Recent Developments in JPL/USC GAIM

Philip Stephens, Brian Wilson, Xiaoqing Pi, Vardan Akopian, Attila Komjathy, B. Iijima, Miguel Dumett and Anthony J. Mannucci

Jet Propulsion Laboratory

California Institute of Technology

M/S 238-600

4800 Oak Grove Drive

Pasadena CA 91109

Outline

- Overview of GAIM
  - Nested Grid
- Data Processing Front-end
  - Ground data
- Real-time system
  - Overview
  - Preliminary Results
- GAIM at CCMC
- Summary

JPL/USC GAIM Overview

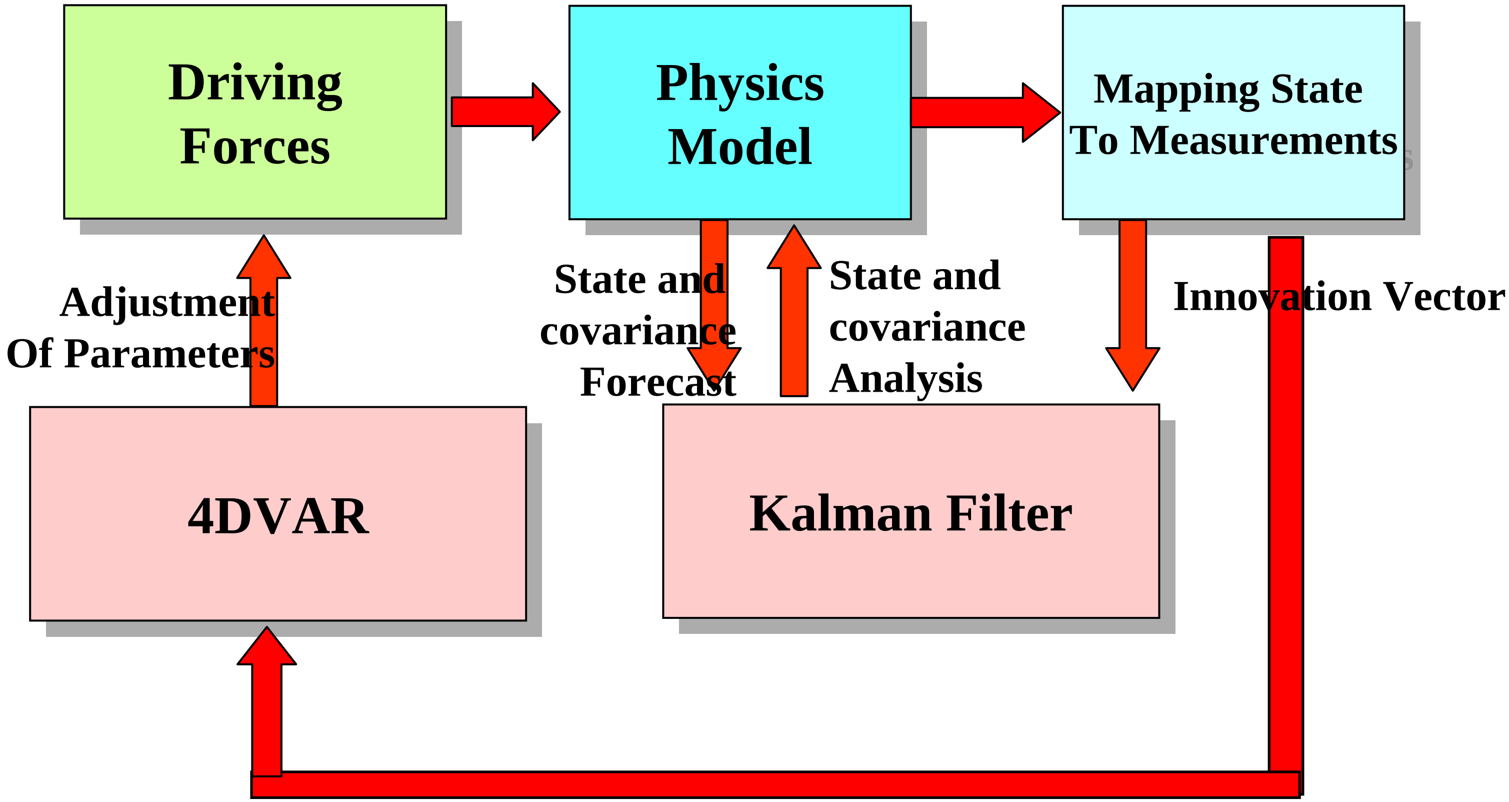
Global Assimilative Ionospheric Model

- **What does GAIM do?**
  - **GAIM is used to model the 3d structure of the ionosphere**
  - **Assimilates data from multiple sources using a Kalman Filter algorithm**
    - 4DVar assimilation also possible but currently too sensitive for general use
- **What does GAIM produce?**
  - **3d density of electron content in the ionosphere**
  - **This can be integrated and the vertical total electron content (VTEC) can be generated**
- **What is the input to GAIM?**
  - **Most common input is the integrated total electron content along the line-of-sight between a ground GPS receiver and a GPS satellite (slant TEC).**
    - At least 200 ground stations to provide global coverage

GAIM Input Data Types

- **Ground GPS Data (Absolute TEC)**
  - **>150 5-min. to Hourly Global GPS Ground Stations**
  - **Assimilate >300,000 TEC points per day (**
- **Space GPS Data (Relative TEC)**
  - **CHAMP (@ 440 km)**
  - **SAC-C (@ 700 km)**
  - **IOX (@ 800 km)**
  - **GRACE (@ 350 km)**
  - **Topex/Poseidon (@1330 km) (Upward looking only)**
  - **Jason 1 (@1330 km) (Upward looking only)**
  - **C/NOFS & COSMIC constellation**
- **UV Airglow: Limb & Nadir Scans**
  - **LORAAS on ARGOS, GUVI on TIMED**
  - **SSUSI/SSULI on DMSP and NPOESS**
- **Other Data Types**
  - **TEC from TOPEX & JASON Ocean Altimeters**
  - **Ionosonde**
  - **DMSP, CHAMP, C/NOFS in situ density**
  - **C/NOFS Electric fields**
  - **GRACE Cross links**
  - **ISR**

Global Assimilative Ionospheric Model
Data Assimilation Process

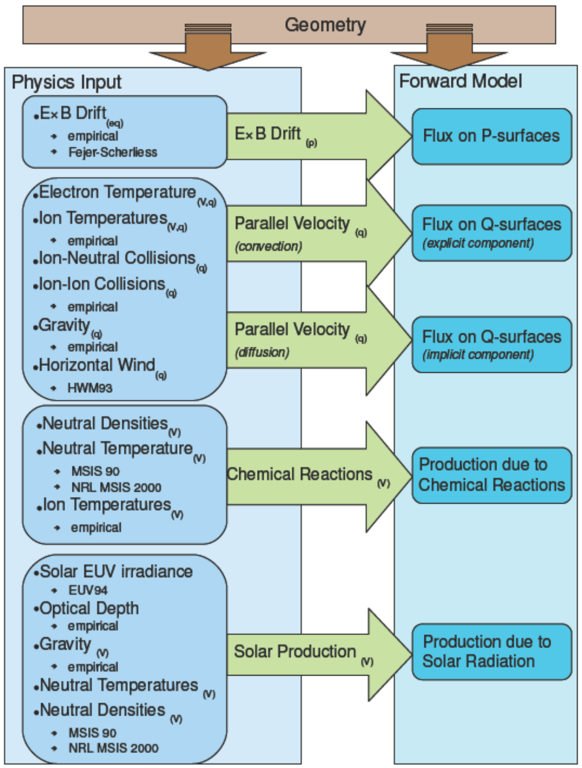


- **4-Dimensional Variational Approach**
  - **Minimization of cost function by estimating driving parameters**
  - **Non-linear least-square minimization**
  - **Adjoint method to efficiently compute the gradient of cost function**
  - **Parameterization of model "drivers"**
- **Kalman Filter**
  - **Recursive Filtering**
  - **Covariance estimation and state correction**

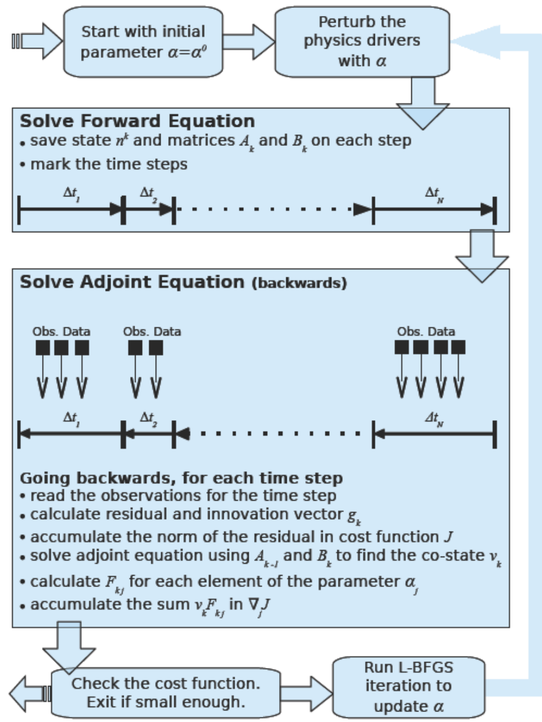
  Optimal interpolation

  Band-Limited Kalman filter

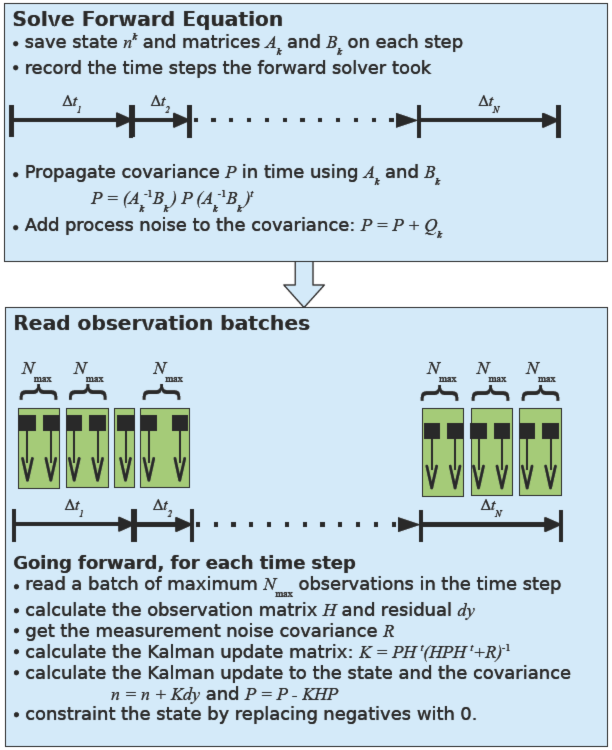
JPL/USC GAIM++



Forward Model With Adjoint

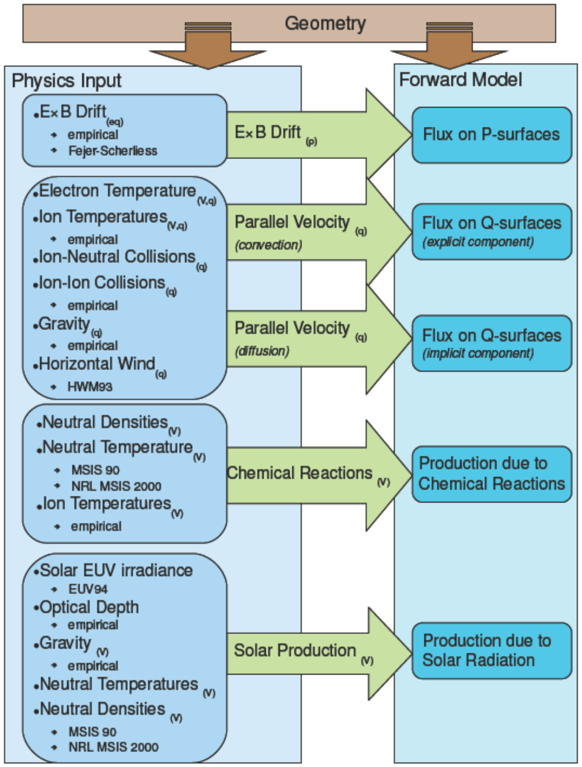


4DVAR

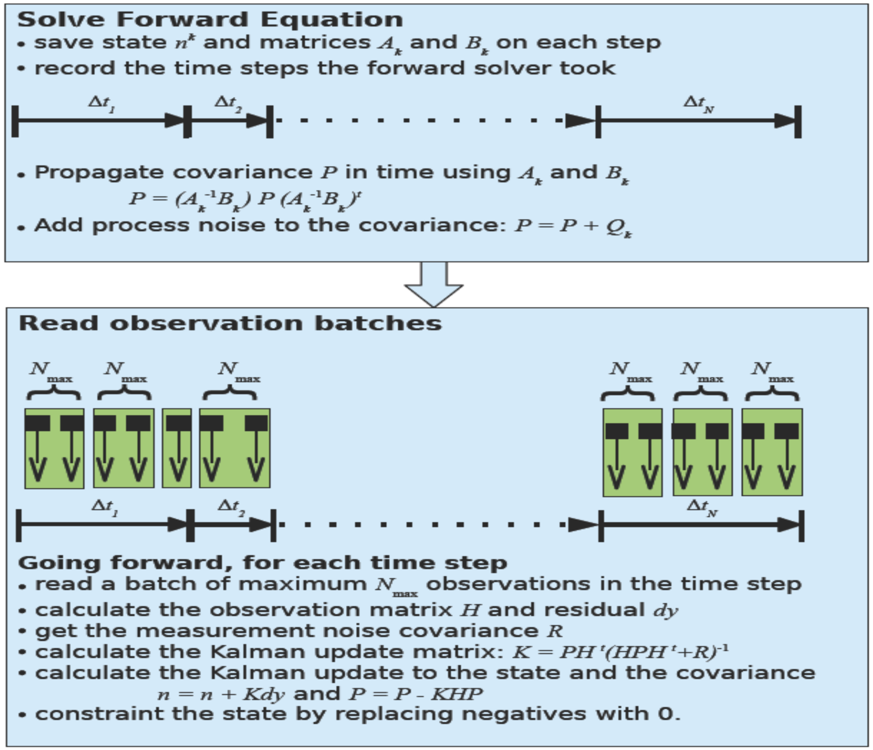


Kalman Filter

JPL/USC GAIM++



Forward Model



Kalman Filter

Kalman Filter Equations

State Model

$$x_{k+1}^t = \Psi_k x_k^t + \varepsilon_k^q$$

Measurement Model

$$m_k^o = H_k x_k^t + \varepsilon_k^o$$

Noise Model

$$\varepsilon_k^o = \varepsilon_k^m + \varepsilon_k^r$$

$$E\left(\varepsilon_k^m, \varepsilon_k^{mT}\right) = M_k$$

$$E\left(\varepsilon_k^r, \varepsilon_k^{rT}\right) = R_k$$

$$E\left(\varepsilon_k^q, \varepsilon_k^{qT}\right) = Q_k$$

$$x_k^a = x_k^f + K_k\left(m_k^o - H_k x_k^f\right)$$

$$K_k = P_k^f H_k^T \left(H_k P_k^f H_k^T + R_k + Q_k\right)^{-1}$$

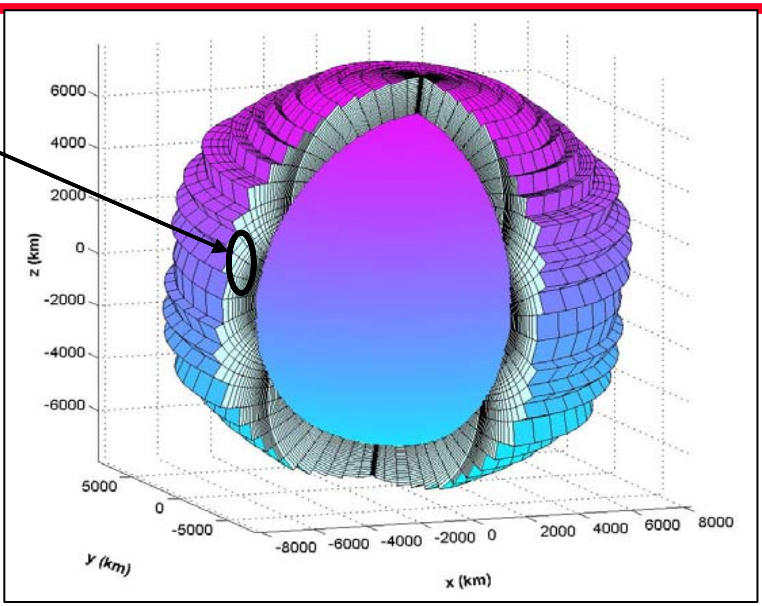
$$P_k^a = P_k^f - K_k H_k P_k^f$$

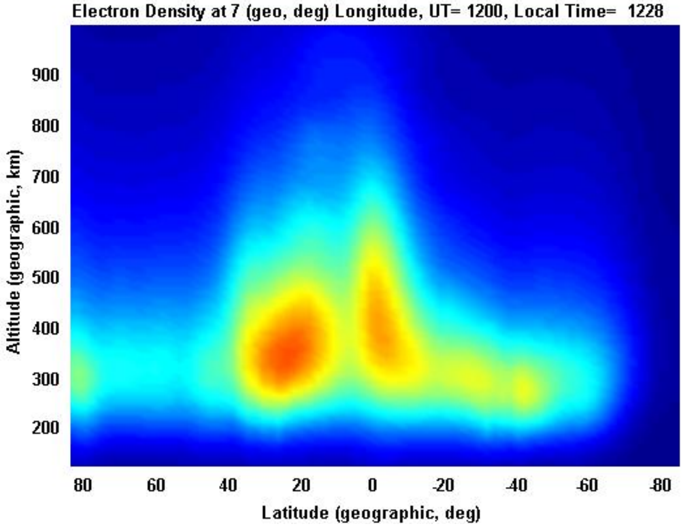
$$x_{k+1}^f = \Psi_k x_k^a$$

$$P_{k+1}^f = \Psi_k P_k^a \Psi_k^T + Q_k$$

Band-Limited Kalman Filter

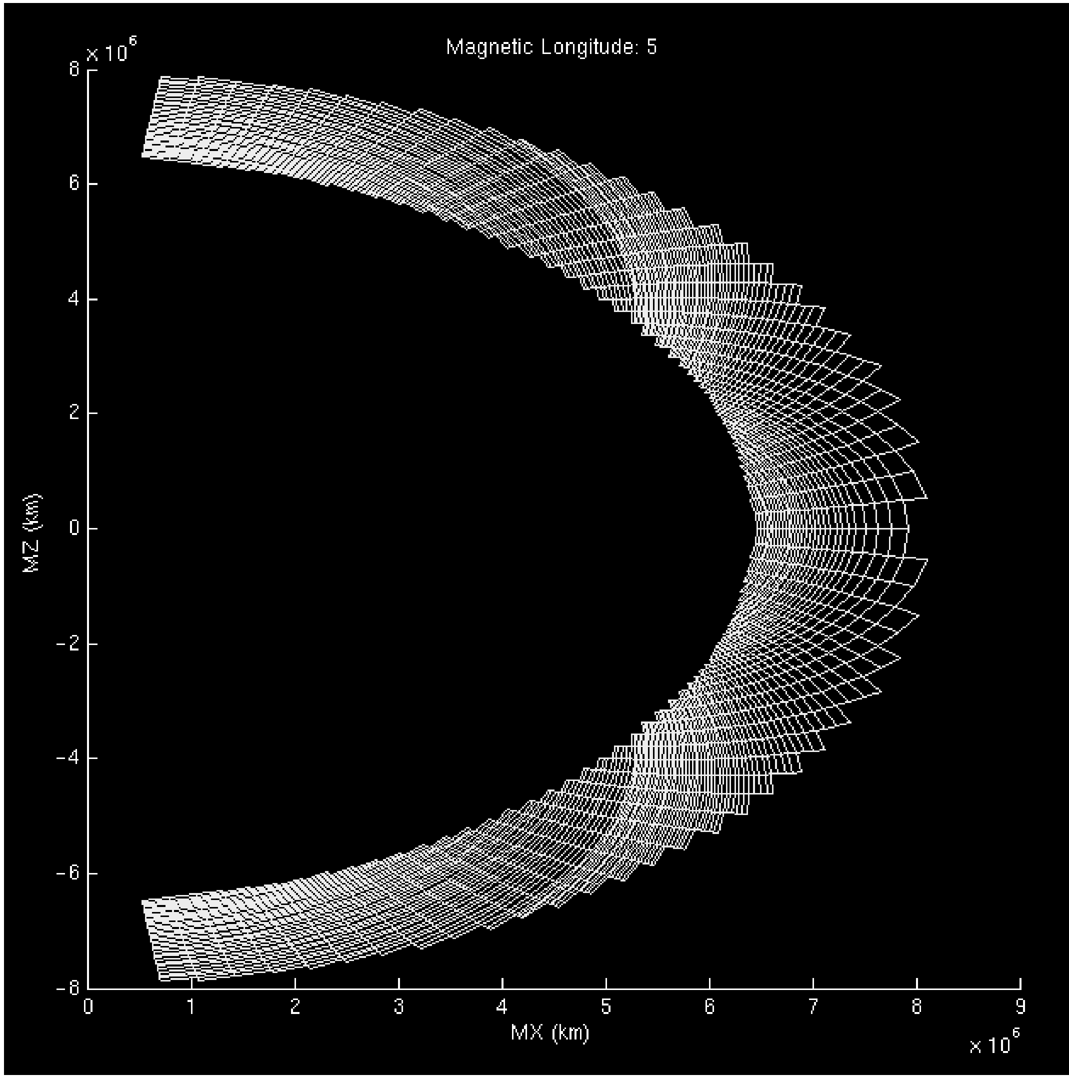
- **Approximate Kalman**: Save only part of covariance matrix based on physical correlation lengths.
- Tested extensively with real data: Ground GPS TEC from 100-200 global sites.
- Validate densities against:
  - Vertical TEC obs. From TOPEX
  - Ionosonde FoF2, HmF2, & bottomside profiles
  - Slant TEC obs. from independent ground GPS sites.
  - Density profiles retrieved from space-based GPS occultations





Medium Resolution Kalman Assimilation

- **Resolution:** **2.5 deg. Lat.**
  10 deg. Lon.
  40 km Alt.
- **No. of grid cells: 100,000**
- **Sparse Kalman filter:**
  - Update & propagate covariance
  - Truncate off-diagonal covariance that is beyond physical correlation lengths



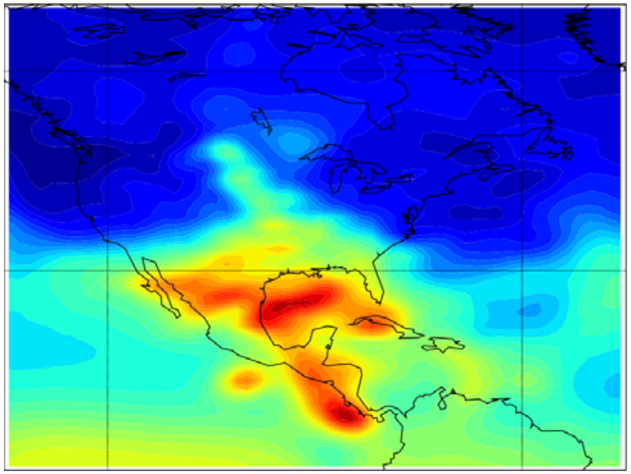
Nested Grid (NGAIM)

- NGAIM grid resolutions:
  - Global grid resolution: 2.5 x 10 degrees lat/lon, 40 km altitude
  - Nested grid resolution: 1 x 2 degrees lat/lon, 40 km altitude
- Forward physics models are coupled at boundary of nested region
  - Global grid provides density and flux on nested boundary using ghost cells
  - Two forward physics models, two Kalman filter runs
  - Both density grids used to properly model TEC links
- Two datasets investigated:
  - Quiet day, July 7, 2009
    - 200 GPS stations for outer region
    - About 50 GPS station for inner region including 8 stations near radar installation
  - Halloween storm (Oct 29-31, 2003)
- Advantages: near real-time execution; can run multiple nested grids at one time (on separate CPUs)

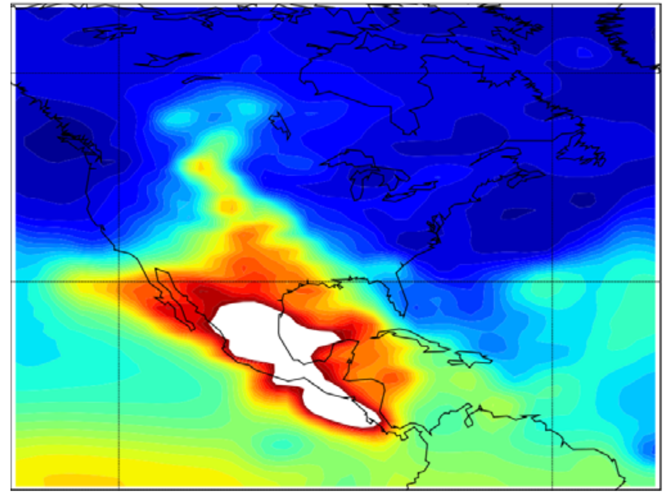
October 29, 2003 NGAIM VTEC Movie

(White > 200 TECU)

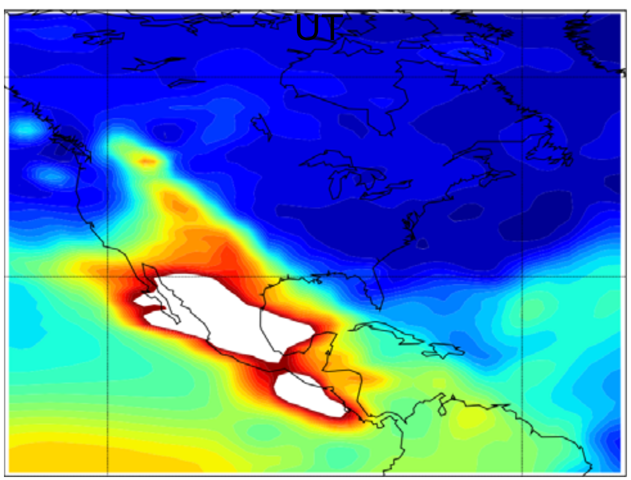
20:12 UT



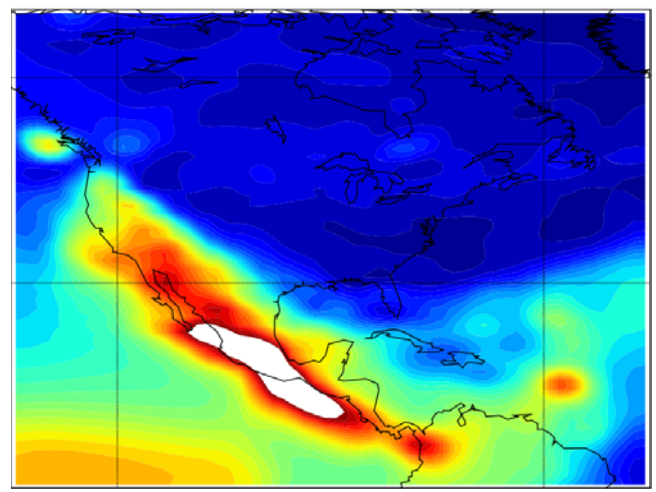
21:12 UT



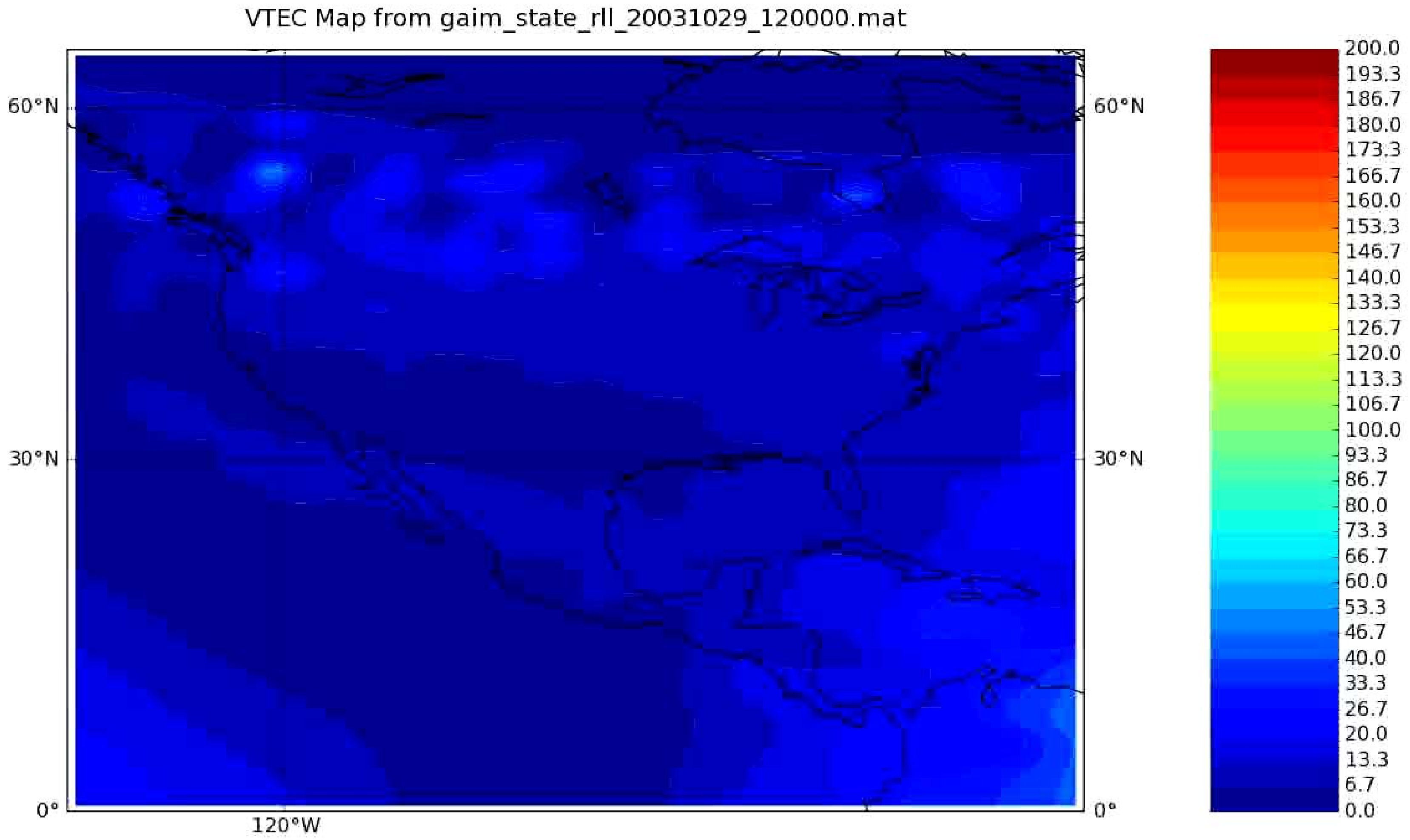
21:48 UT



22:48



October 29, 2003 NGAIM VTEC Movie



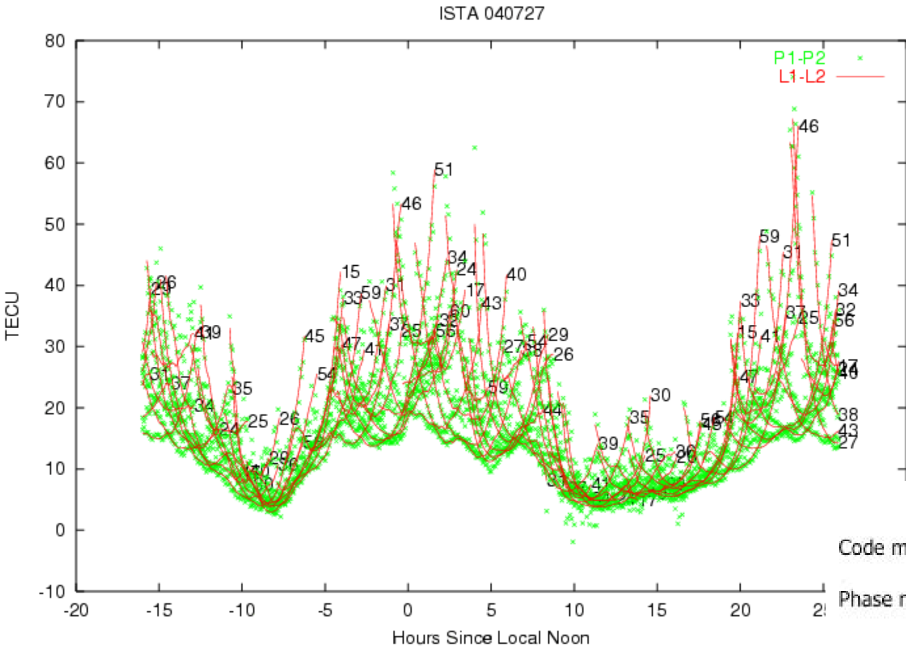
Data Processing System (PyTEC)

- New code, written from scratch in Python
- End-to-end code to take raw data (RINEX format) as input and produce GPS TEC links (in internal tecdump format) in one monolithic code.
  - Only external dependencies are common Python modules (numpy, scipy, matplotlib)
- Developed for processing COSMIC data as well as ground GPS measurements

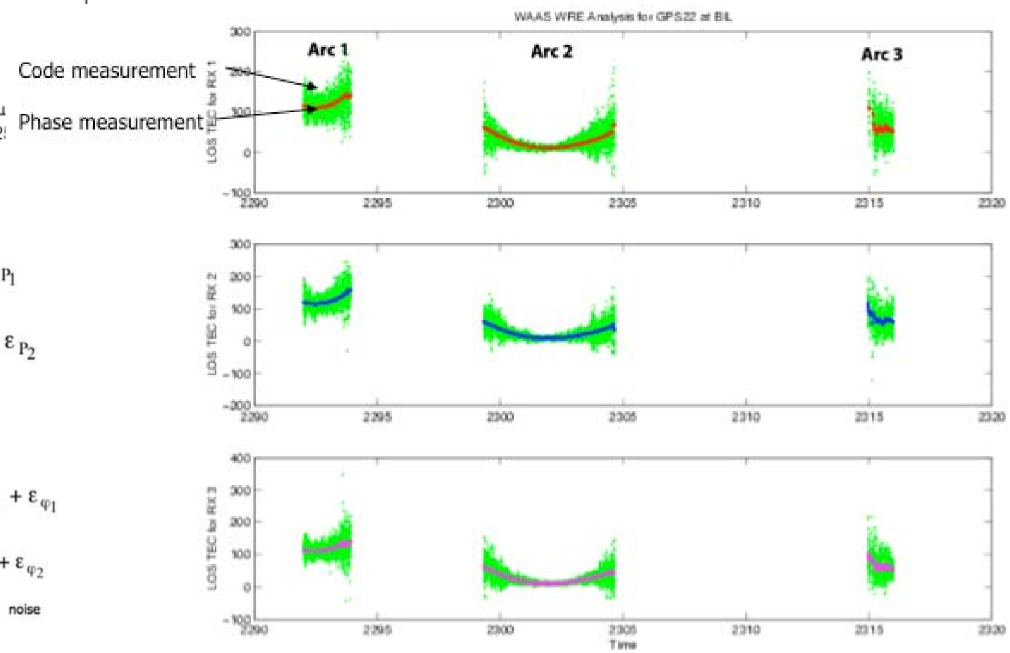
PyTEC Design

- Uses GIPSY's ninja data editing algorithm
  - Written in Fortran, compiled and 'glued' into python
- Uses rnxrdrpro library, written in C/C++ 'glued' into python
- Glued codes execute quickly, but native Python interface is exposed
- Additional compiled code for geometry and compressed file reading algorithms

Information Content of GPS Slant TEC (mid-latitude)



Examples of leveling phase ionospheric measurements using GIM as ground-truth



GPS pseudorange observation equation:

$$P_1 = \rho + c\cdot(dT - dt) + d_{ion,L_1} + d_{trop} + b^{s_i,L_1} + b_{r_j,L_1} + mp_{P_1} + \varepsilon_{P_1}$$

$$P_2 = \rho + c\cdot(dT - dt) + \gamma\cdot d_{ion,L_1} + d_{trop} + b^{s_i,L_2} + b_{r_j,L_2} + mp_{P_2} + \varepsilon_{P_2}$$

GPS carrier phase observation equation:

$$\Phi_1 = \rho + c\cdot(dT - dt) + \lambda_1 N_1 - d_{ion,L_1} + d_{trop} + b^{\varphi,s_i,L_1} + b_{\varphi,r_j,L_1} + mp_{\varphi_1} + \varepsilon_{\varphi_1}$$

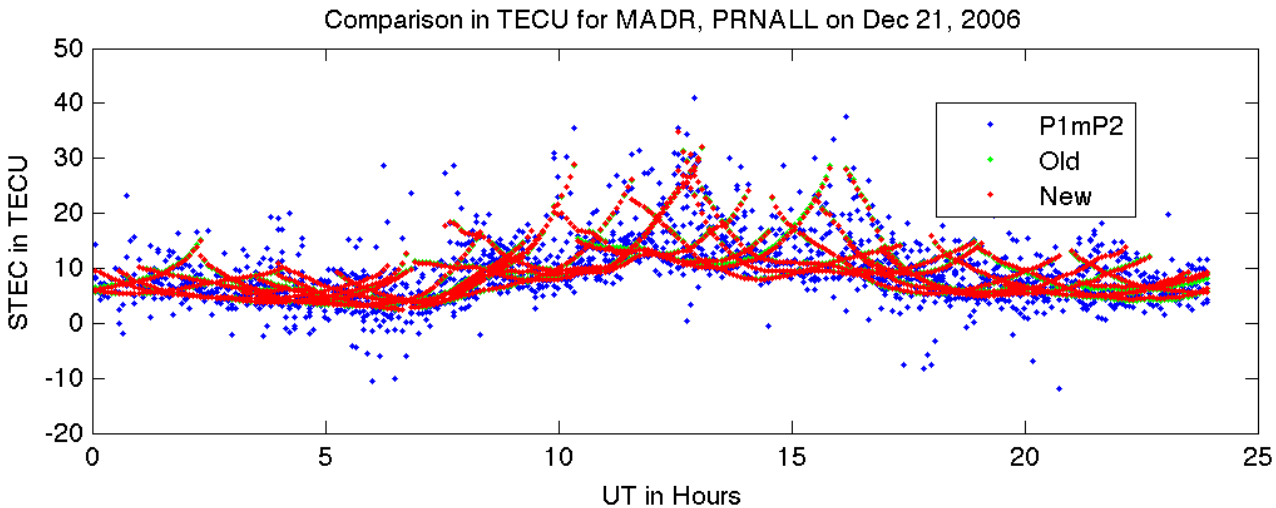
$$\Phi_2 = \rho + c\cdot(dT - dt) + \lambda_2 N_2 - \gamma\cdot d_{ion,L_1} + d_{trop} + b^{\varphi,s_i,L_2} + b_{\varphi,r_j,L_2} + mp_{\varphi_2} + \varepsilon_{\varphi_2}$$

Range, clock, ambiguity, ionosphere, troposphere, satellite bias, receiver bias, multipath, noise

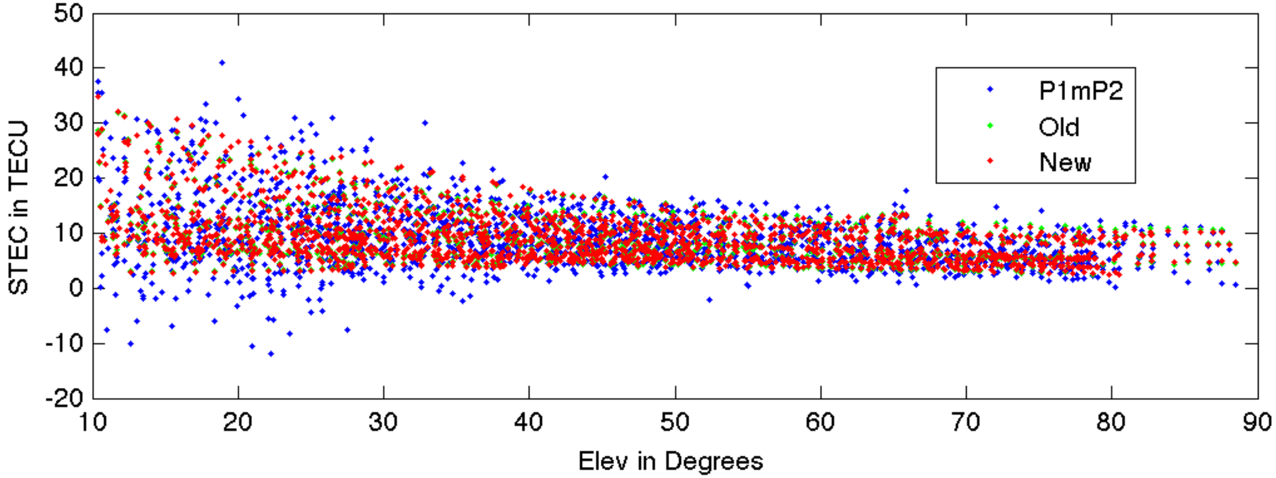
PyTEC Considerations

- Ground data – we have an operational code to validate algorithms against. Results to follow.
- COSMIC/LEO data
  - Leveling algorithms were developed for ground data, new algorithm developed for LEO data
  - Bias estimation must be performed for LEO satellites, non-trivial
- Analysis module design for implementing user specific analysis during processing, i.e.
  - TECDump module for writing results in tecdump format
  - BiasEstimator for LEO bias estimation

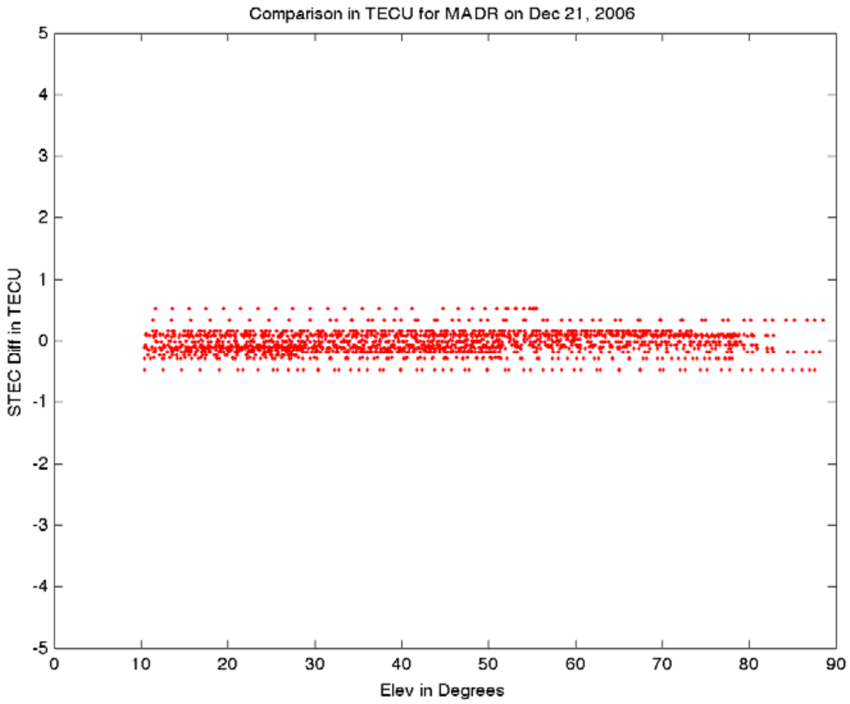
Comparison of Old and New Slant TEC Processing for Ground GPS Measurements

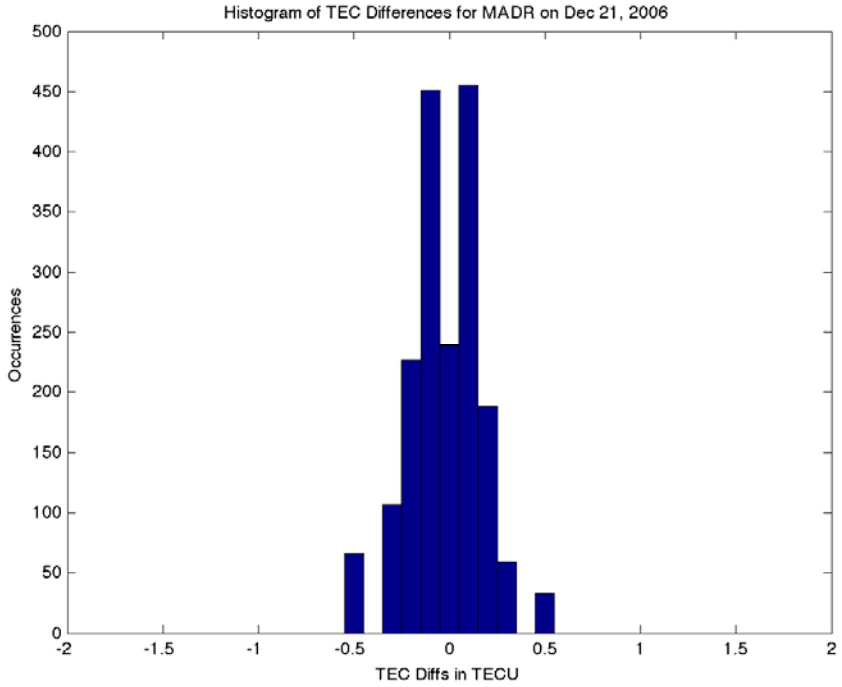


RMS Old = 3.63 TECU RMS New = 3.62 TECU



TEC Differences Between Old and New Processing Techniques at Station MADR on Dec 21, 2006





Processed ground GPS data comparison using station MADR

Leveling Algorithm

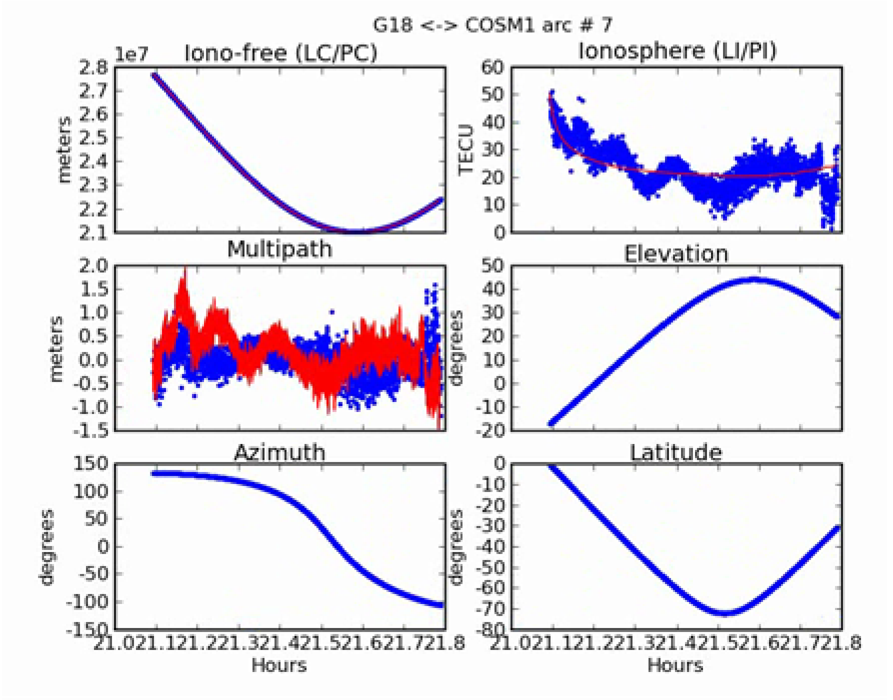
- We have noisy P1-P2 combination and biased L1-L2 combination. We use the P1-P2 to determine the level (offset) for the L1-L2 combination.
- Noise (multipath) for P1-P2 combination for ground data is highly correlated to the elevation angle of the link
  - Thus we consider lower noise data as more valid than higher noise data in determining the level of the P1-P2 combination.
- For LEOs, in particular COSMIC, we do not have such good behavior

Multipath in COSMIC

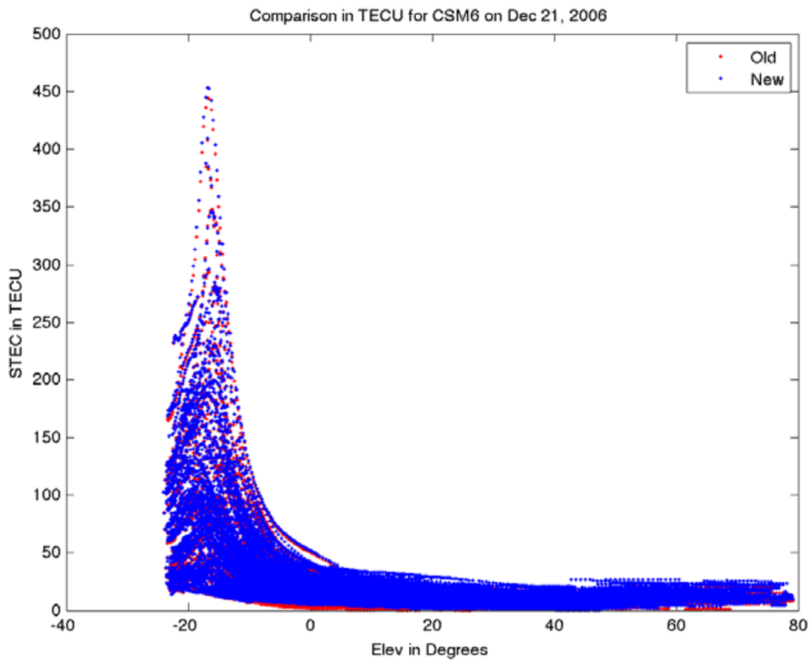
- Noise in PI combination correlated to multipath behavior
- We consider the multipath in the leveling
  - Compute the mean and stdev of x = MP1-MP2
  - Weight for point *i* in leveling is then

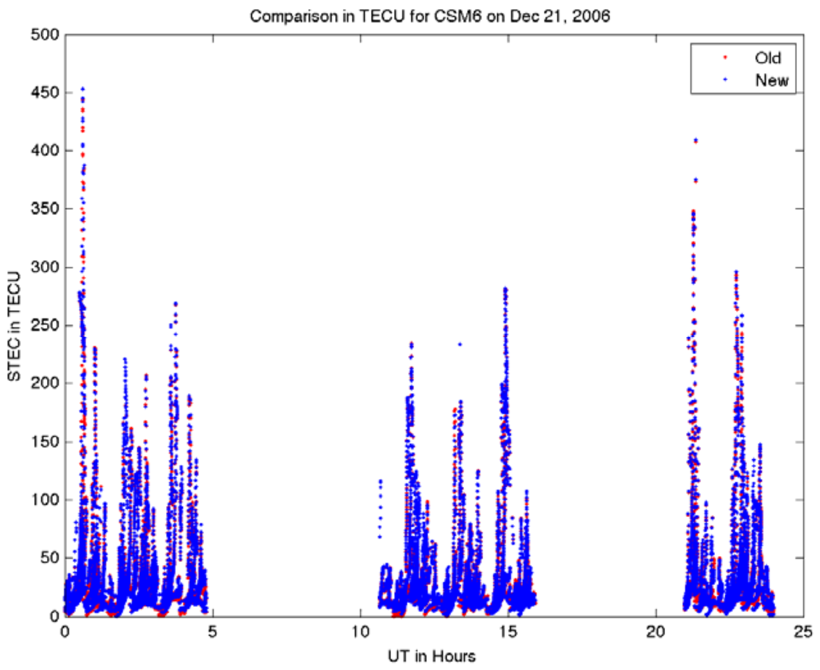
$$w_i = \sqrt{\sqrt{2\pi}\exp\left(\frac{(x_i - \langle x \rangle)^2}{\sigma_x^2}\right)}$$

i.e. the the weight squared is the inverse of the Gaussian



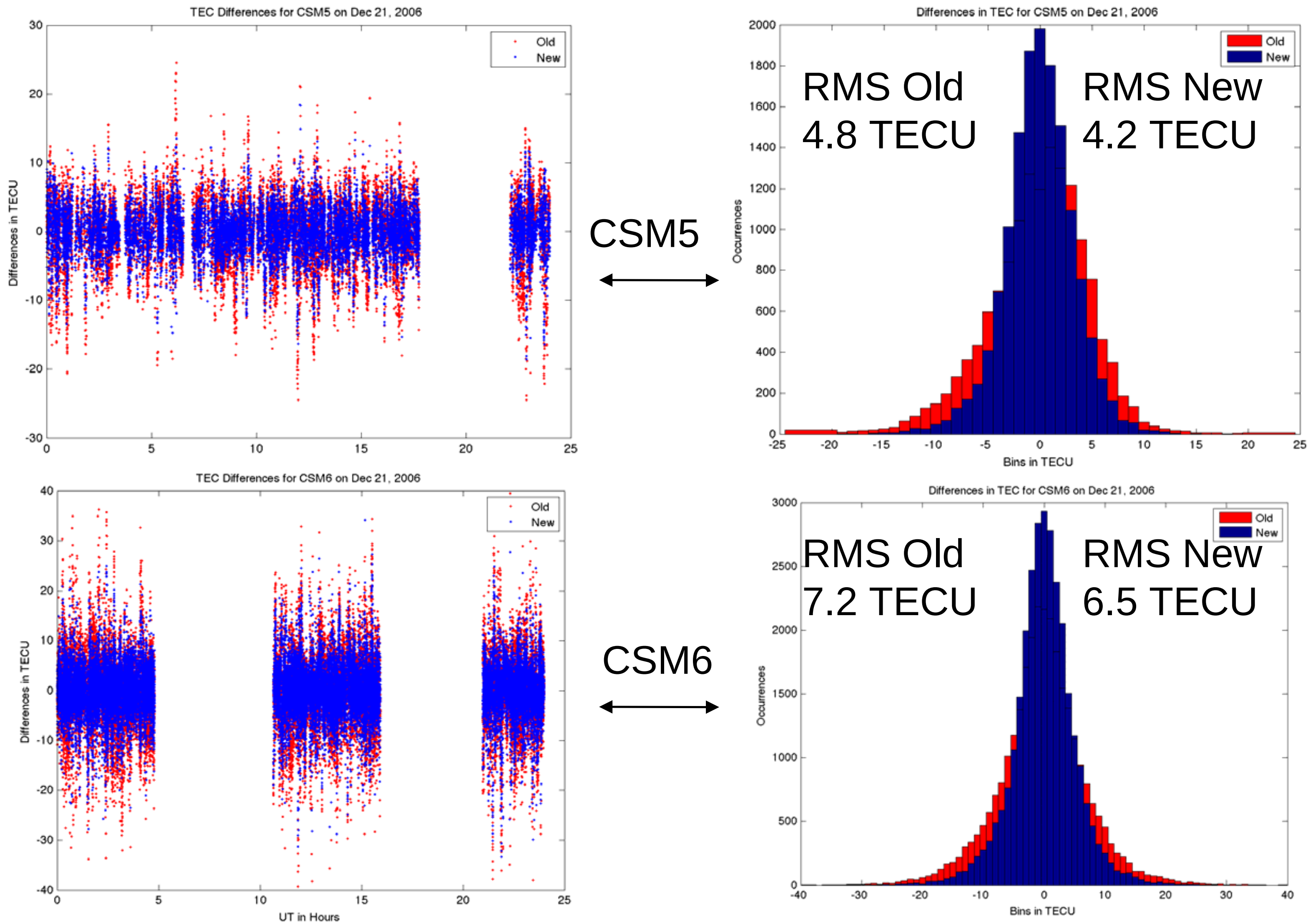
TEC Differences Between Old and New Processing Techniques for Entire CSM6 on Dec 21, 2006





To validate our newly processing of COSMIC and ground GPS data we selected the data set of Dec 21, 2006 because of our prior experience and published results for that day

All CSM5 and CSM6 Residuals and Histogram for Dec 21, 2006

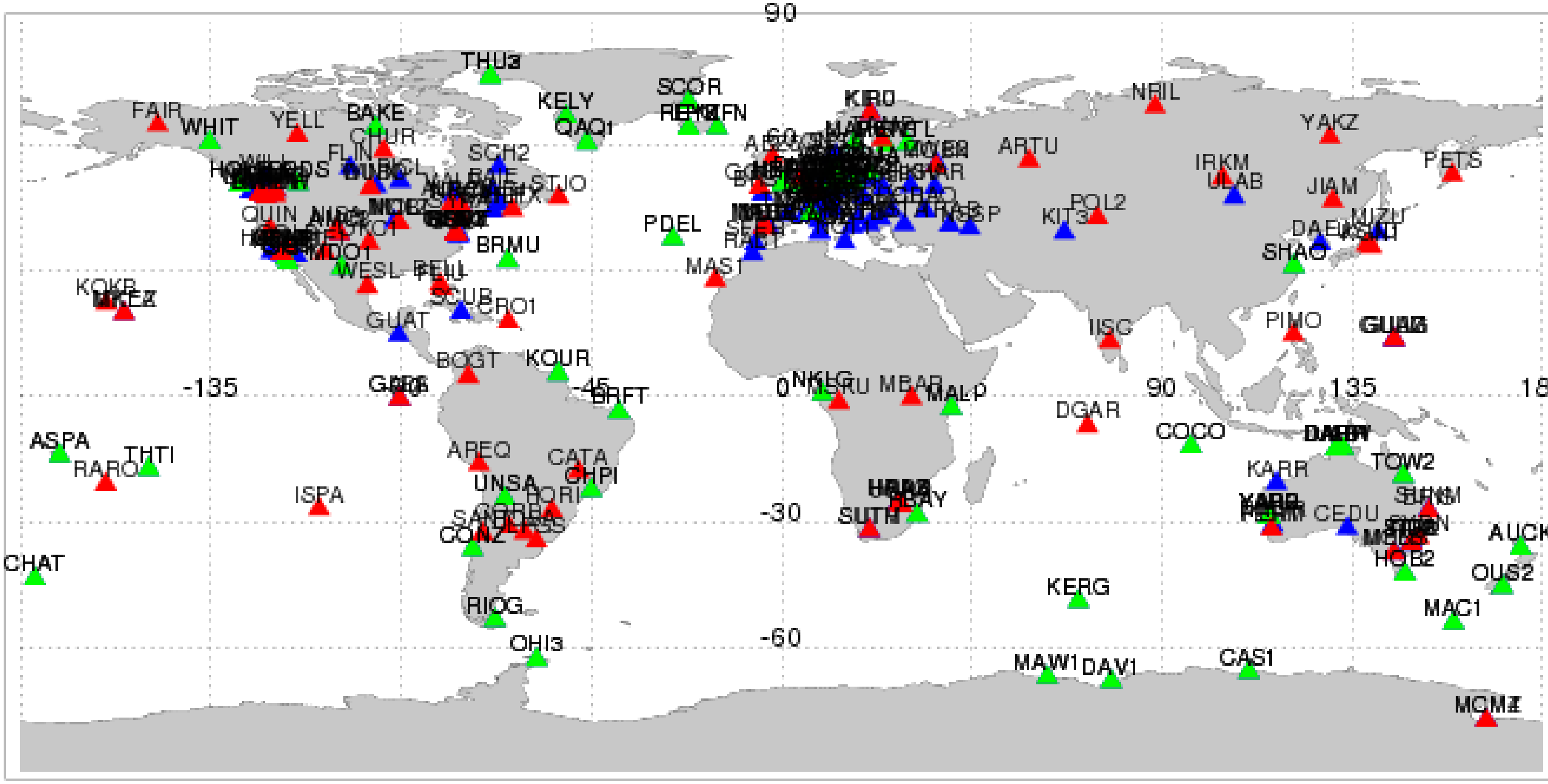


Real-Time System

- Designed to be an end-to-end product
  - Takes RINEX data as input, in real-time
    - Processed to TEC data via the PyTEC software
  - TEC data used in Kalman Filter
- 15 min update from 5 minute ground data
  - Currently running
- 2 hour update from COSMIC data, this is then propagated to 15 minute update thread to maximize data usage
  - Under development

Ground GPS streaming & Hourly Sites

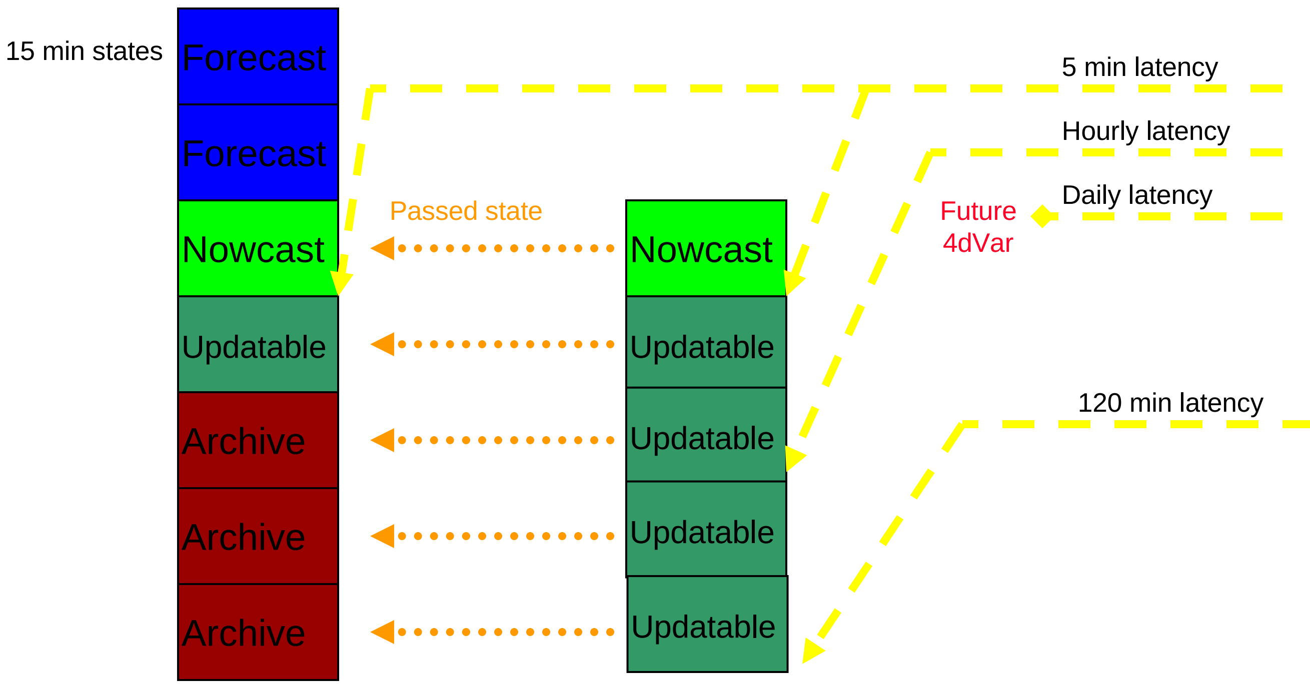
Current Streaming & Hourly GPS Sites (red=streaming, blue/green=hourly)



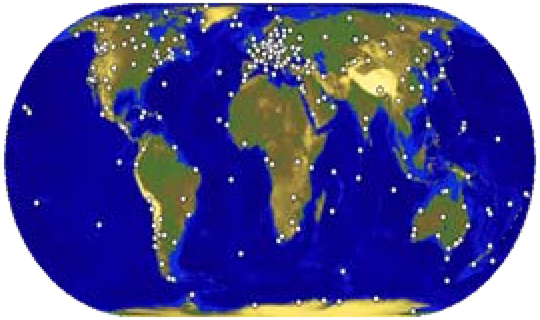
Multiple Latency Data Feeds

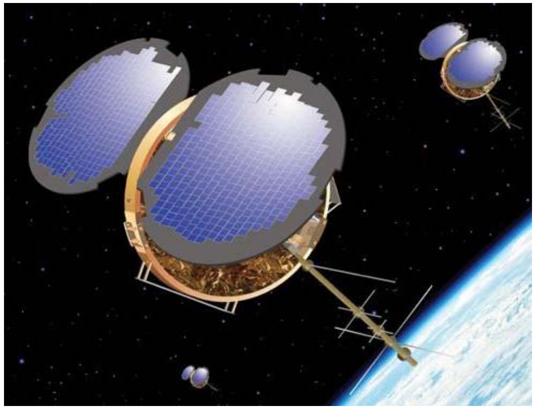
Main run proceeds with minimal latency tolerance, with states updated by catch-up processes

Catch-up run will process highly latent data and pass state once done



Global ground network



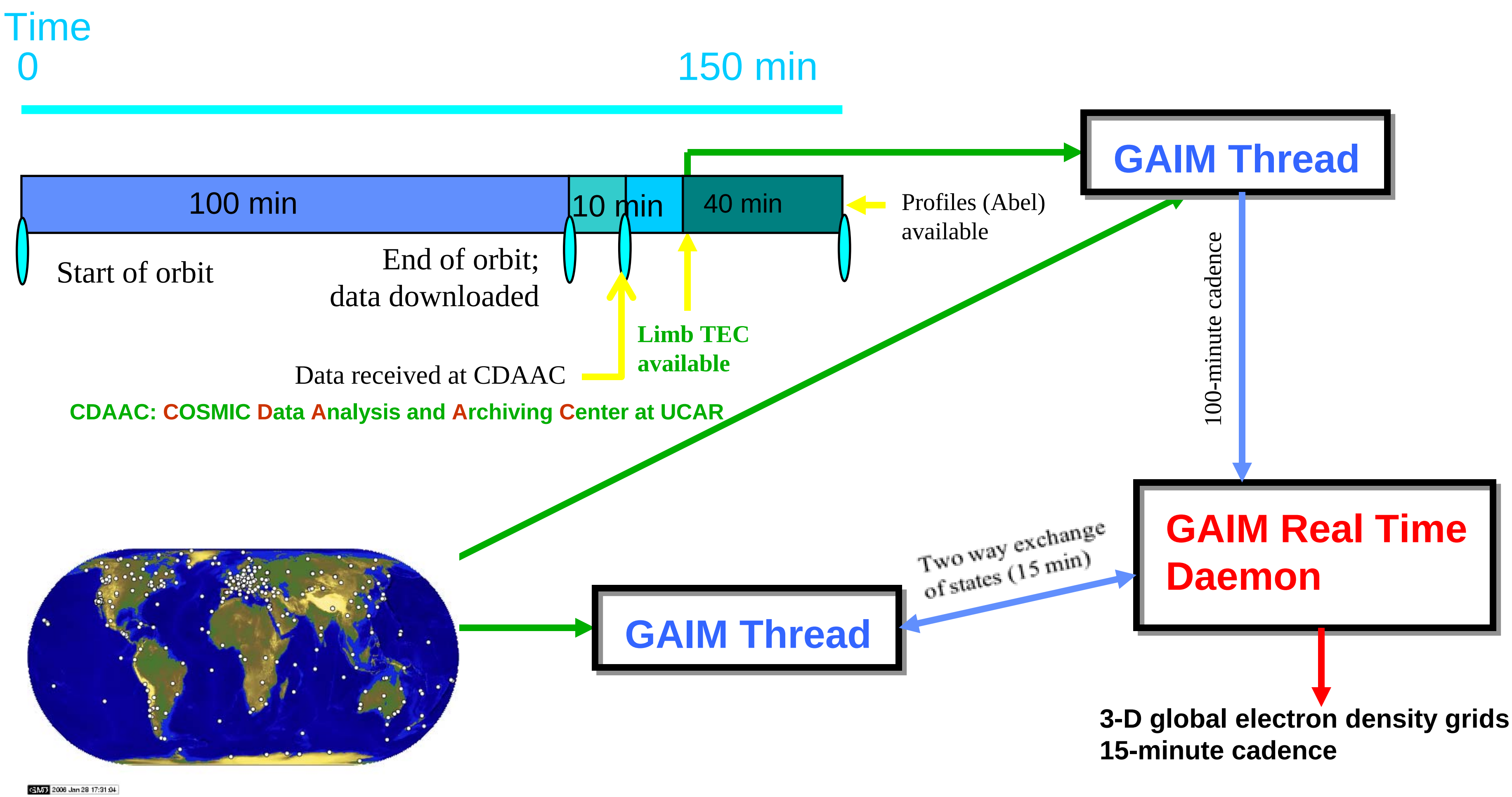


COSMIC occultations

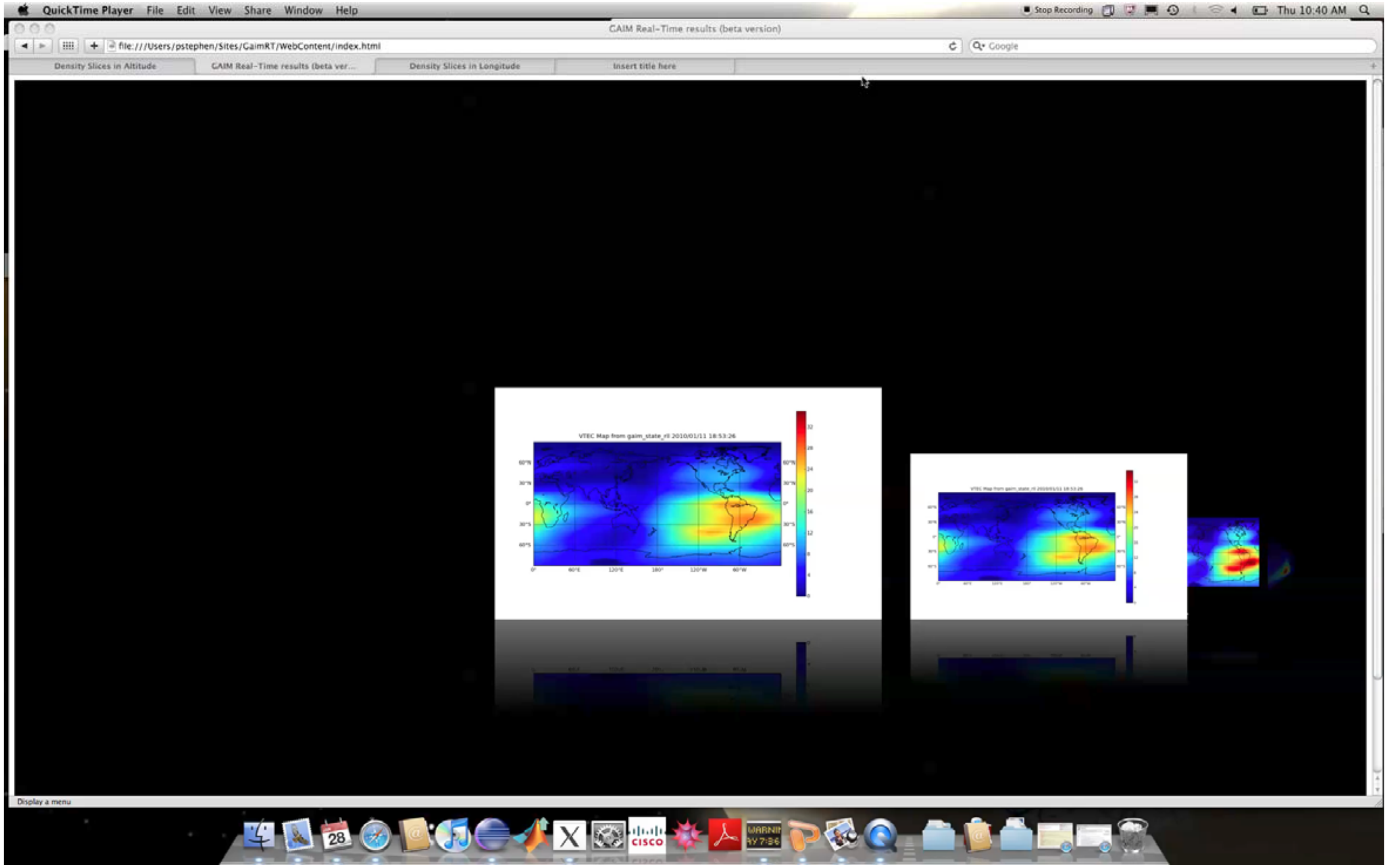
Real-Time COSMIC Schematic

Global ground network data: 5-minute and 1-hour latency

COSMIC data: 120+ minutes latency



Preliminary Results



GAIM at CCMC

- Plan to have ready by end of FY '10
  - Still must make some technical decisions before deployment
- Plan to have run-on-request version that can assimilate ground data from any station
  - End-to-end with data processing front-end to process the RINEX data before assimilation
  - Future enhancement: Add nested-grid capability for high resolution over areas of interest
- Have shown results from a functioning real time code (previous slide)
  - Code ran continuously for 22 days (12/18/09-01/11/10)
  - Similar approach (technologies) would be used for run-on-request version

Conclusions

- JPL/USC GAIM plans to be added to CCMC this FY
  - Includes data processing front end
    - End-to-end code to take raw data (RINEX format) as input and produce GPS TEC links (in internal tecdump format) in one monolithic code.
  - Plan to be able to run-on-request assimilation of several hundred ground station
  - Future goal to add nested grid capability to CCMC run-on-request code
- Real-Time GAIM operational,
  - We assimilate real-time GPS data;
  - Web interface nearly complete
  - COSMIC data assimilation is under developmen
  - Similar code-base to be used for run-on-request version

Acknowledgements

- This research was performed at the Jet Propulsion Laboratory/California Institute of Technology under contract to the National Aeronautics and Space Administration