Contaminant-Driven Simulations

or How Polluted Can Your Simulation Become over Time

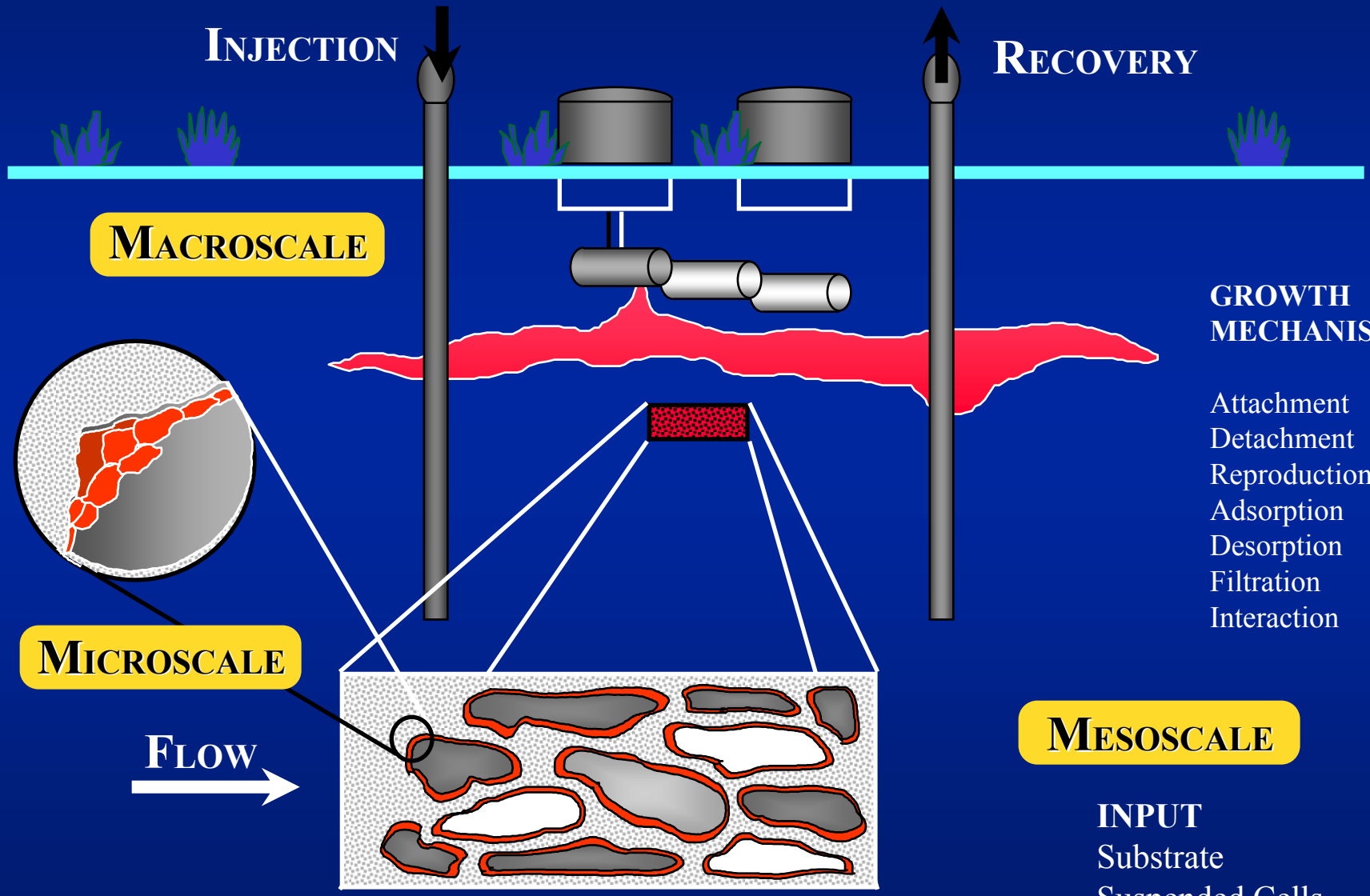
Craig C. Douglas
University of Kentucky and Yale University
douglas-craig@cs.yale.edu
http://www.dddas.org

Martin Cole, Yalchin Efendiev, Richard Ewing, Victor Ginting, Chris Johnson, Greg Jones, Raytcho Lazarov, Deng Li, and Jenny Simpson

Joint project of the University of Kentucky, Texas A&M, and the University of Utah

Supported in part by the National Science Foundation (ITR-DDDAS)

Bioremediation Strategies



GROWTH MECHANISMS

Attachment
Detachment
Reproduction
Adsorption
Desorption
Filtration
Interaction

INPUT
Substrate
Suspended Cells
Oxygen

Forward versus Backwards in Time

- Forward
  - Standard pollutant tracking problem in some sense
  - Start by knowing who is the polluter and where it began
- Backward
  - Might not know who the polluter is or where it began
  - Determine it correctly
    - Report it so remediation can be started
    - Get sued
      - Spend 5-10 years dealing with lawyers
  - Virtual telemetry solves legal problems until software used in the field by others who are used to lawsuits

Data to Drive Application

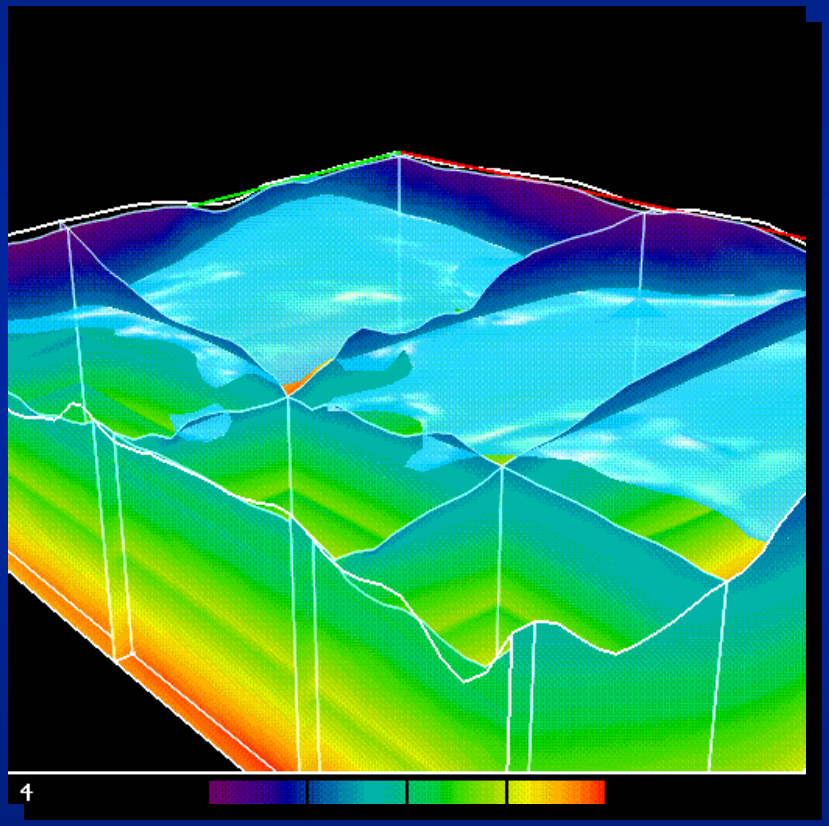
- Where is the contaminant?
  - **Use remote sensing data to locate media, update positions, and find new spread directions.**
    - **Open water bodies**
      - **Buoys and/or submerged stations**
        - **Chemical sensors**
        - **Visible, near IR, and IR scanning**
        - **Radiation detection**
    - **Underground**
      - **Well sensors**

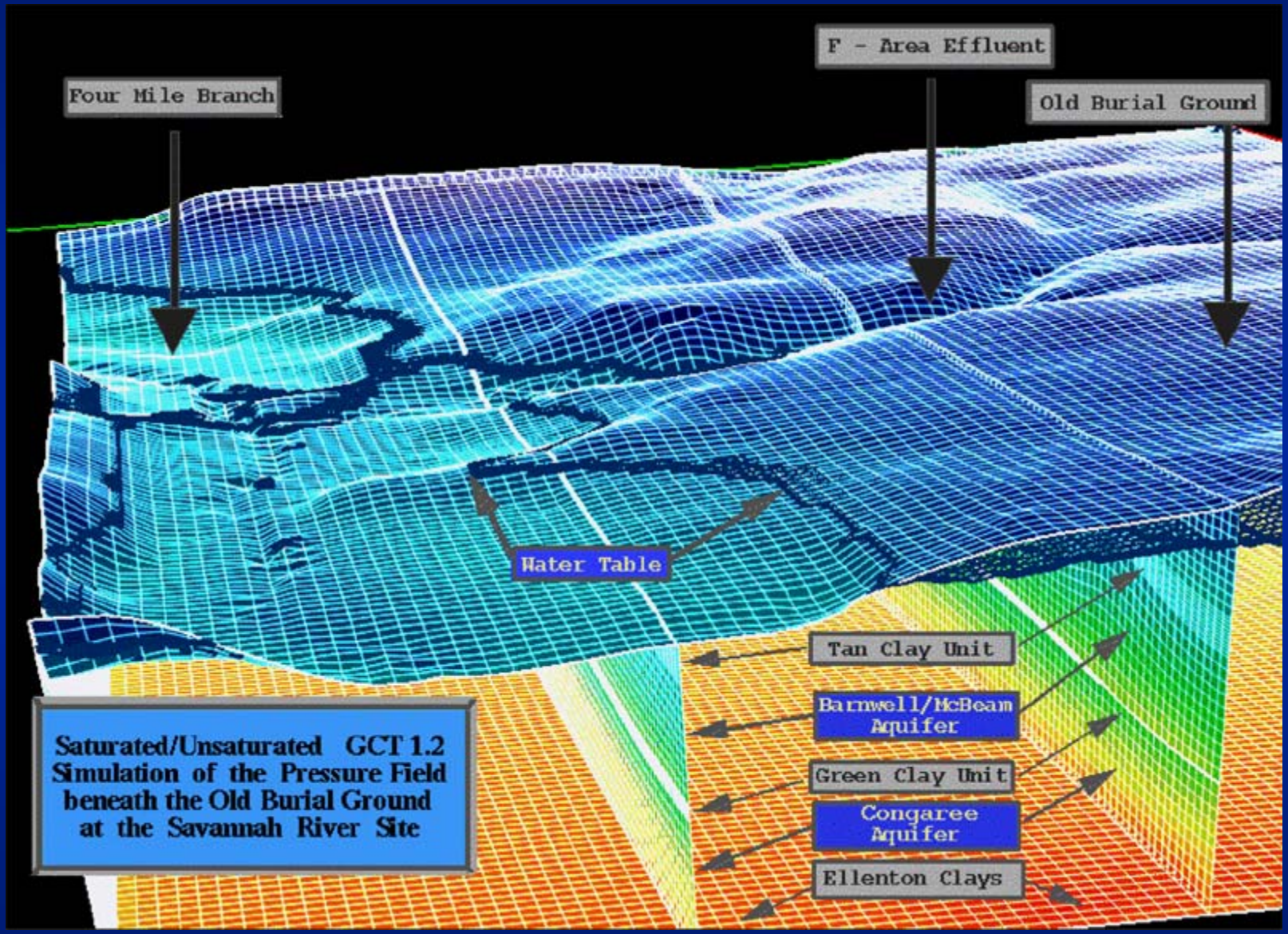
Data to Drive Application (cont.)

- What is the terrain like in that area? What small-scale features are there?
  - **New topography sets give world topography at 30 arcsec (~ 1 km), US at 3 arcsec (~100 m).**
  - **Better local sources usually available from fire and police departments.**
- What are the changing weather conditions?
  - **Large-scale data (current analyses or forecasts) used for initial conditions and for updating boundary conditions.**
- **People as data sources and decision makers.**
  - **Hot spots, scales, subproblems, recombining subproblems, …**

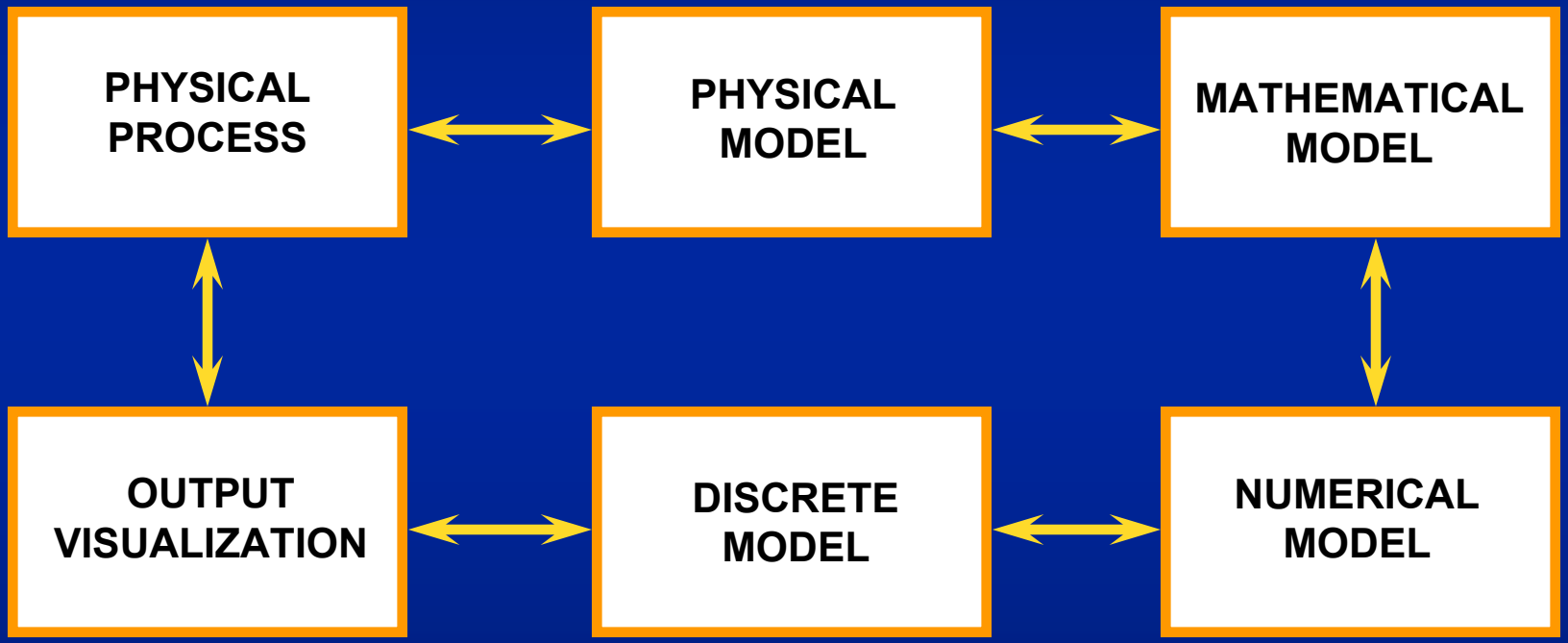
Savannah River Site

- Difficult topography
- Highly Heterogeneous Soils
- Saturated and Unsaturated Flows
- Reactions with disparate time scale
- Transient/Mixed Boundary Conditions

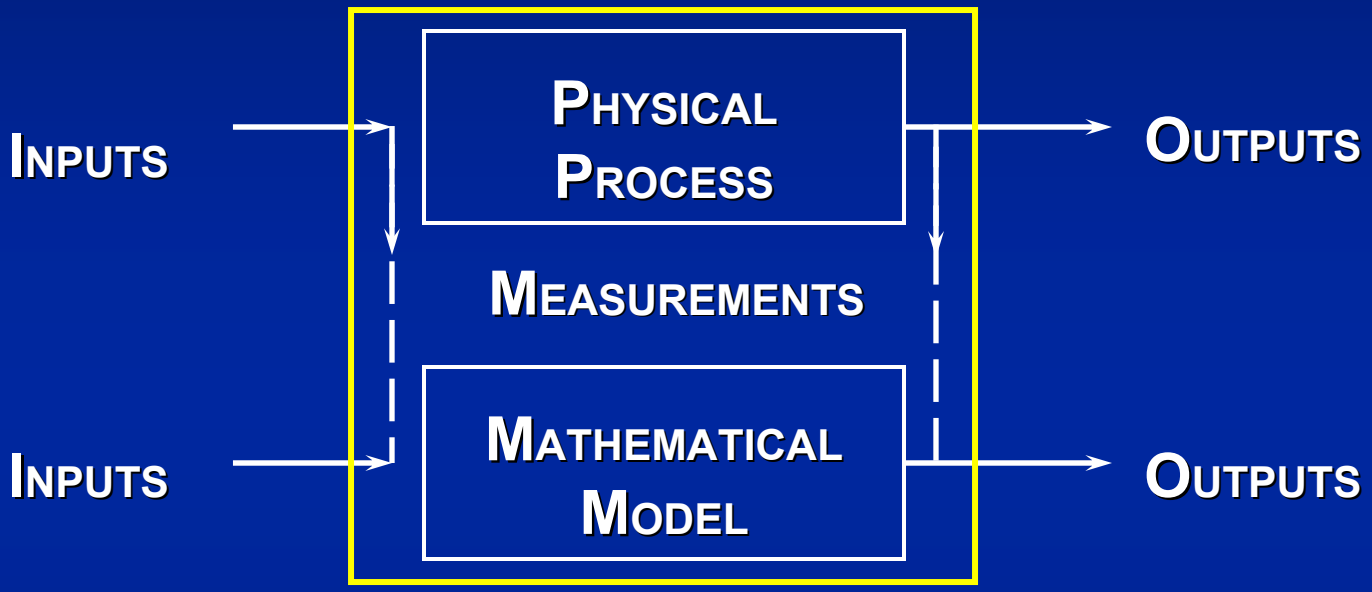



F - Area Effluent
Four Mile Branch
Old Burial Ground
Water Table
Tan Clay Unit
Barnwell/McBean Aquifer
Green Clay Unit
Congaree Aquifer
Ellenton Clays
Saturated/Unsaturated GCT 1.2 Simulation of the Pressure Field beneath the Old Burial Ground at the Savannah River Site

Modeling Process



Identification (Inverse) Problem



- Determine suitable mathematical model
- Estimate parameters within mathematical model
- Long term simulations provide lots of data that can be used to approximate the solution to the actual inverse problem.

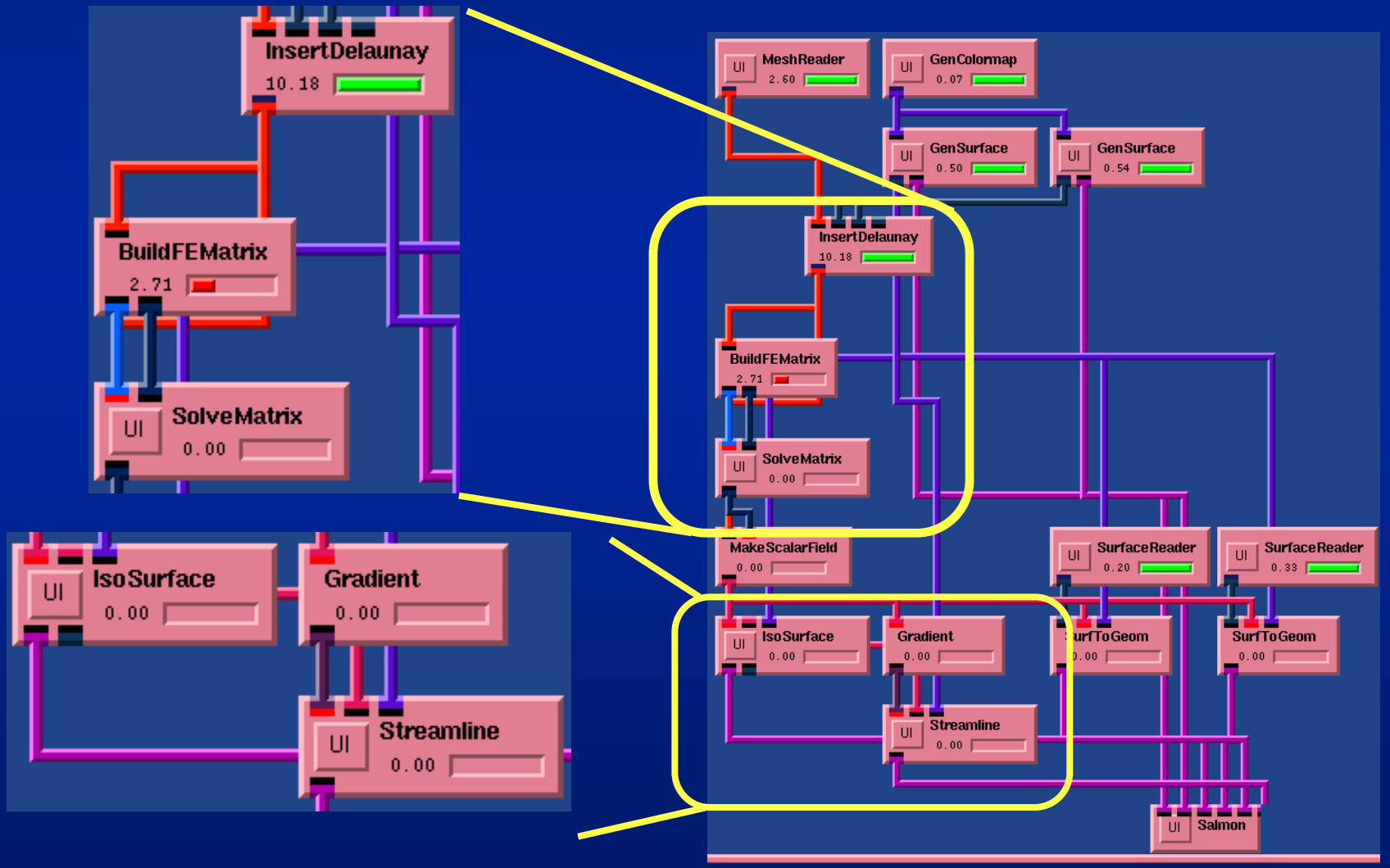
Issues of Perturbations from On-Line Data Inputs

- Solve: $F(x+\Delta x(t)) = 0 \leftrightarrow$ Choice of new approximation for x
  - Do not need a precise solve of equation at each step
    - Incomplete solves of a sequence of related models
    - Effects of perturbations (either data or model)
    - Convergence questions?
  - Premium on quick approximate direction choices
    - Lower-rank updates
    - Continuation methods
  - Interchanges between algorithms and simulations
  - *Local* boundaries and conditions not known *a priori*
- Fault-tolerant algorithms

Many Different Components that We Want to Vary *Quickly* during Project

- We need graphics, solvers, and specialized codes
  - SCIRun has great potential and will be used
    - *Open source and free*
    - Extensible since owners of code are project members
    - Easy to construct complicated codes using cut and paste plus connect the boxes techniques
  - Already added to SCIRun
    - General, mixed finite element basis functions
    - Sensor input (virtual or real telemetry)

SCIRun Methodology



SCIRun and Telemetry

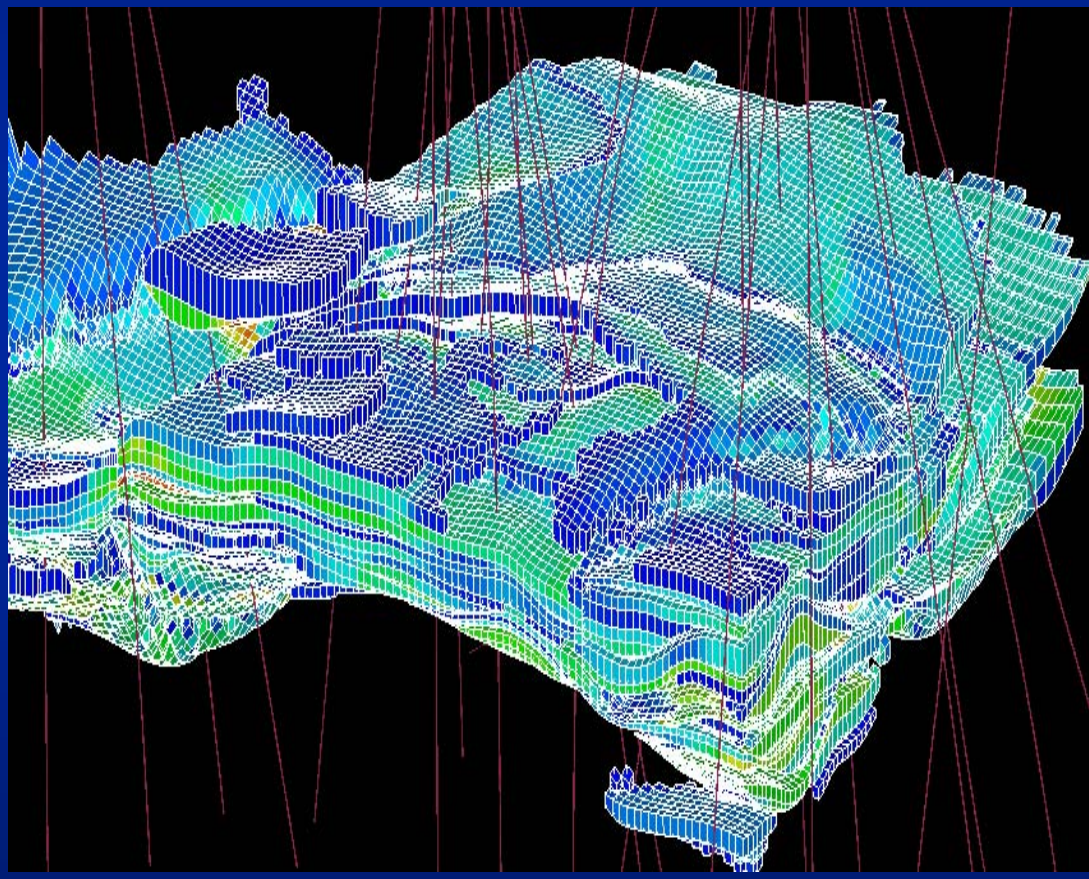
- Streaming paradigm very useful
  - Add/delete remote clients trivially
  - Well understood for audio and somewhat less for video
- Streaming implementations problematic for DDDAS
  - Standard formats lossy (filter out frequencies that cannot be heard or seen)
  - Implementations of lossless formats (e.g., Ogg Vorbis) are lossy, too ☹

SCIRun and Telemetry

- Adding streaming modules
  - Line points to a stream, either incoming or outgoing, instead of another SCIRun module
  - Multicasting is an alternative, but it is banned by almost all Internet providers and may be outlawed shortly
- Fill in data (interpolation as needed) as data comes in
- Timestamp data so that running backwards in time possible and reverting to an earlier time step (warm restart) is possible (locally or globally)

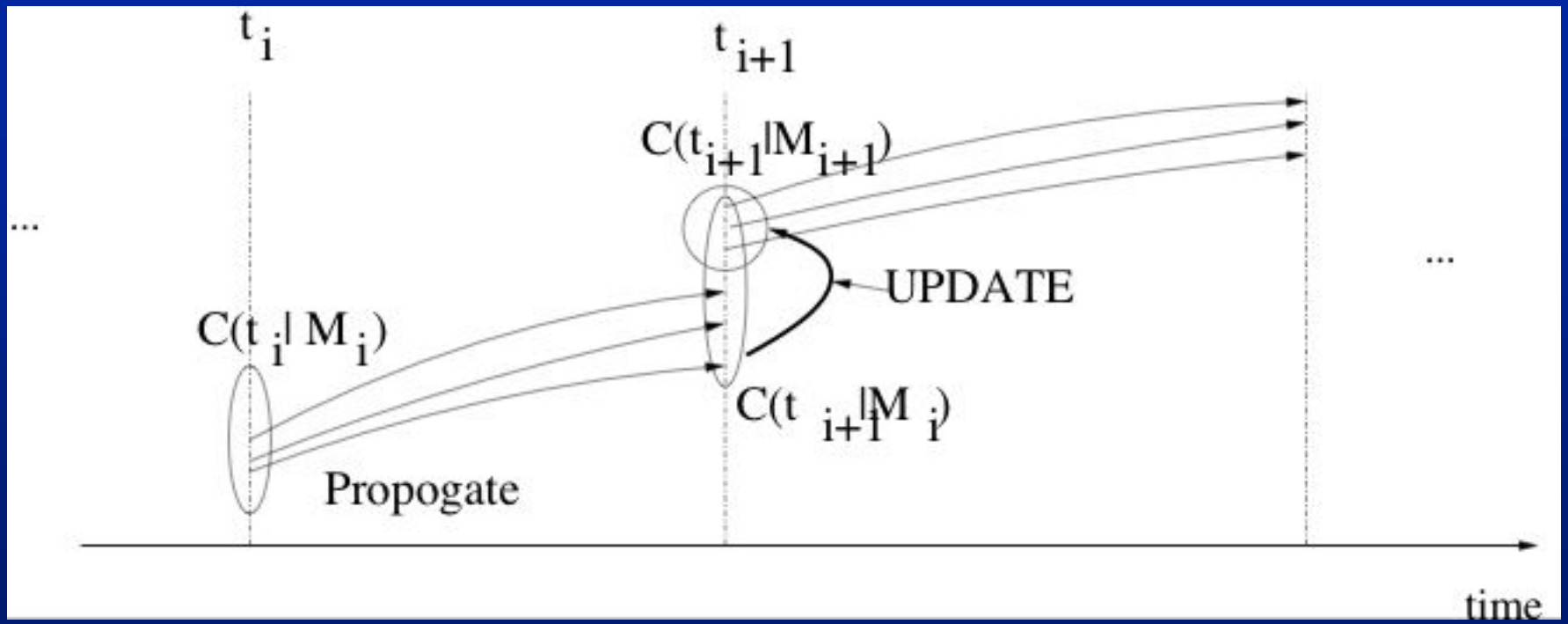
Why Interpolation Is Required

- Data available only at subset of mesh points
- Data not available on mesh at all
- Multiscale interpolation required



Sensor Data Insertion

- C ($t_i|M_i$) object of interest at time t_i conditioned by data M_i. Sensor data from subset of nodes of a coarse triangulation. Geostatistical information filters the updates.



Nonlinear parabolic case

- General equation of form
  (1) $D_t u_\varepsilon = \mathrm{div}(a_\varepsilon(x,t,u_\varepsilon,D_x u_\varepsilon)) + a_{0,\varepsilon}(x,t,u_\varepsilon,D_x u_\varepsilon)$,
  in $Q_0 \times [0,T]$, where Q_0 is the spatial domain and ε represents small hetereogenieties.
- S^h space of piecewise linears on coarse mesh.
- We construct mapping $E: S^h \rightarrow V^{h,\varepsilon}$ by constructing mapping that satisfies in $K \times [t_i, t_{i+1}]$, K a coarse element,
  (2) $D_t u_{\varepsilon,h}(x,t) = \mathrm{div}(a_\varepsilon(x,t,\eta,u_\varepsilon,D_x u_{\varepsilon,h}))$
  where is the average of $u_\varepsilon \cdot u_{\varepsilon,h}(x,t)$ and is the solution of (2) on the fine scale mesh.

Multiscale Comparison Method

- Boundary conditions in (2): We use the sensor data, if available, or the last known data otherwise. In some cases we cannot fix the values since this would cause artificial discontinuities in the solution of (2).
- We compare data at sensor locations using

$$\int_{Q_0} (u_h(x,t_{i+1}) - u_h(x,t_i))v_h dx + \sum_K \int_{t_i}^{t_{i+1}} \int_K ((a_\varepsilon(x,t,\eta,D_x u_{h,\varepsilon}), Dv_h) +$$

$$a_{0,\varepsilon}(x,t,\eta,D_x u_{h,\varepsilon})v_h)dxdt = \int_{t_i}^{t_{i+1}} \int_{Q_0} f v_h dxdt$$

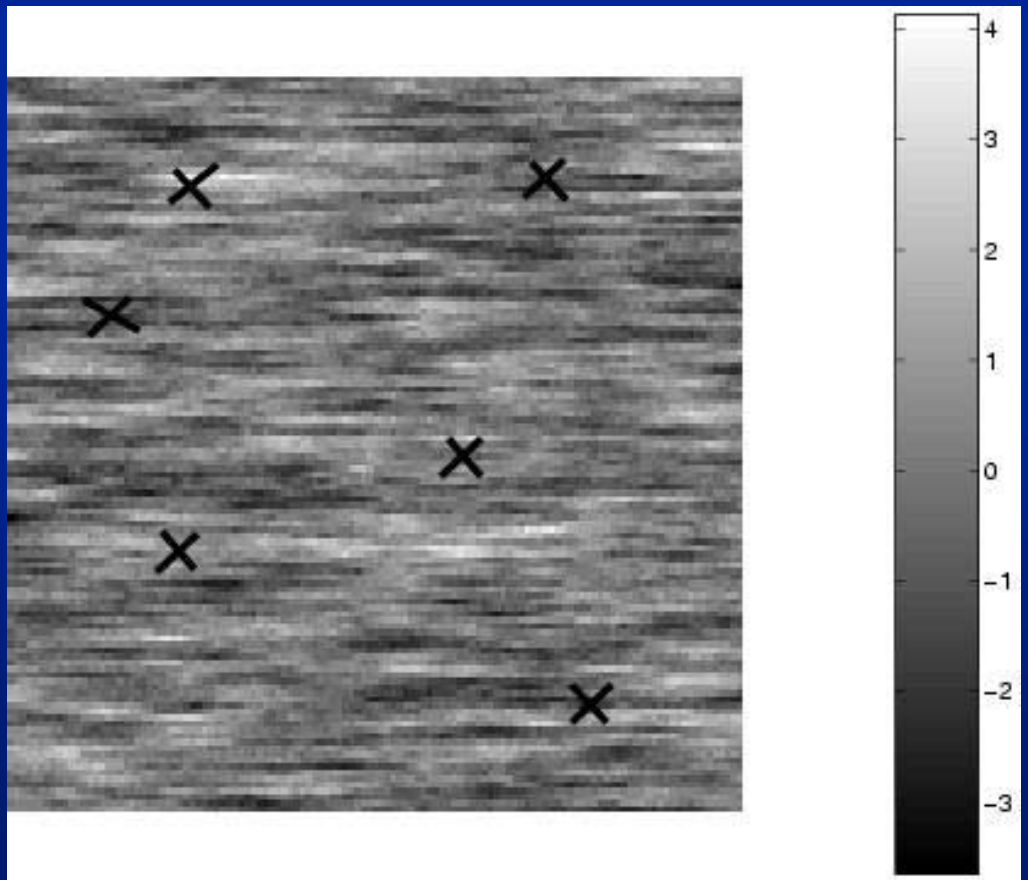
- Coarse mesh used for most of the computation, making the method very effective.

Numerical Examples: Configuration

- Represent cross section in *x-z* plane of subsurface
- System lengths L_x and L_z
- System length in $x=5z$
- *121x121* realizations of overall variance σ^2 and correlation structure
- Dimensionless lengths l_x and l_z, where each correlation length is nondimensionalized by the corresponding system length. Hence, $l_x = 0.3$ means the actual permeability field is $0.3\ L_x$

Numerical Examples: Sample

- l_x=0.2 and l_z=0.02, sensors marked by *X*'s

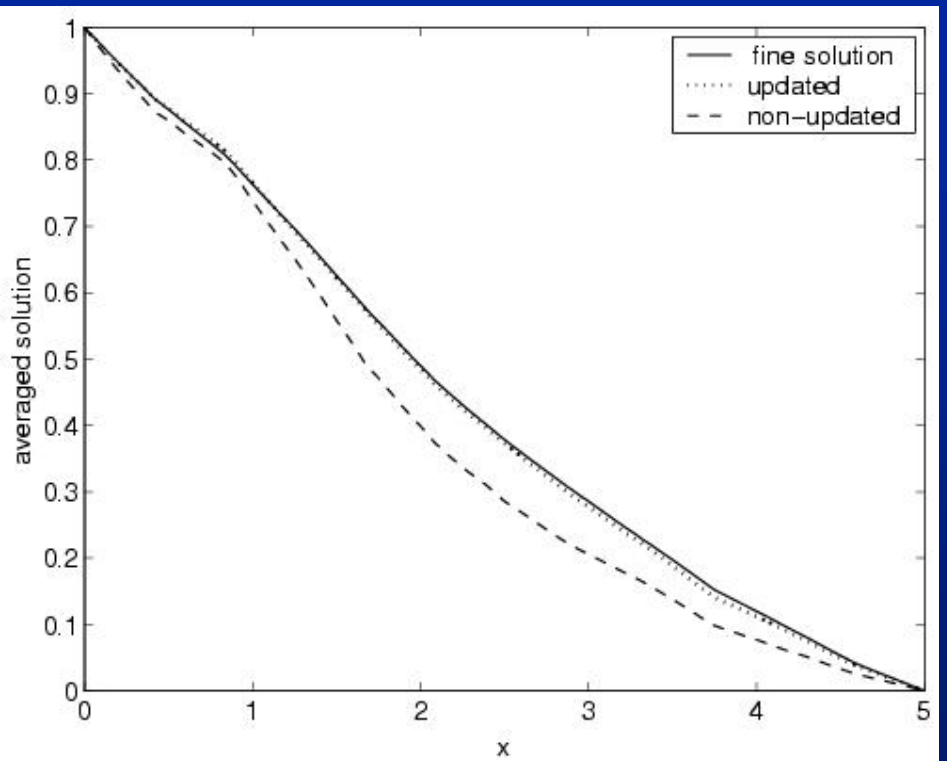


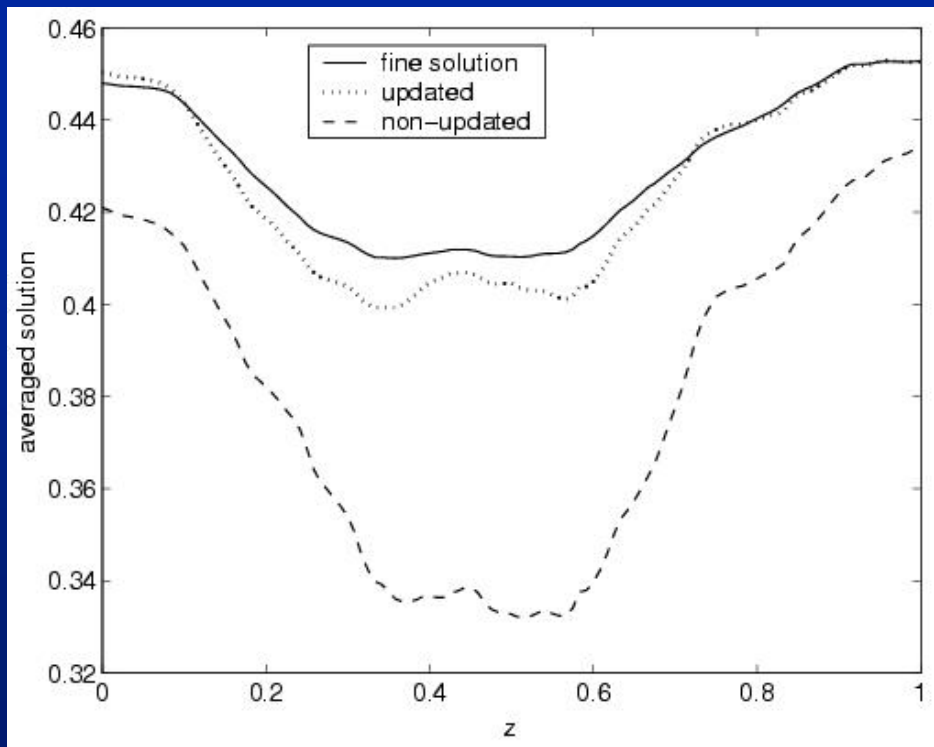
Nonlinear Examples

- Simplify equation to $D_t u_\varepsilon = \mathrm{div}(a_\varepsilon(x, u_\varepsilon) D_x u_\varepsilon)$, where $a_\varepsilon(x, \eta) = k_\varepsilon(x)/(1+\eta)^{\alpha(x)}$, $k_\varepsilon(x) = \exp(\beta_\varepsilon(x))$ is chosen so that $\beta_\varepsilon(x)$ is a realization of a random field with exponential variogram with some correlation structure, and $\alpha(x)$ is chosen so that $\alpha(x) = k_\varepsilon(x) + \text{constant}$. We only specify $\beta_\varepsilon(x)$.
- In figures,
  - solid lines are true solutions
  - Dotted lines are computed solutions *with* updates
  - Dashed lines are computed solutions *without* updates

20 Updates

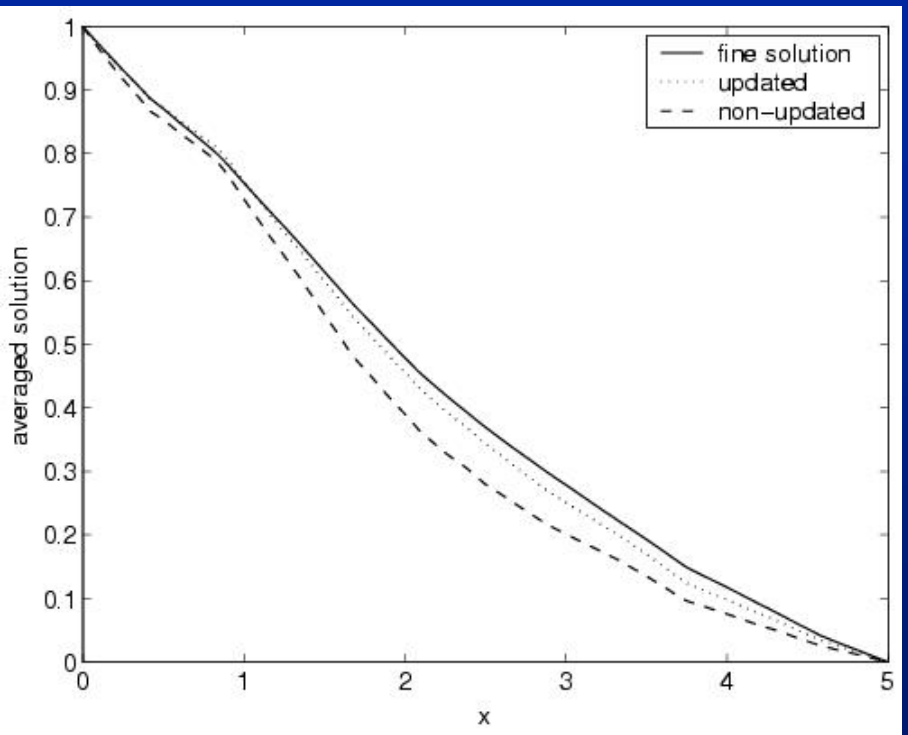
- l_x=0.2 and l_z=0.01, σ=1 while the random field used for simulations is σ=2. l_2 error norms are 2.5% (dotted) and 14% (dashed).

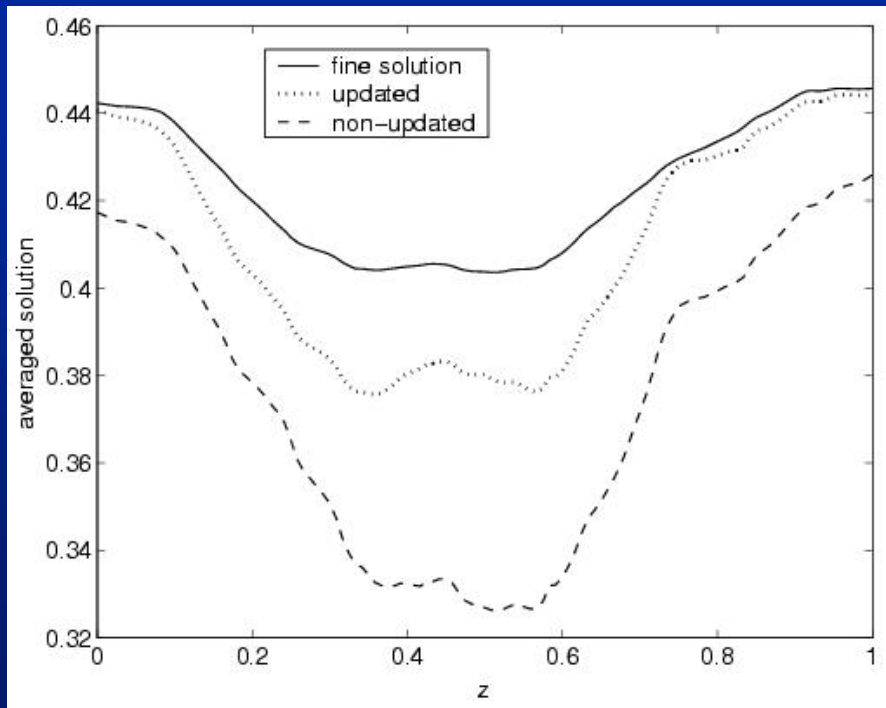




5 Updates

- l_x=0.2 and l_z=0.01, σ=1 while the random field used for simulations is σ=2. l_2 error norms are 5.7% (dotted) and 14% (dashed).





More Typical Example

- Use 3 different hetereogeneous fields with exponential variograms with different probablities
  - True field: $l_x=0.2$ and $l_z=0.01$ with probability 0.1
  - 2nd field: $l_x=0.2$ and $l_z=0.01$ with probability 0.8
  - 3rd field: $l_x=0.4$ and $l_z=0.02$ with probability 0.1
- $\sigma=1$ in all cases
- l_2 error norms are 5% (updated solution) and 9% (non-updated solution).

Why Virtual Telemetry?

- We are doing contaminant tracking. Suppose you have real telemetry data to offer us…
  - Would you *really* admit you are a significant polluter?
  - Do you want the extent of your pollution *known*?
- Oil simulation experts in academia rarely have access to long term actual data from reservoirs. The data is usually sanitized and only for starting simulations from simple datasets.
- ***It is hard to beat the cost of virtual telemetry.***
- We can vary quantity and quality of data using multiple streams.

Out of Country Cooperation

- Offered real telemetry from Rio de Janeiro.
- Cannot afford the phone bill to get the data on current grants.
- Personal pickup *is* an option.
- Has to be a better way (except in January)

FEEMA and LNCC

- Two possible sites that we can history match and then try to predict for the future:
  - The Lagoon in Rio de Janeiro which has sensors in place and is one of the most polluted spots in a major city.
  - River system crossing the border with a neighboring state. Chemical polluters in one of the two states problematic. No cooperation on how much pollution is upstream and flowing into the State of Rio de Janeiro.
- Conventional data feed: FEEMA to LNCC.
- Then use virtual telemetry software to get data to the rest of research team.

Virtual Telemetry: The Other Code

- We have another code running in real time with small time steps for possibly duration of project
  - Known to be very accurate from past history matching.
  - Probably not close to state of the art in terms of runtime.
  - Computes everything each time step.
  - Is computing on a very different mesh than our DDDAS code is.
  - Code streams out data to whoever is interested (and authorized).
  - Data lands on the floor when no one is listening to it or can be saved.

Virtual Telemetry Meets Telemetry

- Design SCIRun modules for telemetry so that real telemetry data can be used in the *near* future.
  - This is a data representation problem
  - Requires a hook to do interpolation correctly into computational mesh
    - Both virtual and real telemetry need hook
    - Requires looking at several applications to do right
    - GUI interface to a simple sensor data editor to build XML based libraries of sensors
    - SCIRun reads sensor library and dynamically builds a sensor reader

Conclusions

- Designing general purpose components that will be useful by DDDAS community.
- Telemetry components inexpensive to set up and use.
  - Transparent when switching between real and virtual telemetry and now integrated into SCIRun.
  - Allows us to investigate easily DDDAS aspects that are not in static data set formulations.
- Inexpensive multiscale methods are being developed for interpolation problems.
- Dissemination through http://www.dddas.org
  - Links to other projects there, too.