EO-LDAS: Some thoughts on the system

NCEO* Reading
NCEO Swansea
NCEO UCL
NCEO Exeter
JRC Ispra
FastOpt
Science Systems
U. Jena

A study being done for the Support to Science Element of ESA's Earth Observation Envelope Programme

NCEO = (UK) National Centre for Earth Observation

National Centre for Earth Observation

Rationale: From the Statement of Work

"Inconsistencies of products derived from different sensor systems are considered the stumbling block for the insertion of different EO data into assimilation schemes and thus for making full use of their synergies.

Consequently a dedicated activity was initiated for the development of a generic Earth Observation - Land Data Assimilation Scheme (EO_LDAS)"

Idea not completely new: earlier studies were done for the SPECTRA candidate Earth Explorer

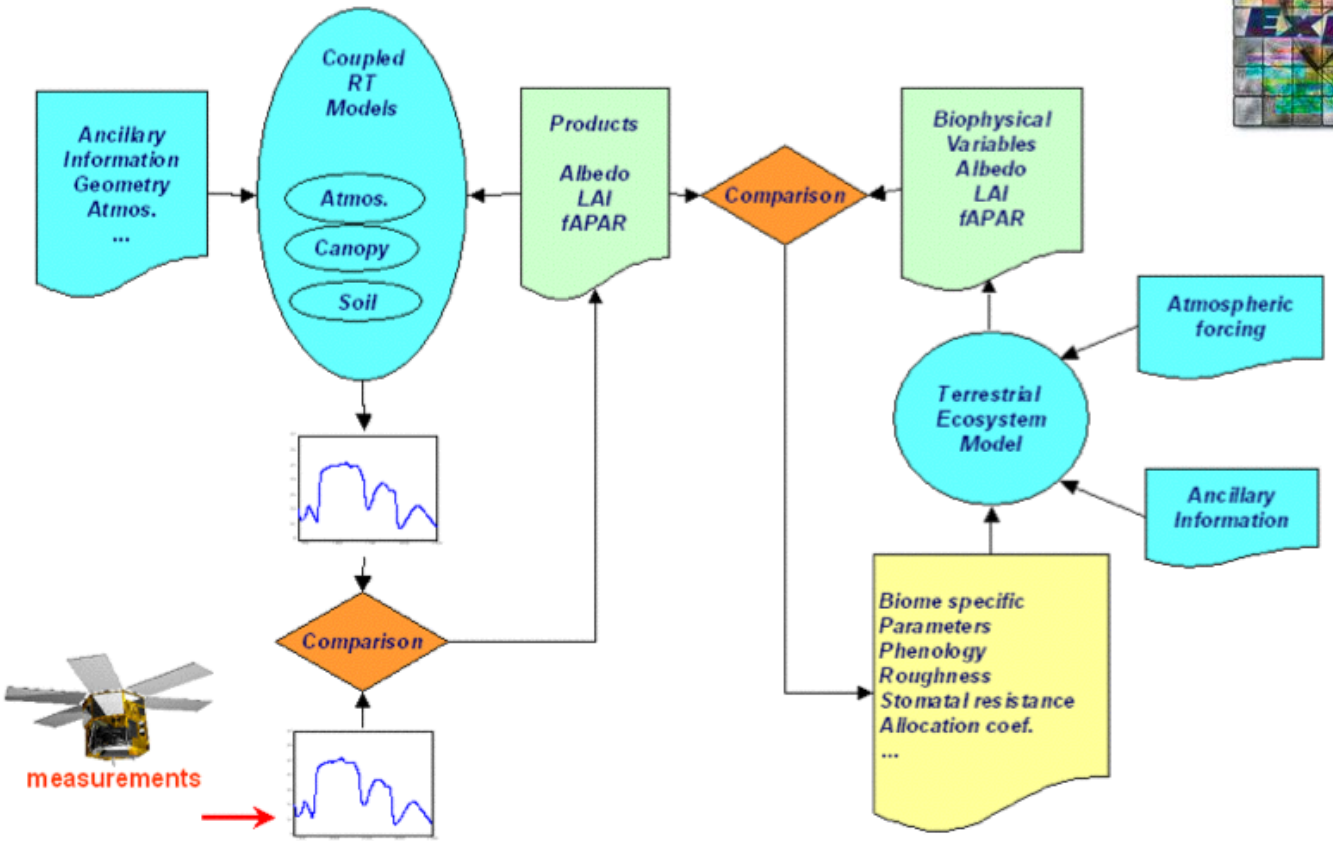


Figure 5.3: *Conceptual scheme of data assimilation procedure to use observations of geo-biophysical variables provided by SPECTRA to constrain regional scale Terrestrial Ecosystem Models and estimate parameter values (Courtesy F. Baret).*

(From the SPECTRA report for mission selection, April 2004)

ESA's requirements

- Improved retrievals (reported customer dissatisfaction with 'products')
- Tools for working with constellations of imaging sensors (Vis-NIR)
  - Importance of consistent retrievals
  - Therefore important to assimilate radiances, not products
- Tool for catching calibration drift
  - or cross-calibration in a constellation system
- provide a prototype and a case study.

- Others (e.g. ecosystem modellers) may make use of a stand-alone system for DA experiments. This may depend on how modular the system is.
- The LDAS should contribute towards the development of an OSSE scheme.

Components of a DA system

Process model (*M*), to advance the system in time

Simulator (Observation operator, *h*)

- Geometric interpolator(s)
- Radiative Transfer functions

Suitable cost function (*J*)

Assimilation methodology

Not to be forgotten:

Sensor Errors

Auxiliary data

Data handling routines

QA routines

Components of a DA system

Process model (M), to advance the system in time

Simulator (Observation operator, h)

- Geometric interpolator(s)
- Radiative Transfer functions

Suitable cost function (J)

Assimilation methodology

Not to be forgotten:

Sensor Errors

Auxiliary data

Data handling routines

QA routines

Components of a DA system

Process model (*M*), to advance the system in time

Simulator (Observation operator, *h*)

Geometric interpolator(s)

Radiative Transfer functions

Suitable cost function (*J*)

Assimilation methodology

The Essential Components

Not to be forgotten:

Sensor Errors

Auxiliary data

Data handling routines

QA routines

Components of a DA system

Process model (*M*), to advance the system in time

Simulator (Observation operator, *h*)

- Geometric interpolator(s)
- Radiative Transfer functions

Suitable cost function (*J*)

Assimilation methodology

Not to be forgotten:

Sensor Errors

Auxiliary data

Data handling routines

QA routines

Data Assimilation and Regularised Estimation

Minimising a cost function

$$J(\mathbf{x}) = \underbrace{J_{model}(\mathbf{x})}_{\text{fit to forward model}} + \underbrace{J_{obs}(\mathbf{x})}_{\text{fit to observation}}$$

Excellent model — Uncertain model

DA — RE

But we are still doing the same thing

Usual Shape of Cost Functions (3DVAR)

$$J_{model} = \left(\mathbf{x} - \mathbf{x}_B\right)^T \mathbf{B}^{-1} \left(\mathbf{x} - \mathbf{x}_B\right) \qquad \mathbf{x}_B = M\left(\mathbf{x}_{-1}\right)$$

x represents our 'state' variables, but could include anything uncertain or variable that affects the radiative signals we see.

M is a dynamic (vegetation) model.

B is the background covariance matrix, containing the error characteristics of *M*, for all variables at all sites.

The ‘state vector’

x comprises:

Parameters of the vegetation canopy: LAI, other biophysical parameters, plant biochemical parameters, soil reflectance properties, leaf reflectance properties

Parameters relating to atmospheric scattering and transmission: aerosol loading & type, column concentrations of important absorbers

Parameters of the space craft/sensor system. We consider calibration, and pointing errors.

p variables in all.

Usual Cost Functions (3DVAR)

$$J_{obs} = \left(\mathbf{y} - h(\mathbf{x})\right)^T \mathbf{R}^{-1} \left(\mathbf{y} - h(\mathbf{x})\right)$$

y is a set of at-satellite radiances, in a series of images. Footprint at ground: 5m – 1km typically

h is the observation operator, or simulator, a mixture of interpolation and radiative transfer functions.

R is the covariance matrix of the observations, often taken to be diagonal. Note that this assumes no error in *h*.

Geometry

Our state variables are defined on a model grid Γ_m, say.
This is related to the geometry of the eventual product grid.

Referenced to lat/long, locally a grid of squares, separation r_m.

State variables at each model point represent some average of that variable over the local square.

An image has a different reference system (Γ_o, say).

Comparisons are evaluated on an assimilation grid Γ_a

Geometrical Considerations

Traditionally:

Resample/interpolate **x** to obtain **x*** : cf. $h(\mathbf{x}^*)$ to **y** $\quad \Gamma_a = \Gamma_o$

But, can we do arithmetic on the **x** values?

Better

Resample/interpolate $h(\mathbf{x})$ to obtain $h^*(\mathbf{x})$: cf. h^* to **y** $\quad \Gamma_a = \Gamma_o$

Why not:

Resample/interpolate **y** to **y***, cf $h(\mathbf{x})$ to **y***. $\quad \Gamma_a = \Gamma_m$

Interpolation error

Measurements, $y(\mathbf{s}_i)$ and simulations $h(\mathbf{r}_j)$: $\mathbf{s}_i \in \Gamma_o$, $\mathbf{r}_j \in \Gamma_m$

$$y_i = \int f(\mathbf{s}-\mathbf{s}_i) L[\mathbf{x}(\mathbf{s})] d\mathbf{s} \qquad \int f(\mathbf{s}-\mathbf{s}_i) d\mathbf{s} = 1$$

$$h_j = \int g(\mathbf{r}-\mathbf{r}_j) L[\mathbf{x}(\mathbf{r})] d\mathbf{r} \qquad \int g(\mathbf{r}-\mathbf{r}_j) d\mathbf{r} = 1$$

$$g(\mathbf{r}-\mathbf{r}_i) g(\mathbf{r}-\mathbf{r}_j) \propto \delta_{ij}$$

f is the point spread function of the system
g is a top hat function for the cell in question

Should we be trying to model y as a function of the h's? or v.v?

Interpolation error

Which is better?

$$h_j \sim \sum \alpha_{ji} y_i$$ Resample image values

$$y_i \sim \sum \beta_{ij} h_j$$ Resample simulated radiances

Knowing the weighting function (*g* and *f*) we can estimate optimal coefficients and determine the error, for a random stationary field. This error depends on:

(1) Autocorrelation length of *L*,
(2) Separation length of samples in model grid
(3) FWHM of Surface Response Function (psf) and pixel separation.

Some experiments under way to establish criteria for choosing which way to interpolate.

Resolution of Model grid, Γ_m

What should the spatial sampling interval be?

>1km Users will go elsewhere - they (may) have become used to products at this scale and might expect nothing coarser.

<100m Horizontal transport, other 3D surface effects, become important (J-L. W). Our forward RT models break down in some cases.

High resolution can lead to problems of dimensionality.

Dimensionality Trouble

If r is the grid cell size, and D the linear dimension of an area being operated on, then we have $N \sim (D/r)^2$ cells. We have $N \times p$ unknowns in all. The required final state is that which satisfies:

$$\nabla_{\mathbf{x}} J = \mathbf{0}$$

a system of Np equations in Np unknowns. Solving a *linear* set of equations of this size is an $O(N^3 p^3)$ operation in general.

Dimensionality, ctd.

If the observation operator is linear, $h = \mathbf{Kx}$, then the one-step solution to the 3DVAR problem is

$$\hat{\mathbf{x}} = \mathbf{H}\mathbf{B}^{-1}\mathbf{x}^B + \mathbf{H}\mathbf{K}^T\mathbf{R}^{-1}\mathbf{y} = \mathbf{x}^B + \mathbf{H}\mathbf{K}^T\mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{K}\mathbf{x}^B\right)$$

$$= \mathbf{x}^B + \mathbf{B}\mathbf{K}^T\left(\mathbf{R} + \mathbf{K}\mathbf{B}\mathbf{K}^T\right)^{-1}\left(\mathbf{y} - \mathbf{K}\mathbf{x}^B\right)$$

where

$$\mathbf{H}^{-1} = \mathbf{B}^{-1} + \mathbf{K}^T\mathbf{R}^{-1}\mathbf{K}$$

$\mathbf{H}^{-1}$ is the Hessian matrix, same size as $\mathbf{B}$ (N times p)
$\mathbf{K}^T\mathbf{R}^{-1}\mathbf{K}$ is likely to be sparse, for our problem.

B & H

Use of the background covariance matrix **B** can improve estimates of **x** in the absence of any data

$$\hat{\mathbf{x}} = \mathbf{x}^B + \mathbf{B}\mathbf{K}^T\left(\mathbf{R}+\mathbf{K}\mathbf{B}\mathbf{K}^T\right)^{-1}\left(\mathbf{y}-\mathbf{K}\mathbf{x}^B\right)$$

The Hessian's inverse, **H**, can be used to accelerate convergence in an iterative search for the **x** that minimises the cost function.

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \mathbf{H}\left[\nabla_{\mathbf{x}} J(\mathbf{x}_n)\right]$$

Both v. important matrices in DA. But both are very big.

Dimensionality Trouble

If the scheme runs on a k-teraflop system, the time taken to invert either **B** or the Hessian is

$$t \sim \left(\frac{D}{100r}\right)^6 p^6 k^{-1}$$

If D = 100km, r=1km, t=$1/k$ seconds

If D = 1000km, r=1km, or

If D = 100km r=100m t = 116/k days

If D = 290km r=100m t = 19/k years (all for p=1)

Dimensionality Trouble

May not be feasible to work with both **B** and $\mathbf{B}^{-1}$ (in extreme cases, we might not even be able to store them) (This problem exists at operational centres). A way round the **B** problem is possible (parameterise the correlation matrix).

May not be able to use Hessian to accelerate convergence of minimisation routine. May be able to approximate the inverse Hessian, then update this at each iteration without ever inverting a really big matrix.

So, sub-optimal procedures and compromises are necessary. Atmospheric community well-practiced in these arts. We must learn them.

Ignoring Correlations

One possibility is to carry out all retrievals ignoring all the spatial correlations i.e. do it on a cell-by-cell basis, with all **B** matrices diagonal, and all simulated radiances depending on **x** of just one model cell.

We can then exploit the full power of e.g. a Levenberg-Marquardt solver.

This may be the only practical way to carry out multitemporal (4DVAR) assimilations/inversions in the prototype.

But the correlations are information that should be exploited.

Variables to be retrieved

		Model Type	Model form	Correlated ε?
Veg. canopy phys:	LAI	DVM	$M(x)=M(x_0,ppt,E...)$	Strong (same type)
	Soil Albedo	M(**r**, ppt)	M=M(x,ppt…)	Strong
	LAD params	M(pft, t)	M(x)=x	Weak
	more…			
chem:	Aerosol τ	Climatology?	M(x) = const.	Strong
Atmos	Angstrom parameter	Climatology?	M(x) = const.	Very strong
	Column WV	ECMWF?	M(x) = const.	Strong
	Column O3	ECMWF?	M(x) = const.	Strong
	more…			
Sensor	Calibration	Zero order?	$M(x) = f(T_{sat},x,emr)$	
	Pointing	Random	M(x) = 0	Image to image

Correlations

We expect many of the errors to be spatially correlated, perhaps over 10s of km.

We do not expect atmospheric variable errors to be correlated with vegetation

We do not expect O3 errors to be correlated with WV

In fact, do we expect *any* correlation between errors at a given location?

If no correlations between different variables at each site,…

Then (rearranging $\mathbf{x}$)

$$\mathbf{x} = \left(\mathbf{x}_1^T, \mathbf{x}_2^T, ..., \mathbf{x}_p^T\right)^T \qquad \mathbf{B} = \begin{pmatrix} \mathbf{B}_1 & & & \\ & \mathbf{B}_2 & & \\ & & \ddots & \\ & & & \mathbf{B}_p \end{pmatrix}$$

And:

$$\left(\mathbf{x}-\mathbf{x}^B\right)^T \mathbf{B}^{-1}\left(\mathbf{x}-\mathbf{x}^B\right) = \left(\mathbf{x}_1-\mathbf{x}_1^B\right)^T \mathbf{B}_1^{-1}\left(\mathbf{x}_1-\mathbf{x}_1^B\right) + \ ... \ + \left(\mathbf{x}_p-\mathbf{x}_p^B\right)^T \mathbf{B}_p^{-1}\left(\mathbf{x}_p-\mathbf{x}_p^B\right)$$

$$J_B = J_{LAI} + ... + J_{O3} + J_{wv} + J_{cal} + J_{point}$$

Treatment of **B**

In a 1-d Markov process, the *correlation* matrix is given by: $C(i,j) = \rho^{|i-j|}$

$$\mathbf{B} = \mathbf{DCD}$$

$$\mathbf{C} = \begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \rho^2 & \rho & 1 & & \vdots \\ \vdots & & & \ddots & \rho \\ \rho^{N-1} & \cdots & \cdots & \rho & 1 \end{pmatrix}$$

D is a diagonal matrix of variances

Treatment of **B**

For this matrix, the inverse is tridiagonal:

$$\mathbf{C}^{-1} = \frac{1}{1-\rho^2}\begin{pmatrix} 1 & -\rho & 0 & & \\ -\rho & 1+\rho^2 & -\rho & 0 & \\ 0 & -\rho & 1+\rho^2 & -\rho & \\ 0 & 0 & -\rho & & \ddots \\ & & & & \ddots \end{pmatrix}$$

$$\left(\mathbf{C}^{-1}\mathbf{x}\right)_j = \frac{1}{1-\rho^2}\left[-\rho x_{j-1} + \left(1+\rho^2\right)x_j - \rho x_{j+1}\right]$$

$$= \frac{1-\rho}{1+\rho}x_j - \frac{\rho}{1-\rho^2}\left(x_{j-1} - 2x_j + x_{j+1}\right)$$

Treatment of **B**

We can re-write this as:

$$\mathbf{C}^{-1}\mathbf{x} \sim \frac{1-\rho}{1+\rho}\left[\mathbf{x} - \frac{\rho}{(1-\rho)^2}(\Delta t)^2 \frac{\partial^2}{\partial t^2}\mathbf{x}\right] \qquad \left(\Delta t = x_j - x_{j-1}\right)$$

So, the inverse of a long range, smooth correlation matrix can be short range operator. For 2 spatial dimensions, therefore try to parameterise **C** thus:

$$\mathbf{C}^{-1} = \gamma\left(\mathbf{I} - l^2\nabla^2\right)$$

l is a correlation length, γ a constant to be determined

Treatment of **B**

When we do this, we find that each component of

$$\frac{\partial}{\partial \mathbf{x}} J_q = 2\mathbf{B}^{-1}\left(\mathbf{x} - \mathbf{x}^B\right)$$

is a sum of just 5 terms, for each component of the cost function except $J_{\text{obs.}}$

We believe we have discovered an analytic form for **C**, when $\mathbf{C}^{-1}$ takes this simple form.

Observation Operator error

Refer to the cost function for the observations *(m* in number)*:*

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\eta_i\left(\mathbf{R}^{-1}\right)_{ij}\eta_j$$

$$\eta_i \;=\; y_i \;-\; h_i(\mathbf{x})$$ "Observation increment"

R is the covariance matrix of errors in the $\{y_i\}$. Usually taken to be diagonal, which can be useful.

Observation operator error is usually ignored. **R** is then assumed to absorb all the unknown unknowns. And to still be diagonal.

Observation Operator Errors

Interpolation errors - result of initial observation layout and model grids being misaligned.

Canopy RT errors

- Clumping
- Incorrect parameter selection
- Model specification error

Adjacency effect (3d effect; 1d approximation)

Approximated coupling (atmosphere-canopy)

Terrain effects - canopy RT usually assumes level surface (*)

How do we characterise these? They will usually be bigger than the simple observation errors captured in **R**

Observation Cost Function

Propose: to make estimates of OO error to add to existing **R**. Observation errors are assumed uncorrelated between grid cells, and between channels in a multi-channel image. So keep **R** diagonal

$$J_{obs} = \sum_i \eta_i^2 \sigma_i^{-2}$$

$$\eta_i = G_i(\mathbf{y}) - h_i(\mathbf{x})$$

Sensor Errors

Pointing

Calibration

Keystone

Smile

…

Wavelength Calibration Errors

…

Calibration

$$Radiance = (counts - darkcurrent)/electronic\_gain$$

An error in dark current correction, or a drift in gain, means we work with a 'radiance' L_{calc}, that is related to the true radiance via:

$$L_{true} = a + b.L_{calc}$$

where a and $|1-b|$ are both fairly small.

Can modify h so that we have $h^* = a+bh(\mathbf{x})$

Or can modify $\mathbf{y}$ so that $\mathbf{y}^* = a^*+b^*h(\mathbf{x})$

Calibration: cost function

$$J_{calib} = \left(\mathbf{b}-\mathbf{b}_f\right)^T \mathbf{B}_c^{-1}\left(\mathbf{b}-\mathbf{b}_f\right) + \left(\mathbf{a}-\mathbf{a}_f\right)^T \mathbf{A}_c^{-1}\left(\mathbf{a}-\mathbf{a}_f\right)$$
$$\left\{+\mathbf{a}^T \mathbf{A}_2^{-1}\mathbf{a} + \left(\mathbf{b}-\mathbf{j}\right)^T \mathbf{B}_2^{-1}\left(\mathbf{b}-\mathbf{j}\right)\right\}$$

$$\mathbf{b}_f = f\left(\mathbf{b}\_\right) \qquad \text{or}$$

$$\mathbf{b}_f = f\left(spacecraft \;\; temperatures,...\right)$$

Something like this currently being done to account for bias in the assimilation of altimeter data in ocean models (UKMO) (**j** is a vector of units)

Pointing Error

Pointing errors and possible attitude problems mean that an image can be displaced and perhaps slightly rotated. If the true ground location (pixel centre) of a sample is (u_{true}, v_{truw}) and its notional position are (u_{not},v_{not}) then:

$$u_{true} = \Delta u + \cos\theta u_{not} + \sin\theta v_{not}$$
$$v_{true} = \Delta v - \sin\theta u_{not} + \cos\theta v_{not}$$

$$y(u_t,v_t) = y(u_{not},v_{not}) + (u-u_{not})\frac{\partial y}{\partial u} + (v-v_{not})\frac{\partial y}{\partial v} + O(\Delta^2)$$
$$\sim y(u_{not},v_{not}) + (\Delta u+\theta v)\frac{\partial y}{\partial u} + (\Delta v-\theta u)\frac{\partial y}{\partial v} + O(\Delta^2)$$

Pointing Error

So, instead of using the notional observed value, we modify it by adding unknown amounts of the local image gradients.

$$y(u_t, v_t) = y(u_{not}, v_{not}) + (\Delta u + \theta v)\frac{\partial y}{\partial u} + (\Delta v - \theta u)\frac{\partial y}{\partial v}$$

We *could* do it the other way, look for displacements so that the resampled $h(\mathbf{x})$ best matches our observation, but that means calculating gradients at every iteration. This way, just once.

Pointing Error

We need the displacements to be small. We cannot expect DA techniques to recover gross errors (several pixels) in pointing. A first estimate of the displacements should first be made by

(1) cross-correlation of images with each other (time series).

(2) cross-correlation of image with a first-guess simulation of TOA radiances.

This would give a measure of gross error. We can then use the derivatives of the image to try to retrieve more accurate values.

Pointing Error

The errors are expected to be random, with zero-mean, so an appropriate cost function for just these terms is:

$$J_{point} = \frac{(\Delta u)^2 + (\Delta v)^2}{\sigma^2} + \frac{\theta^2}{V_\theta}$$

Perhaps some other effects, which are largely geometric distortions (smile, keystone) could be handled in a similar way.

The "Data Operator" G(**y**)

Latest calibration correction factors are applied.

Resampled from image space to assimilation grid.
Gradients (w.r.t. spatial coordinates) are calculated.
Possibly other geometric un-distortions applied.

[Atmospheric correction to top-of-canopy reflectance values].

$G_i(\mathbf{y})$ for comparison with $h_i(\mathbf{x})$ at the ith grid cell.

The Data Operator

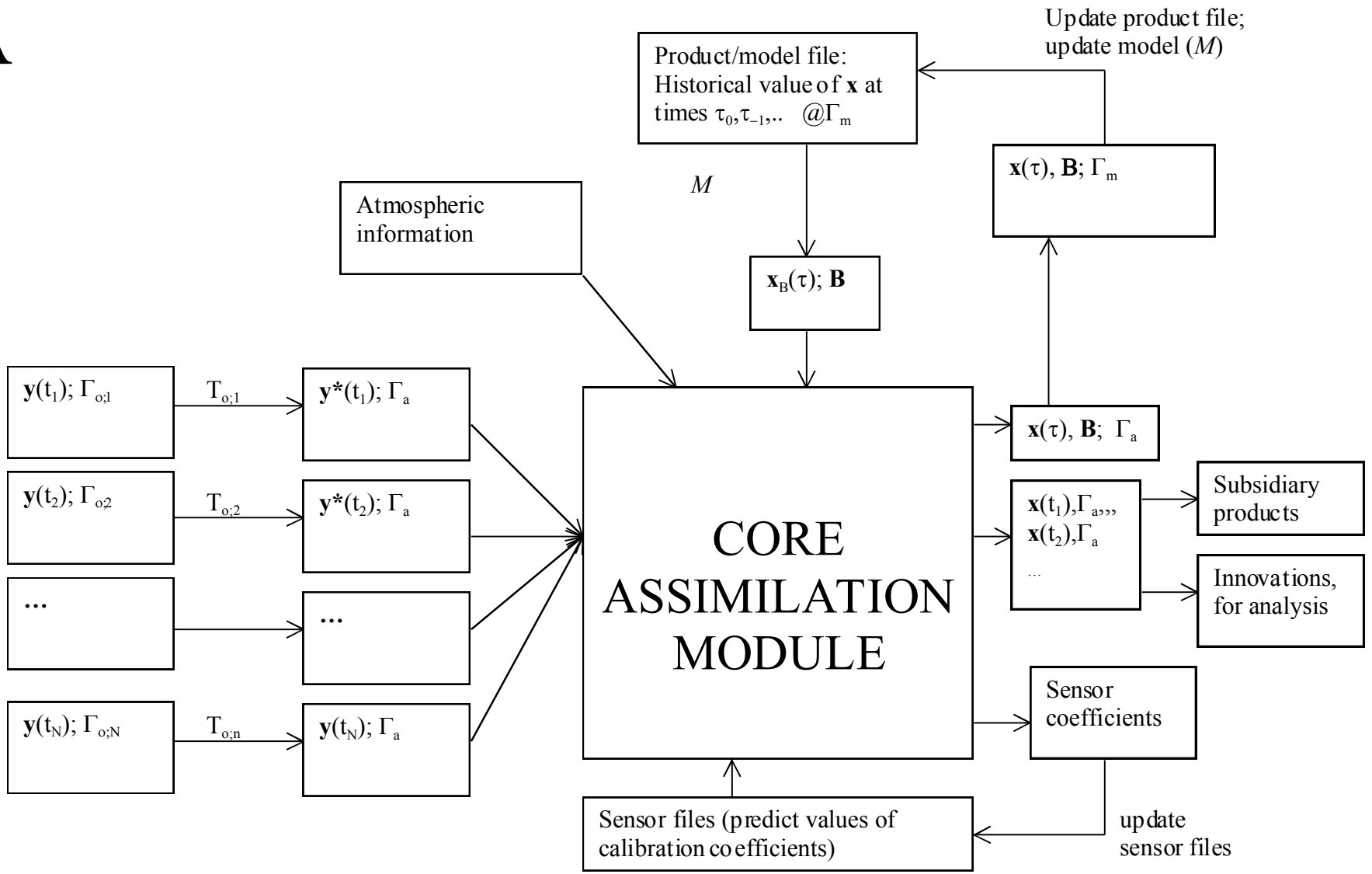
Advantages (to interpolate **y** to model grid):

- Simpler to handle pointing error (once-only gradient calculation)
- Expected to be lower interpolation error unless pixel size is much bigger than model grid size
- Rather simpler calibration error fix (a relevant term can be pre-computed)
- The matrix $\mathbf{K}^T\mathbf{R}^{-1}\mathbf{K}$ is more sparse (should make Hessian easier to work with).

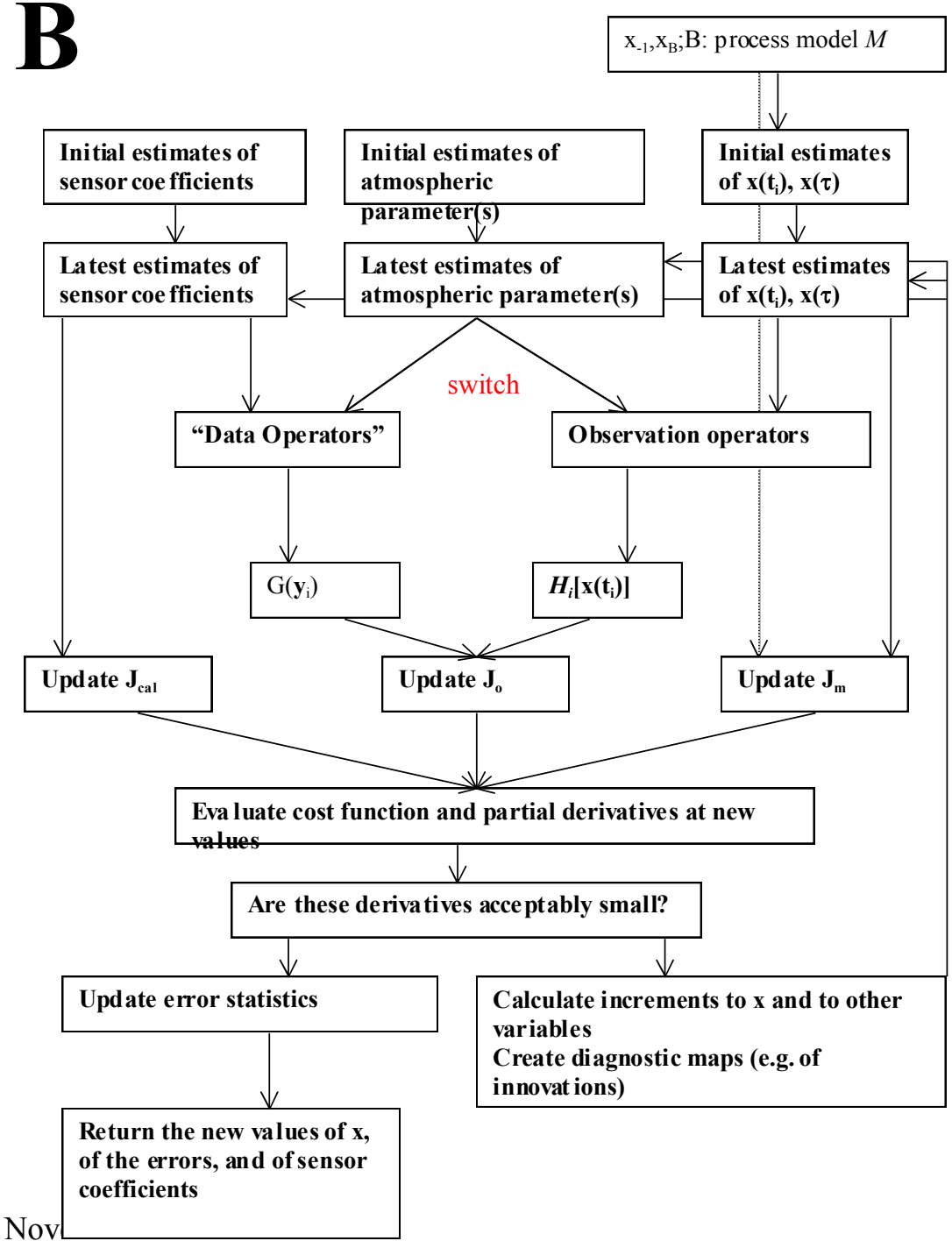
Disadvantage

- **R** no longer diagonal. But it never was (model error) so treating it as diagonal is in keeping with established practice.

A


Update product file;
update model (M)
Product/model file:
Historical value of x at
times $\tau_0,\tau_{-1},..$ @Γ_m
M
$\mathbf{x}(\tau)$, $\mathbf{B}$; Γ_m
Atmospheric
information
$\mathbf{x}_B(\tau)$; $\mathbf{B}$
$\mathbf{y}(t_1)$; $\Gamma_{o;1}$
$T_{o;1}$
$\mathbf{y}^*(t_1)$; Γ_a
$\mathbf{y}(t_2)$; $\Gamma_{o;2}$
$T_{o;2}$
$\mathbf{y}^*(t_2)$; Γ_a
…
…
$\mathbf{y}(t_N)$; $\Gamma_{o;N}$
$T_{o;n}$
$\mathbf{y}(t_N)$; Γ_a
CORE
ASSIMILATION
MODULE
$\mathbf{x}(\tau)$, $\mathbf{B}$; Γ_a
$\mathbf{x}(t_1),\Gamma_a,,,$
$\mathbf{x}(t_2),\Gamma_a$
…
Subsidiary
products
Innovations,
for analysis
Sensor
coefficients
Sensor files (predict values of
calibration coefficients)
update
sensor files

B


x_{-1},x_B;B: process model *M*
Initial estimates of sensor coefficients
Initial estimates of atmospheric parameter(s)
Initial estimates of $x(t_i)$, $x(\tau)$
(perhaps from per-cell inversions)
Latest estimates of sensor coefficients
Latest estimates of atmospheric parameter(s)
Latest estimates of $x(t_i)$, $x(\tau)$
switch
(pointing)
"Data Operators"
Observation operators
$G(y_i)$
$H_i[x(t_i)]$
Update J_{cal}
Update J_o
Update J_m
Evaluate cost function and partial derivatives at new values
Are these derivatives acceptably small?
Update error statistics
Calculate increments to x and to other variables
Create diagnostic maps (e.g. of innovations)
Return the new values of x, of the errors, and of sensor coefficients

Modularity?

How easy is it to plug-in and play?

What if we want to experiment with different RT codes, or process models, or assimilation schemes?

If any auxiliary data changes, how much else has to change?

Modularity?

How easy is it to plug-in and play?

Not very easy.
State vector must be compatible with what's happening both in the OO and in the PM .

Routines all need to know about auxiliary data, and any definitions. e.g. where is aspect measured from?

If we switch to a different solver for atmospheric RT (e.g. based on Fourier decomposition in azimuth) there are consequences for the canopy scheme.

Modularity

Some modularity might still be possible. e.g. At some point in the architecture we make a call to a function called BRDF (say), which generates reflectance fields. A connecting routine just below this could be tied to whatever routine a given experimenter wanted to use.

SIM-10

SUBROUTINE BRDF (XR, PI, I_I, I_R, MU_I, PHI_I, MU_R, PHI_R, LAMDA, REFRAD, IFAIL)

DESCRIPTION:

This is the heart of the canopy RT scheme, and gives the monochromatic reflected radiation in directions (mu_r,phi_r) when the canopy is illuminated by directional radiance of unit strength from the directions mu_i, phi_i).

ARGUMENTS

On entry:

xr	real	vector	State variables for present pixel (real valued)
pi	real	vector	Local parameters (e.g. slope and aspect)
I_i	integer	scalar	Number of directions for incident direction
I_r	integer	scalar	Number of directions for which reflected radiance is required
mu_i & phi_i	real	vector	Direction(s) of incident beam, 2 vectors each of length I_i
mu_r & phi_r	real	vector	Direction(s) of reflected beam, 2 vectors each of length I_r
lamda	real	scalar	The wavelength for which the calculation is made

On exit:

refrad	real	matrix	The required reflected radiance distribution: size I_i by I_r.
ifail	integer	scalar	No value on input, has value 0 for successful exit.

COMMENTS

This could directly call a radiative transfer function itself, or it could use look-up-tables, for instance. Any parameters that relate to any strap-on hot-spot contrivance should be within the brdf routine. Since this may depend on cover type, that needs to be passed as one of the local parameters.

The ifail variable is standard in NAG routines, with non-zero values being returned for different error conditions encountered. I think our scheme should have this in place by default, but I won't be explaining it anywhere else in this strawman.

I see no reason why we shouldn't also calculate within this routine the gradient of these radiances with respect to the state variables, if this is possible via the RT function being analytic. Otherwise, it would be calculated by repeated calls with slightly different xr values, and the gradient approximated by differences.

e.g. BRDF routine

Inputs:

n_i, n_r, ω_i, ω_r, λ, **x**, **p**

$\omega = (\theta, \phi)$

Returns

$r(n_i, n_j)$

$r(l, m)$ is the radiance reflected in direction $\omega_r(m)$ from a beam of light incident from $\omega_i(l)$. The routine doesn't need to know where the sun is, for example.

Modularity

Call a routine to calculate the surface irradiance for a black ground (fn of atmosphere only) $L_0\downarrow(\theta,\phi)$.

Call BRDF function (Canopy RT, function, LUT)

Call coupling routine to get $L\downarrow(\theta,\phi)$ (actual) and $L\uparrow(\theta,\phi)$, given BRDF and $L_0\downarrow(\theta,\phi)$

Conclusions

Well tested,well-defined forward Atm RT models.

Biggest uncertainty is aerosol loading.

Excellent canopy RT models available, still an active research area.

Need to investigate further

methods for clumped surfaces

terrain effects on reflection.

Conclusions

The heterogeneity of the surface (and the desire to apply these methods to moderate resolution imaging systems) calls for high-resolution assimilation grids; this entails dimensionality problems.

The background covariance matrix is large, inverting it impossible (storing it may even be a problem). Can model it as the inverse of a local operator, or simply ignore the correlations, and work at per-cell level (but atmospheric correlations are important).

The inverse Hessian is large, similar considerations. May mean having to use first order gradient descent minimisation methods, or work per-cell, or approximate it and update it (as is done operationally now).

Heterogeneity of the surface entails OO errors larger than those on the observations. Work to be done to account for these.

Conclusions

Advantages exist to making the comparisons "in model space" rather than in "image space", but both possibilities should be included in the prototype. An analysis of interpolation errors should guide us.

A plausible calibration corrector exists.

A speculative corrector for pointing error exists.

Finally, modularity of an eventual, comprehensive scheme may be problematic.