Interval Field for Spatially and Temporally Dependent Uncertainty—

Machine Learning Approach

David Betancourt, PhD Student CSE
Rafi L. Muhanna CEE
Robert L. Mullen CEE
Background
Uncertainty in engineering systems

Insufficient modeling of uncertain parameters in engineering systems can lead to:

- Catastrophic events
- Inaccurate expectations of systems performance
- Increased engineering schedules and costs.

In the context of computational mechanics:

- Uncertainty can be spatially-varying or temporally varying.
  - Loadings, stiffness, boundary conditions.
Uncertainty in engineering systems (SFEM)

- In computational mechanics:
  - Spatially or Temporally dependent uncertainty is traditionally modeled with random field theory within the Stochastic Finite Element method (SFEM).
  - SFEM makes assumptions about the probability distribution of the uncertainty, usually Gaussian.
Uncertainty in engineering systems (IFEM)

- In computational mechanics:
  - When we don’t have enough data to select a probability distribution $\rightarrow$ SFEM is inadequate. Is the world Gaussian?
  - Interval Finite Element method (IFEM) is an alternative.

- The IFEM:
  - does not make assumptions about probability distributions.
  - Can model epistemic and aleatory uncertainty.

- The IFEM:
  - Does not include an “interval field” that can model spatially- or temporally-varying uncertainty as is the case with SFEM.
Uncertainty in engineering systems (IFEM)

- In computational mechanics:
  - Need a method to model temporally or spatially varying uncertainty.
  - IFEM is a natural candidate but lacks interval field that can be used for any domain dimension and is scalable.

- In other disciplines:
  - Directed/Undirected graphical models
  - Bayes Nets, MRFs, Gibbs Fields, Hidden MRFs...
  - Graphical Models + Deep Neural Nets
Method
Supervised Interval Field

● IFEM:
  ○ Does not have a unified interval field framework that can model spatially- or temporally- varying uncertainty as in the case with SFEM.

● Solution:
  ○ Use supervised machine learning to infer the uncertain properties in the field:
    ■ *Supervised Interval Field.*
  ○ Works for any domain dimension (1D, 2D, 3D, 4D).
  ○ Independent of IFEM mesh discretization.
Supervised Learning

- Supervised Learning crash course:
  - Learn a predictive model
    - $f : X \rightarrow Y$
  - Train the model with a training set $(X, Y)$ $X \in \mathbb{R}^{n \times d}$ (features) and $Y \in \mathbb{R}^d$ (targets or labels)
  - Approximation of function $Y$ by minimizing a loss function $L(\Theta)$
Supervised Learning

- Supervised Learning (SL) crash course:
  - After model is trained:
    - Make predictions on unseen data.
- Goal of SL:
  - Thou shalt not overfit
  - Generalization
- SL $\rightarrow$ induction.
Supervised Interval Field

- Two ML methods for SIF introduced:
  - Extreme Gradient Boosting (XGBoost)
  - Deep LSTMs
- Will talk about Deep LSTMs
Deep Neural Networks
Neural Nets - a bit of history

- 1958: Psychologist Frank Rosenblatt introduces the perceptron, single-layer NN.
- 1970s: AI winter
- 1989: Yann LeCunn pioneers CNNs to recognize handwritten Zip codes
- 1991: Sepp Hochreiter and Jurgen Schmidhuber pioneer RNNs.
- Mid 1990-2011: other methods such as SVMs, GPs were superior or competitive.
- June 2012: Google’s cat experiment is successful.
Neural Nets - a bit of history


Source: wikipedia
Neural Nets - training

Neural Nets - training

Neural Nets (Convolutional Neural Networks)

- Trained with backpropagation algorithm
- Backprop → parameters optimized with stochastic gradient descent (weights updates).
- Regularization → “dropout”
- More Components:
  - Pooling layer for CNNs
  - Activation: ReLu nonlinearity between layers
  - Loss Functions: Softmax

Source: wikipedia
Implications for Interval Neural Nets

- Must prove convergence for interval input for new deep architectures.
The SIF main tool: Deep Recurrent Neural Networks (using LSTMs)

- Designed for sequential data.
- Have “memory” or choose to “remember” or “forget” information.
- Applications: Speech recognition, translation, DNA sequence analysis, time series, interval field!
SIF-IFEM
SIF-IFEM a unified framework

- SIF provides to the IFEM the uncertain property with respect to the model’s mesh.
- SIF is independent of mesh, therefore, must discretize and map into IFEM mesh.
- Mid-point interval value from SIF prediction assigned to each FE in the mesh.
- Average of the prediction taken over the size of FE.
Experiments
Experiments: Predict soil layer properties

- The problem:
  - Two tests to obtain soil properties.
  - Test 1: \(\{q_c, f_s, u_2\}\)
  - Test 2: \(\{V_s\}\)
  - What if Test 2 is missing or incomplete? What if we want to save time by not performing Test 2 for entire domain?

- Solution:
  - Predict Test 2 Properties with SIF
  - Perform mechanics calculations with SIF-IFEM

- Does it have to be Test 1 (features) obtained at same site?
  - NO

- Greatest benefit: Big data from state-sponsor, company-wide program.

- Data Challenge: Database of labeled engineering data.
Experiments: Training Data

Sample Features

Sample Targets
Experiments: SIF Results

Table 1: $R^2$ Coefficients for Testing Set

<table>
<thead>
<tr>
<th>Model</th>
<th>$R^2$ Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM ($X$)</td>
<td>0.915</td>
</tr>
<tr>
<td>LSTM ($\bar{X}$)</td>
<td>0.909</td>
</tr>
<tr>
<td>LSTM ($\bar{X}$)</td>
<td>0.875</td>
</tr>
<tr>
<td>LSTM ($X + \text{noise}$)</td>
<td>0.896</td>
</tr>
<tr>
<td>XGBoost*</td>
<td>0.228</td>
</tr>
</tbody>
</table>

*XGBoost is only tested with raw features.*
Experiments: SIF-IFEM Results

- SFEM prior ($\mu, \sigma^2$) obtained from the same training set data as for SIF.
- Deterministic solution uses actual measured data (ground truth)
- SIF-FEM closely encloses the solutions without being overly conservative.
- SFEM has 22.7% deviation from deterministic displacement at tip.
- SFEM could have been even worse.
Future Work

- Experiments of Interval Field with image data
- Autonomous decision-making in partially observable environments
  - Reinforcement Learning
  - Imitation Learning