mean temperature from 2000–2099 under SRESA1B 720 ppm CO_2 stabilization scenario. Temperature scales differ.

Our "Holy Grail" is to make Distributed Data Reduction and Analysis (DDRA) more practical, less painful, and quicker for non-computer scientists

2. Three Levels of Optimization

To achieve high performance in data processing, NCO identifies and exploits parallelism on multiple-levels, and uses smart arithmetic algorithms:

- Parallelize across commands within script:
  - Script Workflow Analysis for MultiProcessing (SWAMP)
  - Schedule execution based on dependency tree analysis
  - Server-side processing reduces bandwidth for distributed data
  - Store intermediate files in RAM and transfer results only
- Parallelize arithmetic across variables within file:
  - Symmetric Multi-Processing (OpenMP)
  - Message Passing Interface (MPI)
- Optimize low-level, repetitive arithmetic algorithms:
  - Most Rapidly Varying (MRV)
  - Weight Re-use (WRU)

Table 1: **NCO Operator Summary** (*Zender*, 2006)

Command	Name (primary functionality)	Type	MFO	Par.
`ncap`	Arithmetic Processor (algebra, scripts)	A		
`ncatted`	Attribute Editor (change attributes)	M		
`ncbo`	Binary Operator (subtraction, addition)	A		✓
`ncea`	Ensemble Averager (means, min/max)	A	✓	✓
`ncecat`	Ensemble Concatenator (join files)	M	✓	✓
`ncflint`	File Interpolator	A		✓
`ncks`	Kitchen Sink (sub-set, hyperslab)	M		
`ncpdq`	Pack Data, Permute Dimensions	A/M		✓
`ncra`	Record Averager (means, min/max)	A	✓	✓
`ncrcat`	Record Concatenator (join time-series)	M	✓	✓
`ncrename`	Renamer (rename any metadata)	M		
`ncwa`	Weighted Averager (average, integrate)	A		✓

3. File-level Paradigm

One command Local Data Analysis:

```
ncra 19??.nc average.nc
```

and Distributed Data Analysis:

```
ncra -n 100,2,1 1900.nc
-p http://data.edu/opendap
-o average.nc
-v 'H2O.?'
-d time,1900.,1957.
-d lat,-10.,10.
```

1. Ingests multiple files (e.g., 100)
2. Remote access methods: OPeNDAP, FTP, MSS, `scp`, SFTP
3. Processes (e.g., averages) *all* variables in file
4. Input data all automatically:
   (a) Threaded over variables
   (b) Unpacked
   (c) Type-promoted
   (d) Broadcast
   (e) Masked for missing values
5. Selected run-time options:
   (a) Subset variables
   (b) Hyperslab inter/intra-file

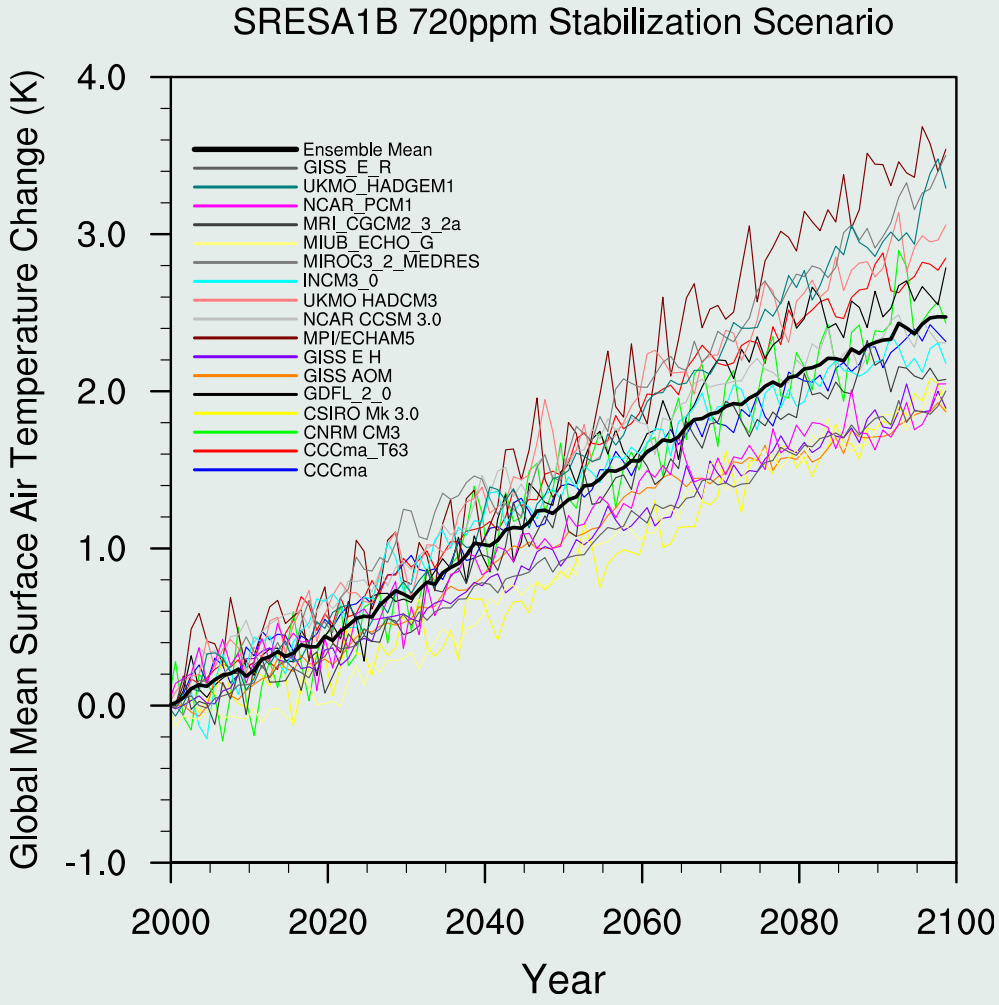

Figure 2: Predicted Global annual-mean temperature from 2000–2099 under SRESA1B 720 ppm CO_2 stabilization scenario.

4. NCO Analysis Script Example

IPCC Distributed Data Reduction Shell Script for Figure 2:

```
models='cccma_cgcm3_1 cccma_cgcm3_1_t63 cnrm_cm3 csiro_mk3_0 \
        gfdl_cm2_0 gfdl_cm2_1 giss_aom giss_model_e_h giss_model_e_r \
        iap_fgoals1_0_g inmcm3_0 ipsl_cm4 miroc3_2_hires miroc3_2_medres \
        miub_echo_g mpi_echam5 mri_cgcm2_3_2a ncar_ccsm3_0 ncar_pcm1 \
        ukmo_hadcm3 ukmo_hadgem1'
variables='tas pr'
scenarios='sresa1b sresa2 sresb1'
for scn in $scenarios; do
  for mdl in $models; do
    ncwa -O -v $variables -w area -a lat,lon \
         -p http://user:password@climate.llnl.gov/cgi-bin/dap-cgi.py/ipcc4/$scn/$mdl \
         pcmdi.ipcc4.$mdl.$scn.run1.atm.mo.xml $scn_$mdl_200001_209912.nc
    ncwa -F -d time,1,12 $scn_$mdl_200001_209912.nc $scn_$mdl_2000.nc
    ncdiff $scn_$mdl_200001_209912.nc $scn_$mdl_2000.nc $scn_$mdl_anm.nc
  done # end loop over model
  ncea *_200001_209912.nc $scn_avg_200001_209912.nc
  ncwa -F -d time,1,12 $scn_avg_200001_209912.nc $scn_avg_2000.nc
  ncdiff $scn_avg_200001_209912.nc $scn_avg_2000.nc $scn_avg_anm.nc
done # end loop over scenario
```

5. Computational Model with Optimizations

The NCO computational model (*Zender and Mangalam*, 2007) accurately predicts cost and throughput for high-level arithmetic and I/O solely from file metadata. For example, weighted averaging costs:

$$I(\text{Un-Opt}) \propto N[34R + 8R_{\text{w}} + 25 + W + (W+11)N_{\text{A}}^{-1}] + B \quad (1a)$$

$$I(\text{WRU}) \propto N[28R + 0 \quad + 23 + W + (W+11)N_{\text{A}}^{-1}] + B \quad (1b)$$

$$I(\text{MRV}) \propto N[\ 6R + 8R_{\text{w}} + 17 + W + (W+11)N_{\text{A}}^{-1}] + B \quad (1c)$$

$$I(\text{WRU+MRV}) \propto N[\quad 0 + 0 \quad + 15 + W + (W+11)N_{\text{A}}^{-1}] + B \quad (1d)$$

where $I(\text{Un-Opt})$ is the un-optimized operation count, and $I(\text{WRU})$, $I(\text{MRV})$, and $I(\text{WRU+MRV})$, include the WRU, MRV, and both optimizations, respectively. Here R and R_{w} are the ranks of the variable and weight (e.g., gridcell area), respectively, W is the data wordsize (i.e., four or eight-bytes), and N_{A} and N_{w} are the products of the sizes of the averaged and weight dimensions, and $B = (W+2)N_{\text{w}}$.

Table 2: **Test File Geometries**

	Satellite	GCM
Max. Rank R	2	4
Variables	8	128
Time	—	8
Level	—	32
Latitude	2160	128
Longitude	4320	256
Elements N [#]	75×10^6	285×10^6
Total Size [MB]	299	1143

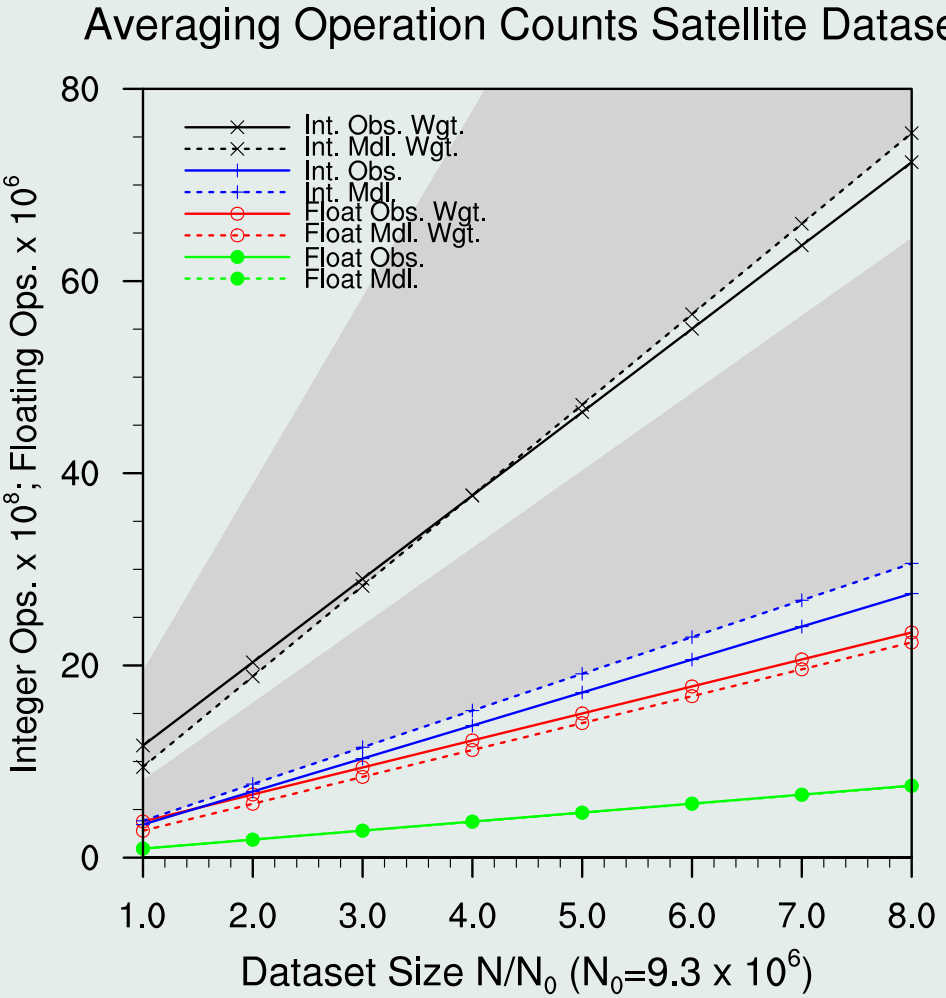

Figure 3: Observed (solid) and predicted (dashed) integer I and floating point F operations necessary to average an N element dataset with (upper two sets of curves) and without (lower two sets of curves) weighting. Grey areas indicate I predicted for rank $R = 2$–5 datasets (*Zender and Mangalam*, 2007).

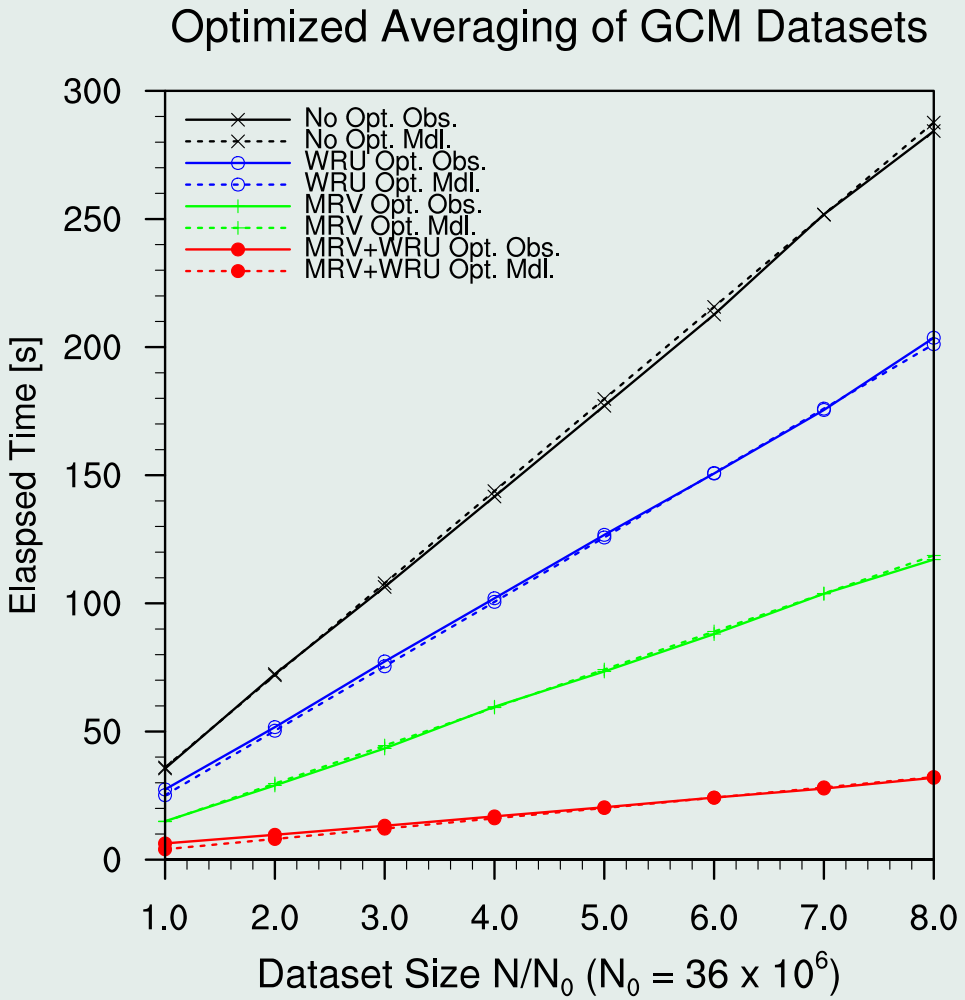

Figure 4: Observed (solid) and predicted (dashed) elapsed times to perform a weighted average of an N element GCM-geometry ($R = 4$) dataset with and without (1a) WRU (1b) and MRV (1c) and both (1d) optimizations. (*Zender and Mangalam*, 2007).

6. Lessons Learned

Design for High Performance Distributed Terascale Data Reduction/Analysis:

- Support shell scripting for power, flexibility
- Optimize independently at script, operator, algorithm levels
- Use standard protocols (DAP, THREDDS) for network transparency
- File level paradigm works, eases SMP/MPI implementation
- Metadata alone predict I/O, arithmetic costs

Remaining Challenges:

- Use computational model to allocate/schedule Grid resources
- Exploit MPI2 parallel I/O (pnetcdf, netCDF4)
- Install/test NCO/SWAMP for DDRA on your data servers (please contact me if interested!)

7. References

References

Zender, C. S. (2006), netCDF Operators (NCO) for analysis of self-describing gridded geoscience data, *Submitted to Environ. Modell. Softw.*, available from http://dust.ess.uci.edu/ppr/ppr_Zen07.pdf.

Zender, C. S., and H. J. Mangalam (2007), Scaling properties of common statistical operators for gridded datasets, *In Press in Int. J. High Perform. Comput. Appl.*, available from http://dust.ess.uci.edu/ppr/ppr_ZeM07.pdf.