Multi-objective Optimization and Multimodal Prediction in the Design of Materials System

Ankit Agrawal and Alok Choudhary

Center for Ultra-Scale Computing and Information Security (CUCIS),
Department of Electrical Engineering and Computer Science,
Northwestern University
{ankitag,choudhar}@eecs.northwestern.edu

MURI Program Review
January 21, 2013
Outline

❖ Introduction

❖ Project Collaboration
  ❖ Multi-objective Structure-Property Optimization
  ❖ Multimodal Prediction of Localization Relationships
  ❖ Exploring Composition-Processing-Property Linkages in NIMS Steel Database

❖ Summary & Future Work
“Data Mining”
  – the act of extracting information from data and transform it into understandable structures.

“Scalability”
  – the ability of a system to handle a growing amount of work in a capable manner.

... coupled with materials science/informatics
  – Describe process-structure-property-performance relationships
  – Extract microstructure features
  – Predict properties
  – Materials design
Introduction (cont.)

What we do
– make use of computational techniques to solve data problems
– parallelization

Problems we study
– classification, regression, clustering
– dimension reduction
– optimization

Collaborators

- Multi-Scale Structural Simulations Laboratory
- Materials Informatics for Engineering Design
- TATA Consultancy Services
PSPP Linkages

Goal/means

Processing

Structure

Properties

Performance

Cause and effect

Project Collaboration


Project 1. Multi-objective Structure-Property Optimization
Project Collaboration

Project I. Multi-objective Structure-Property Optimization

Project II. Multimodal Prediction of Localization Relationships

Project Collaboration

Project I. Multi-objective Structure-Property Optimization

Project II. Multimodal Prediction of Localization Relationships

Project III. Exploring Composition-Processing-Property Relationships

Project I. Multi-Objective Optimization of Structure-Property Relationships

**Motivation** is to explore the structure–property relationships in polycrystals.

**Collaboration** is achieved by incorporating data mining in computer science and structural simulation in materials.

**Objective** is to obtain structures with desired optimized property.

---

**Microstructure** 

**Property**

---

**Microstructure Data**

**Property Data**

---

Multi-Scale Structural Simulations Laboratory

Center of Ultra-Scale Computing and Information Security

---

Computational data mining models
Project I. Multi-objective Structure-Property Optimization

Problem

How can one identify the microstructure, or set of microstructures, that yield a desired property?

How about a number of desired properties?

Structure Representation

Crystal orientations in the form of an orientation distribution function (ODF), discretized using a finite element approximation in Rodrigues space.

Property Representation

Linear/nonlinear functions
Structure Representation

Volume Fraction Representation

Mathematical representation of all possible ODFs using FE degrees of freedom.

Three constraints define the space of first order microstructural feature (ODF):

- Normalization, $q^T A = 1$
- Lower bound, $A \geq 0$
- Crystallographic Symmetry, $r' = G r$
Property Representation

Property Representation

Magnetism (M), Yield strength (Y), Young’s Modulus (E),

- Magnetism (M): linear
- Yield strength (Y): linear
- Young’s Modulus (E): nonlinear

And their combinations: Y*M/E, Y/T_cr (Critical Temperature for buckling)

- Y*M/E: nonlinear
- Y/T_cr: nonlinear

Objectives

- Given the function/data, optimize the desired property.
- Inversely, retrieve textures with desired properties.

Solution

- Global optimization
- Data mining
Global Optimization

A global optimization problem

\[
\begin{align*}
\text{maximize } & \quad F(X) \\
X & = \{x_1, x_2, \ldots, x_D\} \subseteq \mathbb{R}^D \\
\text{subject to } & \quad \alpha^T X = 1, \quad X \geq 0
\end{align*}
\]

Traditional Method (infeasible)

- Exhaustive search

Optimization Methods

- Linear programming
- Genetic algorithms
- Simulated annealing
Global Optimization (cont.)

Linear Programming (LP)

LP is used to efficiently solve an optimization problem with a linear objective function, subject to linear equality and linear inequality constraints.

Genetic Algorithm (GA)

GA searches for useful solutions to optimization using a heuristic that mimics the process of natural evolution, with operators as crossover and mutation.

Why we need data mining?

High dimension! High dimension! High dimension!
Structure-Property Optimization with Data Mining: Flowchart

Microstructure Representation
- Features that mathematically or statistically describe microstructures

Traditional Method

Database Construction
- Randomly generated microstructure-property pairs with most desired and most undesired objectives

Feature Selection
- Select a small set of “critical” microstructure features

Global Optimization
- Find the value of microstructure that leads to the extremal properties

Data Mining Method
Structure-Property Optimization with Data Mining

Database Construction

• Randomly generate data instances under constraints.
• Keep the most desired class and the most opposing class.

Feature Selection:

*Data mining helps reduce the searching space.*

• Search only valuable variables (76 $\rightarrow$ 10?)
• Search only valuable regions ($odf3 \in [0, 122] \rightarrow [100, 122]$?)

Feature Selection Methods

• Information gain
• Chi-square
• Correlation
• SVM
Feature Selection Results for Combination Problem 1

Problem
Maximize $Y*M/E$.

Feature Rankings

<table>
<thead>
<tr>
<th>Correlation</th>
<th>Chi-square</th>
<th>Info Gain</th>
<th>SVM</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>odf2</td>
<td>odf3</td>
<td>odf3</td>
<td>odf3</td>
<td>odf3</td>
</tr>
<tr>
<td>odf3</td>
<td>odf2</td>
<td>odf2</td>
<td>odf2</td>
<td>odf2</td>
</tr>
<tr>
<td>odf16</td>
<td>odf65</td>
<td>odf65</td>
<td>odf75</td>
<td>odf65</td>
</tr>
<tr>
<td>odf30</td>
<td>odf66</td>
<td>odf66</td>
<td>odf75</td>
<td>odf66</td>
</tr>
<tr>
<td>odf32</td>
<td>odf64</td>
<td>odf66</td>
<td>odf73</td>
<td>odf64</td>
</tr>
<tr>
<td>odf57</td>
<td>odf70</td>
<td>odf70</td>
<td>odf73</td>
<td>odf65</td>
</tr>
<tr>
<td>odf60</td>
<td>odf72</td>
<td>odf72</td>
<td>odf72</td>
<td>odf70</td>
</tr>
<tr>
<td>odf64</td>
<td>odf30</td>
<td>odf30</td>
<td>odf70</td>
<td>odf70</td>
</tr>
<tr>
<td>odf65</td>
<td>odf10</td>
<td>odf10</td>
<td>odf16</td>
<td>odf64</td>
</tr>
<tr>
<td>odf66</td>
<td>odf24</td>
<td>odf24</td>
<td>odf16</td>
<td>odf66</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Structure-Property Optimization:
Optimum Found & Time Saved

Experiment Result: Solution found/Performance vs. Number of Variables

Optimum Solution

Time Consumption (s)

Number of Variables

Optimum found
Time Consumed
Project II: Multimodal Prediction of Structure–Response Localization

**Problem:** Given a microstructure, predict the spatial distribution of the response at the microscale.

**Solution:**
- Divide into multiple training groups based on volume fraction
- Extract neighborhood information at voxel scale
- Construct additional features from neighbor input
- Key feature selection
- Generate regresional response output

This is a collaborative work between Surya’s group at Georgia Institute of Technology and CUCIS group at Northwestern
Project II. Multimodal Prediction of Structure–Response Localization

Problem

How can one model the microstructure-response relationship and predict the response at microscale?

The responses within a cube is constrained.

Data

- A large number of 21x21x21 3D cubes of a two-phase composite material
- Corresponding 3D cubes of its strain fields response
- Contrast ratio 10: 2500 cubes
- Contrast ratio 50: 2500 cubes
Steps for Structure–Response Prediction

Step 1: Divide training cubes into groups
- Cubes within a group have higher similarity
- Use volume fraction as the criterion

Step 2: Neighbor information extraction
- Enumerate neighbor voxels up to the 12th level
- Add novel composite features
- Supervised feature selection process to identify key features

Step 3: Response prediction with regression analysis
- Build regression tree with highly influencing features
- Model ensemble to achieve greater accuracy
Feature Selection Flowchart

Define an original (large) set of neighbor features.

Direct neighbors

Feature Selection

Obtain a ranking of the features and determine a final (small) set.

Composite features
Direct Neighbor Extraction

Extract the neighbors’ information

• Neighbors are distinguished by center-to-center distance.

• A voxel in a 3D cube has
  - 1\textsuperscript{st} neighbors – 6
  - 2\textsuperscript{nd} neighbors – 12
  - 3\textsuperscript{rd} neighbors – 8
  - 4\textsuperscript{th} neighbors – 6
  - 5\textsuperscript{th} neighbors – 24
  - 6\textsuperscript{th} neighbors – 24
  - 7\textsuperscript{th} neighbors – 12
  - ...

Extra Composite Feature Creation

• Besides direct neighbor voxel inclusions, extra features are created and added
  • by aggregating sets of neighbor voxels to form composite features

• Composite features
  ▪ Weighted 0/1 neighbors – (2)
    ▪ Sum of (1/distance) of all 0/1 neighbors
  ▪ Number of 0s/1s in a certain level – (12)
  ▪ Number of 0s/1s up a certain level – (12)
  ▪ Encoded level 1/2/3/4 neighbors – (4)
    ▪ Consider neighbors’ physical placements by converting it into a binary-coded decimal
# Feature Selection Results

## Feature Rankings

<table>
<thead>
<tr>
<th>Rank</th>
<th>Feature</th>
<th>Rank</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Self</td>
<td>17</td>
<td>Weighted_0_nbrs</td>
</tr>
<tr>
<td>2</td>
<td>Nbr1.2</td>
<td>18</td>
<td>Num.1s.sofar.nbr1</td>
</tr>
<tr>
<td>3</td>
<td>Nbr1.3</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>4</td>
<td>Nbr1.1</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>5</td>
<td>Nbr1.0</td>
<td>24</td>
<td>Num.0s.sofar.nbr1</td>
</tr>
<tr>
<td>6</td>
<td>Nbr1.4</td>
<td>25</td>
<td>Nbr2.6</td>
</tr>
<tr>
<td>7</td>
<td>Nbr1.5</td>
<td>26</td>
<td>Nbr5.6</td>
</tr>
<tr>
<td>8</td>
<td>Nbr2.2</td>
<td>27</td>
<td>Nbr5.10</td>
</tr>
<tr>
<td>9</td>
<td>Nbr2.3</td>
<td>28</td>
<td>Nbr2.9</td>
</tr>
<tr>
<td>10</td>
<td>Nbr2.0</td>
<td>29</td>
<td>Nbr8.28</td>
</tr>
<tr>
<td>11</td>
<td>Nbr2.1</td>
<td>30</td>
<td>Nbr5.11</td>
</tr>
<tr>
<td>12</td>
<td>Nbr4.4</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>13</td>
<td>Nbr2.4</td>
<td>82</td>
<td>Weighted_1_nbrs</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>83</td>
<td>Encode.nbr1</td>
</tr>
</tbody>
</table>

...
Validation of Feature Selection

For comparison purpose, we run experiments on a small portion of the data (4 training, 1 testing) with contrast ratio 10, with various features.

<table>
<thead>
<tr>
<th>Features used</th>
<th>Number of features</th>
<th>Error rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1: Neighboring information up to the 12th level</td>
<td>233</td>
<td>0.101</td>
</tr>
<tr>
<td>F2: Neighboring information up to the 5th level</td>
<td>69</td>
<td>0.1006</td>
</tr>
<tr>
<td>Selected feature out of F1</td>
<td>69</td>
<td>0.0962</td>
</tr>
</tbody>
</table>
Prediction Results

Data Split
Contrast 10: no predefined train/test split. We use 4 varieties.
Contrast 50: out of 2500, 2000 are held for training, and 500 for testing.

Prediction Results

<table>
<thead>
<tr>
<th>Contrast 10 Models</th>
<th>MASE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>500/500 split</td>
<td>3.74</td>
</tr>
<tr>
<td>500/2000 split</td>
<td>5.86</td>
</tr>
<tr>
<td>1000/1000 split</td>
<td>3.25</td>
</tr>
<tr>
<td>1000/1500 split</td>
<td>4.22</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contrast 50 Models</th>
<th>MASE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression tree only</td>
<td>13.67</td>
</tr>
<tr>
<td>Regression tree with feature selection</td>
<td>11.86</td>
</tr>
</tbody>
</table>

\[
MASE = \frac{1}{N} \sum_{N} \left| \frac{y - \hat{y}}{y_{imposed}} \right| \times 100\%
\]
**Objective:** Employ data-driven approaches to the NIMS public domain materials database for exploring composition-processing-property relationships and constructing predictive models for fatigue strength of steels.
NIMS Database Attributes

**Fatigue Data Sheet Information:**

**Chemical composition** - %C, %Si, %Mn, %P, %S, %Ni, %Cr, Cu %, Mo% (all in wt. %)

**Upstream processing details** - Ingot size, Reduction ratio, Non-metallic inclusions

**Heat treatment conditions** – Temperature, Time and other process conditions for Normalizing, Carburizing-Quenching and Tempering processes

**Mechanical properties** - YS, UTS, %EL (Elongation), %RA (Reduction in Area), Vickers Hardness, Charpy impact value (J/cm²), Rotating bending fatigue strength @ $10^7$ cycles

Total - 437 data records
Carbon and low alloy steels - 371 observations,
Carburizing steels - 48 observations and Spring steels - 18 observations

Steel Fatigue Strength Prediction Framework

Preprocessing → Feature Selection → Predictive modeling → Evaluation

Rotating Bending Fatigue Testing Data from NIMS

Preprocessed Data

Fatigue Strength Prediction Database

Training Split

Leave One Out Cross Validation (LOOCV)
Cluster Visualization

kmeans clustering

- Cluster 1
- Cluster 2
- Cluster 3
- Centroids
Information Gain Based Feature Ranking

Relative Attribute Importance

- TT
- NT
- Cr
- THCr
- C
- CTH
- T
- TCr
- DT
- D
- Qt
- C
- Ch
- T
- Th
- Mo
- N
- Mn
- RedRatio
- dA
- Si
- dC
- Cu
- S
- dB
- P
Evaluation Metrics

- Compare vectors of actual and predicted values
  - Coefficient of correlation (R)
  - Coefficient of determination ($R^2$)
  - Mean Absolute Error (MAE)
  - Root Mean Squared Error (RMSE)
  - Standard Deviation of Error (SDE)
  - Mean Absolute Error Fraction (MAE)
  - Root Mean Squared Error Fraction (RMSE)
  - Standard Deviation of Error Fraction (SDE)

\[ R = \frac{\sum_{i=1}^{N} (y_i - \bar{y})(\hat{y}_i - \bar{y})}{\sqrt{\sum_{i=1}^{N} (y_i - \bar{y})^2 \sum_{i=1}^{N} (\hat{y}_i - \bar{y})^2}} \]
\[ MAE = \bar{e} = \frac{1}{N} \sum_N |y - \hat{y}| \]
\[ RMSE = \sqrt{\frac{1}{N} \sum_N (y - \hat{y})^2} \]
\[ SDE = \sqrt{\frac{1}{N} \sum_N (|y - \hat{y}| - \bar{e})^2} \]
\[ MAE_f = \bar{e}_f = \frac{1}{N} \sum_N \left| \frac{y - \hat{y}}{y} \right| \]
\[ RMSE_f = \sqrt{\frac{1}{N} \sum_N \left( \frac{y - \hat{y}}{y} \right)^2} \]
\[ SDE_f = \sqrt{\frac{1}{N} \sum_N \left( \left| \frac{y - \hat{y}}{y} \right| - \bar{e}_f \right)^2} \]
Results Comparison

![Bar chart showing comparison of performance metrics for different models.](chart.png)
## Results Comparison

<table>
<thead>
<tr>
<th>Method</th>
<th>$R$</th>
<th>$R^2$</th>
<th>MAE</th>
<th>RMSE</th>
<th>SDE</th>
<th>$MAE_f$</th>
<th>$RMSE_f$</th>
<th>$SDE_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecisionTable</td>
<td>0.9494</td>
<td>0.9014</td>
<td>34.8762</td>
<td>58.5932</td>
<td>47.1371</td>
<td>0.0584</td>
<td>0.0806</td>
<td>0.0557</td>
</tr>
<tr>
<td>IBk</td>
<td>0.9589</td>
<td>0.9195</td>
<td>46.0320</td>
<td>53.2749</td>
<td>26.8499</td>
<td>0.0859</td>
<td>0.0940</td>
<td>0.0382</td>
</tr>
<tr>
<td>KStar</td>
<td>0.9702</td>
<td>0.9413</td>
<td>36.9986</td>
<td>45.3779</td>
<td>26.3029</td>
<td>0.0706</td>
<td>0.0857</td>
<td>0.0487</td>
</tr>
<tr>
<td>SVM</td>
<td>0.9795</td>
<td>0.9594</td>
<td>24.2820</td>
<td>37.6250</td>
<td>28.7736</td>
<td>0.0400</td>
<td>0.0530</td>
<td>0.0349</td>
</tr>
<tr>
<td>LRTtrans</td>
<td>0.9796</td>
<td>0.9596</td>
<td>22.3336</td>
<td>37.4748</td>
<td>30.1272</td>
<td>0.0370</td>
<td>0.0514</td>
<td>0.0357</td>
</tr>
<tr>
<td>RobustFitLSR</td>
<td>0.9804</td>
<td>0.9612</td>
<td>22.2152</td>
<td>37.2188</td>
<td>29.8960</td>
<td>0.0369</td>
<td>0.0520</td>
<td>0.0366</td>
</tr>
<tr>
<td>LinearRegression</td>
<td>0.9815</td>
<td>0.9633</td>
<td>25.6006</td>
<td>35.7168</td>
<td>24.9345</td>
<td>0.0456</td>
<td>0.0581</td>
<td>0.0360</td>
</tr>
<tr>
<td>PaceRegression</td>
<td>0.9816</td>
<td>0.9635</td>
<td>25.0302</td>
<td>35.5733</td>
<td>25.3065</td>
<td>0.0439</td>
<td>0.0565</td>
<td>0.0356</td>
</tr>
<tr>
<td>ANN</td>
<td>0.9861</td>
<td>0.9724</td>
<td>19.7778</td>
<td>31.0545</td>
<td>23.9695</td>
<td>0.0343</td>
<td>0.0470</td>
<td>0.0322</td>
</tr>
<tr>
<td>REPTree</td>
<td>0.9862</td>
<td>0.9726</td>
<td>22.5671</td>
<td>30.9401</td>
<td>21.1907</td>
<td>0.0414</td>
<td>0.0542</td>
<td>0.0349</td>
</tr>
<tr>
<td>M5ModelTree</td>
<td>0.9890</td>
<td>0.9781</td>
<td>19.3760</td>
<td>27.6065</td>
<td>19.6870</td>
<td>0.0353</td>
<td>0.0484</td>
<td>0.0332</td>
</tr>
<tr>
<td>MPR</td>
<td>0.9900</td>
<td>0.9801</td>
<td>18.5529</td>
<td>26.4378</td>
<td>18.8563</td>
<td>0.0350</td>
<td>0.0556</td>
<td>0.0432</td>
</tr>
</tbody>
</table>
I. Structure-property optimization
  – Global optimization methods are used to solve the problem.
  – Dimension reduction speeds up the search dramatically.
  – It is able to discover answers that would be otherwise difficult to find.

II. Microstructure-response modeling
  – Feature selection is able to discover valuable localized information.
  – A classification ensemble based meta model is able to approximate the FEM process precisely.

III. Steel Fatigue Strength Prediction
  – Neural networks, decision trees, multivariate polynomial regression able to achieve high $R^2$ values of $>0.97$. 
Future Work

• A generalized data mining tool for global optimization
  – Random data generation subject to constraints
  – Searching space reduction
  – Starting point selection

• A multi-scale structure-response modeling
  – Macro-scale: consider the geometry of 0/1 placement within a cube
  – Micro-scale: constrained regression within a cube

• Enhanced prediction of mechanical properties in steels
  – Ensemble predictive modeling
  – Hierarchical predictive modeling
  – Predict other properties like %Elongation


Thank you!