A PARAMETRIC INVESTIGATION OF THE DIAGNOSTIC ABILITY OF PROBABILISTIC NEURAL NETWORKS ON TURBOFAN ENGINES

C. Romessis
Research Assistant

A. Stamatis
Research associate

K. Mathioudakis
Assistant Professor

Laboratory of Thermal Turbomachines
National Technical University of Athens
A PARAMETRIC INVESTIGATION OF THE DIAGNOSTIC ABILITY OF PROBABILISTIC NEURAL NETWORKS ON TURBOFAN ENGINES

- Formulation of the diagnostic problem
- Features of Probabilistic Neural Networks
- Study of factors affecting diagnostic performance of PNN
  - Effect of Training Set
  - Effect of Measurements’ noise
  - Effect of the severity of a fault
  - Effect of the operating conditions
  - Effect of the selected diagnostic parameters
- Summary - Conclusions
Formulation Of The Diagnostic Problem (I)

**OBJECTIVE:** PNN to diagnose component faults of High-by-Pass ratio, partially mixed, TURBOFAN engine

*Layout of a turbofan engine and station numbering for positions of interest.*
Formulation Of The Diagnostic Problem (II)

Information Provided

- Quantities defining operating Condition (4)
  - Ambient Conditions
  - Flight Speed
  - Set Point Variable (Fuel flow rate)

- Quantities for Deducing Engine Condition (7)
  - Shaft speeds
  - Pressure, Temperatures

Information Required

- Condition of individual engine components
  - Kind, Location, Severity (magnitude) of faults
PNN Generation

(1) Architecture

• **Input Layer:**
  Each node represents a measured quantity
  (Inputs : $\Delta Y = \frac{(Y - Y_{ref})}{Y_{ref}} \times 100\%$)

• **Hidden Layer:**
  Each node represents a training pattern
  (a-priori given information)

• **Output Layer:**
  Each node represents a fault parameter
  (probability values derived by NN)
A Parametric Investigation of The Diagnostic Ability Of Probabilistic Neural Networks On Turbofan Engines
PNN Generation

(II) A priori information

• Training Patterns

• Generated by Engine Performance model
  • Reference information (“healthy” - $Y_{ref}$)
  • Faulty Engine Information (Y)
Turbofan Engine Modeling

**Quantities defining the operating Conditions (4):**

- Ambient Pressure
- Fuel consumption
- Engine Inlet Conditions (pressure, temperature)

**Fault Parameters (11):**

- Flow factors of positions of interest: \( SW_i = \frac{W_i \cdot \sqrt{T_i}}{p_i} \left( \frac{W_i \cdot \sqrt{T_i}}{p_i} \right)_{ref} \)

- Efficiency factors of positions of interest: \( SE_i = \frac{\eta_i}{(\eta_i)_{ref}} \)

- Exhaust area (numbered station 8)

**Measured quantities (7):**

- Shafts’ speed (low and high pressure)
- Pressures and temperatures at stations along the engine
Creating Fault Patterns

(Containing noise)

Procedure for generating noise-free patterns by EPM and adding noise on them

\[ \Delta Y = \frac{(Y^* - Y_o^*)}{Y_o^*} \times 100 \]
Generating the Training – Testing patterns

Patterns (generated by EPM)

Deltas of measurements for a specific set of deltas of the fault parameters

Considered faults

• 11 Single faults: Delta of 1 fault parameter deviates within a range of +/- 3%

• 5 Combined faults: Deltas of 2 fault parameters, of the same module, deviate.

* Delta of a value $Y$: $\Delta Y = \frac{Y - Y_{ref}}{Y_{ref}} \times 100\%$
Noise Situations Considered
(reproduce realistically the data collected from an actual engine)

5 noise levels considered

<table>
<thead>
<tr>
<th>noise level</th>
<th>noise in u</th>
<th>noise in y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>( \sigma_y/2 )</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>( \sigma_y )</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>( 3\sigma_y/2 )</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>( 2\sigma_y )</td>
</tr>
<tr>
<td>5</td>
<td>( \sigma_u )</td>
<td>( \sigma_y )</td>
</tr>
</tbody>
</table>

• \( \sigma_u \): noise due to Operating Condition uncertainty
• \( \sigma_y \): noise due to sensor inaccuracies
## Considered Probabilistic Neural Networks

<table>
<thead>
<tr>
<th>ID</th>
<th>No. of input nodes</th>
<th>Operating point</th>
<th>Noise level included</th>
<th>No. of training patterns</th>
<th>No. of output nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>cruise</td>
<td>No noise</td>
<td>112</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>cruise</td>
<td>1st,2nd, 3rd, 5th</td>
<td>2240</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>cruise</td>
<td>5th level</td>
<td>1120</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>take-off</td>
<td>1st,2nd, 3rd, 5th</td>
<td>2240</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>take-off</td>
<td>5th level</td>
<td>1120</td>
<td>11</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>cruise</td>
<td>5th level</td>
<td>1120</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td>cruise &amp; take-off</td>
<td>5th level</td>
<td>2240</td>
<td>11</td>
</tr>
</tbody>
</table>
Effect of noise level on the training patterns
Performance of a network tested on patterns partially used for training and on non-training patterns
Effect of noise level in the patterns presented to the network.

Network trained using data from all noise situations.
Effect of magnitude of fault on the diagnostic performance of PNN.
Effect of noise magnitude on test patterns

Fault 3: SW2
Fault 8: SE41
Diagnostic performance of three networks, each one trained and tested at a different operating point.
Diagnostic performance of two networks with different output configuration.

![Graph showing diagnostic performance](image)
Conclusions

• The influence of noise is important. Training with noisy data improves the performance.

• When a network is used, the performance is better for less noise in the data. Filtering procedure of measurements before they are fed to such a network may thus be beneficial.

• Possibility of correct detection of a fault is better for more severe faults, namely when the magnitude of deviation is bigger.

• It is safer to make diagnosis based on observations at higher power settings (e.g. take-off versus cruise conditions).