Variational data assimilation for morphodynamic model parameter estimation

Conference or Workshop Item

Published Version

Presentation slides

Variational data assimilation for morphodynamic model parameter estimation

Department of Mathematics, University of Reading:
Polly Smith*, Sarah Dance, Mike Baines, Nancy Nichols,

Environmental Systems Science Centre, University of Reading:
Tania Scott

*email: p.j.smith@reading.ac.uk

This project is funded under the Natural Environment Research Council (NERC) Flood Risk From Extreme Events (FREE) programme, with additional funding provided by the Environment Agency as part of the CASE (Co-operative Awards in Science and Engineering) scheme. Thanks also to HR Wallingford for visits and useful discussions.

Outline

- Background/ motivation
  - what is morphodynamic modelling?
  - why do we need morphodynamic models?
- A simple 1D morphodynamic model
- Data assimilation and parameter estimation
  - how can we use data assimilation to estimate uncertain model parameters?
  - how do we model the background error covariances?
- Results
- Summary

Terminology

- Bathymetry - the underwater equivalent to topography
  - coastal bathymetry is dynamic and evolves with time
  - water action erodes, transports, and deposits sediment, which changes the bathymetry, which alters the water action, and so on
- Morphodynamics - the study of the evolution of the bathymetry in response to the flow induced sediment transport
- Morphodynamic prediction
  - why?
  - how?

Picture courtesy of Nigel Cross,
Lancaster City Council

18km

Kent channel

1960s

1970s

1997

- Channel movement
  - impacts on habitats in the bay
  - affects access to ports
  - has implications for flooding during storm events

Morphodynamic modelling

- Operational coastal flood forecasting is limited near-shore by lack of knowledge of evolving bathymetry
  - but it is impractical to continually monitor large coastal areas
- Modelling is difficult
  - longer term changes are driven by shorter term processes
  - uncertainty in initial conditions and parameters
- An alternative approach is to use data assimilation

Parameter estimation

- Model equations depend on parameters
    - exact values are unknown
    - inaccurate parameter values can lead to growth of model error
    - affects predictive ability of the model
- How do we estimate these values *a priori*?
    - theoretical values
    - calibration

or ...

- data assimilation
    - choose parameters based on observations
    - state augmentation: model parameters are estimated alongside the model state

Simple 1D model

Based on the sediment conservation equation

$$\frac{\partial z}{\partial t} = -\left(\frac{1}{1-\varepsilon}\right)\frac{\partial q}{\partial x}$$

where $z(x,t)$ is the bathymetry, t is time, q is the sediment transport rate in the x direction and ε is the sediment porosity.

For the sediment transport rate we use the power law

$$q = Au^n$$

where $u(x,t)$ is the depth averaged current and **A and n are parameters whose values need to be set**

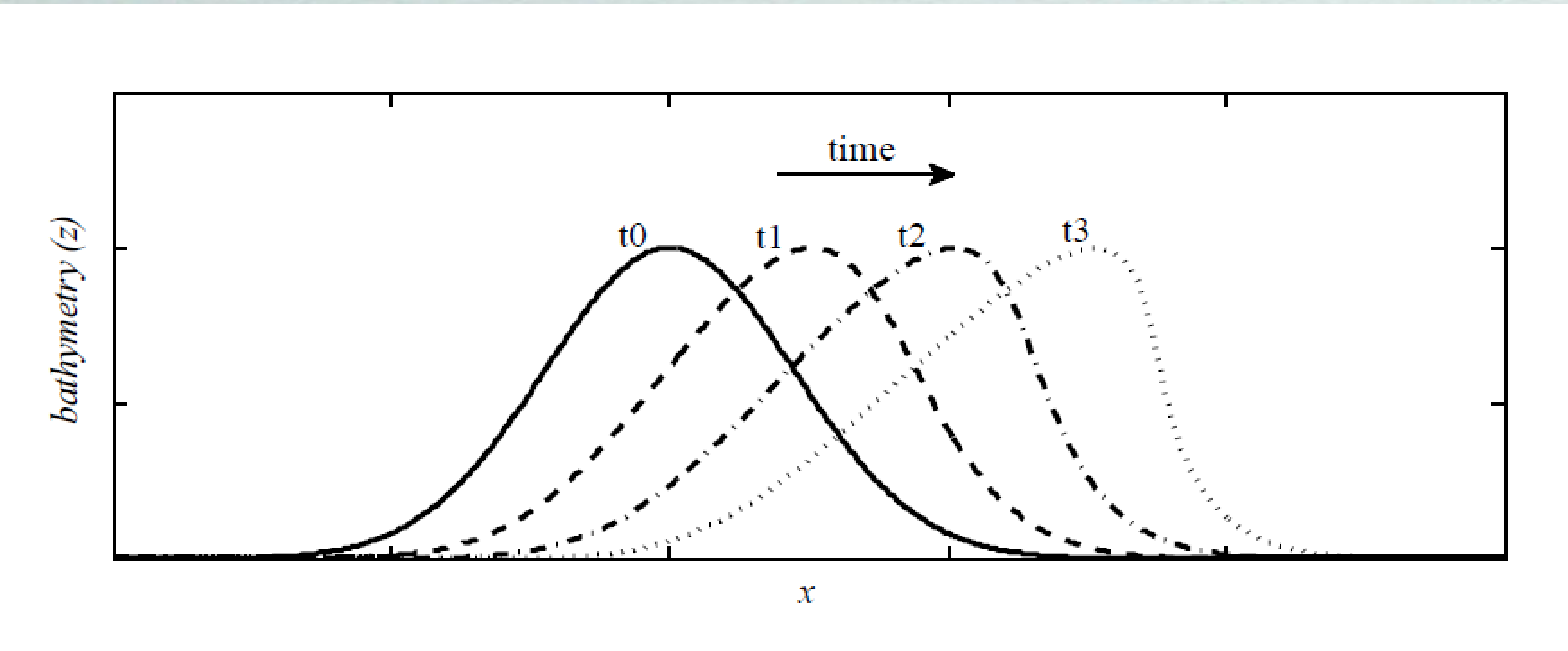
If we assume that water flux (*F*) and height (*H*) are constant

$$F = u(H - z)$$

we can rewrite the sediment conservation equation as

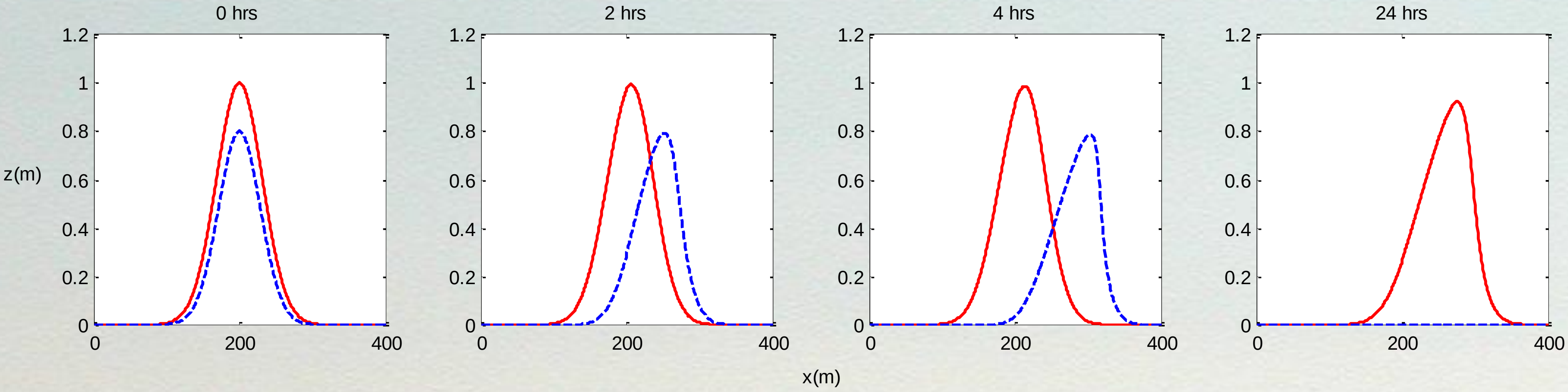
$$\frac{\partial z}{\partial t} + a(z, H, F, \varepsilon, A, n)\frac{\partial z}{\partial x} = 0$$

where $a(z, H, F, \varepsilon, A, n)$ is the advection velocity or bed celerity.


time
t0
t1
t2
t3
bathymetry (z)
x

Can we use data assimilation to estimate the parameters A and n?

model run with incorrect parameters & without data assimilation



red line = correct parameters

blue line = incorrect parameters (A over estimated, n under estimated)

State augmentation

- Dynamical system model

$$\mathbf{z}_{k+1} = \mathbf{f}(\mathbf{z}_k, \mathbf{p}_k)$$

(discrete, non-linear, time invariant)

- Parameter evolution

$$\mathbf{p}_{k+1} = \mathbf{p}_k.$$

- Augmented system model

$$\mathbf{w}_{k+1} = \begin{pmatrix} \mathbf{z}_{k+1} \\ \mathbf{p}_{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}(\mathbf{z}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix} = \tilde{\mathbf{f}}(\mathbf{w}_k)$$

- Observations

$$\mathbf{y}_k = \mathbf{h}(\mathbf{z}_k)$$

in terms of the augmented system ...

$$\mathbf{y}_k = \tilde{\mathbf{h}}(\mathbf{w}_k)$$

where

$$\tilde{\mathbf{h}}(\mathbf{w}) = \tilde{\mathbf{h}}\begin{pmatrix}\mathbf{z}\\ \mathbf{p}\end{pmatrix} = \mathbf{h}(\mathbf{z}).$$

3D Var

Cost function:

$$\tilde{J}(\mathbf{w}) = (\mathbf{w} - \mathbf{w}^b)^{\mathrm{T}} \tilde{\mathbf{B}}^{-1} (\mathbf{w} - \mathbf{w}^b) + (\mathbf{y} - \tilde{\mathbf{h}}(\mathbf{w}))^{\mathrm{T}} \mathbf{R}^{-1} (\mathbf{y} - \tilde{\mathbf{h}}(\mathbf{w}))$$

$\tilde{\mathbf{B}}$ and $\mathbf{R}$ are the covariance matrices of the background and observation errors.

$$\tilde{\mathbf{B}} = \begin{pmatrix} \mathbf{B_{zz}} & \mathbf{B_{zp}} \\ (\mathbf{B_{zp}})^{\mathrm{T}} & \mathbf{B_{pp}} \end{pmatrix}.$$

$\mathbf{B_{zz}}$ state background error covariance

$\mathbf{B_{pp}}$ parameter background error covariance

$\mathbf{B_{zp}}$ state parameter error cross covariance augmented gain matrix:

$$\begin{aligned}\tilde{\mathbf{K}} &= \tilde{\mathbf{B}}\tilde{\mathbf{H}}^{\mathbf{T}}\left[\tilde{\mathbf{H}}\tilde{\mathbf{B}}\tilde{\mathbf{H}}^{\mathbf{T}} + \mathbf{R}\right]^{-\mathbf{1}} \\ &= \begin{pmatrix} \mathbf{B_{zz}H^T} \\ \mathbf{B_{zp}^T H^T} \end{pmatrix} \left[\mathbf{HB_{zz}H^T} + \mathbf{R}\right]^{-1} \\ &\stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{K_z} \\ \mathbf{K_p} \end{pmatrix}\end{aligned}$$

state & parameter updates:

$$\begin{aligned}\mathbf{z}^{\boldsymbol{a}} &= \mathbf{z}^{\boldsymbol{b}} + \mathbf{K_z}(\mathbf{y} - \mathbf{h}(\mathbf{z}^{\boldsymbol{b}})) \\ \mathbf{p}^{\boldsymbol{a}} &= \mathbf{p}^{\boldsymbol{b}} + \mathbf{K_p}(\mathbf{y} - \mathbf{h}(\mathbf{z}^{\boldsymbol{b}}))\end{aligned}$$

State-parameter cross covariances

The Extended Kalman filter (EKF)

State forecast:

$$\mathbf{w}_{k+1}^{f} = \tilde{\mathbf{f}}_k(\mathbf{w}_k^a)$$

Error covariance forecast:

$$\mathbf{P}_{k+1}^{f} = \mathbf{F}_k \mathbf{P}_k^a \mathbf{F}_k^T$$

where

$$\mathbf{F}_k = \left.\frac{\partial \tilde{\mathbf{f}}}{\partial \mathbf{w}}\right|_{\mathbf{w}_k^a} = \left.\begin{pmatrix} \frac{\partial \mathbf{f}(\mathbf{z},\mathbf{p})}{\partial \mathbf{z}} & \frac{\partial \mathbf{f}(\mathbf{z},\mathbf{p})}{\partial \mathbf{p}} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}\right|_{\mathbf{z}_k^a,\mathbf{p}_k^a} = \begin{pmatrix} \mathbf{M}_k & \mathbf{N}_k \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$$

Error covariance forecast:

$$\mathbf{P}^f_{k+1} = \begin{pmatrix} \mathbf{M}_k \mathbf{P}^a_{\mathbf{zz}_k} \mathbf{M}_k^T + \mathbf{N}_k \mathbf{P}^a_{\mathbf{pp}_k} \mathbf{N}_k^T & \mathbf{N}_k \mathbf{P}^a_{\mathbf{pp}_k} \\ \mathbf{P}^a_{\mathbf{pp}_k} \mathbf{N}_k^T & \mathbf{P}^a_{\mathbf{pp}_k} \end{pmatrix}$$

a new hybrid approach ...

$$\tilde{\mathbf{B}}_k = \begin{pmatrix} \mathbf{B}_{\mathbf{zz}} & \mathbf{N}_k \mathbf{B}_{\mathbf{pp}} \\ \mathbf{B}_{\mathbf{pp}} \mathbf{N}_k^T & \mathbf{B}_{\mathbf{pp}} \end{pmatrix}$$

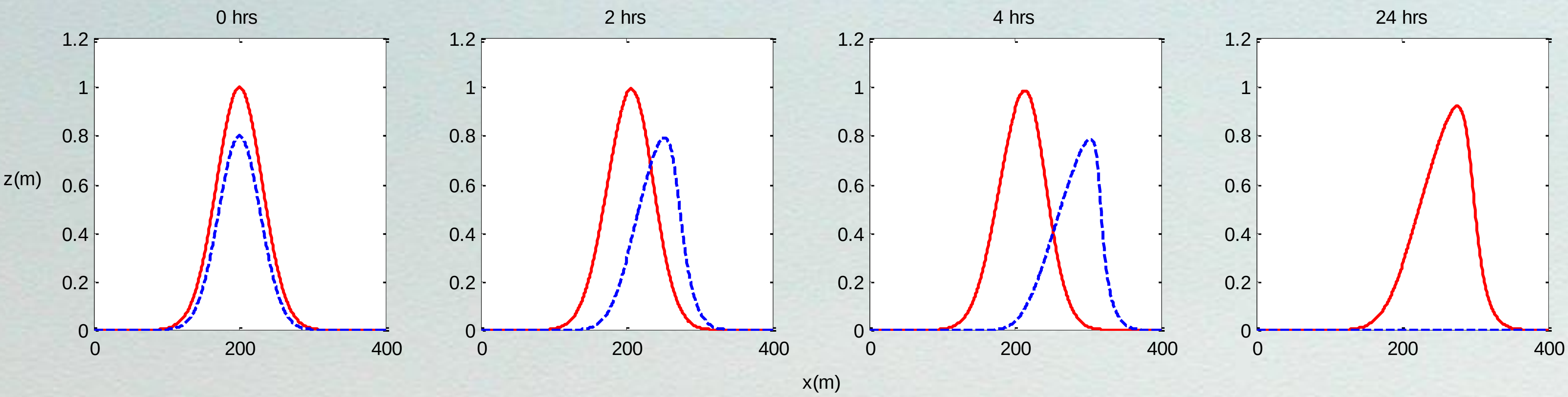
for our simple 2 parameter model

$$
\begin{aligned}
\mathbf{B}_{\mathbf{zp}_k} &= \mathbf{N}_k \mathbf{B}_{\mathbf{pp}} \\
&= \begin{pmatrix} \frac{\partial \mathbf{f}_k}{\partial A} & \frac{\partial \mathbf{f}_k}{\partial n} \end{pmatrix} \begin{pmatrix} \sigma_A^2 & \sigma_{An} \\ \sigma_{An} & \sigma_n^2 \end{pmatrix} \\
&= \begin{pmatrix} \sigma_A^2 \frac{\partial \mathbf{f}_k}{\partial A} + \sigma_{An} \frac{\partial \mathbf{f}_k}{\partial n} & \sigma_n^2 \frac{\partial \mathbf{f}_k}{\partial n} + \sigma_{An} \frac{\partial \mathbf{f}_k}{\partial A} \end{pmatrix}
\end{aligned}
$$

Model setup

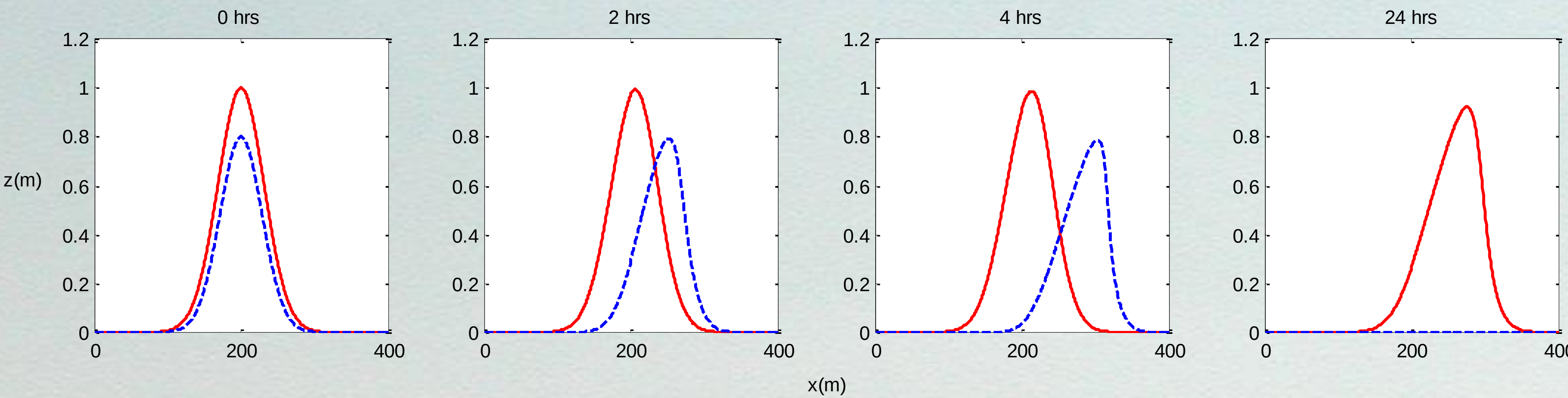
- Assume perfect model and observations
- Identical twin experiments
  - reference solution generated using Gaussian initial data and parameter values $A = 0.002\ ms^{-1}$ and $n = 3.4$
- Use incorrect model inputs
  - inaccurate initial bathymetry
  - inaccurate parameter estimates
- 3D Var algorithm is applied sequentially
  - observations taken at fixed grid points & assimilated every hour
  - the cost function is minimized iteratively using a quasi-Newton descent algorithm
- Covariances
  - $\mathbf{B_{zz}}$ fixed
  - $\mathbf{B_{zp}}$ time varying

without data assimilation

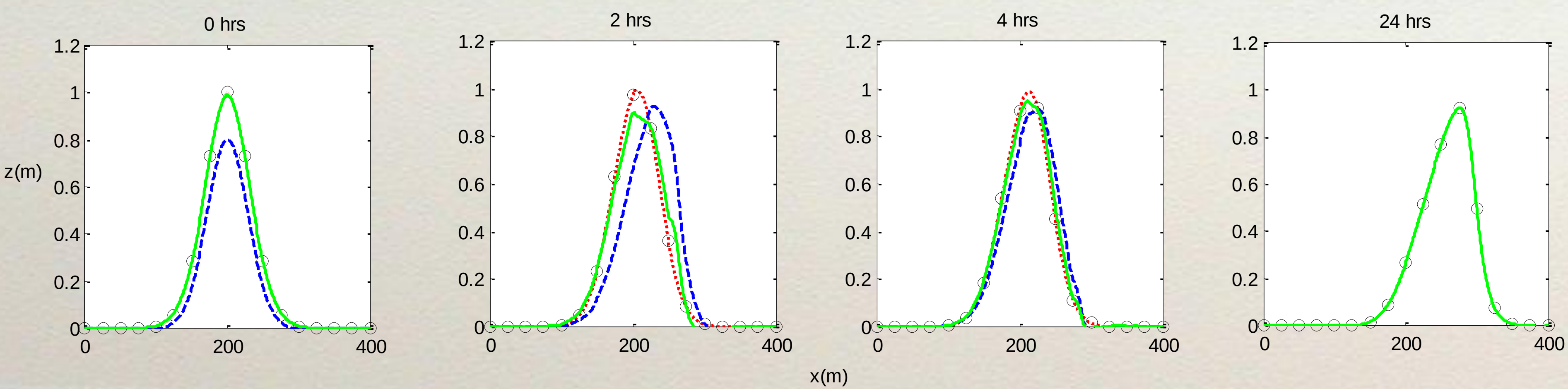


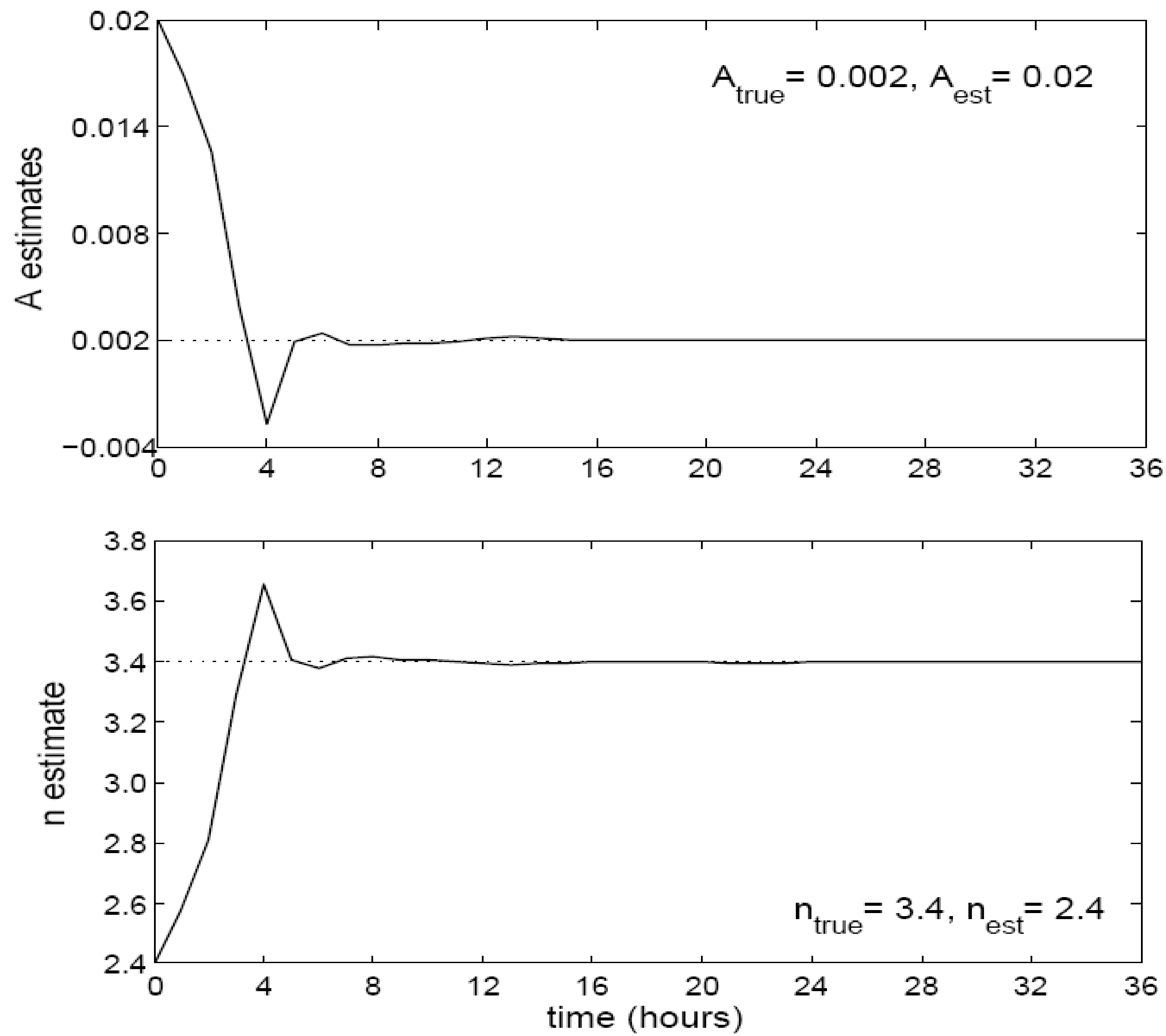
with data assimilation …

without data assimilation

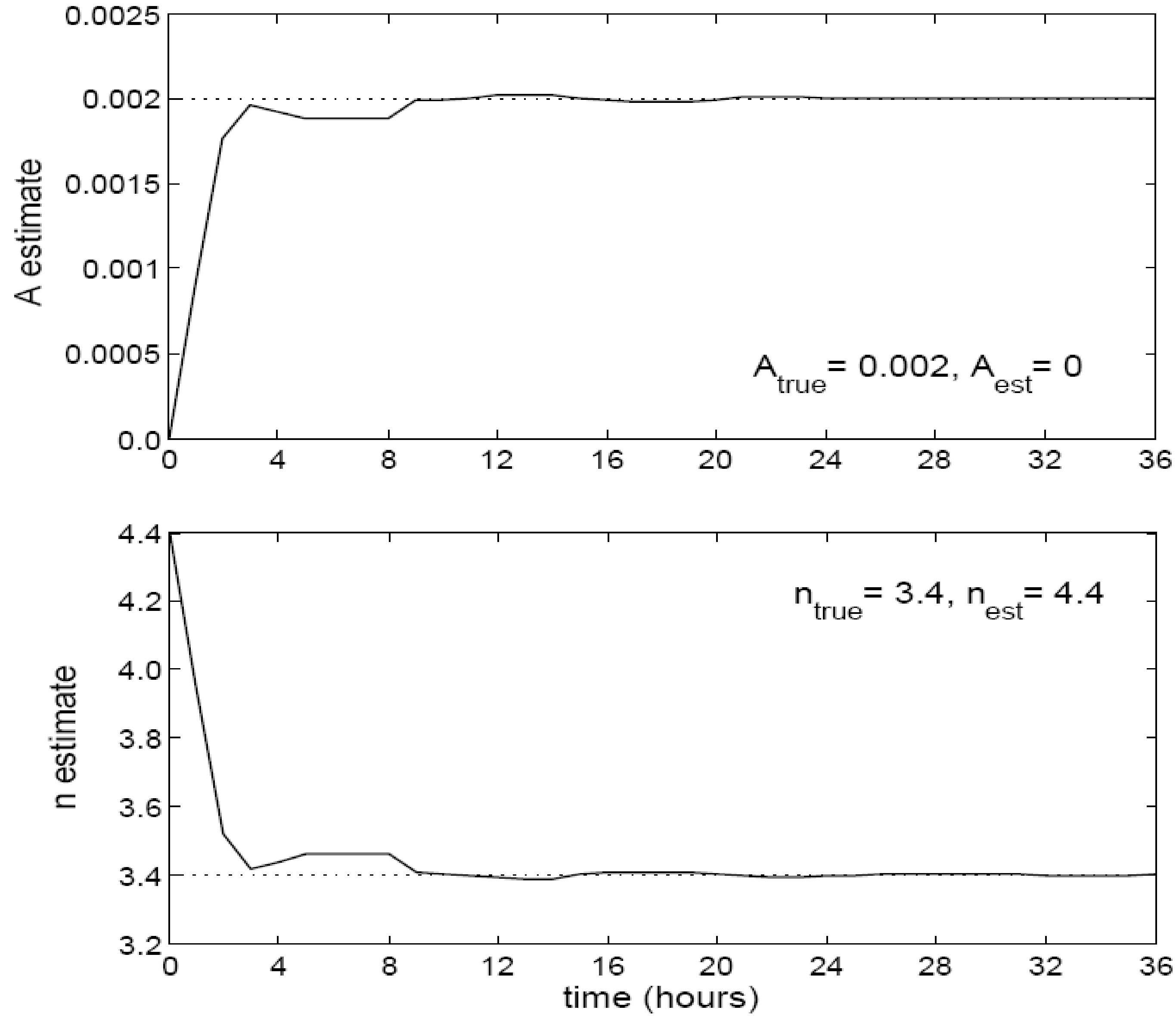


with data assimilation …

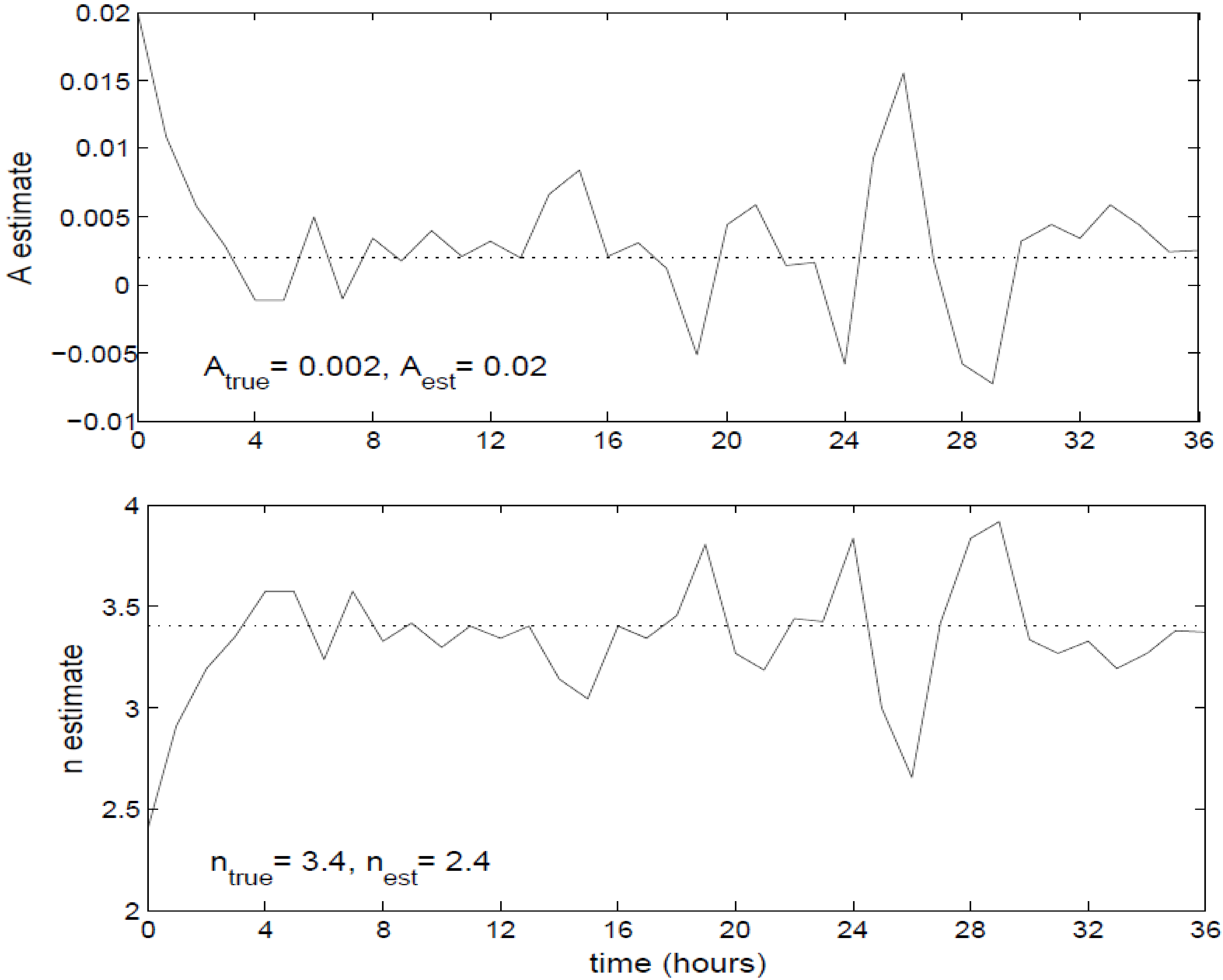




A estimates
A_{true}= 0.002, A_{est}= 0.02
0.02
0.014
0.008
0.002
−0.004
0
4
8
12
16
20
24
28
32
36
n estimate
n_{true}= 3.4, n_{est}= 2.4
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.4
time (hours)



A estimate
0.0025
0.002
0.0015
0.001
0.0005
0.0
A_{true}= 0.002, A_{est}= 0
0
4
8
12
16
20
24
28
32
36
n estimate
4.4
4.2
4.0
3.8
3.6
3.4
3.2
n_{true}= 3.4, n_{est}= 4.4
time (hours)



A estimate
0.02
0.015
0.01
0.005
0
−0.005
−0.01
A_{true}= 0.002, A_{est}= 0.02
0
4
8
12
16
20
24
28
32
36
n estimate
4
3.5
3
2.5
2
n_{true}= 3.4, n_{est}= 2.4
time (hours)

Summary

- Presented a novel approach to model parameter estimation using data assimilation
  - demonstrated the technique using a simple morphodynamic model
- Results are very encouraging
  - scheme is capable of recovering near-perfect parameter values
  - improves model performance
- What next …?
  - can our scheme be successfully applied to more complex models?
  - can we say anything about the convergence of the system?

Questions?

Simple Models of Changing Bathymetry with Data Assimilation

P.J Smith, M.J. Baines, S.L. Dance, N.K. Nichols and T.R. Scott

Department of Mathematics, University of Reading

Numerical Analysis Report 10/2007*

Data Assimilation for Parameter Estimation with Application to a Simple Morphodynamic Model

P.J Smith, M.J. Baines, S.L. Dance, N.K. Nichols and T.R. Scott

Department of Mathematics, University of Reading

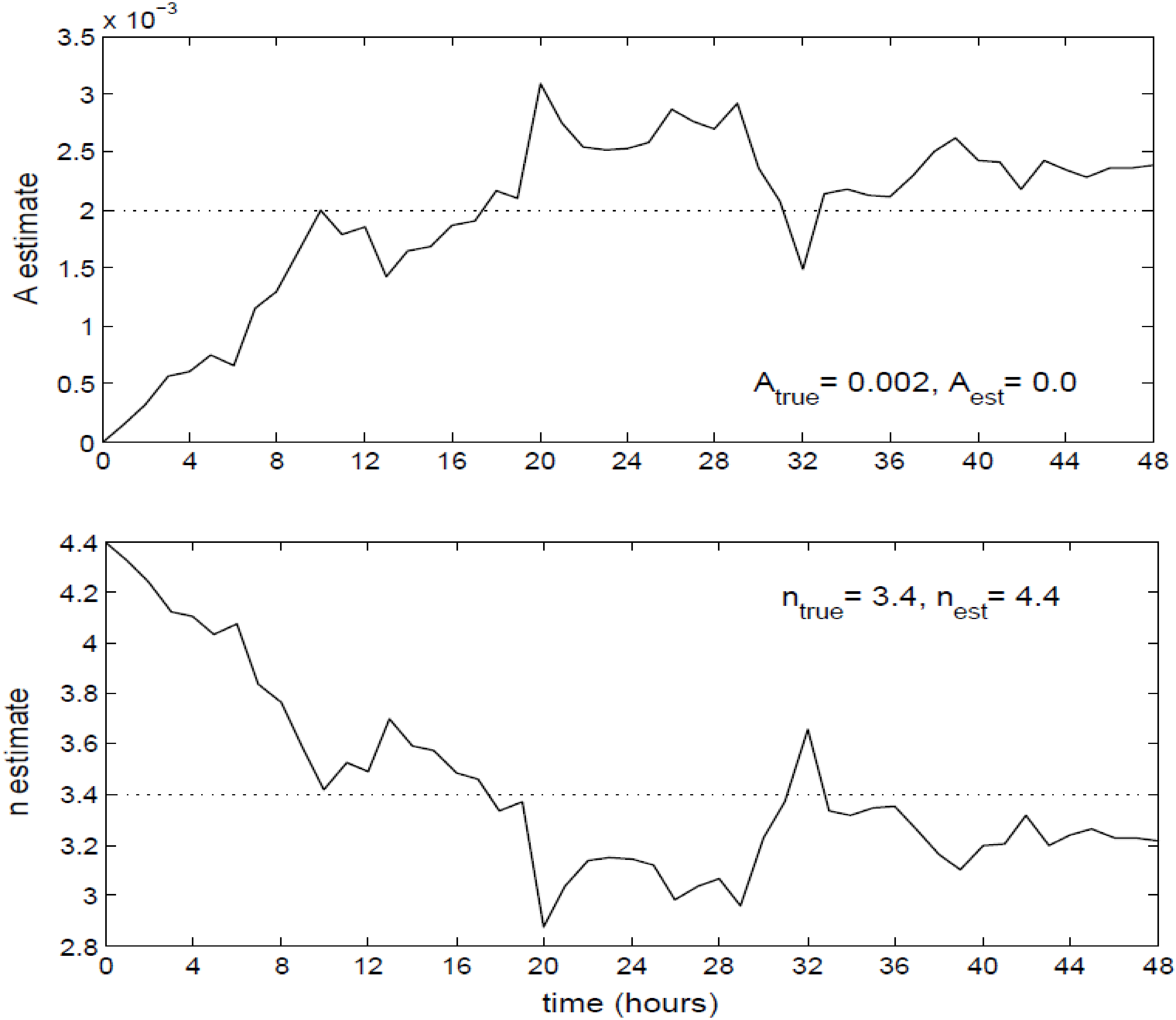
Mathematics Report 2/2008*

Variational data assimilation for parameter estimation: application to a simple morphodynamic model

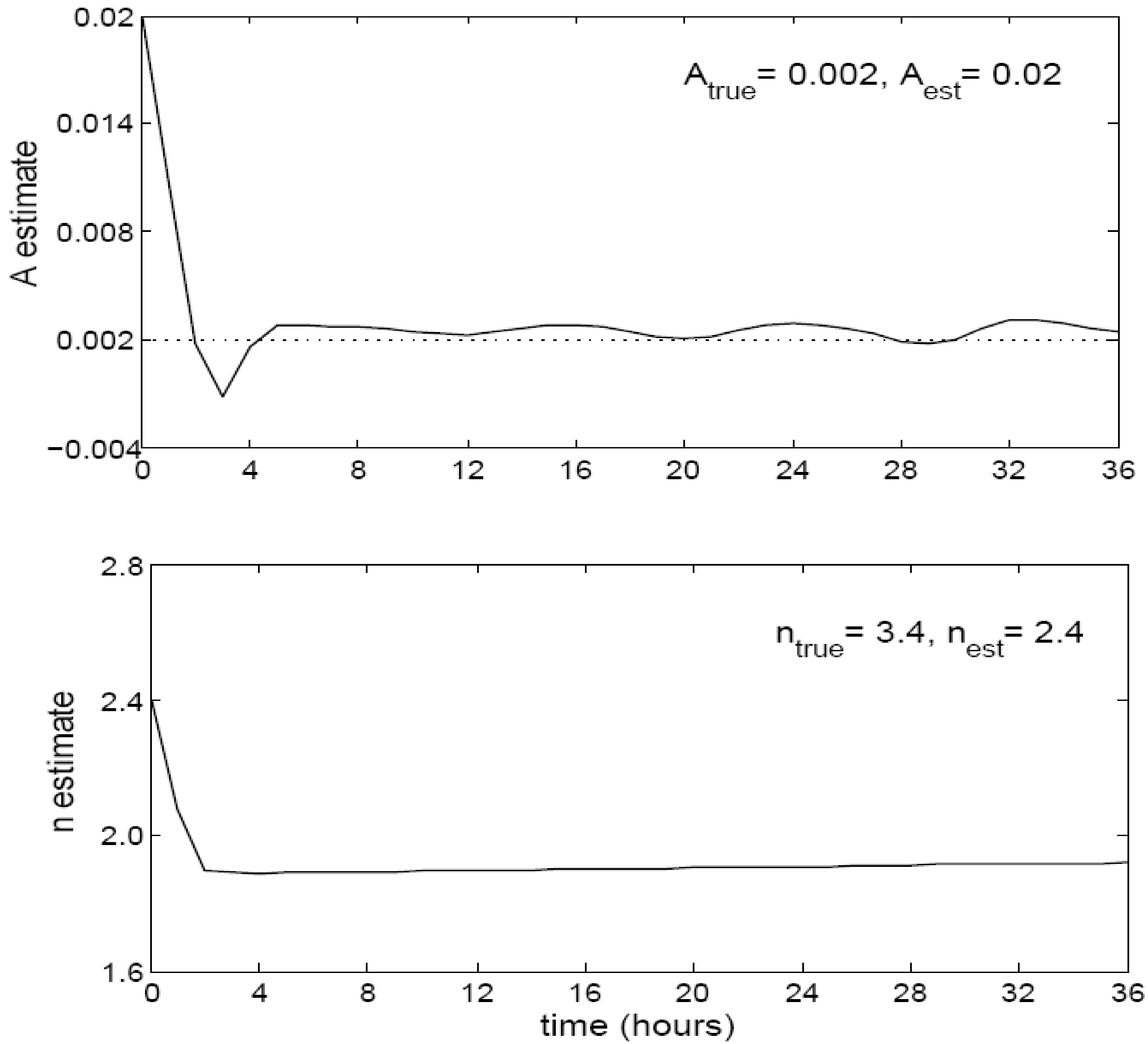
P.J Smith, M.J. Baines, S.L. Dance, N.K. Nichols and T.R. Scott

Submitted to Ocean Dynamics PECS 2008 Special Issue*

*available from http://www.reading.ac.uk/maths/research/



x 10^-3
A estimate
A_true = 0.002, A_est = 0.0
n estimate
n_true = 3.4, n_est = 4.4
time (hours)



0.02
0.014
0.008
0.002
−0.004
A estimate
A_{true}= 0.002, A_{est}= 0.02
0
4
8
12
16
20
24
28
32
36
2.8
2.4
2.0
1.6
n estimate
n_{true}= 3.4, n_{est}= 2.4
time (hours)